PUBLIC

2018-02-16

Bank Analyzer (FS-BA)

Content

1 Bank Analyzer (FS-BA)

Use

Bank Analyzer supports you in your risk and return management by calculating, measuring, and analyzing financial products. Bank Analyzer can also assist you in the implementation of appropriate regulatory requirements (for example, International Accounting Standards (IAS) / International Financial Reporting Standards (IFRS) or Basel regulations). The structure of Bank Analyzer is based on the Integrated Finance and Risk Architecture (IFRA).

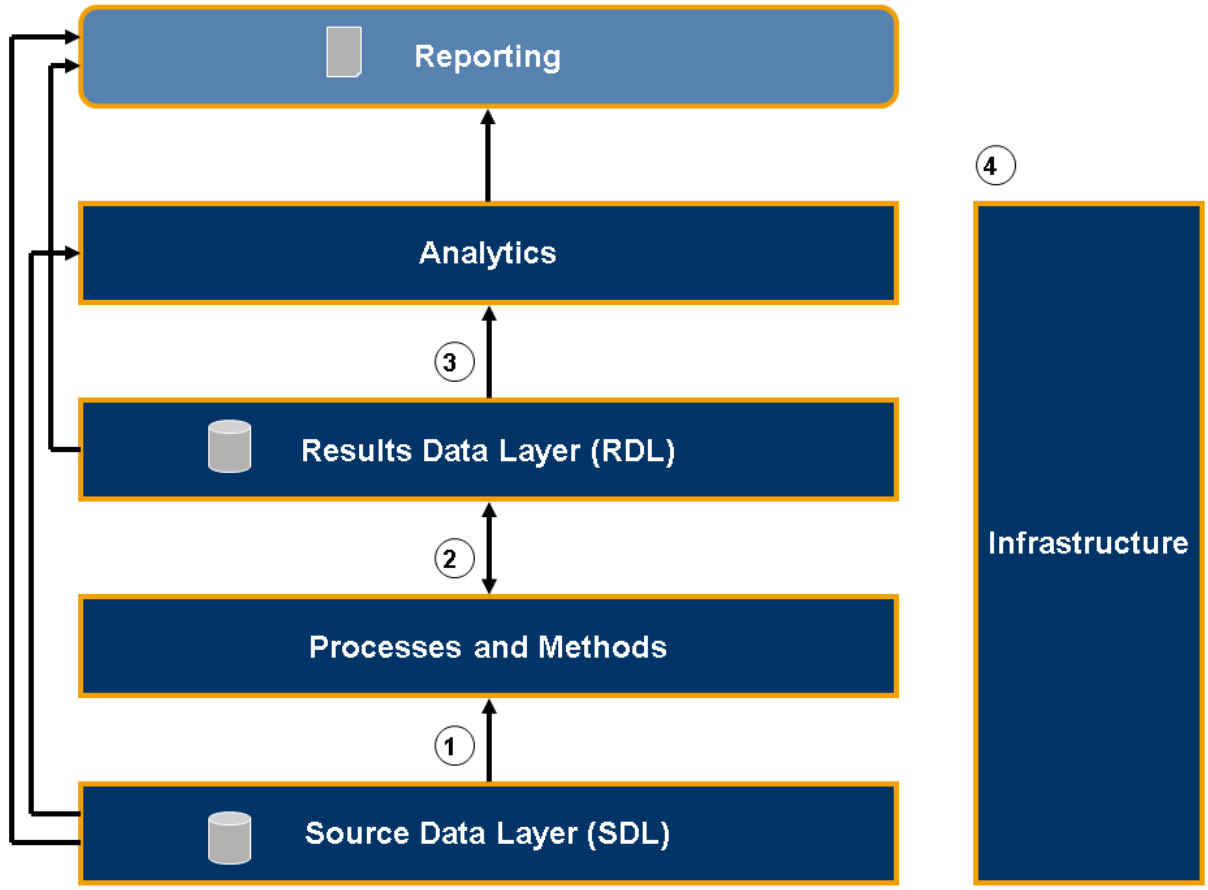
Bank Analyzer consists of the following components:

- Data Load Layer (FS-BA-DL) [page 1581]
- Source Data Layer (FS-BA-SD) [page 100]
- Processes and Methods (FS-BA-PM) [page 366]
- Results Data Layer (FS-BA-RD) [page 1364]
- Analytics (FS-BA-AN) [page 1410]
- Infrastructure (FS-BA-IF) [page 1581]
- Tools (FS-BA-TO) [page 1821]

The division of the components ensures that data is stored in an integrated and consistent way. The system loads original data from operational systems or source systems into the Source Data Layer (SDL). The SDL is the original data basis for the processes and methods of Bank Analyzer. The valuation results of processes and methods are stored in the Results Data Layer (RDL). This structure ensures that original data, methods, and valuation results are clearly separated. The open, modular structure of Bank Analyzer supports a gradual implementation into existing system landscapes.

Bank Analyzer provides a consistent view of a bank's operational data and enables you to process data promptly so that you are always in a position to provide current financial and risk information. Results data is therefore always available for decision-making and for day-to-day business.

The figure below shows the structure of Bank Analyzer:



Structure of Bank Analyzer

1. The SDL manages the basic data for the measurement of financial products. This data is loaded from the operational source systems by means of extraction, transformation, and loading (ETL) processes.
   The SDL is the source for semantically integrated data for all valuation processes that are based on financial products, and is also a central consolidated source for analyses.
   The SDL is not used to store data that has already been analyzed completely. Instead, this data is stored in the RDL.
2. The RDL manages consistent and reusable financial and risk data from various calculation and valuation processes for financial instruments and financial transactions.
3. Reporting and Analytics read results data from the RDL. The Analytics layer contains analytical applications that call results from the RDL and process them as required. This means that results data is analyzed specifically for each application.
4. Infrastructure and Tools provide central services and utilities for the various Bank Analyzer components.

> **Note**
>
> In addition to the RDL, Bank Analyzer also has a Result Database [page 1823] (RDB). RDL and RDB are two different results databases where the system can store results data. The RDB is found in a variety of forms in Bank Analyzer. These forms depend on the various areas (Financial Accounting, Credit Risk). The RDL is

a standardized results data store for accounting and risk-based analyses of financial transactions or financial instruments.

Integration

The integrated data store for product-based source and results data is based on SAP NetWeaver Business Intelligence technology. SAP NetWeaver is the basis for integrating Bank Analyzer in various IT environments and internal bank solutions.

Features

Bank Analyzer contains the following solutions:

SAP Financial Database

The *SAP Financial Database* solution offers an extensive database infrastructure for analytical data and accompanying data processing systems. It is technically compatible with other SAP applications and with third-party applications.

SAP Financial Database uses the following Bank Analyzer components:

- SDL (FS-BA-SD) [page 100]
- RDL (FS-BA-RD) [page 1364]
- Cash Flow Generation (FS-BA-PM-GM-CFG) [page 440]
- Correction Server(FS-BA-IF-CS) [page 1719]

The system uses ETL processes to load original data from other systems or source systems into the SDL in the form of primary objects. Primary objects are a flexible way of storing master data and flow data in entities that belong together logically from a business perspective.

Results data from financial calculations and valuations are stored in the RDL in results data areas in the form of result types. The SAP Financial Database uses the SDL and RDL to support the extensive versioning and authorization concept. In the SDL it provides functions to support the principle of dual control. This means that you can define special release rules to protect certain processes.

Cash flow generation generates cash flows that are made up of a number of flows (for example, disbursement, interest, payment).

The correction server enables data flow management and records corrections to find and display any inconsistencies. If the relevant system settings have been made, the correction server can find and display any entities belonging to these corrections.

SAP Credit Risk Management for Banking

SAP Credit Risk Management for Banking provides an infrastructure that you can use to calculate exposure key figures, which serve as the basis for calculating the capital requirement, and therefore supports you in complying with supervisory guidelines such as Basel III. A special algorithm takes care of the optimal distribution of collateral to exposures in order to minimize the capital requirement. *SAP Credit Risk Management for Banking* offers a central data store for the credit risk data required by the reporting processes and supervisory review processes.

The system can run the calculation not only for real data, but also for stress data (for example, changes in the rating data of a sovereign or business partner).

In addition to the calculation of default risks, *SAP Credit Risk Management for Banking* also supports the limiting and analysis of risks. You can control risk key figures using central limits. The reporting functions for internal and external purposes allows you to monitor risks and their limits. You can use an online limit check when you enter a transaction to make sure that the new transaction does not violate any existing portfolio limits.

SAP Credit Risk Management for Banking uses the following Bank Analyzer components:

- Account Pooling [page 383]
- Free Line for Facilities [page 395]
- Collateral Distribution [page 404]
- Determination of Default [page 410]
- Credit Exposure [page 1192]
- Historical Database [page 1478]
- Disclosure and Reporting [page 1527]
- Limit Manager [page 1529]

> ⚠ Caution
>
> The productive use of *SAP Limit Manager* is only permitted for volumes not exceeding 300,000 transactions. If the volume of your business exceeds 300,000 transactions, an SAP consultation is required. For more information, contact your SAP account executive, or create a customer incident under component FS-BA.

SAP Accounting for Financial Instruments

In the *SAP Accounting for Financial Instruments* solution, you use Bank Analyzer as a subledger for the accounting of financial instruments.

Subledger Scenario

You transfer financial instrument data to the Bank Analyzer system here. You can then post and price the related business transactions, aggregate documents, and transfer them to the general ledger. You can also create the financial statements for the end of the period. You can link the hedging relationships between financial instruments, test the effectiveness of the hedging relationships as per the accounting rules, and create accounting documents for the hedged items.

You can import subledger documents from external systems directly into the Bank Analyzer system and copy these to the general ledger.

In addition to the SDL and the RDL, the subledger scenario uses the following functions:

- Accounting Processes [page 693]
- Hedge Processes (FS-BA-PM-HP) [page 975]
- General Ledger Connector (FS-BA-AN-GL) [page 1411]
- Financial Statement Preparation (FS-BA-AN-FSP) [page 1425]
- Use of Imported Subledger Documents in the Subledger Scenario [page 958]

Smart Accounting for Financial Instruments

Smart Accounting for Financial Instruments (Smart AFI) provides a modular subledger for financial instruments. Smart AFI consistently follows an underlying accounting process model in its process steps, in the application menu, and in Customizing.

Smart AFI follows a central GAAP approach. This means that the system only creates postings that are not dependent on GAAP once. The system creates postings that are dependent on GAAP once for each GAAP. Operational process steps create central GAAP documents accordingly, while the remaining process steps create delta GAAP documents.

Operational master data, flow data and exchange rates are imported to the Source Data Layer (SDL) in simplified form. Accruals need to be imported to the Results Data Layer (RDL). The subledger documents created by Smart AFI are persisted in the RDL and provided to subsequent applications from there (for example, the general ledger or Financial Data Mart). The general ledger documents created for the general ledger connection also follow the central GAAP approach, which is compatible with SAP General Ledger's (FI-GL) leading ledger approach.

In this way, Smart AFI provides the required functions for a subledger optimized for performance and data volume.

SAP Hedge Management

The *SAP Hedge Management* solution supports you in your hedging activities under IAS 39. Bank Analyzer supports fair value hedges and cash flow hedges as well as portfolio fair value hedges. The system identifies hedged objects and hedging instruments, and maps these as hedging relationships. Bank Analyzer provides prospective and retrospective effectiveness tests, and extensive functions for hedge accounting.

Additional Components

Impairment Processes (FS-BA-PM-IP) [page 963]

> ⚠ Caution
>
> Before going live with *SAP Accounting for Financial Instruments*, a gap analysis and an SAP consultation are required. In particular, it is imperative that the planned processing volumes and the timeframes available are analyzed. For more information, contact your SAP account executive, or create a customer incident under component FS-BA.

2 Basic Settings

You have to make basic settings before you can make other settings in the Bank Analyzer system. Basic settings include:

- System Landscape for Bank Analyzer [page 13]
- Characteristic Repository [page 20]
- Key Figure Repository [page 72]
- Connection to the Garbage Collector [page 97]

2.1 System Landscape for Bank Analyzer

Use

The Bank Analyzer system landscape comprises all of the Bank Analyzer systems that are linked together as a result of transports of metadata and application data or client copies and RFC calls.

Structure

The system landscape is made up of Bank Analyzer Core systems (development, consolidation, and productive system) and the Tool BIs [page 16] assigned. A Tool BI system must be assigned to each client in the core Bank Analyzer system. If a Tool BI system is not explicitly assigned to a client in the core Bank Analyzer system, the relevant client accesses the local BI and uses this as the Tool BI system. The Reporting BI system is used for reporting in Bank Analyzer and other applications.

> **Note**
>
> See Data Load Layer [page 1581] for information about settings in the system landscape for the Data Load Layer (DLL).

Integration

In the core Bank Analyzer system, data can be stored separately in different clients. This is not currently possible in the Tool BI system. You can use the following scenarios to keep data separate in Bank Analyzer:

- Scenario 1: Local Tool BI system with source system ID
- Scenario 2: Remote Tool BI system with source system ID
- Scenario 3: Remote Tool BI system without source system ID

> **Caution**
>
> We recommend you use a local Tool BI system with source system ID (scenario 1).

You cannot switch to another scenario after you have made Customizing settings.

Scenario 1: Local Tool BI System with Source System ID:

- **Definition:** In this scenario, the core Bank Analyzer system and the Tool BI system are mapped in one system. In doing so, Tool BI systems are not assigned to the core Bank Analyzer clients. The BI meta data is valid in all of the core clients involved. This client data is separated using the source system ID. You have to enter a two-character value as the source system ID for each core Bank Analyzer client. To prevent the metadata in the Tool BI system becoming inconsistent, one of the core clients is identified as the main client. You can replicate some metadata (key figure definition, local characteristics, segment hierarchy types) only from the leading client into the Tool BI system.
- **Settings in the Tool BI system:** In the Data Warehousing Workbench, choose *Maintain Master Data* and assign the source system ID of InfoObject 0SOURSYSTEM to the relevant source system. In addition, include the *source system ID* characteristic in all the InfoObjects for which client-specific master data exists.

  > **Note**
  >
  > If metadata from the core Bank Analyzer system is created in the Tool BI system, the characteristic is included automatically in the source system ID if it has client-specific master data.

- **Settings in the core Bank Analyzer system:** In Customizing for *Bank Analyzer*, make the settings for the system landscape under *Basic Settings* > *Global Settings* > *Define System Landscape for Bank Analyzer*.
- **Master data hierarchies:** The data in master data hierarchies cannot be kept separate using source system IDs. Since the master data contains the reference to the source system ID, ensure that you use master data related to the same source system ID when you create master data hierarchies.
- **Authorizations:** To enable authorization checks to be carried out, you have to set the *Authorization Relevant* indicator in the InfoObject 0SOURSYSTEM for the source system ID.

Scenario 2: Remote Tool BI System with Source System ID:

- **Definition**: In this scenario, you can assign more than one core client to the same tool BI system. In this case, the BI metadata is valid in all of the core clients involved. If you do not want the master data to be client-independent, they need to be kept separate in regard of the core client. You do this using the source system ID. Since there is at least one logical BI system that is assigned to several core clients, you have to enter a two-character value as the source system ID for each core client. To prevent the metadata in the Tool BI system becoming inconsistent, one of the core clients with the same BI system is identified as the main client. You can replicate some metadata (key figure definition, local characteristics, segment hierarchy types) only from the leading client into the Tool BI system.
- **Settings in the Tool BI system:** In the Data Warehousing Workbench, choose *Maintain Master Data* and assign the source system ID of InfoObject 0SOURSYSTEM to the relevant source system. In addition, include the source system ID characteristic in all the InfoObjects for which client-specific master data exists.

  > **Note**
  >
  > If metadata from the core Bank Analyzer system is created in the Tool BI system, the characteristic is included automatically in the source system ID if it has client-specific master data.

- **Settings in the core Bank Analyzer system:** In Customizing for *Bank Analyzer*, make the settings for the system landscape under *Basic Settings* > *Global Settings* > *Define System Landscape for Bank Analyzer*.
- **Master data hierarchies:** The data in master data hierarchies cannot be kept separate using source system IDs. Since the master data contains the reference to the source system ID, ensure that you use master data related to the same source system ID when you create master data hierarchies.

- **Authorizations:** To enable authorization checks to be carried out, you have to set the *Authorization Relevant* indicator in the InfoObject 0SOURSYSTEM for the source system ID.

Scenario 3: Remote Tool BI System Without Source System ID:

- **Definition:** A separate Tool BI system is assigned to each core client, which means that a given Tool BI system is not assigned to more than one client. The various Tool BI systems and, therefore, core clients are completely independent of one another.
- **Settings in the Tool BI system:** In this scenario, you do not have to make any settings for the source system ID in the Tool BI systems.
- **Settings in the core Bank Analyzer system:** In Customizing for *Bank Analyzer*, make the settings for the system landscape under *Basic Settings* > *Global Settings* > *Define System Landscape for Bank Analyzer*. The Source System ID field is not relevant in this scenario.

Advantages and Disadvantages of the Scenarios

	Advantages	Disadvantages
Scenario 1	• Prevention of performance problems as a result of Remote Function Calls. • Prevention of application errors when data is not available. • Less administration as a result of fewer BI systems. • Lower configuration requirements, since the part of the data model contained in the tool BI system only has to be set up once and is then available in all the connected clients. • Characteristic master data can be client specific or cross client.	• Clients are not completely independent because they contain the same metadata. • User administration applies to one Tool BI system. Detailed data protection settings must be made by assigning authorizations. • A consistent approach must be adopted regarding the master data hierarchies.
Scenario 2	• Less administration as a result of fewer BI systems. • Lower configuration requirements, since the part of the data model contained in the tool BI system only has to be set up once and is then available in all the connected clients. • Characteristic master data can be client specific or cross client.	• Clients are not completely independent because they contain the same metadata. • User administration applies to one Tool BI system. Detailed data protection settings must be made by assigning authorizations. • A consistent approach must be adopted regarding the master data hierarchies.
Scenario 3	• Clients in the core Bank Analyzer system are completely independent. • Separate user administration for each Tool BI system.	• High administration requirements for several Tool BI systems.

2.1.1 Tool BI

Definition

Tool BI is an SAP NetWeaver Business Intelligence (BI) system that provides Bank Analyzer with service functions.

Use

In the Tool BI system, you can use the BI functions that are required to map the operative processes (index function [page 266] for primary objects, characteristic repository [page 20], and key figure repository [page 72]). You can use the Tool BI system as a local BI system.

The following table shows how objects are created in the Tool BI system:

Object Types	Generated/Created Objects	Objects in Tool BI
Custom characteristics [page 62]	InfoObjects	Created manually
	Master data	Created manually
	Master data hierarchies	Created manually
Local [page 59] and fixed characteristics [page 25]	InfoObjects	Created automatically
SAP key figures [page 76] and custom key figures [page 96]	InfoObjects	Created automatically
Segment hierarchies	InfoObjects	Created automatically
Technical fields	InfoObjects	Created automatically
Index tables [page 344]	DataStore objects	Created manually
Relevancy tables	DataStore objects	Created automatically
InfoSets [page 1616]	InfoSets	Created manually
BI DataSource	DataSources	Created automatically

Note the following points with regard to configuration in the Tool BI system.

Background Users in the Standard BI Client

Calls are necessary in the standard BI client for all clients of the Tool BI system (with the exception of the standard BI client itself). These calls can be made via RFC or by direct access.

➔ Recommendation

We recommend that you use direct access. You define this in Customizing for *Bank Analyzer* under *Basic Settings* > *Global Settings* > *Define System Landscape for Bank Analyzer* by leaving the *Tool BI System* field blank.

RFC Connection for the Standard BI Client

You set an RFC connection in the standard BI client from the *SAP Easy Access* screen by choosing *Tools* > *Administration* > *Administration* > *Network* > *RFC Destinations*. To create the background user, on the *SAP Easy Access* screen choose *Tools* > *Administration* > *User Maintenance* > *Users*. Enter this user on the *Users by Logon Data* tab page in the *Logon* section. To enable the Tool BI system to be accessed in the different logon languages of Bank Analyzer, and to allow a language comparison, do **not** enter a language here. Use the logical system identifier of the standard BI client as the name of the RFC destination. In Customizing for *SAP NetWeaver*, choose *Application Server* > *IDoc Interface/ Application Link Enabling (ALE)* > *Basic Settings* > *Logical Systems* > *Assign Logical System to Client*.

Logical System for the Standard BI Client

You define the logical system for the Tool BI system in Customizing for *SAP NetWeaver* under *Application Server* > *IDoc Interface / Application Link Enabling (ALE)* > *Basic Settings* > *Logical Systems* > *Define Logical System*.

Generating Entries in Control Table RSADMINA

To generate entries in control table RSADMINA, on the *SAP Easy Access* screen choose *Tools* > *ABAP Workbench* > *Overview* > *Data Browser* and edit the following fields in table *RSDAMINA* for the entry *CUSTOMIZID:BW*:

BWUSER: Background users in the standard BI client

BWMANDT: Standard BI client

BWMANDTRFC: Logical system name of the standard BI client

Since the entries in the *RSADMINA* table are cross-client, it does not matter which client you use to make the entry. As a result, you cannot use different background users or different logical system identifiers. The remaining table fields are not relevant for the system landscape of Bank Analyzer [page 13].

Consistent Setting of the Source System ID

In the Tool BI system, you have to create the source system IDs from the system landscape. Assign the same ID to each client in the core Bank Analyzer system as in the core Bank Analyzer system landscape. To do so, on the *SAP Easy Access* screen in the standard client choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *SAP NetWeaver Business Intelligence* > *Data Warehousing Workbench* and then *Assignment of Source System to Target System ID (Table RSSOURSYSTEM)*. If you want to load data (that is valid only for one client in the core Bank Analyzer system) from an external source into the Tool BI system, you also have to create source systems with appropriate source system IDs. The content of the *Source System ID* field in the Tool BI system determines for which core Bank Analyzer client the individual data records are valid. In the Tool BI system, you can assign the same source system ID to several source systems. This means that data that is to be valid for the same client in the core Bank Analyzer system can be loaded from different sources.

Example

You load master data from source system A into the custom characteristic *Organizational Unit* and from source system B in the custom characteristic *Product*. The master data of both characteristics is client dependent. Although the data comes from two different systems, you provide the source system ID of the characteristics with the same values.

Source System ID in the Metadata of the Tool BI System

All metadata in the Tool BI system to which client-dependent master data or flow data belongs must be linked by the source system ID. If you create metadata from within the core Bank Analyzer system, the system automatically considers the characteristic for the source system ID. If the metadata does not contain the source system ID, the application data can be used in all core Bank Analyzer system clients.

2.1.2 Reporting BI

Definition

The Reporting BI system is an SAP NetWeaver Business Intelligence system that you use to map the processes and functions required for analytical reporting (OLAP Reporting).

Use

The Reporting BI system is used by both Bank Analyzer and other applications. Unless other restrictions, such as performance considerations, exist, you can use the Reporting BI system company-wide as a central Business Intelligence system.

The following table provides an overview of how Bank Analyzer objects arrive in the Reporting BI system.

Bank Analyzer Objects in Reporting BI

	Created/Generated Object	Reporting BIs
Custom characteristics [page 62]	InfoObjects	Transported from the Tool BI system
	Master data	Transported from the Tool BI system
	Master data hierarchies	Transported from the Tool BI system
Local [page 59] and fixed characteristics [page 25]	InfoObjects	Transported from the Tool BI system
	Master data	Automatically replicated from the Bank Analyzer core system
SAP key figures [page 76] and custom key figures [page 96]	InfoObjects	Transported from the Tool BI system
Segment hierarchies	InfoObjects	Transported from the Tool BI system
	Master data	Automatically replicated from the Bank Analyzer core system
	Hierarchies	Automatically replicated from the Bank Analyzer core system

When you set up the Reporting BI system, note the following:

Local Characteristics

You generate InfoObjects for local characteristics in the Tool BI system [page 16]. You must then transport the InfoObjects into the Reporting BI system. In contrast to this, you can replicate master data for local characteristics directly in the Reporting BI system.

Extraction of Data to the Reporting BI System

- **Consistent Setting of the Source System ID**
  As in the Tool BI system, the source system IDs you create in the Reporting BI system must be consistent with the settings in the system landscape. Assign the same ID to each client in the core Bank Analyzer system as in the core Bank Analyzer system landscape. To do so, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *SAP NetWeaver Business Intelligence* > *Data Warehousing Workbench* and then choose *Tools* > *Assignment of Source System to Source System ID* (table RSSOURSYSTEM).
- **Inclusion of the Source System ID in Metadata for the Reporting BI System**
  Note that all metadata in the Reporting BI system belongs to the client-dependent master data or flow data that are linked to the source system ID.
- **Authorizations**
  If you want users to see only certain data in the Reporting BI system, assign the relevant authorizations for the values of the source system ID. To enable authorization checks to be carried out when queries are being executed, you have to set the *Authorization Relevant* checkbox for the source system ID (0SOURSYSTEM) in the Reporting BI system. Authorization objects can then be automatically generated for reporting with this characteristic. This permits you to run search requests for selected value ranges of this characteristic (for example, within the data of a core client). In Customizing for *Bank Analyzer* choose *Basic Settings* >

Settings for Metadata > *Characteristics* > *Edit InfoObjects for Custom Characteristics* . Enter the source system ID 0SOURSYSTEM in the InfoObject field and then choose *Maintain* and *Business Explorer*.

- **Master Data Replication**
  The Reporting BI system is not stored in the system landscape of the core Bank Analyzer system. You replicate the master data for the Reporting BI system using a report for master data comparison or using the master data monitor. For master data replication, you have to manually enter the corresponding logical system for the Reporting BI system.
  To replicate the master data of fixed and local characteristics from the master data monitor, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Master Data and Hierarchies* > *Call Master Data Monitor* .
  To start mass processing, on the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Master Data and Hierarchies* > *Replicate Master Data* . The report replicates the master data of all local and fixed characteristics from the core Bank Analyzer system to the Reporting BI system. Use this report regularly, and start it as a background job, to ensure that the master data in BI remains consistent.

See also Call Master Data Monitor [page 338] and Replicate Master Data [page 339].

2.2 Characteristic Repository

Definition

A repository used for the central management of all the characteristics in Bank Analyzer. From a technical perspective, it manages the metadata and master data of the characteristics in the core Bank Analyzer system.

Use

The characteristic repository is used as follows:

- It enables standardized semantics to be used in Bank Analyzer, which ensures that several components can use the same characteristic.
- It can be extended to include local and custom characteristics, which ensures that customer-defined characteristics can be used.
- It provides central services for all the characteristics and their master data, such as services for determining metadata, calling up input help, checking master data.

2.2.1 General Information About Characteristics

Definition

Characteristics are a means of classifying data. They contain metadata and can also have master data. You can create master data hierarchies [page 337] for master data [page 337].

Use

Characteristics are used in Bank Analyzer, for example:

- For assigning attributes to primary objects
- For communication between individual layers
- For controlling purposes within the layers
- For storing results
- For reporting

Structure

In Bank Analyzer, a distinction is made between:

- Custom Characteristics [page 62]
- Local Characteristics [page 59]
- Fixed Characteristics [page 25]

Comparison of Characteristic Types

	Fixed Characteristics	Local Characteristics	Custom Characteristics
Source	Fixed characteristics are provided by SAP.	Local characteristics are created by the customer with reference to existing data elements.	Custom characteristics are created freely by the customer.
Creation of characteristics in the Tool BI [page 16] system	InfoObject catalog and InfoObjects are generated in the Tool BI system automatically when initial configuration is run in the core Bank Analyzer system.	InfoObjects can be generated in the Tool BI system once they have been created in the core Bank Analyzer system.	You first create an InfoObject catalog. You then create the InfoObjects and assign them to the InfoObject catalog.
Prefix for the characteristics in the core Bank Analyzer system	`/BA1/` (in a few exceptional cases, no prefix)	`/1BA/, /1BA/_BA1C_`	`/BIC/, /1FB/, /B20C/S_`, No prefix

	Fixed Characteristics	Local Characteristics	Custom Characteristics
Creation of characteristics in the core Bank Analyzer system	Initial configuration is run in Customizing to register the fixed characteristics.	You create and register local characteristics in Customizing for the core Bank Analyzer system. You can then generate InfoObjects in the Tool BI system.	Once it has been created in the Tool BI system, the InfoObject catalog for the custom characteristics has to be registered in the core Bank Analyzer system. Then the characteristics themselves have to be registered.
Prefix of the characteristics in BI	`/BA1/`	Possibly customer-defined prefix	Possibly customer-defined prefix
Metadata and master data	Fixed characteristics based on data elements: Metadata and master data is read in the core Bank Analyzer system from the data elements. When the initial configuration is run, the metadata is transferred automatically to the Tool BI system. You use the master data monitor to transfer the master data to the Tool BI system (see Calling the Master Data Monitor [page 338]). Fixed characteristics based on InfoObjects: The metadata must first be activated in BI. When the initial configuration is run, InfoObjects are registered as characteristics.	Metadata and master data is read in the core Bank Analyzer system from the data elements. Metadata can be transferred automatically to the Tool BI system. You use the master data monitor to transfer the master data to the Tool BI system.	You create metadata and master data manually in the Tool BI system.
Client dependency	Cross-client but must be registered in each client by running initial configuration.	Client-specific	InfoObjects in the Tool BI system are not client-specific. Characteristics in the core Bank Analyzer system are client-specific.

There is no distinction between custom characteristics, local characteristics, and fixed characteristics as regards the ways in which they can be used.

Recommendation

When deciding whether to create a characteristic, we recommend you proceed as follows:

1. Check whether a suitable fixed characteristic is provided by SAP. If so, use this characteristic.
2. Check whether a suitable data element exists in the core Bank Analyzer system. If so, register this data element as a local characteristic.

3. If you cannot use a fixed characteristic, and if there is no suitable data element that you can use, create a new InfoObject in the Tool BI system, and register this InfoObject as a custom characteristic in the core Bank Analyzer system.

Data Types and Field Lengths of Characteristics

Possible data types for characteristics in Bank Analyzer are: *CHAR* (character string), *NUMC* (numeric string), *DATS* (date), and *TIMS* (time). Characteristics of types *CHAR* and *NUMC* can be up to a maximum of 60 characters long. However, the following technical restrictions apply:

- Characteristics with master data (client-specific and non-client-specific) can only be up to 45 characters long. The functions that require master data (for example, input help, value checks, text display) can be used in conjunction with these characteristics. These functions cannot be used for characteristics with longer field lengths. These characteristics have to be defined without master data.
- In selection functions (for example, selection screens for reports) and in authorization checks, only characteristic values that do not exceed 45 characters are processed.

Note that the field lengths specified above refer to the total length of the entire characteristic key. In the case of compound characteristics, this includes the total of the field lengths of each of the characteristics involved.

2.2.2 Roles of Characteristics

Definition

You assign roles to characteristics to allow the characteristics in the characteristic repositoryto be identified as keys of primary objects.

Example

You use roles of characteristics for the following, for example:

- The input help of the primary object to be provided for the characteristic
- The value check to be carried out with the master data for the primary object
- Navigation to the primary object from the general initial screen for corrections

Structure

The following is a list of the fixed characteristics [page 25] that are already assigned roles when they are shipped. You should use these characteristics as keys in those functions where primary objects are fixed objects.

Fixed Characteristics with Assigned Roles

Characteristic	Text	Role
/BA1/C10UOID	Analysis Object	KEY 10

Characteristic	Text	Role
/BA1/C15SCTYP	Spread Curve Type	KEY 15_1
/BA1/C20BPART	Business Partner	KEY 20
/BA1/C21DFVAL	BP/Diff.Cat.Value	KEY 21
/BA1/C30POS	Position	KEY 30
/BA1/C35BTRAN	Business Transaction	KEY 35_1
/BA1/C35BTSRC	Source System	KEY 35_2
/BA1/C40FTRAN	Fin. Transaction	KEY 40
/BA1/C41FINST	Financial Instrument	KEY 41
/BA1/C42SACC	Settlement Account	KEY 42
/BA1/C43CLACC	ClassPos.SecuritAcct	KEY 43
/BA1/C44GLACC	G/L Account	KEY 44
/BA1/C45CUACC	Customer Positn Act	KEY 45
/BA1/C46CNTRY	Country	KEY 46
/BA1/C47HEDGE	Hedge Management	KEY H0
/BA1/C47HDGFDB	Hedging Relationship	KEY 47
/BA1/C48PASSET	Physical Asset	KEY 48
/BA1/C49POSMD	Position Master Data	KEY 49
/BA1/C62UASSID	Analysis Object	KEY 10
/BA1/C62UCGARI	Analysis Object	KEY 10
/BA1/C62UCOLLI	Analysis Object	KEY 10
/BA1/C62UEXPOS	Analysis Object	KEY 10
/BA1/C62UGUARI	Analysis Object	KEY 10
/BA1/C62UPIDCG	Analysis Object	KEY 10
/BA1/C62UPIDCL	Analysis Object	KEY 10
/BA1/C62UPIDGA	Analysis Object	KEY 10

Bear the following aspects in mind when you customize the roles of characteristics:

- If you require different characteristics with the same role (for example, different characteristics for business partner), you can assign roles to local [page 59] and custom characteristics [page 62] as well.In Customizing for Bank Analyzer, choose *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Assign Roles.* The role is also used to identify this characteristic as the key of a primary object.
- You cannot assign roles to fixed characteristics [page 25] because SAP has already assigned roles to these characteristics.
- In some services, you have to additionally register characteristics that are used as keys to objects, regardless of the role that is assigned to them.The characteristic can be one of the fixed characteristics (see above table).To register characteristics, go to Customizing for *Bank Analyzer* and choose *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Index Function* > *Assign InfoObjects to Components.*

2.2.3 Fixed Characteristics

Definition

A characteristic whose master data and properties are defined by a data element in the core Bank Analyzer system or by an InfoObject in the Tool BI system.

Example

An example of a fixed characteristic is the financial transaction ID.

Use

Fixed characteristics are used in the same way as all other characteristics in Bank Analyzer (see General Information About Characteristics [page 21]). You can register fixed characteristics for specific clients in the core Bank Analyzer system when you run initial configuration in Customizing for *Bank Analyzer* by choosing *Basic Settings* > *Global Settings* > *Run Initial Configuration*.

Note

There are also fixed characteristics that are registered for technical reasons only, as well as fixed characteristics that are simply used as templates for InfoObjects. These fixed characteristics cannot be selected in all application functions. Each application specifies how the technical fixed characteristics are used.

There are differences between the following fixed characteristics:

- Fixed characteristics based on data elements:
  Fixed characteristics and the related data elements are predefined by SAP and shipped in the form of a system table. The system table already contains the fields for the names of the characteristic, data element, and InfoObject. Alternatively, you can define fixed characteristics in Customizing for the Analyzers.
  During the initial configuration the appropriate InfoObject catalog (/BA1/CFIXSAP) and InfoObjects are created in the background in the Tool BI system, provided that an InfoObject has been assigned to the fixed

characteristic. The InfoObjects include the predefined prefix namespace /BA1/ (in a few exceptional cases, the prefix is not included).

> **Note**
>
> You can execute the initial configuration run as often as you require –if, for example, you have just upgraded to a new release with new fixed characteristics.
>
> You can display fixed characteristics in the characteristic monitor [page 65] and in Customizing for *Bank Analyzer* by choosing *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Register Characteristics*.

- Fixed characteristics based on InfoObjects:
  SAP predefines this type of fixed characteristic. The system table contains the InfoObject catalog and the names of the InfoObject.

  > **Note**
  >
  > You have to activate the InfoObjects in the BI system.
  >
  > The system does not display fixed characteristics based on InfoObjects in the characteristic monitor.

Integration

From a technical perspective, fixed characteristics based on data elements are similar to local characteristics [page 59] in the core Bank Analyzer system, since the information about metadata and master data is read from the predefined data elements in the system. Metadata and master data of custom characteristics [page 62] is, however, read from the InfoObjects in the Tool BI system and resembles the fixed characteristics based on InfoObjects. However, these differences do not affect how the characteristics can be used.

2.2.3.1 Secondary InfoObjects for Fixed Characteristics

Definition

See Secondary InfoObject.

Use

You use secondary InfoObjects to replicate the master data [page 337] of fixed and local characteristics to the relevant InfoObjects in the Reporting BI system [page 18].

Before you replicate the master data of fixed characteristics to the Reporting BI system, you have to make the relevant InfoObjects known to the characteristic repository. In contrast to the (primary) InfoObjects for fixed

characteristics, which are generated during the initial configuration run, the following options are available for secondary InfoObjects:

- If the name of a secondary InfoObject is known when the Bank Analyzer software is shipped to you, it will already have been included in the table for fixed characteristics. In this case, you do not need to enter any additional information.
- If the name of the secondary InfoObject is **not** known when the Bank Analyzer software is shipped to you, you can assign secondary InfoObjects to fixed characteristics later on in Customizing for *Bank Analyzer* by choosing *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Assign Secondary InfoObjects to Fixed Characteristics*.

> **Note**
>
> If you have specified a name for the secondary InfoObject in Customizing, and the name is supplied to the table of fixed characteristics at a later date, then the secondary InfoObject name in the table is used (in other words, master data is written to the InfoObject from the table of fixed characteristics).
>
> If SAP does not ship the name of the secondary InfoObject, and you do not specify one in Customizing, the master data will be written to the primary InfoObject.

2.2.3.2 Fixed Technical Characteristics

SAP delivers the following characteristics (invisible) that are technically fixed:

Technical Name	Text	Short Description
/BA1/C11BTSELD	Date	Selection date for business transaction
/BA1/C12BREF	Reference for Calculation Basis	Determines the reference calculation basis for the flow
/BA1/C12BUSDAY	BusRecDate	Cash flow export: Business validity date
/BA1/C12CCURR	Payment Currency of Condition	Cash flow export: The currency of the condition (used for mapping multicurrency bonds)
/BA1/C12CDATE	Calculation Date	Specifies the start date for the calculation
/BA1/C12CEXPO	Exponential Interest Calc.	Cash flow export: Exponential interest calculation indicator
/BA1/C12CFCAT	Cash Flow Category	Cash flow export: Category of cash flow
/BA1/C12CFGTYP	Cash Flow Generation Type	Cash flow export: Cash flow generation type

/BA1/C12CFKEYD	Generation Key Date	Cash flow export: Key date for cash flow generation
/BA1/C12CFSIDE	Flow Direction	Cash flow export: Flow direction
/BA1/C12CFSIGN	Reference Int. Rate Operator	Cash flow export: Reference interest rate operator
/BA1/C12CFSYSD	System Date of Cash Flow Generation	Cash flow export: System date of cash flow generation
/BA1/C12CFSYST	System Time of Cash Flow Generation	Cash flow export: System time of cash flow generation
/BA1/C12CFTYPE	Flow Type	Cash flow export: Flow type
/BA1/C12CSEQ	Processing Sequence for Conditions	The processing sequence is mainly determined by the internal financial mathematics settings and these cannot be changed. However, if the internal settings are the same, you can use this field to decide their processing sequence.
/BA1/C12DCALF	Calculation From	Cash flow export: Start date of the calculation period
/BA1/C12DCALT	Calculation To	Cash flow export: End date of the calculation period
/BA1/C12DFIX	Interest Fixing Date	Cash flow export: Interest fixing date
/BA1/C12DMATUR	Maturity	Cash flow export: Due date of flow
/BA1/C12DVAL	Value Date	Cash flow export: Payment day (value date) of flow
/BA1/C12DVALF	Cash Flow Valid From	Cash flow export: Validity start for cash flow
/BA1/C12DVALT	Cash Flow Valid To	Cash flow export: Validity end for cash flow
/BA1/C12EXTID	External Number	Cash flow export: External identifier of the master data object
/BA1/C12FREF	Reference f. Complex Formula	Cash flow export: Reference for complex formulas
/BA1/C12ICALF	*Calculate From* Inclusive Indicator	Cash flow export: *Calculate From* inclusive indicator

/BA1/C12ICALS	Inclusive Indicator for Settlement Date (Cash Flow)	
/BA1/C12ICALT	Inclusive Indicator *Calculation to*	Cash flow export: *Calculate To* inclusive indicator
/BA1/C12IMATUR	*Due Date* Inclusive Indicator	Cash flow export: *Due Date* inclusive indicator
/BA1/C12INCAL	Calendar	Cash flow export: Factory calendar for interest calculation
/BA1/C12INMETH	ICM	Cash flow export: Day count convention
/BA1/C12INTREF	Reference Interest Rate	Cash flow export: Reference interest rate for flow
/BA1/C12INTTYP	Interest category	Cash flow export: Indicator of the type of interest: variable, variable fixed, fixed
/BA1/C12MDCODE	Market Data Area	Cash flow export: Market data area of generated cash flow
/BA1/C12NODENO	Node Number	Cash flow export: Node number (subobject)
/BA1/C12ORIGIN	Cash Flow Origin	Cash flow export: Cash flow origin
/BA1/C12PERI	Calculation Period	
/BA1/C12PEXPO	Int. Sttlmnt Freq	Cash flow export: Interest settlement frequency for exponential interest calculation. Interest settlement frequency (in months) is used for exponential interest calculation as follows: Amount = Calculation base amount x (q ** (days / base days * IS) - 1) For yearly interest settlement: IS = 12 / interest settlement frequency in months q = 1 + percentage rate / 100 * IS (compounding factor)** = power operator. For the daily interest settlement, IS = 360 or IS = 365.
/BA1/C12PRIM	Parent Templ. (CFGT)	Cash flow export: Parent template for cash flow generation type
/BA1/C12PRVARI	Template Variant for Cash Flow Generation	Cash flow export: Template variant for cash flow generation

/BA1/C12PT	Template Category	Cash flow export: Template category
/BA1/C12SEC	Template (CFGT)	Cash flow export: Template for cash flow generation type
/BA1/C12SYSDAY	Date of System Time	Cash flow export: Date of system time
/BA1/C12SYTIME	System Time	Cash flow export: System time
/BA1/C12TUNIN	Time Unit for Interest Rate	Determines the period which the interest rate refers to, for example a year or a month
/BA1/C12UCAL	Month-End Indicator for Value Date	
/BA1/C12UPERI	Calculation Period (Time Unit)	
/BA1/C12UCALF	"Due Date" Month-End Indicator *Calc. From*	Cash flow export: *Calculate To* month-end indicator
/BA1/C12UCALS	Inclusive Indicator for Settlement Date (Cash Flow)	Month-end indicator for settlement date in cash flow
/BA1/C12UCALT	"Due Date" Month-End Indicator *Calculation to*	Cash flow export: Month-end indicator *Calculate To*
/BA1/C12UMATUR	"Due Date" Month-End Indicator *Maturity*	Cash flow export: *Due Date* month-end indicator
/BA1/C12VIRTU	Indicator: Fictitious Flow	Cash flow export: Indicator: Fictitious flow
/BA1/C12XFORM	Formula	Cash flow export: Complex formula triggered
/BA1/C12XFORM2	Formula 2	Cash flow export: Complex formula triggered supplement
/BA1/C30ACNT	Account Number	General external account number to be used in cases where the field can relate to accounts with various account types.
/BA1/C30ACTCAT	Account Category	You use *account categories* to classify accounts. All attributes that may potentially be needed to describe an account are drawn together under an account category. Account categories are part of the template categories that classify accounts.

/BA1/C30KFGRMN	Key Fig. Gr. Minus	Designates a group of position key figures, each of which is reduced by a business transaction key figure.
/BA1/C30KFGRPL	Key Fig. Gr. Plus	Designates a group of position key figures, each of which is increased by a business transaction key figure.
/BA1/C30POSCL	Position Class	*Position classes* are used to categorize and to map, edit, and define settings for positions. A position class is a means of grouping together optional defining characteristics, descriptive characteristics, and key figures. Positions in a position class have the same characteristics and key figures.
/BA1/C30POSKFMN	Key Fig. ID Minus	Name of a position key figure, which is reduced by a business transaction key figure.
/BA1/C30POSKFPL	Key Fig. ID Plus	Name of a position key figure, which is increased by a business transaction key figure.
/BA1/C35BTCL	Business Transaction Class	Is used to categorize business transactions. Defining a business transaction class also involves assigning the implementation-dependent characteristics for the header data and assigning the item classes and their cardinality.
/BA1/C35BTKFID	Key Figure ID	Name of a key figure in a business transaction
/BA1/C35ITEMCL	Item Class	An *item class* in business transactions is a means of grouping together items and is defined by assigning implementation-dependent characteristics and key figures. Item classes form the basis for customizing business transaction classes.
/BA1/C55ACAG8F	Control Indicator for Redetermining Effective Interest	
/BA1/C55ACCALC	AC Calculation Date	Date of the last calculation of the amortized cost
/BA1/C55ACNEWF	Control Indicator for Redetermining Effective Interest	

/BA1/C55ADJPTI	Event	Logical time when a correction is processed in balance processing.
/BA1/C55ADJSRC	Source	Corrections made in balance processing: Internal, technical use within the package.
/BA1/C55AOID	Accounting Object GUID	Globally unique key (GUID) of an accounting object. The GUID represents a combination of defining characteristic values.
/BA1/C55ARESET	Automatic Reset	Automatic reset has already been created
/BA1/C55BSOCAT	Balance Subobject Category	Category of a balance subobject type, for example: The main part or the currency leg
/BA1/C55BTACT	Bus. Transaction Cat.	The *business transaction category* structures the business transactions that are to be posted. The business transaction attributes are predefined by SAP and you cannot change them.
/BA1/C55BTCSK	Sort Key	Used when the system updates the secondary business transactions. Together with other fields it controls the sort sequence of the business transactions to be posted.
/BA1/C55BT_ID	Business Transaction Key	The *business transaction ID* field is filled by Balance Analyzer when business transactions are posted. It is used to compound external business transactions and internal business transactions derived from these by assigning the relevant posting documents the same value in this field.
/BA1/C55CHGAL	FPO Change Allowed	*Change to Descriptive Characters Allowed*: This indicator controls whether the descriptive characteristics of the financial position object are allowed to be changed using a business transaction.

/BA1/C55CPROC	Calculation Proced.	Determines the calculation steps that are followed to generate an internal business transaction. The calculation procedures are determined in Customizing and are derived from a calculation procedure template.
/BA1/C55CP_NO	Component Number	Sequence number that differentiates component position objects for the same primary position object. Primary position objects and simple position objects have the number 0.
/BA1/C55CRDAT	Gener. Date	Value date of the business transaction that generated the position object. From a business perspective, this date is useful for single position objects only.
/BA1/C55CTIME	Event	Time stamp used as a basis for calculation in Accounting.
/BA1/C55CURCLN	Single Currency Transaction	
/BA1/C55DBCDF	Debit/Credit	
/BA1/C55DESCAT	Designation (System Time)	This specifies whether the hedge was designated simultaneously or retrospectively.
/BA1/C55DOCDAT	Payment Date	Specifies the date recorded on the (paper) document relating to the business transaction.
/BA1/C55DOCNO	Document Number	The *ACC Doc. No* field is assigned in internal accounting and is unique for every posting document.
/BA1/C55DOCTXT	Document Header Text	Contains a short description of the document
/BA1/C55FINEWF	Control Indicator for Redetermining FI Spread	
/BA1/C55FVCALC	Fair Val. Calc. Date	*Fair value calculation date*
/BA1/C55HACALC	Hedge Adj. Calc. Date	*Hedge adjustment calculation date*
/BA1/C55ISREV	Reverse Flag	The indicator is set for all reversal business transactions.

/BA1/C55ITEMNO	Number of Item	Within accounting, this number is assigned for each item in the posting document
/BA1/C55ITMPT	Item Template	The *item template* controls the posting of accounting documents. In Customizing, each item type must be assigned to an item template. When the document is posted, the system determines the item template from the position item contained in the business transaction. SAP supplies basic item templates. You can, however, add your own item templates. The customer namespace must be populated with item templates that start with the characters 9, X, Y, or Z.
/BA1/C55ITMTPI	Hdg Inflow Item Type	This specifies the inflow item type of the hedge posted to the target position.
/BA1/C55ITMTPO	Hedge Outflow Item	This specifies the outflow item type of the hedge posted to the source position.
/BA1/C55JRNL	Journal	Accounting in Balance Analyzer uses several specialized journals (in which each item in the document is stored) for classifying accounting data from a business perspective and for boosting technical performance. The specific journals are financial position journal, cost and revenue journal, and the general journal. The different display transactions allow you to display the complete document or only the display items that are assigned to a journal.
/BA1/C55KFCLA	Key Figure Class	You can use the *key figure classes* that are defined by SAP to define key figures in the key figure repository of Bank Analyzer. Each key figure must be assigned to a key figure class. In addition to the SAP key figure classes, you can define additional key figures. In accounting, the logic for posting business transactions is defined at key figure class level.
/BA1/C55LINNO	Document Row Number	The *row number* field is assigned by Accounting and splits the rows in a posting document technically into several parts.

/BA1/C55MDS_FI	FI Spread Mkt Data Set	*Market Data Set for the Financial Instrument Spread*: for the secured portion of the financial transaction.
/BA1/C55MDS_FU	Uncovered Market Data Set for FI Spread	*Market Data Set for the Financial Instrument Spread*: for the secured portion of the financial transaction.
/BA1/C55MDS_HS	Market Data Set for Hedge Spread	
/BA1/C55MPOST	Man. Posting Allowed	
/BA1/C55PDFIX	Fixed Posting Date	If this indicator is set for a rule for determining the posting date, the posting date that is determined from the rule is not changed. If the posting date falls in a closed period, the system issues an error message during the posting process. If the indicator is not set, the posting process for the document determines whether the posting date that was determined using the rule falls in an open period. If so, the determined posting date is retained. If the posting date falls in a closed period, the system determines a new posting date, which is the first day of the next open period.
/BA1/C55PDPRP	Default Posting Date	Date proposed by the system as a *default posting date* in the accounting document. Used for updating the secondary business transactions.
/BA1/C55PERIOD	Period	Specifies the posting period during which the posting document was classified from a business perspective.
/BA1/C55POSTD	Posting Date	The *posting date* field is populated by means of the SDL and presents a proposal for Balance Analyzer on the basis of which the fiscal year and the posting period are determined. If this posting period is already closed, the Customizing settings determine whether the business transaction is posted to the next open period or whether an error message is displayed.
/BA1/C55PRMPN	Primary BT	*Primary Business Transaction*

/BA1/C55PR_SQ	Primary Sequence	This field comprises the value date plus an additional sorting number, which is predefined by SAP.
/BA1/C55PUNIT	No. Processing Unit	Processing units determine the sequence of business transactions that are dependent on one another with regard to their chronology and their results. Processing units include business transactions that cannot be processed in parallel or that do not overlap. A from/to range is selected for each position object in order to transfer the business transactions belonging to a processing unit.
/BA1/C55REFPN	Ref. to Source Doc. of Reset Doc.	Reference in the original business transaction to the source document of the reset business transaction. If the transaction is automatically reset, the original business transaction has a reference to the reset business transaction.
/BA1/C55REGTM	Registration Time/Date	This characteristic contains a technical time stamp for every document row, which positions the row together with its posting period in a two-dimensional versioning schema on the basis of technical and business aspects. It is later than the SDL time stamp of the related financial reporting preprocess but earlier than the accounting time stamp of the financial reporting process.
/BA1/C55RESVAL	Indicator for Determination of Resid. Valuation	
/BA1/C55REVDAT	Reversal Date	The *reversal date* field contains the date on which the reversed posting document was reversed.
/BA1/C55REVDOC	Reversed Document	
/BA1/C55REVIND	Reversal Document Flag	The *reversal document* indicator shows whether a document has been reversed or not.
/BA1/C55RVSKEY	Sender Key Reversal	External key of the reversal business transaction

/BA1/C55SECSQ	Secondary Process Sequence	The *secondary processing sequence* is determined when the internal business transactions are updated and is documented in the generated documents. You can make settings for the determination rules in the transaction *rules for secondary processing sequence* (/BA1/B1_US_CUS_SECSQ).
/BA1/C55SNDID	Sender ID	The *sender ID* field contains the name of the source system from which the business transaction originates.
/BA1/C55SNDKEY	Sender Document Key	
/BA1/C55SP_ID	ID	Order number for the part of an accounting document that refers to a particular line item.
/BA1/C55SYSDAT	System Date	
/BA1/C55SYSTIM	System Time	
/BA1/C55UNAME	Created By	The user in whose name a business transaction was created.
/BA1/C55VALDAT	Valuation Date	The *Valuation Date* field saves the date of the last assessment of the financial position object.
/BA1/C55VARNT	Fiscal Year Variant	
/BA1/C55YEAR	Fiscal Year	The *fiscal year* field, together with the *fiscal period* field, specifies the fiscal period to which postings are made.
/BA1/C55YPER	Fiscal Year and Period	
/BA1/C56EFT_RULE	Fair Value Effectiveness Test Rule	
/BA1/C57VALID	Validity of Portfolio Item	
/BA1/C63SRCIDX	Index	
/BA1/C65EXGUID	External ID of Run	GUID of the associated run in the Historical Database
/BA1/C65JCTIME	Timestamp of Run	Time stamp of the associated run in the Historical Database

/BA1/C65RELSTA	Release Status	Specifies the *release status* of the data record
/BA1/C69VALRULE	Valuation Rule	The *valuation rule* is used to process groups of transactions individually within the risk basis.
/BA1/C93TCID	Timestamp ID	
EFT_ENV	EFT Environment	*Fair Value Effectiveness Rule Test Environment*

2.2.3.3 Fixed Standard Characteristics

SAP delivers the following fixed non-technical characteristics (visible):

Technical Name	Text	Data Type	Short Description
/BA1/C10UOID	Unified Object ID	CHAR	Cross-object unique ID of objects in Bank Analyzer
/BA1/C11CEPTPE	Product Type	CHAR	
/BA1/C11CRLCAT	Relationship category	CHAR	You use the *object relationship category* to classify one-way relationships between two master data objects.
/BA1/C11ENDT	End of Term	DATS	
/BA1/C11EXTCON	External Number	CHAR	General external number of objects in the master data framework that are used if the field can refer to objects of different template categories. The external number is a unique ID for a master data object, such as a financial transaction, financial instrument, or account for a specific template category.

/BA1/C11FPROD	Parent template	CHAR	*Template hierarchy –parent template*: You define structured templates using a step-by-step "bottom up" approach. In other words, you move from the elementary templates to the complex, structured templates. Because of this, the maintenance of the structure of a template is limited to the maintenance of the direct child node of the root/parent node and the input of cardinality restrictions.
/BA1/C11HEDCAT	Hedging Relationship Category	CHAR	You use *hedging relationship categories* to classify hedging relationships. Possible hedging relationships categories are *fair value hedge* or *cash flow hedge*.
/BA1/C11KEYDAT	Key Date	DATS	Key date for selection of objects from the SDL. The system takes into account information that is valid on this date from a business perspective (object versions, time-dependent object attributes).
/BA1/C11KEYDAY	Key Date of Calculation Date	NUMC	
/BA1/C11NODENO	Node Number	NUMC	Number of a node in a hierarchy of master data objects that uniquely identifies a subobject in conjunction with the parent and child templates.
/BA1/C11PROD	Template	CHAR	A *template* comprises the entire rule set that is used for *financial transactions* or *accounts*, for example. The template can be tailored to meet each customer's requirements at the level of the individual object.

/BA1/C11PTYPE	Template Category	CHAR	In the master data framework, the *template category* is used for classifying templates. The template category brings together all attributes that could be used to describe templates. From a business perspective, the template category corresponds to an object category in the master data framework (for example, *financial transaction* or *country*).
/BA1/C11REFINT	Reference interest rate	CHAR	This field contains the key of a reference interest rate.
/BA1/C11RFLG	Indicator for Release Relevancy	CHAR	Release relevance of the object
/BA1/C11RUSER	Workflow Recipient in Release	CHAR	Name of the workflow recipient in the release
/BA1/C11SPROD	Child template	CHAR	*Template hierarchy –child template:* You define structured templates using a step-by-step "bottom up" approach. In other words, you move from the elementary templates to the complex, structured templates. Because of this, the maintenance of the structure of a template is limited to the maintenance of the direct child node of the root/parent node and the input of cardinality restrictions.
/BA1/C11STARTT	Start of Term	DATS	
/BA1/C11SYSTIM	System Time/Date	NUMC	*System time* up to which changes to data that was entered in the system prior to this time are made.

/BA1/C15BCURR	Reference currency	CUKY	The currency key used for translating all rates in the rate category. If you have to translate between many different currencies, you can simplify this process by specifying a *reference currency*. You specify the exchange rates of the currencies in relation to this reference currency within the rate category in question. All currencies are then translated using a two-step calculation where the reference currency plays a pivotal role.
/BA1/C15CACTG	Corporate Action Category	CHAR	The category of the corporate action such as dividend, stock split, increase in capital through subscription right, increase in capital through company's resources.
/BA1/C15SCCCY	Currency	CUKY	Spread curve currency
/BA1/C15SCTYP	Spread Curve Type	CHAR	*Spread curve type* comprises spread curves that have the same type but different currencies, and can also be defined as the difference between two yield curve types.
/BA1/C16AREA	Generic Market Data Area	CHAR	Key component of all generic market data. All market data is stored in relation to a market data area. A market data area determines which data a user can display, and which authorizations a user has for accessing all market data.
/BA1/C16FNDDAT	Date	DATS	Validity date for which a data record is found
/BA1/C16FNDTRM	Term	NUMC	Maturity for which a data record is found

/BA1/C16FNDUNT	Time Unit	CHAR	Unit of the term for which a data record was found (days, months, or years)
/BA1/C16TERM	Term	NUMC	*Term* in the unit of the field *Time Unit*
/BA1/C16UNIT	Time Unit	CHAR	The possible time units are days, months, or years.
/BA1/C20BPART	Business Partner	CHAR	External identifier of a business partner
/BA1/C21DFVAL	Object ID of BP Differentiation Type	CHAR	Values of the *Differentiation in Business Partner* field that you use to further differentiate attributes or relationships within a business partner. An example of this is when a customer has different managers in a number of branches.
/BA1/C30POS	Position: GUID - ID	CHAR	Internal unique ID of a position header (GUID).
/BA1/C35BTRAN	Business Transaction ID	CHAR	*External business transaction ID*: The business transaction ID and the source system are used to form the unique external key to a business transaction.
/BA1/C35BTSRC	Source System	CHAR	The *source system* and the business transaction ID are used to form the unique external key to a business transaction.
/BA1/C35CRTST	Generation Time Stamp for a Primary Object in the Source Data Layer	NUMC	

/BA1/C35DATBUS	Business Transaction: Business Record Date	DATS	The business record date of a business transaction and the time the data was entered are used to define from when a business transaction is visible. These values together map the two-dimensional time dependency of the business transaction.
/BA1/C35DATDOC	Business Transaction: Payment Date	DATS	Payment date of a business transaction: The "Payment Date" field is used globally in Customizing (by Balance Analyzer, for example).
/BA1/C35DATXPO	Business Transaction: Posting Date	DATS	Posting date of a business transaction: The "Posting Date" field is used globally in Customizing (by Balance Analyzer, for example).
/BA1/C35DATTRA	Business Transaction: Trade Date	DATS	Trade date of a business transaction: The "Trade Date" field is used globally in Customizing (by Balance Analyzer, for example).
/BA1/C35DATBUS	Business Transaction: Settlement Date	DATS	Settlement date of a business transaction: The "Settlement Date" field is used globally in Customizing (by Balance Analyzer, for example).
/BA1/C35DATSRC	Business Transaction: System Date in Feeder System	DATS	System date in feeder system. This field shows the date on which the business transaction was created in the source system.

/BA1/C35TIMSRC	Business Transaction: System Time in Feeder System	TIMS	System date in feeder system. This field shows the date on which the business transaction was created in the source system.
/BA1/C35TSTCRE	Business Transaction: Time of Creation	NUMC	Time that a business transaction was entered in the Source Data Layer in Bank Analyzer
/BA1/C40FTRAN	Financial Transaction ID	CHAR	External identifier of a financial transaction
/BA1/C41FINST	Financial Instrument - Identifier	CHAR	External identifier of a financial instrument
/BA1/C42SACC	Settlement Account –Identifier	CHAR	External identifier of a settlement account
/BA1/C43CLACC	ID of Securities Position Account	CHAR	External identifier of a securities position account
/BA1/C44GLACC	G/L Account ID	CHAR	External identifier of a G/L account
/BA1/C45CUACC	Customer Position Account ID	CHAR	External identifier of a customer position account
/BA1/C46CNTRY	Country –Identifier	CHAR	External identifier of a country
/BA1/C47HEDGE	Hedging Relationship ID	CHAR	External identifier of a hedging relationship
/BA1/C47HDGFDB	Hedging Relationship ID	CHAR	External identifier of a hedging relationship
/BA1/C47SECRISK	List of Hedged Risks	CHAR	Key that is assigned to a list of *hedged risk categories* (such as interest rate risk and currency risk). You define the list in Customizing and assign keys to it.
/BA1/C47VALFROM	Hedge Date: Valid from Date	DATS	Hedge data: Start date of the validity period
/BA1/C47VALTO	Hedge Data: Valid to Date	DATS	Hedge data: End date of the validity period
/BA1/C48PASSET	Physical Asset ID	CHAR	External identifier for a physical asset

/BA1/C49POSMD	Position Master Data Identifier	CHAR	
/BA1/C50CUSTAT	Status of Customer: Deleted/Merged	CHAR	Migration analysis: Status of a customer
/BA1/C50CUSTNO	Number of Relevant Customers	NUMC	
/BA1/C50HLEVEL	Name of Hierarchy Level	CHAR	Customers on different hierarchical levels (such as consultant, branch, profit center).
/BA1/C50HVALUE	Value of Hierarchy	CHAR	Values of the different hierarchical levels (such as consultant John Miller, branch *101*, profit center *4711*)
/BA1/C50REPROW	Report Row	NUMC	Row in report to display the results
/BA1/C50RESTSP	Save Time for Result Data	NUMC	*Save time for results data* for versioning
/BA1/C50VERSNO	Version Number	NUMC	Version number of basic data
/BA1/C55ACCKF	Posting Key Figure	CHAR	The *posting key figure* characteristic contains a key figure ID for each document line item. This key figure ID is derived from characteristics of the external business transaction in accounting. The Customizing settings control the derivation. When the accounting balances are calculated, the key figure is determined on the basis of this key figure ID and its value from another line item field.
/BA1/C55ACCSY	Accounting System	CHAR	A calculation system that is identified by an accounting principle (for example, German HGB, IAS, US GAAP), a currency, and a periodization(fiscal year variant). A separate accounting document is created for each accounting system.

/BA1/C55ACCALC	AC Calculation Date	DATS	Date of last amortized cost calculation
/BA1/C55ACPCAT	Period Category	NUMC	Type of write-back of reserves (to due date/fixed period)
/BA1/C55ACPLEN	Period	NUMC	Length of period of time of write-back of reserves in months
/BA1/C55AOCL	Accounting Object Class	CHAR	A group of *accounting objects* that can be described using the same defining characteristics. Each accounting object has a unique, fixed *accounting object class* and a (changeable) accounting object type.
/BA1/C55AOID	Accounting object GUID	CHAR	Globally unique key (GUID) of an accounting object. The GUID represents a combination of defining characteristic values.
/BA1/C55AOCURR	Financial position currency	CUKY	Specifies a currency in which key figures of the financial position object and the balance object are managed. The financial position currency relevant for accounting is usually the issue or the contract currency.
/BA1/C55AOTYPE	AO type	CHAR	The *accounting object type* is used for a detailed view of the accounting object class. A number of accounting object types may be assigned to an accounting object class. Each accounting object has a unique, fixed accounting object class and a changeable accounting object type.
/BA1/C55ASSLIA	Asset Liability Indicator	CHAR	

/BA1/C55BPNO	Financial Reporting Process No.	NUMC	Each financial reporting process within a balance sheet receives a sequential number In the application logs for balance processing, the financial reporting process ID is made up of the balance sheet ID and this sequential financial reporting process number
/BA1/C55BSOTYP	Balance Subobject Type	CHAR	Specifies the structure of the *balance subobjects*. Balance subobjects contain the characteristics and key figures that are processed by the balance processing application in Balance Analyzer.
/BA1/C55BTATP	Business Transaction Type	CHAR	The *business transaction type* is part of the header information of the business transaction and describes it as a whole. Enter any necessary values in Customizing for Balance Analyzer. This field is a required field for posting accounting documents from business transactions and must therefore always be filled For primary business transactions, the system checks the combination of business transaction type and item type, and also the completeness of the data. For secondary business transactions, the system checks the completeness of the business transaction data only.
/BA1/C55CHGTIL	Date up to (and including) which the change is valid	DATS	
/BA1/C55COMPCL	Component Class	CHAR	Component class for financial position object

/BA1/C55COMPL	Force Merge	CHAR	Merging is carried out in accordance with the delivery type even if the standard checks fail.
/BA1/C55CP_ID	ID of common position object	CHAR	Unique key for common position objects They can be components of a structured template and collective positions with corresponding single positions. Their FP_ID is identical to their CP_ID.
/BA1/C55CSTEP	Calculation Step	CHAR	A *calculation step* is part of a calculation procedure for generating internal accounting business transactions.
/BA1/C55CONDAT	End Date	DATS	
/BA1/C55DCMET	Day Calculation Method	NUMC	Method for calculating the number of days to the accrual of revaluation reserves
/BA1/C55DLVTY	Delivery Type	CHAR	The *delivery type* determines how the individual key figures and descriptive characteristics are processed during the merge.
/BA1/C55DOCCL	Document class	CHAR	The *document class* is used to group business transactions into external and internal business transactions. The characteristic values of the document class are supplied by SAP and you cannot make changes to them.

/BA1/C55DTCUR	Target currency	CHAR	In an external business transaction, this can contain a currency code. This currency code represents the target currency of the optional exchange rate, which can be used to convert the amount supplied here to the object currency. The exchange rate itself is contained in key figure `55EXRAT`.
/BA1/C55ENDDA	End Date	DATS	
/BA1/C55ENDKD	To Key Date	DATS	
/BA1/C55EVALPR	Alternative Calculation Procedure	CHAR	A defined calculation method for key date valuation of a position object that deviates from the standard method.
/BA1/C55FDDAT	Final Due Date	DATS	Final due date for which the reserves evaluation is entered
/BA1/C55FP_ID	FP_ID Position	CHAR	Globally unique key (GUID) of a financial position object.
/BA1/C55FPCAT	Category of Financial Position	CHAR	Values of the position object category are primary position object, common position object, single position object, and hedge position object.
/BA1/C55FPSTA	Sender/Recipient	CHAR	A position object can be part of a sender-receiver relationship with another position object as a result of a business transaction (for example, transfer posting or redesignation). The role of the position object in this relationship is specified in the business transaction item.

/BA1/C55FSI	Financial Statement Item	CHAR	Represents an aggregation of key figures for reporting purposes. The system summarizes several key figures according to business-oriented criteria. The result is a financial statement item. Characteristics from single transactions (for example, *account number*) are not taken into account.
/BA1/C55FVCALC	FFV Calculation Date	DATS	
/BA1/C55GLCRTP	Currency Type Indicator	CHAR	
/BA1/C55HACALC	Hedge Adj. Calc. Date	DATS	
/BA1/C55HSIMPR	Simplified Procedure	CHAR	This specifies whether the hedge adjustment of the hedged position is to be calculated in accordance with the simplified procedure.
/BA1/C55IGNREC	Ignore Data Record	CHAR	Indicator that specifies whether the current data record is to be ignored.
/BA1/C55IMPAIR	Impairment Indicator	NUMC	The *impairment indicator* field specifies whether a financial position is subject to an impairment (for example, by a management decision). There are two transactions in Balance Analyzer you can use if the financial position is to be impaired by means of a posting in Balance Analyzer. Alternatively, an impairment business transaction can also be supplied via the SDL.

/BA1/C55ITMTP	Item type	CHAR	Describes an item in the business transaction. Using the item type and the item template assigned to it, the system can find journals and key figures and carry out certain checks. You define item types in Customizing.
/BA1/C55KFSEM	Key Figure Semantic	CHAR	Name of a key figure, for example. *book value* or *pro rata accrued interest*. The name indicates the content of a key figure from a business perspective.
/BA1/C55KFTYP	Key Figure Type	CHAR	Posting key figure types denote a group of similar posting key figures. You use posting key figure types to derive the posting key figures to which items from business transactions are to be posted. First item types are assigned from business transaction items. Depending on other characteristics, you can then use the posting key figure type to derive further posting key figures.
/BA1/C55KVTYPE	Key date valuation type	CHAR	The *type of key date valuation* is used to determine possible rating models for a financial position.
/BA1/C55LGENT	Legal Entity	CHAR	Organizational unit for which a balance sheet must be created.
/BA1/C55METHOD	Method Variant	CHAR	Displays the variants of financial reporting methods.

/BA1/C55POSDA	Value Date	DATS	The *Value Date* field is usually copied from the *Trade Date* or the *Settlement Date* field for pending transactions. It plays a key role in determining the sequence in which financial positions are processed when internal business transactions are updated.
/BA1/C55PP_ID	Primary Position Object	CHAR	Unique, artificial key of a *primary position object* that represents the whole of a structured financial instrument or financial transaction.
/BA1/C55QUOTYP	Quote Type	CHAR	
/BA1/C55RECNO	Status Number of No.	NUMC	Number of the current data record
/BA1/C55REFPD	Original Posting Date	DATS	Posting date of the original business transaction
/BA1/C55REPID	Financial statement ID	NUMC	Key for a financial statement, which is defined by the legal entity, accounting system, year, period, and value type
/BA1/C55REPT	Report Type	CHAR	Specifies for which type of reports the assigned characteristic *Financial Statement Item* is to be used. Examples of this include balance sheet, profit and loss statement, and notes to the financial statement.
/BA1/C55RESACQ	Residual Acquisition Value	CHAR	Indicator for determination of residual acquisition value
/BA1/C55RFPNO	Reference to Original Document of Reset Document	CHAR	*Reference in the reset business transaction to the source document of the original business transaction.* If carried out by the customer, the reset business transaction must have a reference to the original business transaction.

/BA1/C55RKFTYP	Type of Reference Key Figure	CHAR	EQ *Key Figure Type Reference*
/BA1/C55RUNID	ID of Financial Report Run	NUMC	
/BA1/C55RUNTYP	Run Type	CHAR	Enables a distinction to be made between balance sheet values from an initial delivery and values from corrections and their related reversals.
/BA1/C55SBTNO	Sub-Business Transaction Number	NUMC	The *sub-business transaction number* is used to structure a business transaction or an accounting document. Sub-business transaction items that have the same sub-business transaction numbers are treated as one unit during accounting.
/BA1/C55SP_NO	Single Position Number	NUMC	Sequence number that differentiates single position objects within the same collective position object.
/BA1/C55STARKD	From Key Date	DATS	Descriptive characteristic of the position object that is assigned the tax rate used to calculate deferred taxes.
/BA1/C55TXRATE	Tax Rate	CHAR	
/BA1/C55USELEG	Post to currency leg	CHAR	Controls whether the component position object of a currency leg is to be identified instead of the primary position object with the same currency when posting external business transactions. For example, for forward exchange transactions.
/BA1/C55VALPR	Valuation Procedure	CHAR	*Valuation procedure* that is derived for each financial position every time you call a key date valuation. It contains the steps that have to be followed to valuate the financial position.

/BA1/C55VALT	Value Type	CHAR	Describes the data generated in *balance processing*. You specify the *value type* when you create a financial reporting process.
/BA1/C55VALTIL	UTC Time Stamp for the Valid to Date of an FPO	NUMC	
/BA1/C55VSIGN	+/- Book value	CHAR	
/BA1/C55DBTSQ	Sequence of Derived Business Transactions	NUMC	Contains the sequence in which derived business transactions are generated for a primary business transaction
/BA1/C55CSTSQ	Sequence of Calculation Steps	NUMC	Contains the sequence in which the calculation steps are carried out for a calculation procedure
/BA1/C55ALSFL	Asset/Liability Status of Flow	CHAR	
/BA1/C55ALSFP	Asset/Liability Status of Financial Position Object	CHAR	
/BA1/C55ALSGRP	Position Group (Asset/Liability)	NUMC	
/BA1/C55SLJRNL	Journal for an Accounting Document	CHAR	
/BA1/C55GFPCAT	Category of a General Financial Position	CHAR	
/BA1/C55IMP_DT	Impairment Indicator From Date	DATS	
/BA1/C55IMP_VD	Impairment Control for Calculation	NUMC	
/BA1/C55REFKF	Reference Key Figure for Foreign Currency Postings	CHAR	Contains the posting key figure to be transferred in the case of a foreign currency transfer posting. This posting key figure is used as a selection criterion during the derivation of posting key figures.
/BA1/C55FVCALC	Date of Last Fair Value Calculation	DATS	

/BA1/C55HACALC	Date of Last Hedge Adjustment Calculation	DATS	
/BA1/C56FEFFLG	Final Effectiveness Indicator	CHAR	*Final effectiveness indicator* of a hedge. It specifies whether a hedge is dissolved.
/BA1/C56IEPDAT	Ineffective as on Date	DATS	Transaction methods/Risk Basis
/BA1/C62ABSCAT	Category of Securitization Exposure	NUMC	
/BA1/C62ABSEAC	Indicator: RWA includes EAC of Securitization Transaction	CHAR	
/BA1/C62ABSNEL	Indicator: Securitization Transaction Is Eligible	CHAR	
/BA1/C62ABSNR	Indicator: Risk-Free Securitization	CHAR	
/BA1/C62ABSTID	Special r	CHAR	
/BA1/C62APPRCH	Calculation Approach	NUMC	
/BA1/C62AREAR	Indicator: Threshold for Days Past Due	CHAR	
/BA1/C62CALCM	Calculation method	NUMC	
/BA1/C62CCFCAT	CCF Category of Counterparty Credit Risk	NUMC	
/BA1/C62CCFCLS	CCF Class	CHAR	
/BA1/C62CECTR	Transaction Category	NUMC	
/BA1/C62COLCAT	Collateral Category	NUMC	
/BA1/C62COPUFA	UOID for Purchase Commitment	CHAR	
/BA1/C62DDFAP	Indicator: Double Default Framework Applied	CHAR	
/BA1/C62DEDRC	Deduction from Capital	CHAR	
/BA1/C62FAVTRT	Indicator: Preferential Treatment of Short-Term Exposures Under Option 2	CHAR	

/BA1/C62GARCT	Category of Guarantee/Credit Derivative	NUMC	
/BA1/C62LGDCLS	LGD Class	CHAR	
/BA1/C62PFOCAT	Portfolio Category	NUMC	
/BA1/C62POOLID	Pool ID	CHAR	
/BA1/C62PRCAT	BII Product Category	NUMC	
/BA1/C62PRDCLS	Product Class	NUMC	
/BA1/C62REDRW	Indicator: Reduced Real Estate Risk Weight r Was Used	CHAR	
/BA1/C62RETGRA	Indicator: Retail Granularity Criterion was Met	CHAR	
/BA1/C62RETLIM	Indicator: Check Whether Retail Exposure Limit Was Exceeded	CHAR	
/BA1/C62RETPFO	Subportfolio Class	NUMC	
/BA1/C62RISKCL	Risk Class	NUMC	
/BA1/C62RPTEXT	Extension to Risk Position Type	CHAR	
/BA1/C62RSKCAT	Risk Category	NUMC	
/BA1/C62SECCAT	Securitization Category	NUMC	
/BA1/C62SGICCF	Segment ID for CCF	CHAR	
/BA1/C62SGILGD	Segment ID for LGD	CHAR	
/BA1/C62SGIPD	Segment ID for PD	CHAR	
/BA1/C62SH150P	Indicator: Short-Term Exposure with Poor Rating Exists	CHAR	
/BA1/C62SH50P	Indicator: Short-Term Exposure with Good Rating Exists	CHAR	
/BA1/C62SPLDCT	Category of Specialized Lending	NUMC	
/BA1/C62SUBAPP	Subapproach	NUMC	
/BA1/C62UASSID	UOID of Asset	CHAR	

/BA1/C62UCGARI	UOID of Counter-Guarantee	CHAR	
/BA1/C62UCOLLI	UOID of Collateral	CHAR	
/BA1/C62UEXPOS	UOID of Exposure	CHAR	
/BA1/C62UGUARI	Portion UOID of Guarantee	CHAR	
/BA1/C62UPIDCG	UOID of Portion of Counter-Guarantee	CHAR	
/BA1/C62UPIDCL	UOID of the Collateral Portion	CHAR	
/BA1/C62UPIDGA	UIOD of Portion of the Guarantee	CHAR	
/BA1/C63AVPDAT	Date of Calculation of Average Purchase Price	DATS	
/BA1/C63COMPEN	SAP Pooling Complete	CHAR	SAP pooling complete
/BA1/C63SPLITR	Relevant for Splitting	CHAR	
/BA1/C65DATRUN	Base Date for Run	DATS	
/BA1/C69CFSPD	Splitting Value for Coverage Split	CHAR	Value of coverage split
/BA1/C69CFSPF	Splitting Value Margin split	CHAR	Value of margin split
/BA1/C69CFSPK	Splitting Value for Consortial Split	CHAR	Value of consortial split
/BA1/C69CFSPV	Splitting Value for Disbursement Split	CHAR	Value of disbursement split
/BA1/C69CFVIEW	Cash Flow View	NUMC	
/BA1/C69COMPEN		CHAR	
/BA1/C69HRLCAT	Item Category for Hedge	CHAR	Transaction category within a hedging relationship. Possible values: HEDIT (underlying), HEDIN (hedge transaction)
/BA1/C69ITMTPE	Item category	CHAR	Possible values: GG (underlying), SG (hedge transaction)

/BA1/C69KEYFIG	Indicator: Fair Value Server	CHAR	Key figure that is to be calculated by the fair value server. Possible values: FFV (full fair value), AC (amortized cost), HFV (hedge fair value), HAC (hedge amortized cost)
/BA1/C69OBJID	Object ID	CHAR	External key of a transaction or instrument
/BA1/C69SSHLNG	Short/Long Indicator	CHAR	Indicator for purchase (*long*) and sale (*short*) for financial instrument positions.
/BA1/C69TMBCKT	Time Bucket	NUMC	
/BA1/C92SCEN	Scenario ID	CHAR	
/BA1/CH0DESDAT	Designation Date	DATS	
/BA1/CH0EPCLSD	Indicator: Valuation Period Completed	CHAR	
/BA1/CH0DESTMS	Designation Time Stamp	NUMC	
/BA1/CH0SYNTMS	Synchronization Time Stamp	NUMC	
/BA1/CH0HEDSTB	Hedging Relationship: Business Status	CHAR	
/BA1/CH0HEDSTT	Hedging Relationship: Status	CHAR	
/BA1/CH0HAGGLV	Hedging Relationship: Aggregation Level	CHAR	
/BA1/CH0FTMIRR	Indicator for Mirror Transaction	CHAR	
/BA1/CH0XLEGRP	Group of Legal Entities	CHAR	
EFT_ENV	Effectiveness Test Environment for Fair Value	CHAR	Groups together all the main Customizing settings for the fair value effectiveness test.

MDCODE	Market Data Area	CHAR	Component of Primary Keys of All Market Data All market data is stored according to the related market data area. A market data area determines which data a user can display, and which authorizations a user has for accessing all market data.
MDSET	Market Data Set		You use the *market data set* to group together all the rate types that you use in an evaluation.

2.2.4 Local Characteristics

Definition

A characteristic whose metadata and master data is determined by a data element in the core Bank Analyzer system (for example, account currency, or object type)

Use

Local characteristics are used in the same way as all other characteristics in Bank Analyzer (see General Information About Characteristics [page 21]). The advantage of creating local characteristics rather than custom characteristics [page 62] is that the master data for local characteristics is filled automatically using the data for the data element and, therefore, does not have to be created manually.

Before you can use local characteristics in the core Bank Analyzer system or in the Tool BI system, you have to work through the following steps:

1. In Customizing for *Bank Analyzer*, choose *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Register Characteristics*. Choose the *Local Char.* tab page and select a data element.
   You can the call up a list of proposals for local characteristics by choosing *Proposal*.
2. Use the input help to select an InfoObject, or enter an InfoObject name of your choice.
3. If you do not want the characteristic and the selected data element to have the same name, you can enter your own text. To do so, set the *Own Text* indicator and enter a short text.
   The characteristic name is created from the InfoObject name.
4. If, in addition, you want to create documentation, choose Documentation. You can use the F1 help to display information about the local characteristic.
5. If you want to translate the short text or the documentation for one or more characteristics, select the characteristic, and then, from the menu choose *GoTo* > *Translation*.

6. If a local characteristic is compounded with another characteristic, you can enter a fixed value (rating agency, for example) for the unit so that the system can display only combinations with this fixed value in the input help.
7. Register the local characteristic and, if necessary, generate the related InfoObject. You can also generate the InfoObjects for local characteristics at a later stage from the characteristic monitor [page 65].

Integration

From a technical perspective, local characteristics are similar to fixed characteristics [page 25], since the information about metadata and master data is taken from the predefined data elements in the system. The metadata and master data of the custom characteristics [page 62], on the other hand, are read from the InfoObjects of the Tool BI system. These differences do not affect how the characteristics can be used.

To transfer the master data for the local characteristics to a reporting BI system [page 18], use the master data monitor [page 338].

2.2.4.1 Changing Local Characteristics

You cannot change the following essential properties of local characteristics that are already registered.

- Data element
- Primary InfoObject
- Secondary InfoObject
- Compounded fixed values

You can determine whether the characteristic has its own text or its own short text at a later date. To do this, you need to adapt the text of the InfoObject using the characteristic monitor.

You can also change the category of the InfoObject at a later date. Bear in mind that, by doing this, you change the behavior during transport and when copying a client. Assigned primary objects are not generated in the target system.

To determine whether local characteristics [page 59] need to be edited, first check the list of the local characteristics in the characteristic monitor [page 65].

To do this, choose Customizing for ▶ *Bank Analyzer* ▶ *Basic Settings* ▶ *Settings for Meta Data* ▶ *Characteristics* ▶ *Characteristic Monitor* ▶.

- To display all local characteristics, choose *Execute*.
- To display only local characteristics or fixed characteristics, set/unset the corresponding checkboxes, and then choose *Execute*.
- If a red traffic light icon is displayed for a characteristic, choose *Compare InfoObject*. Save your entries.

> **Note**
>
> · The red traffic light icon indicates that this registered characteristic does not match the defined characteristic in the Tool BI system [page 16]. As a result, you have to edit the characteristic.

2.2.4.2 Deleting Local Characteristics

Prerequisites

Before deleting local characteristics, use the *where-used list* to check whether the characteristics are already being used in Bank Analyzer. If there are still dependencies to these characteristics, you have to delete them.

> **Note**
>
> You can also use the *where-used list* to search for characteristics that have been deleted, but that are still in use. To do this, enter the required characteristic in the dialog box.

> **Caution**
>
> Delete the local characteristic only if you are certain that you know all of the dependencies and that you can adapt them. If this is not the case, do not delete the local characteristics.

Procedure

To delete a local characteristic, remove the local characteristic from the table of local characteristics and deregister it using *Compare* .

> **Note**
>
> After you have deleted the local characteristics in Bank Analyzer, dependent generated InfoObjects still exist in the Tool BI system. You have to manually delete these InfoObjects. This presupposes that the InfoObjects in the Tool BI system are not yet being used or that you have already deleted the dependencies of these InfoObjects.

2.2.4.3 Secondary InfoObjects for Local Characteristics

Definition

See Secondary InfoObject.

Use

You use secondary InfoObjects to replicate the master data [page 337] of fixed and local characteristics to the relevant InfoObjects in the Reporting BI system.

Before you replicate the master data of local characteristics to the Reporting BI system, you have to make the relevant InfoObjects known to the characteristic repository. When you do this, bear the following in mind:

- Local characteristics for which a (primary) InfoObject was generated in the Tool BI system
  In Customizing for *Bank Analyzer*, you can specify the name of the secondary InfoObject when you create a local characteristic (*Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Register Characteristics*). If you do not specify a name, the primary InfoObject in the Reporting BI is supplied with master data.
- Local characteristics to which a (primary) InfoObject has been assigned from the Tool BI system
  In Customizing for *Bank Analyzer*, you have to specify the name of the secondary InfoObject when you create a local characteristic (*Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Register Characteristics*). You have to specify a name because in this case only the secondary InfoObject is supplied with master data. If you do not assign a secondary InfoObject, the master data cannot be replicated to the Reporting BI system.

2.2.5 Custom Characteristics

Definition

Characteristics whose metadata and master data is determined by an InfoObject in the Tool BI system.

Example

Examples of custom characteristics include region, profit center, branch office, and business area.

Use

Custom characteristics are used in the same way as all other characteristics in Bank Analyzer (For more information, see General Information About Characteristics [page 21].)

Before you can use custom characteristics in the core Bank Analyzer system or in the Tool BI system, you have to work through the following steps:

1. Create an InfoArea in the Tool BI system.
2. Create an InfoObject catalog for custom characteristics in the Tool BI system.
3. Register the InfoObject catalog of the Tool BI system in the core Bank Analyzer system. You do this in Customizing for *Bank Analyzer* by choosing *Basic Settings* > *Edit Global Settings for Characteristics and Key Figures*.
4. Create InfoObjects in the Tool BI system.

   Recommendation

   You can choose any names for the InfoObjects; we recommend assigning separate prefixes to the individual InfoObjects so that you can tell them apart.
5. Register the characteristics of the Tool BI system in the core Bank Analyzer system. You do this in Customizing for *Bank Analyzer* by choosing *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Register*

Characteristics ▸. Custom characteristics in the core Bank Analyzer system have the prefix /BIC/ or /1FB/, or no prefix.

You can display custom characteristics in the *Enhanced Overview* in the characteristic monitor [page 65]. Unlike local and fixed characteristics, you cannot edit custom characteristics in the characteristic monitor.

Integration

Custom characteristics are created in the Tool BI system as InfoObjects with metadata and master data. Local [page 59] and fixed characteristics [page 25] are created in the core Bank Analyzer system. Their InfoObjects and master data have to be replicated in the Tool BI system.

Time-Dependent Master Data for Custom Characteristics

The master data entered for custom characteristics in the Tool BI system is not time-dependent. To enter time-dependent master data, proceed as follows:

1. In Customizing for *Bank Analyzer*, create an InfoObject that will denote the validity of a characteristic value by choosing ▸ *Basic Settings* ▸ *Settings for Meta Data* ▸ *Technical Fields* ▸ *Automatically Create/Register Technical Fields* ▸.Set the *Validity of a Char. Value* indicator.The technical name *BA_VALID* has already been entered and cannot be changed. Once the activity has been carried out, the InfoObject BA_VALID is created in BI.
2. Enter InfoObject *BA_VALID* as an attribute for all the InfoObjects for which time-dependent master data is to be entered.
    1. To do so, choose the InfoObject in BI.
    2. On the *Master Data/Texts* tab page, set the *With Master Data* indicator.
    3. On the *Attributes* tab page, enter InfoObject `BA_VALID` and set the *Time-Dependent* indicator.
    4. Save your entries and activate the InfoObject.
3. Enter time-dependent master data in the Tool BI system. Do this by choosing the *Maintain Master Data* pushbutton for the appropriate InfoObject, and then entering a from date, to date, and the appropriate master data. Also set the BA_VALID attribute for this period to `X`.

2.2.5.1 Changing Custom Characteristics

Prerequisites

To change fundamental properties of the InfoObjects for custom characteristics [page 62] in the Tool BI system [page 16], first use the where-used list to check whether the characteristics are already being used in Bank Analyzer.

> **Note**
>
> You can change the main properties of the InfoObjects in the Tool BI system only if there are no dependencies on the InfoObject in the Tool BI system. If this is not the case, you have to delete the dependencies or make changes so that the InfoObject is not in use.

Procedure

1. To change the InfoObject, go to Customizing for *Bank Analyzer* and choose *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Edit InfoObjects for Custom Characteristics*, and then choose the *Maintain* pushbutton.
2. If you change the metadata for registered InfoObjects in the Tool BI system, inconsistencies arise between this data and the characteristics that are already registered in Bank Analyzer. To correct these inconsistencies, go to Customizing for Bank Analyzer, and choose *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Regenerate Registered Characteristics*. If the system does not display any characteristics, there are no inconsistencies and a regeneration is not necessary.

2.2.5.2 Deleting Custom Characteristics

Prerequisites

You can only delete custom characteristics [page 62] if you have the required authorization. Before you delete custom characteristics, use the where-used list to check whether the characteristics are already being used in Bank Analyzer. If the characteristics are being used, you have to first delete all users of the characteristic or change the user so that the relevant characteristic can no longer be used.

> **Caution**
>
> Delete the custom characteristic only if you are certain that you know all of the users and that you can adapt them. Do not delete the custom characteristics if this is not the case.

Procedure

To delete custom characteristics, proceed as follows:

- Remove the InfoObject from the InfoObject catalog for custom characteristics or delete the InfoObject in the Tool BI system [page 16]. Note that you can only delete the InfoObject from the Tool BI system if it is not yet used there.
- In Customizing, choose *Bank Analyzer* > *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Register Characteristics*.On the tab page, choose *InfoObjects* > *Compare*.You can only use the *Compare*

function if the InfoObject catalog of custom characteristics no longer contains characteristics that are already registered, or if these characteristics are to be deleted. In addition, you have to have the authorization required for comparing.

> **Note**
>
> If you have already deleted a characteristic and want to retrospectively determine whether the characteristic is still used, call transaction SE37, choose function module */BA1/F0_MAP_USAGE_TREE_DISPLAY* and enter the characteristic name in parameter *I_UOBJID*.

2.2.6 Characteristic Monitor

Use

This function enables you to display the fixed [page 25] and local characteristics [page 59] in Bank Analyzer. The characteristic monitor thus allows you to monitor the metadata of characteristics in the core Bank Analyzer system and InfoObjects in the Tool BI system. The master data for the fixed and local characteristics is displayed in the master data monitor [page 338].

Prerequisites

You have registered the local characteristics in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Register Characteristics*.

You have registered the fixed characteristics by running the initial configuration in Customizing for *Bank Analyzer* under *Basic Settings* > *Global Settings* > *Run Initial Configuration*.

Features

The monitor displays whether inconsistencies have occurred between the properties and master data for the InfoObjects and the related local characteristics in the core Bank Analyzer system. The monitor can display the following inconsistencies:

- The InfoObject does not exist
- The properties of the InfoObject and the data element are not the same

The characteristic monitor provides the following functions:

- *Automatic creation of InfoObjects for local characteristics in the Tool BI system*
  To do this, choose *Create InfoObjects* in the characteristic monitor. If the relevant InfoObject catalog does not already exist, it is generated automatically.
- *Update of the InfoObjects if inconsistencies occur*
  Choose *Update InfoObjects*.

- *Display all Characteristics in Bank Analyzer*
  The system displays an overview of all the characteristics available to customers in Bank Analyzer, including the custom characteristics. It is the same as the list for registering the characteristics. To do so, choose one of the following options:
  - Select the *All Visible Characteristics* radio button on the selection screen.
  - After the characteristics have been checked, choose *Extended Overview* on the screen displaying the results list.

> **Note**
>
> This list does not include fixed characteristics that are registered for technical reasons only or fixed characteristics that are simply used as templates for InfoObjects.

Activities

You call the characteristic monitor in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Characteristic Monitor*.

2.2.7 Characteristic-Based Authorization Check in Bank Analyzer

Use

Bank Analyzer has both a global SAP authorization check and an authorization check that is based on characteristics.

You can use the characteristic-based authorization check (which comprises characteristics of your choice from the characteristic repository) to enable only specific user groups to process or display data.

This kind of authorization check can be useful if, for example, you only want your employees to be able to process particular countries or product groups in a live system.

Integration

Characteristic-based authorization checks are run at various points in Bank Analyzer, and are called up directly from the applications. The checks are normally run after data has been selected that only specific groups of users are permitted to display or process.

When it runs an authorization check, the system first of all reads settings from Bank Analyzer Customizing and from the user roles, and interprets the selection data with the help of characteristic master data. The system then calls the global SAP authorization check.

Prerequisites

If you use a characteristic-based authorization check, you have to make settings in Bank Analyzer Customizing, in Customizing for the individual applications where you use the check, and in the user roles that are also used for the global authorization check. For more information, see Setting Up a Characteristic-Based Authorization Check [page 68] .

You use the following Customizing entities to control the characteristic-based authorization check:

- Authorization object
- Authorization type
- Authorization context
- Authorization characteristic profile

Authorization objects, types, and contexts are shipped by SAP. You create authorization characteristic profiles in Customizing for Bank Analyzer.

An authorization characteristic profile tells the system which characteristics it has to use for the authorization check, In the corresponding authorization object, you specify which characteristic values are allowed for these characteristics.

To enable the system to find the authorization characteristic profile that belongs to an authorization object, you have to assign the authorization characteristic profiles to the relevant application. You use authorization types and authorization contexts for this purpose. However, you can also assign the authorization characteristic profiles directly to Customizing entities for a specific application.

To find out which authorization objects belong to which authorization types, and to which Customizing entities you have to assign authorization characteristic profiles, see the table in the document entitled Authorization Objects in Bank Analyzer [page 70] .

Features

The way in which an authorization check is run depends on your Customizing settings and on the architecture of your Bank Analyzer applications. The following scenarios are possible:

- **Scenario where an authorization type is not used**
  The system finds the authorization characteristic profile in the Customizing settings of the application, and passes the profile on directly to the authorization check.
  For example, this is the case in Strategy Analyzer reporting.
  Scenario where an authorization type is used
  The system derives the authorization characteristic profile from the authorization type, and then passes the profile on to the authorization check.
  If there is also an authorization context for this authorization type, the system uses the authorization characteristic profile defined for this context. If an authorization characteristic profile has **not** been defined for the authorization context, the system uses the authorization characteristic profile defined in the authorization type.
  When data is selected from a primary data source, the system uses a characteristic-based authorization check involving an authorization type and authorization context – provided that the primary data source itself has not been assigned to a specific authorization characteristic profile in Customizing for primary data sources.

Example

In Strategy Analyzer, characteristic-based authorization checks are carried out as part of NPV analysis reporting. The authorization check is run as soon as reporting has been called up. The system checks whether the user has the relevant authorization for **all** the selected transactions of the valuation run in question. If the user does not have the relevant authorization, the system outputs an error message and terminates reporting, without displaying any results data.

A maximum of nine characteristics is available for the check. You can choose these from the characteristic repository of Bank Analyzer. However, the characteristics that are checked should also be characteristics that are actually used for the transactions you are valuating In the user roles, you specify the characteristic values for which a user may view the results.

See also the example in the document entitled Setting Up a Characteristic-Based Authorization Check [page 68] .

2.2.7.1 Setting Up a Characteristic-Based Authorization Check

Use

By defining authorizations or characteristics and characteristic values for the characteristic-based authorization check, you can specify that only specific user groups should be able to process or display data. Since this check is an enhancement of the global SAP authorization check, you will need to define settings for it in Customizing and in user roles.

> **Note**
>
> For more information, see Characteristic-Based Authorization Check in Bank Analyzer [page 66].

Procedure

To set up the characteristic-based authorization check, proceed as follows:

Define authorization characteristic profiles

'Authorization characteristic profile' is the main Customizing entity for the characteristic-based authorization check. If you want to perform an authorization check, you have to define authorization characteristic profiles. These profiles contain all of the characteristics that you want the system to check, and the sequence in which the system should check them. You can use any of the characteristics in the Bank Analyzer characteristic repository for this purpose. To do this, choose Customizing for *Bank Analyzer* > *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Authorizations for Characteristic Values* > *Edit Authorization Characteristic Profiles* .

Assign profiles for authorization characteristics

You can perform an authorization check either globally or locally. If you want the system to always check the same characteristics irrespective of the other Customizing settings, assign one authorization characteristic profile to each authorization type. Proceed as follows:

Assign authorization characteristic profiles to the authorization types

The authorization type is the central control element of the characteristic-based authorization check. A separate authorization object that contains up to ten fields exists for each authorization type. The system uses this to check authorizations. You use the authorization characteristic profiles to define the characteristics the system uses for checking authorizations. You assign authorization characteristic profiles to the authorization types to tell the system which characteristics belong to which authorization objects. To assign authorization characteristic profiles to authorization types, choose Customizing for *Bank Analyzer* > *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Authorizations for Characteristic Values* > *Edit Authorization Characteristic Profiles*. To find out which authorization type or context belongs to the authorization check in question, see the document entitled Authorization Objects in Bank Analyzer [page 70].

If you want the authorization check to be performed in individual applications subject to the Customizing settings, do not assign any authorization characteristic profiles to the authorization types and proceed as follows:

Assign authorization characteristic profiles to the applications

If you assign authorization characteristic profiles to individual applications, you will also need to define settings in Customizing for the applications in question.

Edit roles

You assign authorization characteristic profiles to authorization types or to individual applications to determine when the system checks which characteristics. In the user roles, you define the values of these characteristics for which the user has authorization.

> **Note**
>
> In the authorization characteristic profile, make sure that you define a sequence for the characteristics that also applies to the fields in the authorization objects. A second characteristic in the authorization characteristic profile belongs to the second characteristic field in the associated authorization object, for example.

Result

You have set up the characteristic-based authorization check as required.

Example

You want to check in Strategy Analyzer that the user of the role `SA_ANWENDER` only has authorization to display results for the product group 2200 during NPV analysis reporting. For this purpose, you require an authorization characteristic profile that contains the characteristic 'product group', for example. Proceed as follows:

1. You create an authorization characteristic profile with the name `SA_PVREPORTING` and assign the product group to it as a characteristic with the field number 1.

2. You assign your authorization characteristic profile to the authorization type *SA: PV Reporting* and to the authorization object `F_BAR5_RPV`.
3. You assign your authorization characteristic profile to the field instance that is the basis of your valuation runs.
4. You enter the authorization object `F_BAR5_RPV` in the role `SA_ANWENDER` and enter the value 2200 in the characteristic field 1. You then regenerate the role.

Users that belong to the role `SA_ANWENDER` are now allowed to display the results of the product group 2200 only. These users are not allowed to carry out reporting for results data from other product groups.

2.2.7.2 Authorization Objects in Bank Analyzer

A characteristic-based authorization check is used at various points in Bank Analyzer applications. The table below shows you which authorization objects, authorization types, and Customizing information belong together, and which characteristics the system checks.

> **Note**
>
> Note that some of the characteristic-based authorization checks apply across all applications.

Characteristic-Based Authorization Check in Bank Analyzer

Application	Where-Used	Type of Check	Authorization Object	Authorization Type	Customizing entity to which you assign the authorization characteristic profile
All applications	Derivation	Source field characteristics	`F_BAP4_DER`	Derivation Rule	Derivation step ID
Infrastructure	Primary Data Source	Data that is read via the primary data source	`F_BAF3_DSD`	Primary data source data	Primary data source ID
Source Data Layer	Primary objects	Complex authorization concept for primary objects	`F_BAFP_PO`	Primary objects	Processing area of the processing framework
Infrastructure	Ad hoc reporting	Data from secondary data source is checked	`F_BAFP_REP`	Ad hoc reporting	Report ID
Balance Analyzer	Custom characteristics check	Legal entity, accounting system, custom MM	`F_BAB0_SPT`	Process authorization	

Application	Where-Used	Type of Check	Authorization Object	Authorization Type	Customizing entity to which you assign the authorization characteristic profile
Balance Analyzer	Display of accounting object balances, corrections, drilldown	Custom MM	F_BAB2_BDS	Descriptive characteristics for accounting object balances	
Balance Analyzer	Financial reporting process	Value type	F_BAB2_VAL	None	
Balance Analyzer	Manual adjustments In Balance Processing	Correction authorization	F_BAB2_ADJ	None	
Balance Analyzer	All accounting Accounting	Legal entity, accounting system, Application ID	F_BAB2_PRO	Process authorization	
Historical database	Data display in ad hoc reporting and in the test run	Display authorization	F_BAR6_CHR	HDB: Authorization for characteristics	Selection ID
Credit Risk Analyzer			F_BAR2_005	CRA: Reporting	
Limit Manager	Overview of limits and drawings	Limit-relevant characteristics or characteristics of the limit type	F_BAR3_CHR	Limit: Authorization for characteristics	Edit characteristics of Limit Manager and edit limit types
Strategy Analyzer	NPV analysis reporting	Field instance characteristics	F_BAR5_RPV	SA: PV reporting	Field Instance
Strategy Analyzer	Gap analysis reporting	Segment hierarchy characteristics	F_BAF0_HPR	SA: Segment ID	Segment hierarchy

2.3 Key Figure Repository

Definition

A repository used for the central management of all the key figures in Bank Analyzer. Both elementary and complex key figures are registered with their metadata in the key figure repository.

Use

The key figure repository is used as follows:

- It enables standardized semantics for key figures to be used in Bank Analyzer. This ensures that several components can use the same key figure.
- It provides the SAP key figures [page 76] and key figure classes [page 74] in Bank Analyzer.
- You can add custom key figures [page 96] to the repository.
- It provides the central services for all key figures (for example, calculation of the metadata).

2.3.1 General Information About Key Figures

Definition

Key figures are value fields that are stored centrally in Bank Analyzer in the key figure repository [page 72]. This ensures that they are available to every component with the same properties and meaning.

Additionally, Bank Analyzer provides complex key figures.

Use

Key figures are used, for example:

- As balances in positions
- In object transfer structures (OTS)
- As Analyzer results in the Results Database

Structure

Key figures are created in the key figure repository on the basis of the key figure classes [page 74] shipped by SAP. In addition to the key figures that are created in the core Bank Analyzer system, one or two InfoObjects of the category Key Figure are generated in the related Tool BI system.

The InfoObjects generated in the Tool BI system have the same names as the key figures in the core Bank Analyzer system. If the key figure is used in conjunction with a currency, an extra key figure of the category Dimension is generated in the Tool BI system. In this case, the InfoObjects have the following names:

Unit Category	Name of the InfoObject in the Tool BI System
01 Object Currency	OBJ_CURR
02 Evaluation Currency	EVAL_CURR
03 Functional Currency	BIL_CURR
04 Transaction Currency	TRAN_CURR
05 General Unit for Quantities	0UNIT
98 Generic Unit Category	<Key figure>U
99 Generic	<Key figure>C

In Bank Analyzer, a distinction is made between:

- SAP Key Figures [page 76]
- Custom Key Figures [page 96]

Comparison of SAP Key Figures and Custom Key Figures

	SAP key figure	Custom key figure
Source	Delivered by SAP	Defined by customers
Creation in the core Bank Analyzer system	Automatic creation by means of the initial configuration	Creation by activation or transport
Creation in the Tool BI System	Automatic creation by means of the initial configuration	Automatic creation by activation or transport
Prefix in the core Bank Analyzer system 1. In Customizing 2. Field Name 3. Key figure repository	1. No prefix 2. /BA1/K (elementary key figure) /BA1/KX (complex key figure) 3. &	1. No prefix 2. /BIC/ (elementary key figure) /BIC/X (complex key figure) 3. No prefix
Prefix in the Tool BI System: 1. InfoObject key figure 2. InfoObject unit	1. /BA1/ 2. /BA1/...C	1. No prefix 2. ...C
Visibility in the key figure repository	Cat. 1: visible for customers Cat. 2: not visible for customers	Always visible for customers

> **Note**
>
> Key figures supplied with Bank Analyzer content have their own namespace. These key figures have the following prefix in the core Bank Analyzer system: "/B20C/S_" or "_". The prefix in the Tool BI system is: "/BA1C/K" or "/BA1C/K...C".

There is **no** difference in the way the SAP key figures and the custom key figures are used.

> **Recommendation**
>
> When deciding whether to create a key figure, we recommend you proceed as follows:
>
> First check whether an SAP key figure or a custom key figure that has the required metadata and semantics already exists. If there is no appropriate key figure, create a custom key figure.

2.3.2 Key Figure Classes

Definition

Key figure classes are predefined by SAP and enable you to define key figures in the key figure repository [page 72] of Bank Analyzer. You must assign each key figure to a key figure class. You can define any number of key figures for the key figure classes shipped by SAP. Key figure classes are defined and shipped by SAP in accordance with the requirements of the applications.

You can find an overview of the key figure classes delivered by SAP in Customizing for Bank Analyzer under *Basic Settings* > *Settings for Meta Data* > *Key Figures* > *Edit Key Figures* on the detail screen for a key figure.

Structure

Key figure classes contain the following information:

- The key figure category determines the technical properties of a key figure. The following key figure categories exist within Bank Analyzer:

Technical Category	Key Figure Category	Description
01 Amount (with currency type)	AMNT CURR NOADD FLTP	Non-cumulative amount (float)
	AMOUNT CURR ADD FLTP	Cumulative amount (float)
	AMOUNT CURR ADDIT	Cumulative currency field
	AMOUNT CURR NON ADD	Non-cumulative currency field

	LAMOUNT CURR ADDIT	Cumulative currency (long)
	LAMOUNT CURR NON ADD	Non-cumulative currency (long)
02 Number (without unit)	DECNUM ADDITIVE	Cumulative number (DEC)
	DECNUM NON ADDITIVE	Non-cumulative number (DEC)
	FLOAT ADDITIVE	Cumulative number (float)
	FLOAT NON ADDITIVE	Non-cumulative number (float)
	PERCENTAGE	Non-cumulative percentage
	PERCENTAGE ADDITIVE	Cumulative percentage
03 Quantity	QUANTITY ADDITIVE	Cumulative quantity
	QUANTITY NOT ADDIT	Non-cumulative amount
04 Integer	INTEGER ADDITIVE	Cumulative number
	INTEGER NOT ADDIT	Non-cumulative number
99 Complex key figure	COMPLEX	Complex key figure

> **Note**
>
> Bank Analyzer permits amounts of length 23 or 31 in a specific currency to be generated and processed. Note that the reporting BI system only supports amounts in a specific currency of length 17 for key figure categories AMOUNT CURR ADDIT, AMOUNT CURR NON ADD, LAMOUNT CURR ADDIT, and LAMOUNT CURR NON ADD
>
> The system sends an error message if one of these key figures is transferred from Bank Analyzer to the reporting BI system with a value that has more than 17 places.

- An additive indicator signifies a key figure class for cumulative key figures. Furthermore, the *Exception Aggregation* indicator controls whether or not an exception aggregation can be defined for a custom key figure.

  > **Note**
  >
  > The system generates InfoObjects with the SUM aggregation for both cumulative and non-cumulative key figures. Note that you can use formulas to adjust the aggregation behavior of InfoObjects for key figures when you define the query in the reporting BI system. This may be necessary if you use InfoObjects for non-cumulative key figures, for example.

- The origin of a key figure class is documented in the Bank Analyzer component that is assigned to the key figure class. All of the Bank Analyzer components can use key figure classes.

> **Note**
>
> In some Bank Analyzer components (for example, in Balance Analyzer), the key figure classes shipped with the system are pre-programmed and you do not have a completely free choice of key figure classes when you are creating key figures.

- For key figure classes of the category Amount and Quantity (with unit category), the key figure class defines which unit categories are permitted for the key figures that are based on them.
- Complex key figures are always of the category COMPLEX. You define their composition by explicitly specifying the appropriate Dictionary structure in the key figure class.

2.3.3 SAP Key Figures

Definition

SAP key figures are predefined by SAP and shipped in the form of a system table (see General Information About Key Figures [page 72]).

For a list of the key figures delivered by SAP, see Visible SAP Key Figures [page 77] and Invisible SAP Key Figures [page 90].

Use

SAP key figures are used in the same way as all other key figures in Bank Analyzer.

> **Note**
>
> *Invisible SAP key figures* are registered in the key figure repository for technical reasons. As a result, you cannot view or select them in every application function. The individual applications manage the use of these invisible SAP key figures.

To activate SAP key figures within an initial configuration in Customizing, choose *Bank Analyzer* > *Basic Settings* > *Global Settings* > *Run Initial Configuration*. The first time you run the initial configuration, the */BA1/KFSAP* InfoObject catalog for SAP key figures and the InfoObjects in the Tool BI system [page 16] are created. The InfoObjects contain the prefix namespace */BA1/*. The metadata for activating/generating the InfoObjects is derived from the properties of an assigned key figure class.

> **Note**
>
> You can execute the initial configuration run as often as you require. For example, if you have just upgraded to a new release with new fixed characteristics. If InfoObjects already exist, they are overwritten when you rerun the initial configuration.

Client dependency

SAP key figures are defined and shipped **for all clients**. They are activated for **specific clients**, however, since the key figure repository is client-specific and each client can be assigned to a different Tool BI system.

SAP Key Figure Monitor

In the key figure monitor, you can use a report to display the SAP key figures and the custom key figures [page 96] separately. This report returns a status overview of all visible and invisible SAP key figures. The SAP key figures are identified with an *&*. To call the key figure monitor, go to Customizing for *Bank Analyzer*, and choose *Basic Settings* > *Settings for Meta Data* > *Key Figures* > *Key Figure Monitor*, or call transaction */BA1/FU_KFSAPMONITOR*.

In the SAP key figure monitor you can:

- Generate an overview of the status of the SAP key figure for the particular client
- Display the status details for the SAP key figure for the particular client
- Display detailed information about the SAP key figure
- Activate individual SAP key figures in exceptional cases (Normally, SAP key figures are activated during the initial configuration.)
- Transport individual SAP key figures in exceptional cases (This transport does not substitute the initial configuration in the target client.)

2.3.3.1 Visible SAP Key Figures

You can display the visible key figures delivered by SAP in Customizing for *Bank Analyzers* under *Basic Settings* > *Settings for Meta Data* > *Key Figures* > *Key Figure Monitor*.

SAP delivers the following fixed (visible) key figures:

Technical Name	Text	Data Type	Short Description
35BESAM	Position Amount (BT)	CURR	Position amount in business transaction items: Without fixed semantics. Generically applicable.
35NOMAM	Nominal Amount (BT)	CURR	Nominal amount in business transaction items: Amounts that are relevant for cash flow generation (for example, amount from disbursement of a loan) must be provided using this field of the corresponding business transaction item.

35QUANT	Quantity (BT)	QUAN	Quantity in business transaction items: Quantities that are relevant for cash flow generation (for example, quantity when a security is bought) must be provided using this field of the corresponding business transaction item.
35TRAAM	Transact. Amount (BT)	CURR	Transaction amount in business transaction items: Amount without defined semantics, can be used generically.
55ACGDR	Diff G/L: Current Amt	CURR	Cumulated exchange rate gain/loss difference in functional currency: Current fiscal year contribution.
55ACVAL	Amortized Cost at End of Calculation Date	CURR	
55AECLR	APE Intern. Clearing	CURR	Foreign currency concept in Balance Analyzer: Internal clearing account for APE
55AMBAL	Amount in Rep. Crcy	CURR	This field contains the amount in the functional currency. The business transaction item contains the field. The amount in the field is posted in the financial accounting documents.
55_AMDEV	CMD: Asset FYTC	CURR	Contribution from the current fiscal year to the cumulated merge difference for the equivalent values.
55_AMDPR	Cumulated Merge Difference: Cur. FY Contr. Position in RC	CURR	Contribution from the current fiscal year to the cumulated merge difference for the items in the functional currency.

55AMNOM	Nominal Value	CURR	This field contains the nominal amount in the nominal currency and is classed as a quantity field. The business transaction item contains the field. The amount in the field is copied in the financial accounting documents.
55AMOBJ	Amount in Transaction Currency	CURR	This field contains the amount in the transaction currency. The business transaction item contains the field. The amount in the field is posted in the financial accounting documents.
55APCLO	APP Intern. Clearing	CURR	Foreign currency concept in Balance Analyzer: Internal clearing account for APP
55APD30	CDGL: Current FY OC	CURR	The difference (in the transaction currency) between the revenues and expenses for the current fiscal year from the SDL and accounting.
55APDB3	Cumulated profit difference: Current FY contribution in reporting currency	CURR	The difference (in the functional currency) between the revenues and expenses for the current fiscal year from the SDL and accounting.
55CCGDR	Cml. Dif. ER Gain/Loss	CURR	
55CKSUM	Check Sum: Standard in Reporting Currency	CURR	Sum of the debit/credit relevant key figures in the functional currency that are determined for checking purposes when balances are transferred from the SDL and that are stored for each balance sub-object.
55_CMDEV	CMD: Equivalent Vals	CURR	Cumulated merge difference for equivalent values
55_CMDPR	Cumulated Merge Difference: Position in Reporting Currency	CURR	Merging difference for items in the functional currency

55CMPDF	Merge Difference: All in Object Currency	CURR	If the system detects a difference when it runs the debit-credit check but ignores it, however, because of the defined threshold values, it adds this difference in the merge difference key figure at balance object level (in the transaction currency).
55CPD	Cumulated Profit Difference in Object Currency	CURR	The difference (in the transaction currency) between the revenues and expenses from the SDL and Accounting accumulated over several years.
55CPDB	Cumulated Profit Difference in Reporting Currency	CURR	The difference (in the functional currency) between the revenues and expenses from the SDL and Accounting accumulated over several years.
55CPDFB	Merge Difference: Standard in Reporting Currency	CURR	If the system detects a difference when it runs the debit-credit check but ignores it, however, because of the defined threshold values, it adds this difference in the merge difference key figure at balance object level (in the functional currency).
55CURPD	Diff. FX Gain/Loss	CURR	Valuation differences between IAS and local GAAP due to foreign currency transactions.
55DGRID	Res.Mat. Raster	TTYP	Residual maturity raster
55EXRAT	Exchange Rate	DEC	Exchange rate for currency translation (without new foreign currency concept)
55FISPR	Fin. Instr. Spread	FLTP	Financial instrument spread
55FSBIL	KF in Reporting Crcy	CURR	Financial statement items in Balance Analyzer: Non-semantic key figure in functional currency

55FSOBJ	KF in Object Crcy	CURR	Financial statement items in Balance Analyzer: Non-semantic key figure in transaction currency
55GAPER	CPP: AmtCurr. FY EV	CURR	Foreign currency concept in Balance Analyzer: Contribution from the current fiscal year to the equivalent value of the cumulated profit position
55GAPPO	CPP: Amt Curr. FY OC	CURR	Foreign currency concept in Balance Analyzer: Contribution from the current fiscal year to the cumulated profit position in the transaction currency
55GCCDR	ER Profit: Gen.Pos.	CURR	Foreign currency concept in Balance Analyzer: Exchange rate profit deviation from general currency position (functional currency)
55GCCGR	General Position: Currency Gain	CURR	Foreign currency concept in Balance Analyzer: Currency gain for general position
55GCPCR	Cumulated Profit Position: Currency Gain	CURR	Foreign currency concept in Balance Analyzer: Currency gain of the cumulated profit position
55GCPDR	ER Diff. for Prof. Pos.	CURR	Exchange rate profit difference from profit position (functional currency)
55GCPER	CPP: Equivalent Value	CURR	Foreign currency concept in Balance Analyzer: Equivalent value of cumulated profit position
55GCPO	General Position in Object Currency	CURR	Foreign currency concept in Balance Analyzer: General position in the reporting currency

55GCPPO	Cumulated Profit Position in Object Currency	CURR	Foreign currency concept in Balance Analyzer: Cumulated profit position in transaction currency
55GCPPR	Cumulated Profit Position in Reporting Currency	CURR	Foreign currency concept in Balance Analyzer: Cumulated profit position in the reporting currency
55GCPR	General Position in Reporting Currency	CURR	Foreign currency concept in Balance Analyzer: General position in the reporting currency
55GCPRR	Rndg Diff. Profit Pos.	CURR	Foreign currency concept in Balance Analyzer:
55GCRDR	Roundg Diff. Gen.Pos.	CURR	Foreign currency concept in Balance Analyzer:
55GEVR	General Position: Equivalent Value	CURR	Foreign currency concept in Balance Analyzer: Equivalent Value of the General Position
55HAC	Hedged Amrtzd Costs	CURR	Hedged amortized costs
55HFVAL	Hedge fair value	CURR	
55HSPR	Hedge Spread	FLTP	
55IRRAC	Amort. Acq. Csts EIR	FLTP	Effective rate for acquisition costs
55IRRHA	Hdgd Amor AC Eff IR	FLTP	Effective interest rate for hedged amortized acquisition costs
55LCPCR	G/L-related cumulated profit position: Currency gain	CURR	G/L-related cumulated profit position: Currency gain
55LCPER	G/L-related cumulated profit position: Equivalent value	CURR	G/L-related cumulated profit position: Equivalent value
55LCPPO	G/L-Related Cumulated Profit Position in Object Currency	CURR	G/L-Related Cumulated Profit Position in TC
55LCPPR	G/L Accounts: Cumulate Profit Position	CURR	
55LCPR	G/L Accounts: Total Position	CURR	

55LEVR	G/L-Related Total Position: Equivalent Value	CURR	
55MGRPS	Resid. Mat Range Values	TTYP	Values in residual maturity ranges
55NOMB	Nom.Amt in Rep. Crcy	CURR	Nominal amount in functional currency (quantity field)
55NOMPQ	Nom.Amount per Unit	CURR	Nominal amount per unit for unit-quoted financial instruments in the nominal currency (quantity field)
55ORCDR	OP. ER Gain Diff	CURR	Foreign currency concept in Balance Analyzer:
55ORCGR	Object Position: Currency Gain	CURR	Foreign currency concept in Balance Analyzer: Exchange rate gain for transaction currency
55ORCPO	Object Position in Object Currency	CURR	Foreign currency concept in Balance Analyzer: Object Position in Transaction Currency
55ORCPR	Object Position in Reporting Currency	CURR	Foreign currency concept in Balance Analyzer: Object Position in Reporting Currency
55OREVR	Object Position: Equivalent Value	CURR	Foreign currency concept in Balance Analyzer: Equivalent Value of Object Position
55ORRDR	Obj. Pos.: Roundg Diff	CURR	Foreign currency concept in Balance Analyzer: Rounding difference of object position
55OUTFC	Deduction Factor	DEC	Determines the relative amount that matures.
55PFTAC	Cumulated Profit/Loss	CURR	Cumulated profit/loss in functional currency
55PFTAO	Cumulated Profit/Loss	CURR	Cumulated profit/loss in transaction currency

55QUANT	Quantity	QUAN	The business transaction item contains this field. The quantity is copied to the financial accounting documents.
55SRADJ	Dividend Correction	CURR	Used to calculate the value of stock prescription rights Formula:Value of subscription right = (price of old stocks - (price of new stocks + dividend correction)) / (subscription factor+1)
55SRFAC	Subscription Factor	DEC	The relationship between old and new stocks. Determines the number of stocks that have to be owned before new stocks can be purchased.
55SRSPN	New Stock	CURR	Purchase price of new stock
55SRSPO	SR: Old Stock	CURR	Subscription right for old stock: Key figure for the old market value before the increase in capital.
55TCDER	Merge Difference: Equivalent Values	CURR	Foreign currency concept in Balance Analyzer: Merging difference from equivalent values for currency positions
55TCDR	Cumulated Merge Difference: Positions in Reporting Currency	CURR	Foreign currency concept in Balance Analyzer: Merging difference from currency positions in the functional currency
55TCPR	Total Position in Reporting Currency	CURR	Foreign currency concept in Balance Analyzer: Total Position
55TCSER	Check Sum: Equivalent Values	CURR	Check Sum: Equivalent Values
55TCSPR	Check Sum: Position in Reporting Currency	CURR	Check Sum: Position in Functional Currency
55TCSO	Check Sum: All in Object Currency	CURR	Check sum: All in transaction currency

55TEVR	Total Position: Equivalent Value	CURR	Foreign currency concept in Balance Analyzer: Equivalent value for total position
55TFVAL	Fair Val. in Rep. Crcy	CURR	Fair value in functional currency
55TXVAL	Tax Book Value	CURR	Used as a reference for calculation of tax liabilities
55VBVAL	External Book Value	CURR	Used instead of IAS book value in the financial reporting process as a reference for the calculation of fair value.
55VFVAL	Fair value	CURR	
55FVVAL	Fair Value at End of Calculation Date	CURR	
55VPER	Calc.Base of Price %	DEC	Calculation base for price (percentage-quoted): Key figure in the accounting document that contains the price used for the calculation of fair value (for example, the price of a bond).
55VPQUN	Calc.Base of Price	FLTP	Calculation base for price (indirectly quoted): Key figure in the accounting document that contains the price used for the calculation of fair value (for example, the price of a stock).
62ALPHA	Alpha coefficient as a fixed key figure	DEC	
62ASSET	Calc: Asset	STRUCT	Calculation figures for contract: Asset
62CC_RS	Secured/Non-Sec. Portion	STRUCT	Calculation Figures for Secured and Unsecured Parts
62CCFV	Credit Conversion Factor (CCF)	DEC	

62CCRRP	Risk Position Amount for Counterparty Credit Risk	DEC	Risk Position Amount for Counterparty Credit Risk This field is used in the standardized method for counterparty credit risk.
62COL_G	Collat./Guarantees	STRUCT	Calculation figures for collateral and guarantees
62COLL	Calc: Collateral	STRUCT	Calculation figures for contract: Collateral
62CONST	CE Constants	STRUCT	Constants in credit exposure
62CONTR	Calc.: Contract	STRUCT	Calculation figures for contract
62CORCO	Correlation coefficient	DEC	Indicates whether there is an excessive correlation between collateral provider and borrower. Business partners that are in a group relationship in particular are excessively correlated (the same applies for the party selling the exposure for purchased exposures).
62CTGRT	Counter Guarantee	COMPLEX	Calculation figures for contract: counter guarantee
62EAD	Exposure at Default	CURR	Exposure at default
62EEXPT	Expected Exposure	DEC	
62EL	Expected loss	DEC	
62ELSE	Expected loss for each segment of securitization exposure	DEC	
62ELSEG	Expected Loss per Segment	CURR	Expected loss per segment in the event of exposure with private customers
62EPROR	EAD	CURR	
62ETLGD	EAD x LGD	CURR	
62FACTD	Number of Lines for Each Exposure	DEC	

62FRLIN	Free line for facilities	DEC	
62GUAR	Calc: Guarantee	STRUCT	Calculation figures for contract: Guarantee
62HCCC	Haircut C for Collateral	DEC	
62HCCG	Haircut C for Guarantee	DEC	
62HCFXC	Haircut FX for Collateral	DEC	
62HCFXG	Haircut FX for Guarantee	DEC	
62LGDV	Loss Given Default	DEC	
62LIBLT	Calc: Liabilities	STRUCT	Calculation figures for contract: Liability
62MATM	Maturity M as Fixed Key Figure	DEC	
62MATMX	Maximum Original Maturity	QUAN	
62MATNM	Nominal Maturity	DEC	
62MATNO	Nominal maturity	QUAN	Nominal maturity of a contract
62MVCP	Market Value Change Period as Fixed Key Figure	QUAN	
62N_AGR	Calc: Netting Agreement	STRUCT	Calculation figures for contract: Netting agreement
62ORIGT	Original Maturity as Fixed Key Figure	QUAN	
62PARAM	CE Process Parameter	STRUCTURE	Credit exposure process parameter
62PDV	Probability of default	DEC	
62PRBAL	Total Assets x EAD	CURR	Product of the total assets and the exposure at default
62PREAD	An. Sales x EAD	CURR	
62PRMAT	Product of Residual Maturity in Days and Key Date Drawings	CURR	
62PRTN	Calculation figures for portion	STRUCTURE	

62RAT_B	Bad Rating	STRUCT	Bad rating for financial transaction
62RAT_G	Good Rating	STRUCT	Good rating for financial transaction
62RATBP	Rating: Bus. Partner	TTYP	Business partner rating
62RATCT	Country rating	TTYP	Country rating
62RATFT	Rating: Financial Transaction	TTYP	Financial transaction rating
62RCR	Capital Requirement	CURR	Regulatory capital
62RCRS	Regulatory Capital Requirement for Securitization Exposure	DEC	
62REL	Relationship	STRUCT	Relationship Between Two Contracts
62REL_C	Collateral Relnship	STRUCT	Relationship between contract and collateral
62REL_G	Guarantee Relnship	STRUCT	Relationship between contract and guarantee
62RESLT	Secured/Non-Sec. Portion	STRUCT	Calculation Figures for Secured and Unsecured Parts
62RR_BP	Relevant Rating: BP	STRUCT	Relevant business partner rating
62RR_CT	Rel. Rating: Country	STRUCT	Relevant country rating
62RR_FT	Rel. Rating: Financial Transaction	STRUCT	Relevant financial transaction rating
62RWAA	Risk Weighted Assets After Risk Mitigation	DEC	
62RWAB	Risk Weighted Assets Before Risk Mitigation	DEC	
62RWAS	Risk Weighted Asset in a Securitization Exposure	DEC	
62RWGHT	Risk Weight r	DEC	Risk Weight r for Specialized Lending
62S_AST	Calculation: Asset	STRUCTURE	

62SPECR	Spec. Risk Weight r (special case for risk weight r)	DEC	Risk Weight for Specialized Lending
62UTIL	Drawing	CURR	Drawn Amount
63AVPR	Average Purch. Price	DEC	Average purchase price
63BKSD	Amt. Pld Pnd Elig.Cvd	CURR	Pooled amount pending is eligible for cover
63BKSND	Amt. Pool Pnd Not Elig.	CURR	Pooled amount pending is not eligible for cover
63BKVD	ValPoolAmt Elig.Cov	CURR	Pooled value-dated amount is eligible for cover
63BKVND	ValPoolAmt. Not Elig	CURR	Pooled value-dated amount is not eligible for cover
63BKVNR	PoolValAmt cant reg	CURR	Pooled, settled amount that cannot be registered
63BUSD	NPAmtPend Elig.Cov	CURR	Non-pooled amount pending is eligible for cover
63BUSND	NPAmt NoElig.Cov	CURR	Non-pooled amount pending is not eligible for cover
63BUVD	NPValAmt Elig.Cov	CURR	Settled amount that can be covered and is not pooled
63BUVND	NPValAmt NotElig.Cov	CURR	Settled amount that cannot be covered and is not pooled
63BUVNR	NPValAmt cant reg	CURR	Settled amount that cannot be registered and is not pooled
63EFFZ	Effective interest rate	DEC	Effective interest rate for margin split
63FUSPR	Funding Spread	DEC	
63GKOBT	Total Pooled Amount	CURR	Total pooled amount
63KOBT	Pooled amount	CURR	
63NVF	Net volume factor	DEC	
63OPSPR	Option Spread	DEC	

63OZD	Opp.IR Elig. f.Cover	DEC	Opportunity interest is eligible for cover
63OZND	Opp.IR notElig.Cover	DEC	Opportunity interest is not eligible for cover
63UKOBT	Non-Pooled Amount	CURR	Amount that is not pooled
63ZEROP	Zero Bond Int. Rate	DEC	
69NOM	Nominal value	CURR	Nominal amount (risk basis)
69QUANT	Quantity	QUAN	Quantity (for risk basis)
H0HRITM	HR Items	COMPLEX	Hedging relationship items
H0HRDSS	HR Items	COMPLEX	Hedging relationship data sources with selection criteria
H0HRLE	HR: Legal Entity Data	COMPLEX	Hedging relationship data dependent on legal entity
H0HRVP	HR: Legal Entity Data	COMPLEX	Data for qualified positions

2.3.3.2 Invisible SAP Key Figures

SAP delivers the following fixed (invisible) key figures:

Technical Name	Text	Data Type	Short Description
12AMBAL	Residual balance	CURR	Residual balance of cash flow (current residual balance of the flow)
12AMNOM	Nominal value	CURR	Nominal amount used as the basis for calculating the cash flow amount
12AMOUN	Flow Amount	CURR	Flow amount in a cash flow
12CALCB	No. of Base Days	DEC	Number of base days (in a year) in the calculation period
12CFEFF	Effective Interest Rate	DEC	Effective interest rate of cash flow as a percentage
12CFOPP	Opportunity Int. Rate	DEC	Opportunity interest rate of cash flow as a percentage

12EXRAT	Exchange Rate	DEC	If, in the case of a flow, the nominal amount currency differs from the flow currency, this key figure is the exchange rate - presuming a fixed exchange rate was agreed upon. The following applies: Nominal currency <==> From currency Amount currency <==> To currency. The exchange rate is quoted indirectly. Example of indirectly quoted exchange rate: From currency = DEM To currency = EUR rate = 1.95583. To calculate the EUR amount, the DEM amount in is divided by the exchange rate.
12MARGI	Margin	DEC	Margin of cash flow as a percentage
12NUMD	Days in Calc. Period	DEC	Number of days in calculation period
12PERC	Int. Rate as Percent	DEC	Cash flow: Interest rate as a percentage
12PRATE	Payment Rate	DEC	The payment rate denotes a percentage rate that is applied to the condition amount calculated previously. This means that the cash flow does not show the net present margin that has been calculated. Instead, it shows the condition amount multiplied by the payment rate as a percentage. The reason for this is that exponential interest calculation can produce a variety of results when a payment rate is applied to a percentage rate or an amount.
12VOFF	Offset	DEC	Offset to the reference interest rate: Spread (interest markup and markdown) on reference interest rate in the case of variable templates

55ACVAL	ACon Calc. Date	CURR	Amortized Cost at End of Calculation Date
55FVVAL	Fair Value at End of Calculation Date	CURR	
55KFETB	Table of Key Figures	TTYP	Complex key figure
55RPFAC	Repayment Factor	DEC	Intraday repayment factor
63ABVOL	Asset Balance Volume	FLTP	
63ACNDI	Asset Net Int: IntRat	DEC	Asset-side net interest margin
63ACNDV	Asset Net Int: Volume	FLTP	Asset-side net interest margin: Volume-based
63AGPO	Asset Gap by Maturity	FLTP	Asset gap by maturity evaluation
63AIIRI	Asst IR for NII.:	DEC	Asset-side interest rate for net interest income evaluation
63AIIRV	Asst IR for NII.: Vol	FLTP	Asset-side interest rate for net interest income evaluation: Volume-based
63AIOI	Asset Rate for Mat.	DEC	Asset-side interest rate in maturity evaluation
63AIOKI	Asset OI in KD Pos	DEC	Asset-side opportunity interest in key date position evaluation
63AIOOI	Asset OI in Mat. Eval	DEC	Asset-side opportunity interest rate in maturity evaluation
63AIOOV	Asset OI in Mat.: Vol.	FLTP	Asset-side opportunity interest rate in maturity evaluation: Volume-based
63AIORI	Asset OI in NII.:	DEC	Asset-side opportunity interest rate in net interest income evaluation
63AIORV	Asset OI in NII.: Vol	FLTP	Asset-side opportunity interest rate in interest income evaluation: Volume-based
63ASGAP	Asset Gap on Av. Pos.	FLTP	Asset gap on average position evaluation

63ASGPK	Asset Gap on KD Pos.	FLTP	Asset gap on key date position evaluation
63ASIAI	Asset Int. for AP IR	DEC	Asset-side interest rate for average position evaluation
63ASIAV	Asset Int. for AP Vol.	FLTP	Asset-side interest rate for average position evaluation: Volume-based
63ASIKI	Asset Int. for KD Pos	DEC	Asset-side interest rate for key date position evaluation
63ASIKV	Asset Int.for KDP Vol.	FLTP	Asset-side interest rate for key date position evaluation: Volume-based
63ASIOA	Asset OI for AP Vol.	FLTP	Asset-side opportunity interest rate in average position evaluation: Volume-based
63ASIOI	Asset OI for AP Vol.	DEC	Asset-side opportunity interest rate in average position evaluation: Volume-based
63ASIOK	Asset OI for KDP Vol.	FLTP	Asset-side opportunity interest rate in key date position evaluation: Volume-based
63ASIOV	Asset IR in Mat.: Vol	FLTP	Asset-side interest rate in maturity evaluation: Volume-based
63ASO	Asset Maturity	FLTP	
63ASVAV	Asset Vol. in AV. Av.Pos.	FLTP	Asset-side volume: Average position
63ASVOK	Asset Vol. in KD Pos.	FLTP	Asset-side volume in key date position evaluation
63CNDCI	Net Int. Margin:	DEC	Net interest margin
63CNDCV	Net Int. Margin: Vol.	FLTP	Net interest margin: Volume-based
63DISCF	Disbur. in CF Eval.	FLTP	Disbursement in cash flow evaluation
63DISLQ	Disbur. in Liq. Eval.	FLTP	Disbursement in liquidity evaluation

63FSPD	Funding Spread Cov.	DEC	Funding spread is eligible for cover
63FSPND	Funding Spr. Not Cov.	DEC	Funding spread is not eligible for cover
63IEXP	Interest Expense	FLTP	
63IEXPO	Interest Expense: OI	FLTP	Interest expense: Opportunity interest
63INPCF	Inflow in CF Eval.	FLTP	Incoming payment: Cash flow evaluation
63INPLQ	Inflow in Liq. Eval.	FLTP	Incoming payment: Liquidity evaluation
63IRESA	Abs. Net Interest Income	FLTP	Absolute net interest income
63IRESI	Rel. net int.inc.: IR	DEC	Relative net interest income
63IRESV	Rel. Net Int. Inc.: Vol	FLTP	Relative net interest income: Volume-based
63IREV	Interest Income	FLTP	
63IREVO	Interest Revenue: OI	FLTP	Interest revenue: Opportunity interest
63LBVOL	Vol. of Liabilities	FLTP	
63LCNDI	NIM on Liab.: IntRate	DEC	Liability-side net interest margin
63LCNDV	NIM on Liab.: Volume	FLTP	Net interest margin on liabilities: Volume-based
63LGPO	Liability Gap: Maturity	FLTP	Liability gap in maturity evaluation
63LIGAP	Liab. Gap. in Av. Pos.	FLTP	Liability gap in average position evaluation
63LIGPK	Liab. Gap. in KD Pos.	FLTP	Liability gap in key date position evaluation
63LIIAI	AP Liab. Interest: IR	DEC	Liability-side interest rate for average position evaluation
63LIIAV	AP Liab. Interest: Vol.	FLTP	Liability-side interest rate for average position evaluation: Volume-based

63LIIKI	Liab.Int. KD Pos.: Vol	DEC	Liability interest in key date position evaluation
63LIIKV	Liab. Int. in KD Pos. Vol.	FLTP	Liability interest in key date position evaluation: Volume-based
63LIIOA	Liab. OI for AP Vol.	FLTP	Liability opportunity interest rate in average position evaluation
63LIIOI	Liab. OI for AP Vol.	DEC	Liability opportunity interest rate in average position evaluation: Volume-based
63LIIOK	Liab. OI for KDP Vol.	FLTP	Liability opportunity interest rate in key date position: Volume-based
63LIIOV	Liab. IR in MatEv: Vol	FLTP	Liability side interest in maturity evaluation: Volume-based
63LIIRI	Liab.IR in NII:	DEC	Liability-side interest: Net interest income
63LIIRV	Liab.IR in NII: Vol	FLTP	Liability interest in net interest income evaluation: Volume-based
63LIO	Liab.-Side Maturity	FLTP	
63LIOI	Liab. Mat. Eval	DEC	Liability side interest in maturity evaluation: Volume-based
63LIOKI	Liab. OI for KD Pos.	DEC	Liability opportunity interest rate in key date position
63LIOOI	Liab.OI in Mat. Eval	DEC	Liability opportunity interest rate in maturity evaluation
63LIOOV	Liab. OI in Mat. Ev.: Vol.	FLTP	Liability opportunity interest rate in maturity evaluation: Volume-based
63LIORI	Liab. OI in NII:	DEC	Liability opportunity interest rate in net interest income
63LIORV	Liab. OI in NII: Vol	FLTP	Liability opportunity interest rate in net interest income: Volume-based

63LIQGP	Liquidity Gap	FLTP	
63LIVAV	Liab. Vol. in Av.Pos.	FLTP	Volume of liabilities in average position evaluation
63LIVOK	Liab. Vol. in KD Pos.	FLTP	Liability-side volume: Key date position
63LQGPC	Cumulative Liq. Gap	FLTP	Cumulated liquidity gap
63MZD	Market IR Cover Elig	DEC	Market interest rate is eligible for cover
63MZND	Market IR Not Cover	DEC	Market interest rate is not eligible for cover
63OSPD	Option Spread Cover	DEC	Option spread is eligible for cover
63OSPND	Option Spr. Not Cover	DEC	Option spread is not eligible for cover
63PAYGP	Payment Gap	FLTP	
63PRESV	Net Present Value	FLTP	
63PVMB	NPV in Maturity Band	FLTP	NPV in maturity band
63STRCI	P/L Mismatch Spread	DEC	
63STRCV	P/L from MismSp: Vol	FLTP	P/L from mismatch spread: Volume-based
69PVCR	Complx KF Res. PrCal	TTYP	Complex key figure table: Price calculator result
69TSSR	CKF Risk Basis (Tab)	TTYP	Complex key figure table for the risk basis

2.3.4 Custom Key Figures

Definition

You define custom key figures in the core Bank Analyzer system. The properties of the key figures come from the assignment of a key figure class [page 74].

Use

You use custom key figures in the same way as you use other key figures in Bank Analyzer. For more information, see General Information About Key Figures [page 72].

Before you use the custom key figures, you have to do the following:

1. Enter an InfoObject catalog for custom key figures in Customizing for *Bank Analyzer* by choosing *Basic Settings* > *Settings for Meta Data* > *Edit Global Settings for Characteristics and Key Figures*.
2. Create your own key figures in Customizing for *Bank Analyzer* by choosing *Basic Settings* > *Settings for Meta Data* > *Key Figures* > *Edit Key Figures*.
3. Activate the key figures that you have created.

> **Note**
>
> When you define custom key figures, they are automatically activated. InfoObjects are generated in the Tool BI system under the same name as the elementary key figure, and entered in the specified InfoObject catalog (IOC). If the InfoObject catalog for custom key figures does not yet exist, the system creates it automatically in the Tool BI system.

Key Figure Monitor

You use the key figure monitor for status monitoring and mass processing of custom key figures, and optionally for visible SAP key figures [page 76].

You call the key figure monitor in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Meta Data* > *Key Figures* > *Key Figure Monitor*. In the key figure monitor you can:

- Generate an overview of the status of the key figures
- Display key figure details
- Activate one or more key figures
- Transport individual key figures or all key figures (the activation for the transport must be made in the appropriate client).
- Run detailed checks for the key figures with a log

2.4 Connection to the Garbage Collector

The repository objects mentioned below can be used in the .

Use of the Garbage Collector

Generating Object	Generated Object
Fixed characteristic	InfoObject
Local characteristic	InfoObject

Generating Object	Generated Object
InfoObject	Characteristic
Custom key figure	InfoObject for key figure (for unit)
SAP key figure	InfoObject for key figure (for unit)
	Data element for key figure (for unit)
Technical field	InfoObject
Segment hierarchy	InfoObject
RFC destination for Reporting BI	RFC destination *(SM59)*

> **Note**
>
> If the object generated is an InfoObject, the Garbage Collector tells you that the object is no longer used. You then delete it manually in the Tool BI [page 16].

2.5 Settings for Accounting

To use the Accounting Processes and Cost Accounting Processes in Bank Analyzer, you need to enter Customizing settings first. For example, you have to define which accounting scenario you are using, and configure the legal entities and organizational units. You must also define a fiscal year variant and assign it to an accounting system.

The following scenarios are possible:

- Subledger Scenario
  The subledger scenario requires settings in multiple layers in Bank Analyzer. You enter the settings in Customizing for Bank Analyzer under:
  - *Bank Analyzer* > *Basic Settings* > *Settings for Accounting*
  - *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products*

  You can find the activities for accounting on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products*.
  For more information, see Accounting Processes [page 693].
- Merge Scenario (Balance Analyzer)
  You can find the Customizing activities for the merge scenario in Customizing and on the *SAP Easy Access* screen under *Bank Analyzer* > *Analytics* > *Accounting: Merge Scenario*.

The decision about which scenario you use is client-dependent and cannot be changed. You make this setting in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Accounting* > *Choose Accounting Scenarios for Clients*.

See also:

Bank Analyzer (FS-BA) [page 8]

2.6 Blocking and Deletion of Personal Data in Bank Analyzer

Use

The processes and methods in Bank Analyzer use personal data of the analytical business partner. If there is no longer a reason for using this data, it needs to be blocked. Once the data is blocked, only users with additional authorization (authorization object `B_BUP_PCPT`) can access this data.

In the first step, you block the personal data of business partners for which the business purpose has lapsed. In a second step, you delete this data as soon as it is no longer needed:

1. Block
   The blocking of data can impact system behavior in the following ways: The system either does not display the blocked data, displays it in a masked form, or cancels processing.
   Personal data is blocked in the following ways:
   - If you have made the necessary Customizing settings (see *Customizing*), the system blocks the affected data by masking the display of field values or by performing checks during contract creation, for example.
   - During archiving by SAP Information Lifecycle Management (ILM), the system blocks personal data by archiving the data from the Bank Analyzer system.
2. Delete
   You use the functions of SAP ILM to delete personal data:
   - ILM archiving objects
     You can use ILM archiving objects to delete data from the archive.
   - ILM destruction objects

   > **Example**
   >
   > You can use the transaction *Destroy Process Controller Data* to delete data records from the Process Controller's status management for Smart Accounting for Financial Instruments.

Customizing

You make the Bank Analyzer-specific settings for blocking personal data in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Meta Data* > *Block Personal Data*.

The checkbox relating to the completion of the business purpose (`BU_XPCPT`) must also be selected in the analytical business partner. The setting for this checkbox must be transferred to the analytical business partner.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]
Archiving and Destruction of Business Partner Data Using ILM [page 232]
Destroy Process Controller Data [page 1818]
SAP Note 2504648

3 Source Data Layer (FS-BA-SD)

Use

You use this component to manage original data for the Bank Analyzer system.

The system uses the Data Load Layer [page 1581] component to load original data from other operational systems or source systems into the Source Data Layer (SDL) by means of an extraction, transformation, and loading process (ETL process). The SDL saves, consolidates, and manages the original data. At the same time it provides interfaces to additional operational systems.

The primary objects [page 100] of the Source Data Layer (SDL) and their scenario versions are a flexible way of saving master data and flow data. They also group this data into units that belong together logically from a business perspective. This ensures that the Bank Analyzer components that are linked to the SDL have a standard, consistent data source.

In addition to storing primary object data, the SDL provides the following functions for primary objects:

- Access to Source Data [page 265]
- General Functions for Source Data [page 272]
- Methods for Source Data [page 280]
- General Access to Corrections [page 329]
- Tools [page 334]

Integration

The SDL provides both the central original data basis and a part of the underlying infrastructure for linked applications. It is therefore a key element in ensuring the consistency of data and results.

3.1 Primary Objects

Definition

The data that is transferred to the Source Data Layer from the various source systems in your bank is managed and processed in the primary objects. This data can then be made available in a standard and consistent manner to the various Analyzers for the performance of the evaluations and processes. The various primary objects are categorized as follows.

Primary Object	Data Type	Development Tool	Fundamental Customizing Unit	Can Scenario Be Created?	Data Import Buffer Available?
Financial Transaction [page 122]	Master data	Master data framework	Template	X	X
Financial Instrument [page 129]	Master data	Master data framework	Template	X	X
Account [page 134]	Master data	Master data framework	Template	X	X
Country [page 137]	Master data	Master data framework	Template	X	X
Organizational Unit [page 251]	Master data		Template	--	--
Position [page 142]: Position items Position master data	Flow data Master data	Master data framework	Position class Template	X	X
Physical Asset [page 140]	Master data	Master data framework	Template	X	X
Business Partner [page 161]	Master data			X	--
Business Transaction [page 153]	Flow data		Business transaction class	--	X

Versioning

For version management reasons, all of the versions of primary objects are stored on the database. For more information, see Versioning of Primary Objects [page 103].

Automatic Data Processing

In the live system, the primary objects are usually provided with data automatically via BAPIs. For this purpose, the information relevant for Bank Analyzer is extracted from and provided by the source systems.

For reasons of performance, the **delta approach** must be adopted for automatic data processing. This means that the only data transferred is the data pertaining to templates where changes were actually made. No data is

imported to the Source Data Layer for objects that have not been changed. If an object's data has been changed, all of the current data for this object has to be provided again.

Transaction BAPI under *Financial Services* > *Bank Analyzer* contains the BAPIs and their corresponding documentation.

As of Bank Analyzer Release 5.0, you can also use the Data Load Layer [page 1581] to load data. The Data Load Layer uses the SAP NetWeaver Business Intelligence infrastructure.

Manual Data Processing

The process of manually editing data for the primary objects using appropriate transactions is an exception and is largely restricted to making corrections.

Archiving

You can archive the primary object versions. For more information, see Archiving of Primary Objects [page 279].

Scenarios

For certain primary objects (see above table), you can also specify fictitious data versions (scenarios) for the real versions of the primary objects. For more information, see Scenario [page 1731].

Data Import Buffer

To ensure a quality check of the data that is imported into the Source Data layer (before other Bank Analyzer processes access the primary objects), you can first load the primary objects of the master data framework and the business transactions to a data import buffer. You do this by means of the appropriate parameters of the BAPIs. This data is then temporarily unavailable for other Bank Analyzer processes. After the quality check has run, you have to activate or release the objects in the data import buffer for the other processes using the relevant function modules. For the master data objects, you do this using transaction `/BA1/FP_DIB_REP`. For the business transactions, you do so in the Evaluations for Data Import Buffer [page 159] transaction (`/BA1/BT_DIB_REP`).

3.1.1 Primary Object Versioning

Use

The versioning concept works on the principle that a new version of a primary object [page 100] is automatically created when changes and corrections are made to the object in question. Existing versions of this object (even if these are incorrect) cannot be overwritten.

The versioning of the primary objects provides you with a constant overview of the historical changes and corrections.

Integration

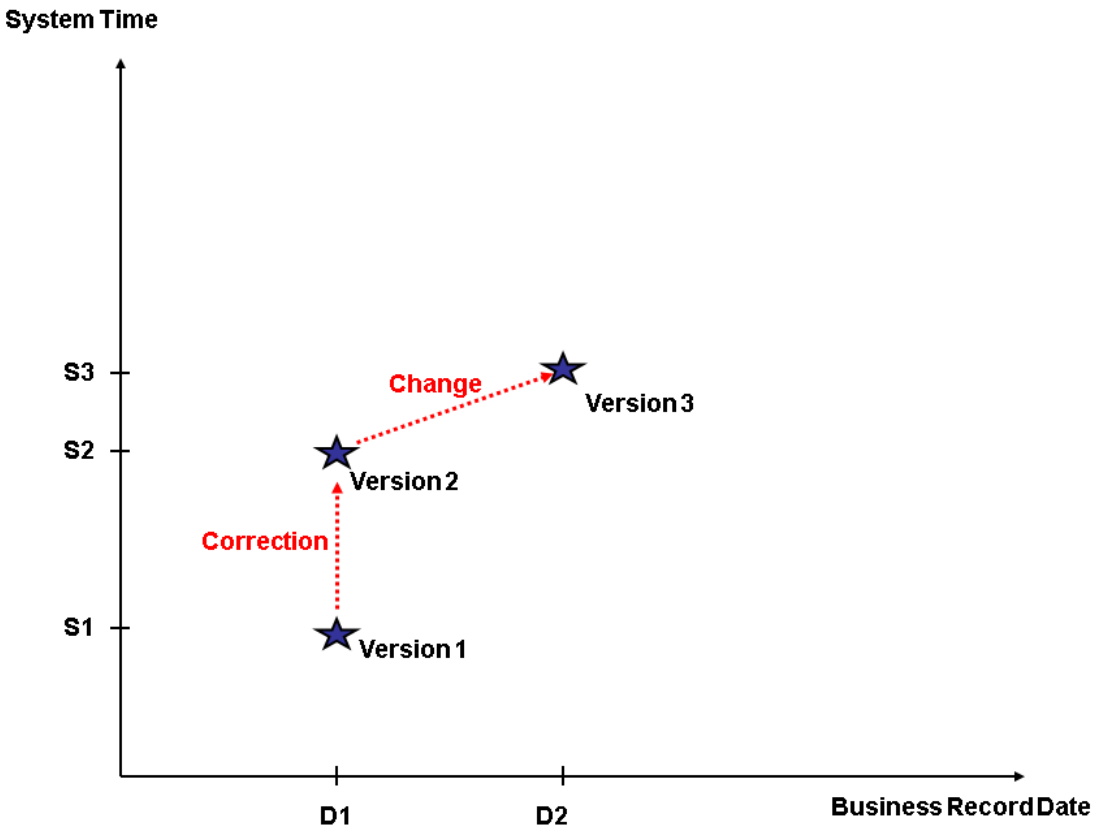
During the evaluation process, the various Analyzers access **exactly one version** of the primary objects that are to be analyzed. The Analyzer does not need to know the other versions.

Features

The system accesses a specific version of a primary object using the following two dates/times:

- Business Record Date
- System Time (time of creation/change in the system)

This system uses this date/time information to version all data of a primary object. The external number and the authorization group [page 277] are the exceptions here.



The figure shows three versions of a primary object. Version 1 was generated for business date D1. This version is incorrect. The correction of the primary object also occurs on business date D1. The system generates version 2. As a result of changes to the conditions for business date D2, a new version 3 is generated.

Selecting a Version

Requests from primary objects must contain both a required business record date and a required system time. The version of the primary object that the system selects is valid for the required key date and was last corrected before the required system time. The system proceeds as follows:

1. The system searches for all the object versions that meet the following condition: business record date <= required key date **and** system time <= required system time.
2. The system selects from the remaining versions the versions that have the latest business record date.
3. From the remaining versions with the latest business record date, the version with the latest system time is selected.

> **Note**
>
> An object can contain time-dependent data. This data is stored in several data records for periods that do not overlap. In cases where only flat data records can be processed, the data record that is valid on the specified business date is the only data record that is selected.

Correcting Versions

If you want to make corrections, you have to make them separately for each version that is affected. The system does not provide a function for correcting several versions simultaneously. If an object version that was valid in the past is corrected, the correction is valid only until the next business change.

Example

A loan concluded on 1 February is extended twice, on 1 March and on 1 May. It is later established that the data for the versions valid as of 1 March and 1 May contain errors. A user now has to find the version dated 1 March and correct it. This correction is, however, valid only until 30 April. The version valid as of 1 May still contains errors. To correct these errors too, the user also has to change the version valid as of 1 May.

The user can change both the validity date and the contents of the object version. If you correct the validity date of an object, you will then have to manually edit any versions dependent on this validity date.

Example

In the above example, it is established that the loan was first extended on 10 March and not on 1 March. This means that the validity date of the version valid as of 1 March first has to be changed to 10 March. An object version valid between 1 and 9 March still remains. The object data from the version dated 1 February must be adapted for this version, since it was not known until 9 March that the loan had been extended.

Special Features for Versions of Positions

The position [page 142] contains values for a key date. If you change the key date, the system generates a new position and not a new version of the same position. For this reason, positions contain only the changes for one particular key date.

Special Features for Versions of Business Partner

See also: Business Partner Versioning [page 209]

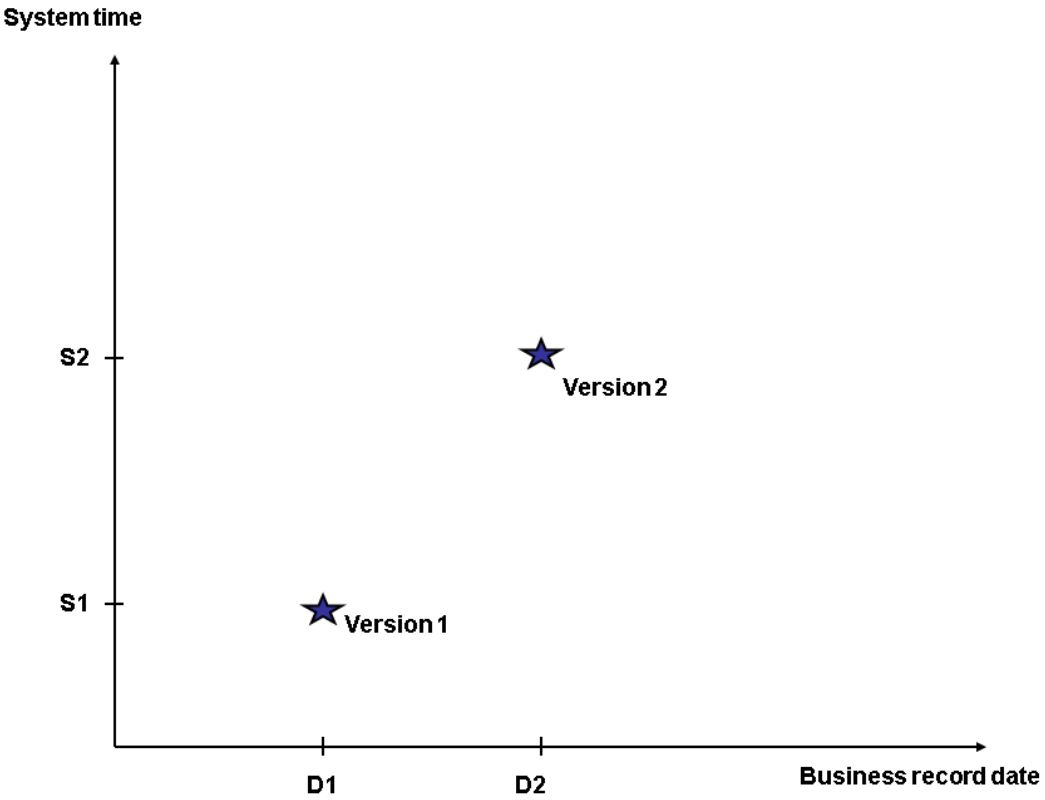
Versions Using Scenarios

If you have created scenarios [page 1731] for the primary objects, the system additionally generates separate scenario versions for a business record date.

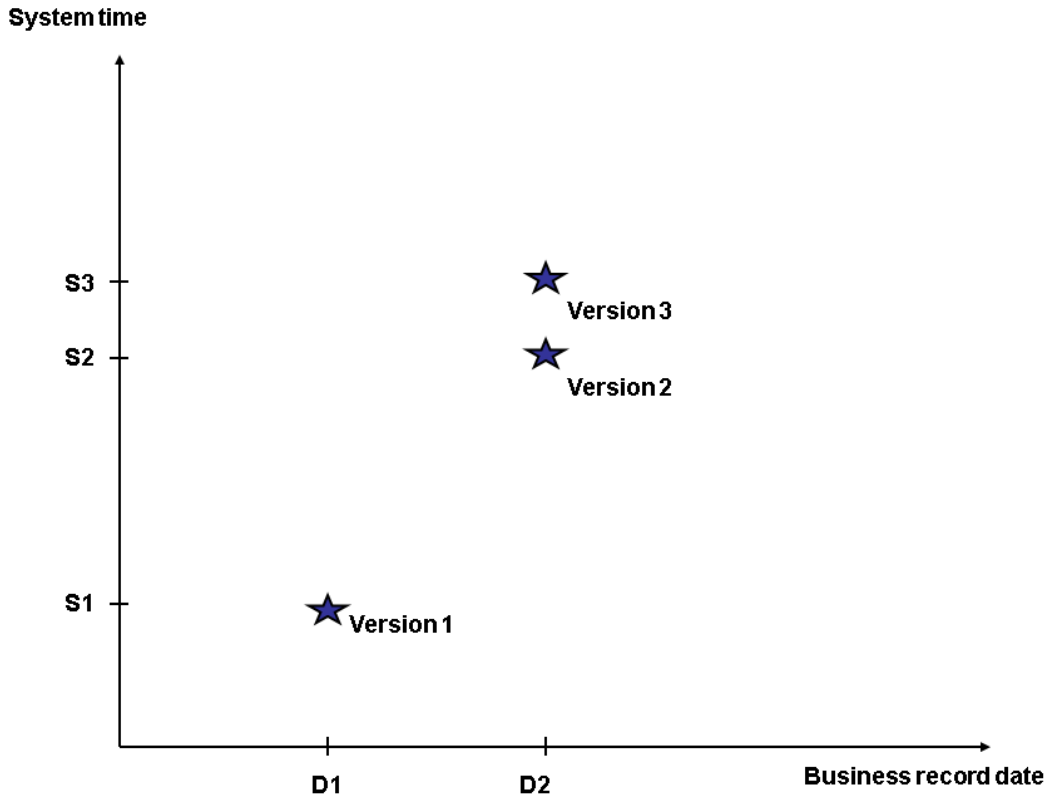
3.1.1.1 Example: Versioning

In this example, the versioning concept is demonstrated on the basis of a loan object (financial transaction) that is processed over a specific period of time. The valid version with the required correction status can be selected for the required key date on the basis of the two time axes "Business Record Date"and "System Date".

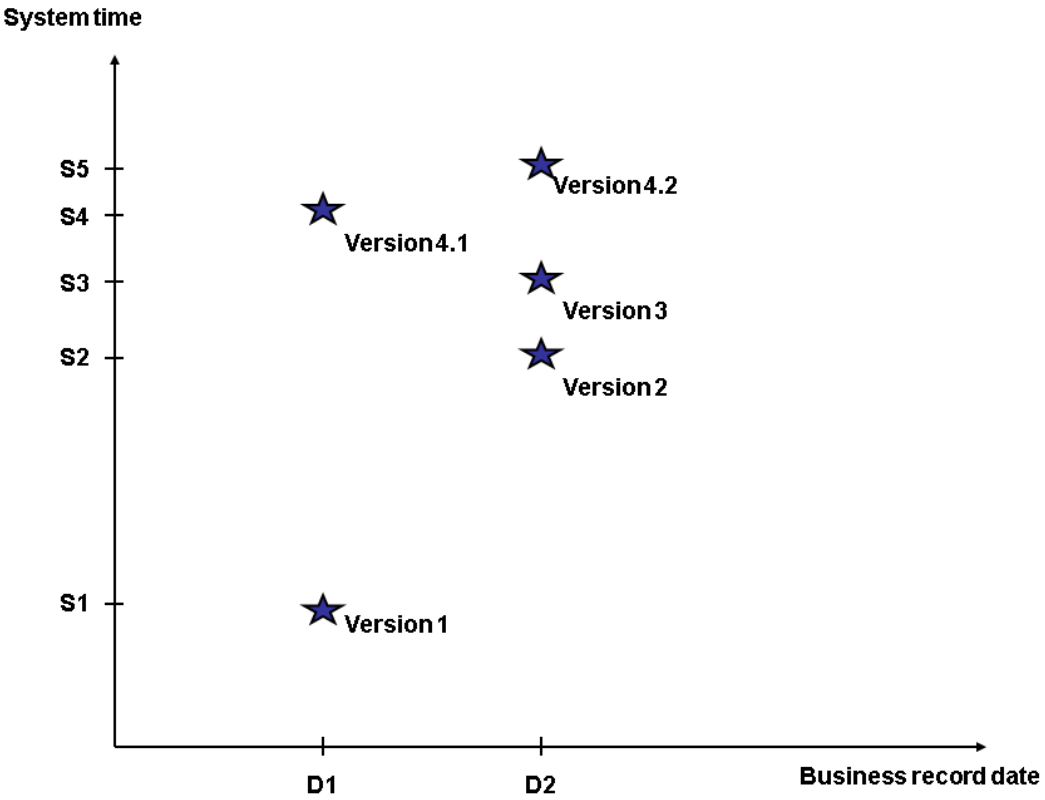
- Version 1:
  The loan contract is concluded on the business record date D1. The nominal interest rate is 6%. The loan contract is entered in the system at system time S1.
- Version 2:
  A change is made to the loan contract: as of the business record date D2, the interest rate is 7%. The change is entered in the system at system time S2.



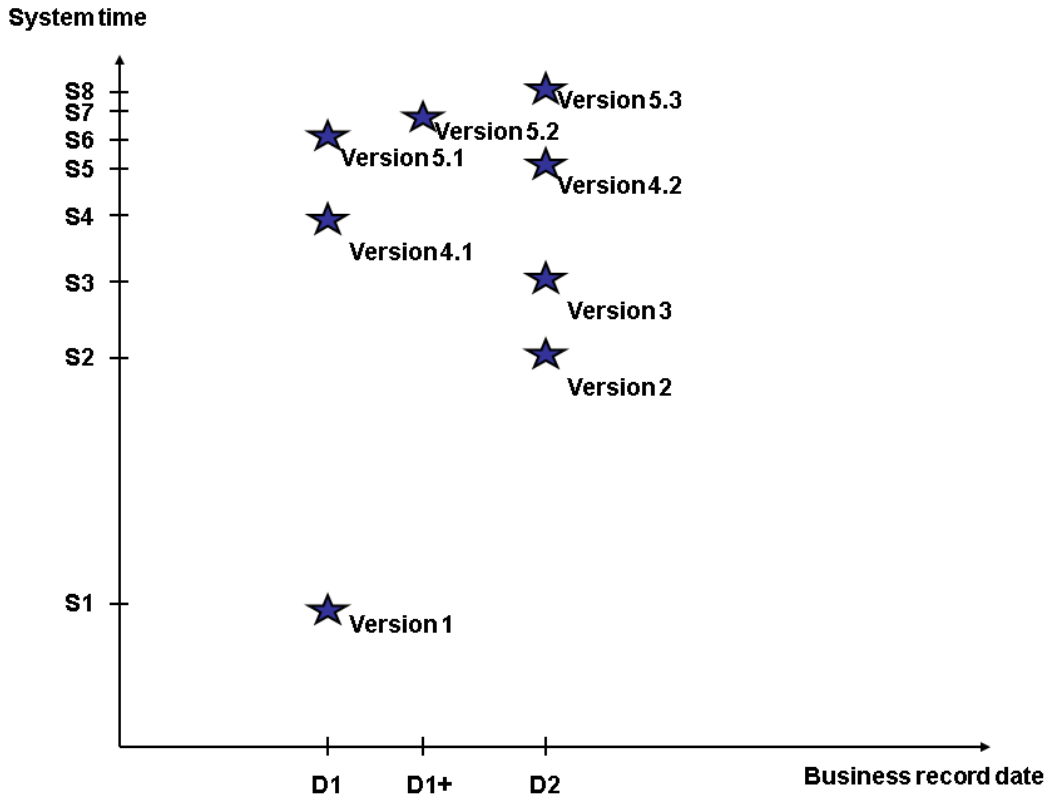
- Version 3:
  It is established that the interest rate of the version valid from D2 is actually 8%, and not 7%. The correction is entered in the system at system time S3.



- Versions 4.1 and 4.2:
  It is established that the interest rate in the first period was 5%, and not 6%. Since this correction concerns the periods D1 and D2, two individual corrections have to be made for this correction. The corrections are entered in the system at system times S4 and S5, since only one version can be recorded in the system at any given time.



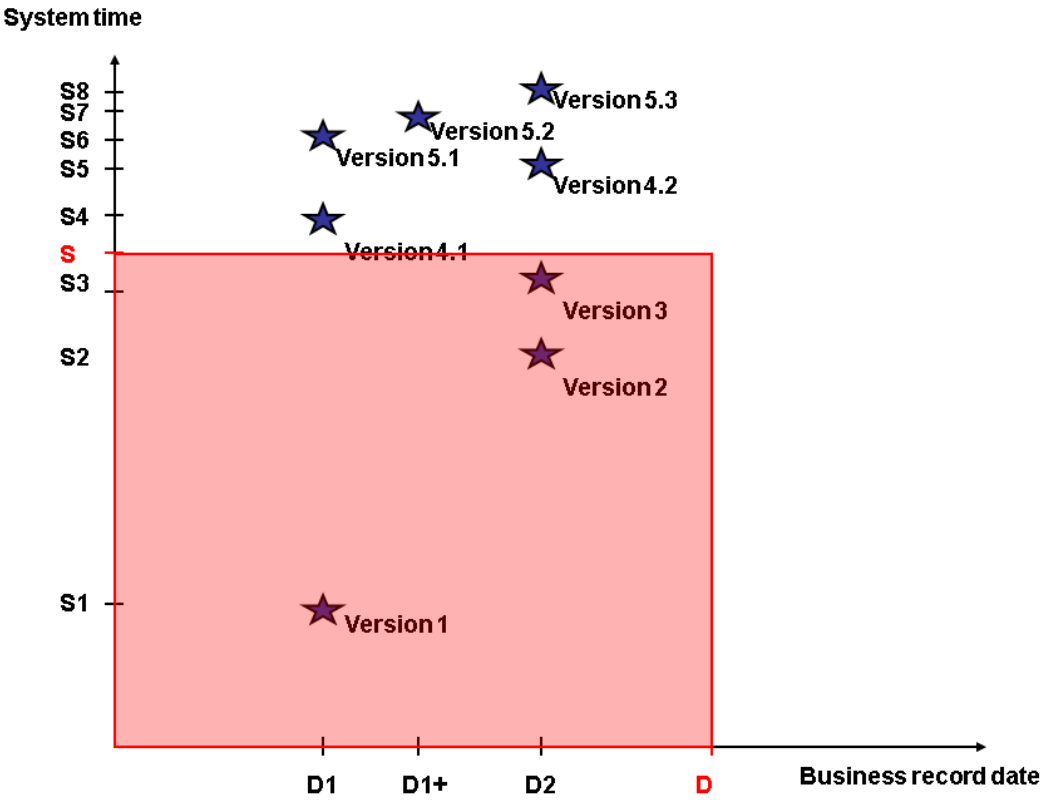
- Versions 5.1, 5.2, and 5.3:
  The conclusion date D1 of the loan contract should have actually been D1+ (>D1). Since this correction concerns the periods D1 and D2 and a business record date is changed as a result, three individual corrections have to be made for this correction. The corrections are transferred to the system at system times S6, S7, and S8.



Selection on the Basis of the Business Date and System Time

A required version contains the date fields D > D2 (business record date) and S3 < S < S4 (system time). The system retrieves this version by carrying out the following selection steps:

1. All the object versions that meet the following condition are selected: business record date <= required key date and system time <= required system time. Versions 1, 2 and 3 remain.



2. In the versions that remain, it finds the version for which the business record date is the maximum value. Versions 2 and 3 remain.
3. In the versions that remain, it finds the version for which the system creation time has the maximum value on the time axis; this is the version with the last relevant correction. The system provides version 3.

3.1.2 Primary Objects of the Master Data Framework (MDF)

The following objects are mapped using the master data framework:

- Financial Transaction [page 122]
- Financial Instrument [page 129]
- Account [page 134]
- Country [page 137]
- Position master data [page 152] (part of the primary object Position)
- Physical Asset [page 140]

3.1.2.1 Basic Principles of the Master Data Framework (MDF)

The following objects are the basis for working in the master data framework:

- Template Category [page 111]
- Template [page 112]
- Template Version [page 117]
- Service Modules [page 118]

3.1.2.1.1 Template Category

Use

You use template categories to classify templates. You assign template categories to templates in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Master Data* > *Templates* > *Edit Template*.

The various template categories are used as follows:

- *Financial transaction*
  You use this template category to map, for example, personal loan contracts, OTC contracts, and facilities.
- *Financial instrument*
  You use this template category to map, for example, contracts for listed securities, and security classes.
- *Accounts*
  Various template categories are available for mapping accounts because customers want to be able to access different positions and also to reference different financial transactions and instruments via an account.
- *Settlement account*
  You can use this template category to map accounts that are used to process one or more financial transactions.
- *Securities position account*
  You use this template category to map accounts that are used for processing a bank's own-account securities trading. You can use these accounts to process own-account trading for one or more security classes.
- *Customer position account*
  You can use this template category to map accounts that record the positions pertaining to a customer. These positions can but do not have to be associated with a financial transaction. An example of a customer position that is associated with a financial transaction is commission for the loans extended to the customer. Examples of customer positions that are not associated with a financial transaction are unit costs for customer support, unit costs for commitments, and accounts that are provided for netting positions.
- *G/L account*
  You use this template category to map accounts that act as internal accounts within a bank (for tangible assets, for example). A G/L account is not associated with any financial transactions or financial instruments.
- *Country*
  You use this template category to map country-specific data, such as ratings.
- *Physical asset*
  You use this template category to map non-financial objects that can be used as collateral in a collateral agreement. An example of a physical asset is a piece of real estate.
- *Position master data*
  You use this template category to map position master data (as part of the primary object position).

- *Aggregation object*
  You use this template category to map aggregated financial transactions.

3.1.2.1.2 Template

Definition

A template defines the values that are allowed for an object of the master data framework. In Customizing, the template is used to determine the attributes available for the primary object.

If several templates are hierarchically organized, structured templates can be created. A structured template consists of several subtemplates. An elementary template, in contrast, is a template that does not contain any subtemplates.

Use

You use templates to create and edit individual primary objects of the master data framework. A template determines which attributes can be assigned to a given object, and what the properties of these attributes are. You have to define the template in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Master Data* > *Templates* and configure a template version.

> **Example**
>
> For example, you can edit a swaption in the transaction *Create Financial Transaction* only if you have already defined the swaption as a template, and configured a template version for this template.

The primary objects are affected by settings that involve templates:

- **Attributes**
  For more information about the various attributes, see Service Modules [page 118].
- **Segmentation Service**
  In template configuration, you can use a characteristic structure to add characteristics that have been defined in the Tool BI system to a template version. You create characteristic structures in Customizing for *Bank Analyzer* under *Basic Settings* > *Segmentation Service* > *Edit Characteristic Structure*. The characteristics that you use to map financial transactions, financial instruments, and accounts are grouped together in the *Attribute Assignment* service module.

Structure

You can create template hierarchies by combining various templates. Template hierarchies show the hierarchical relationships between templates and their subtemplates.

You require product hierarchies for creating financial transactions (such as swaptions) and in particular for editing financial instruments (such as collared floating rate notes). You cannot save a financial transaction, financial

instrument, or physical asset for a structured template unless you have complied with the settings defined for the template hierarchy. This means that if a template contains subtemplates with a minimum cardinality of 1 or higher, then any financial transaction, financial instrument, or physical asset that you create for the subtemplates must also contain subobjects. For more information, see Example: Structured Template (Swaption) [page 113].

You define template hierarchies in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Master Data* > *Templates* > *Template Hierarchy Categories* > *Edit Template Hierarchy*.

> **Note**
>
> You can only create template hierarchies for templates of the following template categories: *Financial Transaction*, *Financial Instrument*, and *Physical Asset*. You can only create a template hierarchy with templates of the same template category.

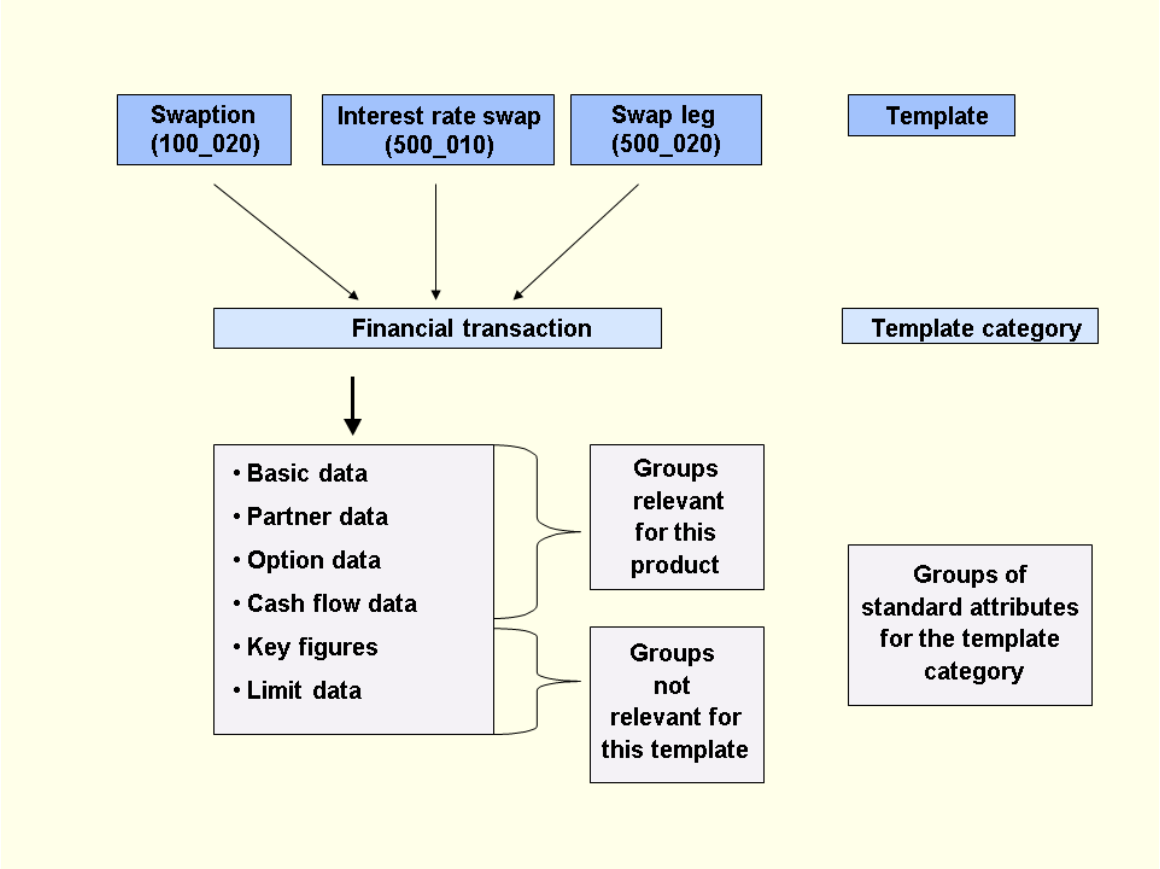
> **Note**
>
> Note that for technical reasons you can create only a maximum of 32767 nodes for a version of a financial transaction/financial instrument/physical asset within a template hierarchy. Regardless of this technical restriction, we recommend that you carefully consider the number of hierarchy nodes and check the performance of your system regularly during the implementation phase because the evaluation of master data with large template hierarchies is very performance intensive.

3.1.2.1.2.1 Example: Structured Template (Swaption)

The swaption is a structured template that consists of three templates: the swaption (option on an interest rate swap), the interest rate swap, and the two swap legs, which are used to map the two swapped interest payments.

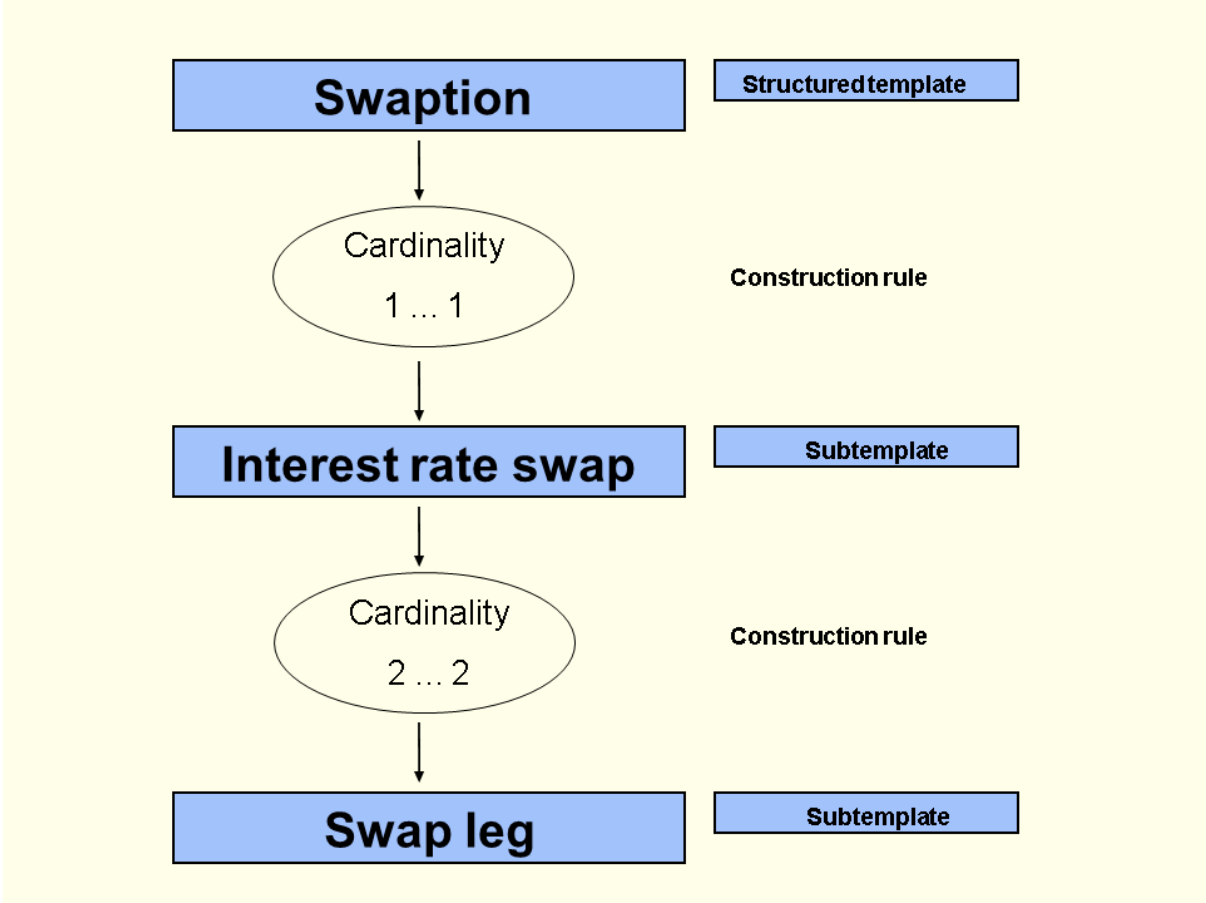
1. When you define the three templates in Customizing, select Financial Transaction as the template category to ensure that all the groups of standard attributes are available for this template category.
   All three templates are defined as elementary templates:
   - `100_020`: Option on interest rate swap
   - `500_010`: Interest rate swap
   - `500_020`: Leg of the interest rate swap

The above scenario can be illustrated as follows:



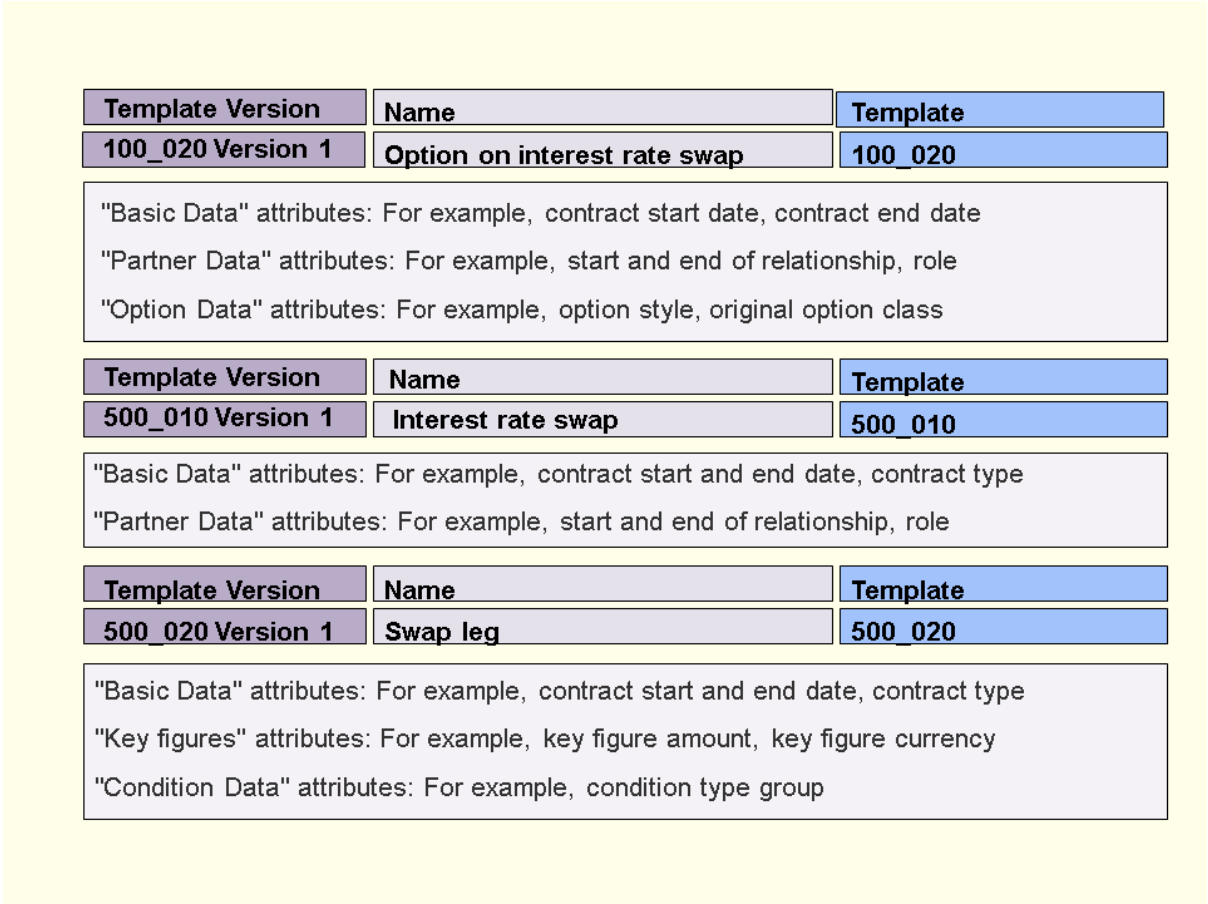
2. The elementary template Interest Rate Swap is the underlying transaction of the swaption. There must, therefore, be an assignment between the two templates. In Customizing, you, therefore, define a template hierarchy between the swaption and the interest rate swap by setting the elementary template Interest Rate Swap as a subtemplate of the swaption. The cardinality defines that exactly one interest rate swap must exist/be created for a swaption.
   The template Interest Rate Swap, in turn, consists of the two swap legs. In Customizing, you therefore extend the template hierarchy of the swaption to include a hierarchical relationship between the interest rate swap and the swap legs. In this relationship, the cardinality defines that exactly two swap legs must exist/be created for an interest rate swap.

These relationships are illustrated in the following diagram:



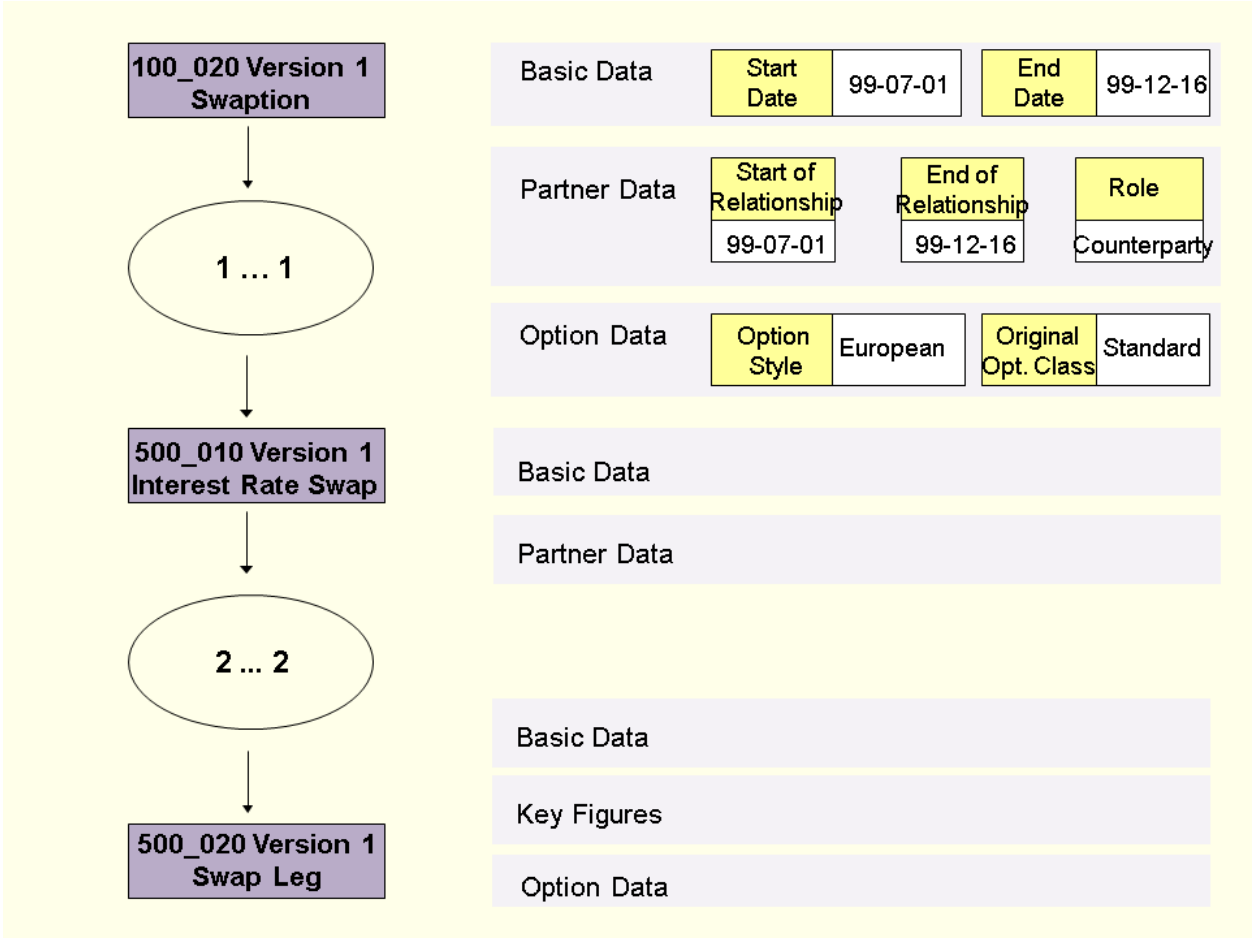
3. Once you have defined the templates and template hierarchy, you configure the relevant template versions for all templates. You select and activate the standard attributes required for mapping the swaption, interest rate

swap, and the two identical swap legs from the attribute groups of the template category financial transaction. This is illustrated in the following diagram:



4. Once you have created the template versions of the swaption, interest rate swap, and swap legs, you can use the template to create financial transaction 4711, which consists of a swaption, the associated interest rate swap, and the swap legs, as well as to edit the associated attribute values.

Financial Transaction 4711:



3.1.2.1.3 Template Version

Definition

A template version contains the actual settings for a template. In particular, template versions are where you make settings for individual service modules, and they can contain a structure of the segmentation service. A template [page 112] can have one or more template versions.

Prerequisites

If you wish to use the segmentation service, you must have set up a suitable structure in Customizing for *Bank Analyzer* > *Basic Settings* > *Segmentation Service* > *Edit Characteristic Structure*.

If you want to use customer-specific validations, you must have set up the relevant checks in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *General Functions for Source Data* > *Validations* > *Edit Environments and Modules for Customer-Specific Validations*.

Use

You define a template version for a template in Customizing for *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Master Data* > *Templates* > *Edit Template Version*. This is where you decide what the settings for the various attributes [page 118] should be. From the total quantity of attributes available for the template category in question, you activate the attributes that are relevant for objects of this template.

The following settings can be available for each attribute:

- Activation
  Only those attributes that you activated when you configured the template version are available for editing a specific primary object.
- Field selection control
  Determines whether an attribute is a *required entry field*, *optional entry field*, or a *display field* when you edit a primary object.
- Default values
  When you edit a primary object, you can specify that the system should propose a default value for the attribute.
- Allowed value set
  Restricts the values of an attribute to a specific range when you edit a primary object.

You can also assign a structure of the segmentation service to a template version, and use the *Grouping* and *Characteristic Profiles* fields to specify this in more detail. If required, you can also assign validation modules.

With these settings, the template version determines whether a financial transaction, financial instrument, account, country, hedging relationship, or physical asset is consistent.

> **Note**
>
> Do not change or delete a template version if there is any application data for the objects of this template version. If you do, these existing objects can become inconsistent, and you will not be able to display or edit them. Recommendation: If you really have to change a template, create a new template version by copying the old template version. You can then change the new template version. When you create new objects in the application, use the new template version. You can change the template version accordingly for existing objects.

3.1.2.1.4 Service Modules

Definition

Service modules are used to manage the concrete attribute values for financial transactions, financial instruments, accounts, countries, hedging relationships, and physical assets. In addition to the management of the attribute values, service modules run consistency checks for a fixed number of related business attributes.

Use

Before you can create a master data object, you have to create a relevant template and template version. In the template version, you specify the attributes that you want to use for the object. You assign these attributes in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Master Data* > *Templates* > *Template Configuration* > *Configure Template Version*.

In the application itself, the service modules are used to structure the attributes of a primary object. The service modules are divided up on the basis of the part they play in the relevant business processes. Each attribute belongs to one, and only one, service module.

The following service modules are provided:

Service Module	Use	Example
Attributes	This service module contains the data for the segmentation service. See Template [page 112].	
Position Management for Balance Analyzer	The attributes of this service module are used for accounting and balance sheet valuation purposes. See Generation of Financial Position Objects for Structured Products [page 750].	
Flow Data	The attributes of this service module enable you to store time-dependent ratings for different rating models.	A rating comprises a rating model, the relevant rating grade, and the period for which the rating applies.
Credit Exposure Data	The attributes of this service module are used for analyzing the object in Credit Exposure.	
Financial Conditions	The attributes of this service module enable you to manage for a financial transaction or financial instrument in a time-based manner. Financial conditions are used for generating cash flows.	Financial conditions are chiefly interest rate conditions, repayment conditions, and conditions for charges.
Basic Data	The attributes of this service module are general attributes that are used for mapping financial transactions, financial instruments, and physical assets.	The transaction type and object currency are pieces of basic data.
Instrument Data	The attributes of this service module are specific attributes that are only used for mapping financial instruments.	You can use the instrument data *index base* and *index-based currency* to map a stock index, for example.

Key Figures	You can use the attributes of this service module to define time-dependent amounts, quantities, and percentages in a key figure category for a financial transaction, financial instrument, country, hedging relationship, or physical asset. You cannot use these attributes for accounts.	You can define the nominal volume of a EUR 20,000 loan as amount information using the key figure category *Nominal Volume*.
Basic Account Data	The attributes of this service module are specific attributes that are only used for mapping accounts. Basic account data applies to all account categories.	The account currency and account opening date are pieces of basic account data.
Limit data	The attributes of this service module enable you to map various time-dependent credit limits for a financial transaction or subtransaction.	You can use general limit data to map the internal and external limits of a checking account. You define these credit limits for specific periods of time.

Object Relationships	The attributes of this service module enable you to map various time-dependent relationships between individual financial transactions, financial instruments, accounts, hedging relationships, and physical assets. The relationships are for the various relationship categories are all time-dependent.	For example, you can map a facility, its drawings, and the collateral provided for the facility. A lender and a borrower agree upon the general conditions for drawings against a credit line. They agree upon a total volume of EUR 300,000. This facility (FAC) is entered as an independent financial transaction. The borrower then draws against the credit line three times (D1 to D3). In addition, collateral (C1) is provided for drawing D3. **D1** • Start of relationship: 2000-01-01 • End of relationship: 9999-12-31 • Relationship category: DRAW (A is a drawing for B) • External number of target object: FAC **D2** • Start of relationship: 2000-03-01 • End of relationship: 9999-12-31 • Relationship category: DRAW (A is a drawing for B) • External number of target object: FAC **D3** • Start of relationship: 2000-05-01 • End of relationship: 9999-12-31 • Relationship category: DRAW (A is a drawing for B) • External number of target object: FAC • Start of relationship: 2000-05-01 • End of relationship: 9999-12-31 • Relationship category: COLL (B is collateral for A) • External number of target object: S1
Option Data	You can use the attributes of this service module to map time-dependent option data for financial transactions and instruments.	You can use the attributes to map a strike, an option classification, or a subscription ratio.

Partner Data	The attributes of this service module enable you to map time-dependent relationships between business partners in specific roles.	For a loan agreement with a term of 10 years, relationships with a time limit can exist for different business partners in different roles. • Start of relationship: 2001-01-01, end of relationship: 2010-12-31, role: borrower • Start of relationship: 2001-01-01, end of relationship: 2010-12-31, role: guarantor
Currencies	You can use the attributes of this service module to map currencies classified by currency class in a time-dependent manner.	You can use currencies to map the national currency of a country. You define this national currency for a specific period of time.
Cash flow data	The attributes of this service module enable you to map cash flows, such as charges and interest payments or repayments, for a financial transaction or financial instrument. The individual flows are grouped into one cash flow. Note that cash flow data involves only planned values.	On December 1, 2001, USD 1 million is purchased at an exchange rate of USD 0.9905 to EUR 1; charges of EUR 100 are incurred on December 5, 2001. This gives rise to the following flows: • Payment date: 2001-12-01, flow type: purchase of currency, amount: 1,000,000, currency: USD • Payment date: 2001-12-01, flow type: sale of currency, amount: 1,009,591.12, currency: EUR • Payment date: 2001-12-05, flow type: fees/commission amount: 100, currency: EUR

> **Note**
>
> Service modules with time-dependent data can contain several data records for periods that do not overlap. In cases where only flat data records can be processed, the data record that is valid on the specified business date is selected.

3.1.2.2 Primary Object: Financial Transaction

Definition

A financial transaction is a specific occurrence of a template [page 112] of the template category [page 111] "Financial Transaction". Each financial transaction refers to one particular template version [page 117], and is based on the information specified for that template version.

If a financial transaction is created for a template, the structure of the template is consolidated for the financial transaction. Financial transactions based on elementary products are denoted as elementary transactions. Financial transactions based on structured products are denoted as structured financial transactions.

Structure

Structured financial transactions are composed of individual sub-financial transactions. The overall financial transaction is the root node of the hierarchical structure, which is derived from the template. Each subtransaction is a structure node, and is assigned to a template version of the corresponding subtemplate. As with the financial transaction, the settings as regards attributes and characteristics for each sub-financial transaction are derived from the product configuration using the reference to a product version.

> **Note**
>
> Structured financial transactions are thus only valid when taken as a whole. The individual financial subtransactions are not independent and can only be used when taken together as an entire financial transaction.

3.1.2.2.1 Creating Financial Transactions

Prerequisites

- You have created the relevant template [page 112] in Customizing.
- You have configured a template version [page 117] for the template in Customizing.

Procedure

1. From the *SAP Easy Access* menu, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Transaction* > *Create Financial Transaction*.
2. Enter the *External Number*, the *Template*, and the *Template Version* and choose *Create*.

   > **Note**
   >
   > An external number is a unique ID for the financial transaction. Use only upper-case characters (capital letters) for external numbers. External numbers cannot contain any wildcard fields because they prevent locks from being set correctly.

   The screen for editing the financial transaction (FT) appears. The FT hierarchy is displayed on the left-hand side of the screen and the attributes and characteristics for editing the FT on the right. For more information, see Template [page 112].

FT Hierarchy	Attributes and Characteristics
The structure of the FT must be based on the structure of the template selected. If the template is an elementary template, you need to create an elementary FT. You cannot select any more financial subtransactions for the elementary FT. If the template is a structured template, you first need to create the root of a structured FT. You can create additional financial subtransactions for the root of a structured FT. **Note** If a financial transaction contains several financial subtransactions, you can choose  and  to edit complex structures of structured financial transactions. **Note** You can use the  function to replace the root or a root node of an FT.	Standard attributes: If you activated the appropriate attributes in Customizing when editing the template version, the following tab pages appear on the right-hand side of the screen: *Basic Data*, *Partner Data*, *General Limit Data*, *Key Figures*, *Object Relationships*, *Option Data*, *Cash Flow*. **Note** The tab pages and attributes that are displayed for editing depend on the template version used. Customer-specific attributes: Attributes that the customer defined in the Customizing activities under *Customer Enhancements* and activated when editing the template version appear on a tab page with a title specified by the customer. Characteristics: SAP NetWeaver Business Intelligence (BI) characteristics that you added to a template version in Customizing under Segmentation Service [page 1843] appear on the *Attribute Assignment* tab page.

3. Enter the authorization group [page 277].
4. Enter the attributes and characteristics of the elementary financial transaction or the root of the structured financial transaction.
5. Choose  to check your entries.

Note

The system checks to see if you still have to create any financial subtransactions.

6. Depending on whether errors occurred during the check, proceed in one of the following two ways:

a) No errors detected in data	b) Errors detected in data
If no errors are detected in the data, *Save* the financial transaction.	A message appears informing you that a subtemplate is still missing for your selected template. This means that you need to create additional financial subtransactions for the structured financial transaction. To create these additional financial subtransactions, proceed as follows: 1. Select the root of the structured financial transaction and choose *Create Node*. **Note** You have to create this node for the next level, since the hierarchy can contain only one root node. 2. In the dialog box, select a template and a template version. **Note** The subtemplate can be a structured template or an elementary template. The new financial subtransaction is included in the hierarchy. Once you have selected a template and a template version, the tab pages for editing appear. 3. Enter the attributes and characteristics of the financial subtransaction and check your entries using the function. 4. If no errors are detected in the check, save the financial transaction. If another error occurs, proceed as described above.

3.1.2.2.2 Creating Financial Transactions by Copying

Prerequisites

You have already created the financial transaction that you want to use, together with the relevant hierarchy, attributes, and characteristics.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Transaction* > *Create Financial Transaction by Copying*.
2. Enter the *External Number*.

   > **Note**
   >
   > An external number is a unique ID for the financial transaction. Use only upper-case characters (capital letters) for external numbers. External numbers cannot contain any wildcard fields because they prevent locks from being set correctly.

3. Select the external number of the financial transaction that you want to use as the template.
4. Enter a *Business Record Date* and a *System Time* to identify the template version you require.

   > **Note**
   >
   > You can use the input help to display all of the existing versions of the template, and select the version you require by double-clicking it. The system then fills the date fields automatically.

5. Choose *Create by Copying*.

   The screen for editing the financial transaction appears. The data on this screen has been defaulted by the template.
6. Enter your changes.
7. Save your entries.

3.1.2.2.3 Displaying and Changing Financial Transactions

Context

Displaying Financial Transactions

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Transaction* > *Display Financial Transaction*.
2. Enter an *External Number*.
3. Enter a *Business Record Date* and a *System Time* to call up the financial transaction version you require.

   > **Note**
   >
   > You can use the input help to display all of the existing versions of the financial transaction, and select the version you require by double-clicking it. The system then fills the date fields automatically.

4. Choose *Display*.

Changing Financial Transactions

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Transaction* > *Change Financial Transaction*.
2. Enter an *External Number*.
3. Enter a *Business Record Date* and a *System Time* to call up the financial transaction version you require.

   > **Note**
   >
   > You can use the input help to display all of the existing versions of the financial transaction, and select the version you require by double-clicking it. The system then fills the date fields automatically.

4. Choose *Change*.
5. Make the required changes.
6. Save your changes.

   > **Note**
   >
   > The system generates a new version automatically.

3.1.2.2.4 Standard Conditions

Definition

A condition is an element of a financial transaction or financial instrument that is defined in a contract. Conditions describe the structure details of a transaction or instrument in terms of time and amount.

Standard conditions are created separately from financial transactions and financial instruments and are subsequently assigned to financial transactions and financial instruments.

> **Note**
>
> Do not change standard conditions retroactively.

Use

You can use standard conditions for several customers in financial transactions and financial instruments. This reusability improves system performance, since standard conditions are only stored in the database once.

See also:

Editing Condition Groups [page 128]

Displaying Condition Groups [page 129]

3.1.2.2.4.1 Editing Condition Groups

Use

You can use this transaction to create or change conditions groups.

Prerequisites

- Prerequisites for creating a condition group:
  You have created *Condition Types* and *Condition Group Types* in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Master Data* > *Settings for Attributes* > *Condition Data*.
- Prerequisites for changing a condition group:
  You have created a condition group.

Procedure

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Transaction* > *Edit Standard Conditions* > *Edit Condition Group*.

To create a condition group, proceed as follows:

1. Select a condition group type from the dropdown list box (input help) for this field.
2. Enter a condition group and a description.
3. Specify a business record date.
4. Choose *Create Condition*.
5. In the *Detail* screen area, enter the required data.
6. Save your entries.
   You have created a condition group and can use this for corresponding financial transactions and instruments.

To change a condition group, proceed as follows:

1. Select a condition group in the frame on the left-hand side of the screen.
2. Select the condition you wish to edit by double-clicking it on the right-hand side of the screen.
3. Change the data that is displayed in the *Detail* screen area.
4. Save your changes.

3.1.2.2.4.2 Displaying Condition Groups

Prerequisites

You have created a condition group.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Transaction* > *Edit Standard Conditions* > *Display Condition Group*.
2. Select a condition group by double-clicking it.
3. Select a condition by double-clicking it.

   You can check the data on the Calculation, Periods, and Adjustment tab pages.

3.1.2.3 Primary Object: Financial Instrument

Definition

A financial instrument is a specific occurrence of a template [page 112] of the template category [page 111] *Financial Instrument*. Each financial instrument refers to one particular template version [page 117], and is based on the information specified for that template version.

If a financial instrument is created for a product, the structure of the product is reflected in the financial instrument. Financial instruments that refer to elementary templates are called elementary financial instruments. Financial instruments that refer to structured templates are called structured financial instruments.

Structure

Structured financial instruments are made up of individual financial subinstruments. The overall financial instrument is the root node of the hierarchical structure, which is derived from the template. Each subinstrument is a structure node, and is assigned to a template version of the corresponding subtemplate. This means that the attribute and characteristic settings can be derived from template configuration.

> **Note**
>
> The structured financial instruments are only valid as a whole. The individual financial subinstruments are not independent and can only be called up via the whole transaction.

3.1.2.3.1 Creating Financial Instruments

Prerequisites

- You have created the relevant template in Customizing.
- You have configured a template version for the template in Customizing.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Instrument* > *Create Financial Instrument*.
2. Enter the *External Number*, the Template, and the *Template Version* and choose *Create*.

   > **Note**
   >
   > An external number is a unique ID for the financial instrument. Use only upper-case characters (capital letters) for external numbers. External numbers cannot contain any wildcard fields because they prevent locks from being set correctly.

   The screen for editing the financial instrument (FI) appears. The FI hierarchy is displayed on the left-hand side of the screen and the attributes and characteristics for editing the FI on the right. .

FI Hierarchy	Attributes and Characteristics

The structure of the FI must be based on the structure of the template selected. If the template is an elementary template, you need to create an elementary FI. You cannot select any more financial sub-transactions for the elementary FI. If the template is a structured template, you first need to create the root of a structured FI. You can create additional financial subinstruments for the root of a structured FI. **Note** If a financial instrument contains several financial subinstruments, you can use the functions and edit complex structures of structured financial instruments. **Note** You can use the function to replace the root or a root node of an FI.	Standard attributes: If you activated the appropriate attributes in Customizing when editing the template version, the following tab pages appear on the right-hand side of the screen: *Basic Data, Partner Data, Instrument Data, Key Figures, Object Relationships, Option Data, Cash Flow.* **Note** The tab pages and attributes that are displayed for editing depend on the template version used. Customer-specific attributes: Attributes that the customer defined in the Customizing activities under Customer Enhancements and activated when editing the template version appear on a tab page with a title specified by the customer. Characteristics: SAP NetWeaver Business Intelligence (BI) characteristics that you added to a template version in the Customizing activities under Segmentation Service appear on the *Attribute Assignment* tab page.

3. Enter the authorization group.
4. Enter the attributes and characteristics of the elementary financial instrument or the root of the structured financial instrument.
5. Check your entries.

   Note

   The system checks to see if you still have to create any financial subinstruments.

6. Depending on whether errors occurred during the check, proceed in one of the following two ways:

a) No errors detected in data	b) Errors detected in data

If no errors are detected in the data, *Save* the financial instrument.	A message appears informing you that a subtemplate is still missing for your selected template. This means that you need to create additional financial subinstruments for the structured financial instrument. To create these additional financial subtransactions, proceed as follows: 1. Select the root of the structured financial instrument and choose *Create Node*. **Note** You have to create this node for the next level, since the hierarchy can contain only one root node. 2. In the dialog box, select a template and a template version. **Note** The subtemplate can be a structured template or an elementary template. The new financial subinstrument is included in the hierarchy. Once you have selected a template and a template version, the tab pages for editing appear. 3. Enter the attributes and characteristics of the financial subinstrument and check your entries. 4. If no errors are detected in the check, *Save* the financial instrument. If another error occurs, proceed as described above.

3.1.2.3.2 Creating Financial Instruments by Copying

Prerequisites

You have already created the financial instrument that you want to use, together with the relevant hierarchy, attributes, and characteristics.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Instrument* > *Create Financial Instrument by Copying*.

2. Enter the *External Number*.

> **Note**
>
> An external number is a unique ID for the financial instrument. Use only upper-case characters (capital letters) for external numbers. External numbers cannot contain any wildcard fields because they prevent locks from being set correctly.

3. Select the *External Number* of the financial instrument that you want to use as the template.
4. Enter a *Business Record Date* and a *System Time* to identify the template version you require.

> **Note**
>
> You can use the input help to display all of the existing versions of the template, and select the version you require by double-clicking it. The system then fills the date fields automatically.

5. Choose *Create by Copying*.

   The screen for editing the financial instrument appears. The data on this screen has been defaulted by the template.
6. Enter your changes.
7. Save your entries.

3.1.2.3.3 Displaying and Changing Financial Instruments

Displaying Financial Instruments

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Instrument* > *Display Financial Instrument*.
2. Enter an *External Number*.
3. Enter a *Business Record Date* and a *System Time* to call up the financial instrument version you require.

> **Note**
>
> You can use the input help to display all of the existing versions of the financial instrument, and select the version you require by double-clicking it. The system then fills the date fields automatically.

4. Choose *Display*.

Changing Financial Instruments

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Instrument* > *Change Financial Instrument*.
2. Enter an *External Number*.

3. Enter a *Business Record Date* and a *System Time* to call up the financial instrument version you require.

> **Note**
>
> You can use the input help to display all of the existing versions of the financial instrument, and select the version you require by double-clicking it. The system then fills the date fields automatically.

4. Choose *Change*.
5. Make the required changes.
6. Save your changes.

> **Note**
>
> The system generates a new version automatically.

3.1.2.4 Primary Object: Account

An account is a specific occurrence of a template [page 112] of the template category [page 111] *Settlement Account*, *Securities Position AccountCustomer Position Account* or *G/L Account*. Each account refers to one particular template version [page 117], and is based on the information specified for that template version.

Various account categories are available because customers want to be able to access different positions via an account (positions pertaining to customer business, say, or internal positions), and also to reference different financial transactions and instruments.

The differences between financial transactions/financial instruments and accounts are as follows:

- Accounts do not have hierarchies
- Accounts can be deleted
- Accounts that have been deleted can be reactivated
- Accounts that are processed always use the current template version

3.1.2.4.1 Creating Accounts

Prerequisites

- You have created the relevant template [page 112] in Customizing.
- You have created a template version [page 117] for the template in Customizing.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Account* > *Create Account*.
2. Specify the account number, account category, and template, and choose *Create*.

> **Note**
>
> You do not need to specify a template version, since the system always uses the current template version.
>
> The *Change Account* screen appears. Here you can edit the attributes and characteristics of the account.
>
> Standard attributes:
>
> If you activated the appropriate attributes in Customizing when editing the template version, the following tab pages appear: *Account Basic Data, Partner Data, and Object Relationships*.

> **Note**
>
> The tab pages and attributes that are displayed for editing depend on the template version used.
>
> Customer-specific attributes:
>
> Attributes that the customer defined in the Customizing activities under *Customer Enhancements* and activated during editing, appear on a tab page with a title specified by the customer.
>
> Characteristics:
>
> SAP NetWeaver Business Intelligence (BI) characteristics that you added to a template version in the Customizing activities under *Segmentation Service* appear on the *Attribute Assignment* tab page.

   - Enter the authorization group.
   - Specify the attributes.
   - Save your entries.

3.1.2.4.2 Creating Accounts by Copying

Prerequisites

You have created the account whose attributes and characteristics you want to use.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Account* > *Create Account by Copying*.
2. Enter the *Account Number*.

   > **Note**
   >
   > An account number is a unique ID for the account.

3. Select the *Account Number* of the account that you want to use as the template. The system selects the *Account Category* automatically once you have selected the account number.
4. Enter a *Business Record Date* and a *System Time* to identify the template version you require.

   > **Note**
   >
   > You can display all the existing versions of the template using input help and select the version you require by double-clicking it. The system then fills the date fields automatically.

5. Choose *Create by Copying*.

   The screen for editing the account appears. This screen already contains the data predefined for the template.
6. Enter your changes.
7. Save your entries.

3.1.2.4.3 Displaying and Changing Accounts

Displaying Accounts

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Account* > *Display Account*.
2. Specify an *account number*.
3. Enter a *Business Record Date* and a *System Time* to call up the account version you require.

   > **Note**
   >
   > You can use the input help to display all of the existing versions of the account, and select the version you require by double-clicking it. The system then fills the date fields automatically.

4. Choose *Display*.

Changing Accounts

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Account* > *Change Account* .
2. Specify an *account number*.
3. Enter a *Business Record Date* and a *System Time* to call up the account version you require.

> **Note**
>
> You can use the input help to display all of the existing versions of the account, and select the version you require by double-clicking it. The system then fills the date fields automatically.

4. Choose *Change*.
5. Enter your changes.
6. Save your entries.

> **Note**
>
> The system generates a new version automatically.

3.1.2.5 Primary Object: Country

Definition

A country is a specific occurrence of a template [page 112] of the template category [page 111] *Country*. Each country refers to one particular template version [page 117], and is based on the information specified for that template version.

Use

You use the *Country* primary object to store country-specific data (such as, internal or external ratings) in Bank Analyzer.

The following data can be stored in this primary object:

- Rating data (for example, internal and external ratings)
- Values for customer-specific characteristics
- Partner data
- Key figures
- Currencies (for example, local currencies)

3.1.2.5.1 Creating Countries

Prerequisites

In Customizing for *Bank Analyzer*, you have created the template for the template category *Country* and a template version [page 112] for this template.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Country* > *Create Country*.
2. Enter an *External Number*.

   > **Note**
   >
   > An external number is a unique ID for the country. Use only upper-case characters (capital letters) for external numbers. External numbers cannot contain any wildcard fields because they prevent locks from being set correctly.

3. Use the input help to select a template.
4. Use the input help to select a template version.
5. Choose *Create*.
6. The *Create Country* screen appears.
7. You can change the template and the template version.
8. Enter the relevant data on the tab page.
9. Save your entries.

3.1.2.5.2 Creating Countries by Copying

Prerequisites

You have created the country whose attributes and characteristics you want to use.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Country* > *Create Country by Copying*.
2. Enter an *External Number*.

   > **Note**
   >
   > An external number is a unique ID for the country. Use only upper-case characters (capital letters) for external numbers. External numbers cannot contain any wildcard fields because they prevent locks from being set correctly.

3. Select the *External Number* of the country that you want to use as the template.
4. Enter the *Business Record Date* and the *System Time* to identify the template version you require.
5. Choose *Create by Copying*. The screen for editing the country appears. This screen already contains the data predefined for the template.
6. Enter your changes.
7. Save your entries.

3.1.2.5.3 Displaying and Changing Countries

Prerequisites

You have created at least one country.

Procedure

To display a country, proceed as follows:

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Country* > *Display Country*.
2. Enter an *External Number*.
3. Specify a *Business Record Date* and a *System Time*.
4. Choose *Display*.

To change a country, proceed as follows:

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Country* > *Change Country*.
2. Enter an *External Number*.
3. Specify a *Business Record Date* and a *System Time*.
4. Choose *Change*.

5. Make the required changes.
6. Save your entries.

3.1.2.6 Primary Object: Physical Asset

Definition

A physical asset is a specific occurrence of a template [page 112] of the template category [page 112] *Physical Asset*. Each physical asset refers to one particular template version [page 117], and is based on the information specified for that template version.

Use

You use physical assets to represent objects that form the basis of a contractual agreement (for example, real estate).

3.1.2.6.1 Creating Physical Assets

Prerequisite

In Customizing for *Bank Analyzer*, you have created the template [page 112] for the physical asset, and a template version [page 117] for this template.

Procedure

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Physical Asset* > *Create Physical Asset*.
2. Enter an *External Number*.

   > **Note**
   >
   > An external number is a unique ID for the physical asset. Use only upper-case characters (capital letters) for external numbers. External numbers cannot contain any wildcard fields because they prevent locks from being set correctly.

3. Use the input help to select a template and a template version.
4. Choose *Create*.
5. Enter a name for the authorization group.
6. You can change the template and the template version.
7. Enter the relevant data on the tab pages.
8. Save your entries.

3.1.2.6.2 Creating Physical Assets by Copying

Prerequisites

You have created the physical asset whose attributes you want to use.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Physical Asset* > *Create Physical Asset by Copying*.
2. Enter an *External Number*.

   > **Note**
   >
   > An external number is a unique ID for the physical asset. Use only upper-case characters (capital letters) for external numbers. External numbers cannot contain any wildcard fields because they prevent locks from being set correctly.

3. Select the *External Number* of the physical asset that you want to use as the template.
4. Enter the *Business Record Date* and the *System Time* to identify the template version you require.
5. Choose *Create by Copying*. The screen for editing physical assets appears. This screen already contains the data predefined for the template.
6. Enter your changes.
7. Save your entries.

3.1.2.6.3 Displaying and Changing Physical Assets

Prerequisites

You have created at least one physical asset.

Procedure

To display a physical asset, proceed as follows:

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Physical Asset* > *Display Physical Asset*.

2. Enter an *External Number*.
3. Specify the business record date and the system time.
4. Choose *Display*.

To change a physical asset, proceed as follows:

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Physical Asset* > *Change Physical Asset*.
2. Enter an *External Number*.
3. Specify the business record date and the system time.
4. Choose *Change*.
5. Make the required changes.
6. Save your entries.

3.1.3 Primary Object: Position

Definition

A position is a unit that contains key figures for a key date. You can assign this unit to an account. Key figures might include the daily balance, the end-of-month balance, or the monthly average value of the account.

Use

You use positions to map all key figures (values and quantities) that are stored in the system based on a key date.

Structure

You store position information in the system using one of the following two objects:

- Primary object category: position
  You use the object category *Position* to store characteristics and key figures that often change (that have transaction data character). You must have already defined position classes in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Flow Data* > *Positions* > *Edit Position Class*. Position classes are used to map different categories of positions. Positions in a position class have the same characteristics and key figures.
- Template category: position master data [page 152] (MDF)
  You use the template category *Position Master Data* of the master data framework (MDF) to store characteristics and key figures that do not often change. You can also use it to map n:m relationships between position information and other primary objects (required, for example, for mapping collateral).

Position master data and the associated positions (identical for the values of the defining characteristics) are linked 1:1 with each other.

You can, however, create positions without associated position master data. Conversely, the system always generates an associated position header when position master data is created.

> **Note**
>
> You can check for inconsistencies, on the *SAP Easy Access* screen under *Bank Analyzer* > *Source Data Layer* > *Tools* > *Flow Data* > *Check Position Master Data* (report /BA1/RF2_CHECK_AND_REPAIR_PMD). Examples of such inconsistencies include the existence of position master data without an associated position, or a position that makes reference to position master data that is no longer in the system.

Integration

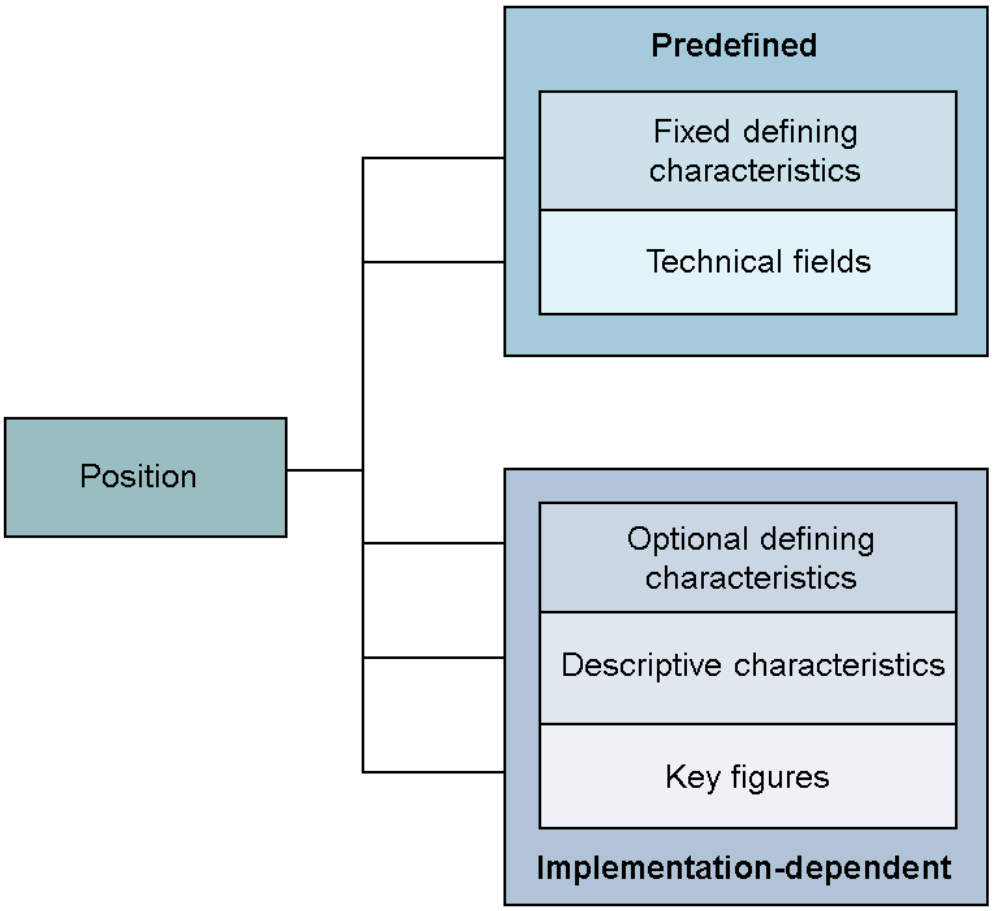
Position information contains key figures for key dates and, as a result, changes to the key figures from one key date to another do not generate new versions as in the Versioning of Primary Objects [page 103]. Instead, these changes generate new positions. The system therefore generates an additional version of the position information for a particular key date.

If you want to create a correction because, for example, a key figure of a position has been created incorrectly for a key date, you can recreate the position for the same key date with the correct value. The system does not then physically change the position version, but generates an additional, new position version containing the correct value and the current system time. You have to provide all of the data and not just the key figures that are to be corrected for this new version.

3.1.3.1 Characteristic, Key Figure, and Technical Field

Definition

A position comprises characteristics and key figures. Certain technical fields are also required for processing and controlling the positions. The technical fields do not depend on the position class, in other words, each position contains all the technical fields.



Structure

- Fixed defining characteristics are the key fields that every position has. These are predefined by SAP.
- Optional defining characteristics are the key fields of a position that are defined during the implementation phase.
- Descriptive characteristics are attributes of a position that provide additional information about the position so that it can be processed by Analyzer-dependent functions, for example.
- Key figures are the value fields that are maintained for the position. The following key figure categories are available:
  - "01" Monetary Amount
  - "02" Percentage
  - "03" Quantity

Example

Field Name	Category
Position class	Fixed defining characteristics
Key date	
Time of entry in system	
Account category	
External account number	
Client key	Optional defining characteristics
PA template	
Object ID	
“Non-performing” checkbox	
Profit center	Descriptive characteristics
Entered by	Technical fields
Time replaced in system	
Invalidity ID	
Pointer to predecessor	
Pointer to successor	
User making change	
Loan commitment interest	Key figures
Commission expense	
Effective interest rate	
Tied-up capital	
Internal refinancing interest rate	
Average margin as amount	
Average margin as percentage	
Average market interest rate as amount	

Average market interest rate as percentage
Interest rate condition contribution
Interest expense relating to other periods
Interest revenue relating to other periods
...

3.1.3.2 Creating a Position

Context

You can create positions by entering the individual characteristic and key figure values of a position.

Procedure

1. Choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Position* > *Create Position*.
2. Enter the required position class, account number, account category, and key date. The *Position Class*, *Key Date*, and *Account* fields are required entry fields. The *Object* fields are optional entry fields.
3. The copy function enables you to use an existing position as a template.

   > **Note**
   >
   > You can use the copy function to create positions of the same position class. The characteristics and key figures are copied, except for the data you have entered above for the position class, account number, account category, and key date.

4. Choose *Create*.
5. The *Create Position* screen appears and contains the following tab pages:
   - Defining Characteristics
   - Descriptive Characteristics
   - Values
   - Quantities
   - Percentages

   > **Note**
   >
   > If a position class does not have the right group for a tab page, the system does not display the tab page in question.

6. Enter the data.

   The entries in the *Position Class, Account Number, Account Category*, and *Key Date* fields are predefined and cannot be overwritten.
7. Save your entries.

Results

The system creates the position.

3.1.3.3 Displaying a Position

Procedure

1. Choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Position* > *Display Position*.
2. You can determine the required position by entering information in the position fields.

   The following fields are required entry fields:
   - Position class
   - Account number
   - Account category
   - Key date
   - Position status

   The following fields are optional:
   - Template category
   - External number
   - Parent template
   - Child template
   - Node number
   - Key date from
   - Entered on (date)
   - Entered at (time)
3. You can also specify selected defining and descriptive characteristics and key figures under *Additional Selection Criteria*.
4. You can set the control parameters to read just the current version or to read all the versions of a position for a particular key date.

   > **Note**
   >
   > You can also use the predecessor and successor functions.

> **Note**
>
> To display the account master data, choose *Display Account*.

Results

The system displays a list of the position versions.

> **Note**
>
> To call up the detailed data, choose *Select Position*. The system displays the tab pages for the characteristics and key figures.

> **Note**
>
> If this is a generated position, you can display the related business transactions from the list. For more information, see Generate Positions [page 304].

3.1.3.4 Changing Descriptive Chars and Key Figures in a Position

Context

You can change descriptive characteristics and key figures.

The old position version remains unchanged in the system. It can be used as a template to create a new position version, in which only descriptive characteristics and key figures can be changed.

Procedure

1. Choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Position* > *Change Position*.
2. You can determine the required position by entering information in the position fields.

   The following fields are required entry fields:

   - Position class
   - Account number
   - Account category

- Key date
- Position status

The following fields are optional:

- Template category
- External number
- Parent template
- Child template
- Node number
- Key date from
- Entered on (date)
- Entered at (time)

3. You can also specify selected defining and descriptive characteristics and key figures under *Additional Selection Criteria*.
4. You can set the control parameters to read just the current version or to read all the versions of a position for a particular key date.

> **Note**
>
> You can also use the predecessor and successor functions.

5. Choose *Execute*.

> **Note**
>
> To display the account master data, choose *Display Account*.

Results

The system displays a list of the position versions that fulfill the selection criteria. To call up the detailed data, choose *Change Position*. The system displays the following tab pages:

- Defining Characteristics
- Descriptive Characteristics
- Key Figure: Values
- Key Figure: Quantities
- Key Figure: Percentages

You can change the fields on the tab pages for descriptive characteristics and key figures.

> **Note**
>
> You can use the *Change Defining Characteristics in Position* function to change the defining characteristics. You can only display the fields for the descriptive characteristics and the key figures, however.

3.1.3.5 Changing Defining Characteristics in a Position

Context

If you change key figures or descriptive characteristics in a position, a new version of the position is created. If you change a defining characteristic, however, a new position is created (rather than a version of an existing position).

Procedure

1. Choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Position* > *Change Defining Characteristics in Position*.
2. You can determine the required position by entering information in the position fields.

   The following fields are required entry fields:

   - Position class
   - Account number
   - Account category
   - Key date
   - Position status

   The following fields are optional:

   - Template category
   - External number
   - Parent template
   - Child template
   - Node number
   - Key date from
   - Entered on (date)
   - Entered at (time)
3. You can also specify selected defining and descriptive characteristics and key figures under *Additional Selection Criteria*.
4. You can set the control parameters to read just the current version or to read all the versions of a position for a particular key date.

   > **Note**
   >
   > You can also use the predecessor and successor functions.
5. Choose *Execute*.

   > **Note**
   >
   > To display the account master data, choose *Display Account*.

Results

The system displays a list of the position versions that fulfill the selection criteria.

> **Note**
>
> To call up the detailed data, choose *Change Position*. The system displays the following tab pages:

- Defining Characteristics
- Descriptive Characteristics
- Values
- Quantities
- Percentages

> **Note**
>
> You can change **only** defining characteristics here.

> **Note**
>
> You can use the *Change Position* function to change the descriptive characteristics and the key figures. You can only display the fields for the defining characteristics, however.

3.1.3.6 Setting a Position to Invalid

Context

If you set a position to invalid manually, the selected position version remains unchanged, and a new version is created with the status "Invalid".

Procedure

1. Choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Position* > *Set Position to Invalid*.
2. You can determine the required position by entering information in the position fields.

   The following fields are required entry fields:

   - Position class
   - Account number
   - Account category

- Key date
- Position status

The following fields are optional:

- Template category
- External number
- Parent template
- Child template
- Node number
- Key date from
- Entered on (date)
- Entered at (time)

3. You can also specify selected defining and descriptive characteristics and key figures under *Additional Selection Criteria*.
4. You can set the control parameters to read just the current version or to read all the versions of a position for a particular key date.

> **Note**
>
> You can also use the predecessor and successor functions.

5. Choose *Execute*.
6. The system displays a list of the position versions.
7. You can select a row in this list and execute the *Set Position to Invalid* function.

3.1.3.7 Position Master Data

Definition

Position master data is a component of the primary object position [page 142] and contains data about positions that have master data character and that are not based on key dates.

> **Note**
>
> You use the position master data to store data that does not often change for the position. As a result, this data does not need to be provided for every change to the position. In the Source Data Layer, you can also create positions without position master data. In some cases, however, the calculation process in Credit Exposure [page 1196] presumes that position master data is being used.

From a technical point-of-view, position master data is an additional template category of the master data framework. As a result, this framework's standard functions and conventions apply. Examples of these are the creation of templates [page 112], template categories [page 111], and template versions [page 117], or the use of service modules [page 118].

Structure

The following service modules can be used for position master data:

- Basic Data
- Object Relationships
- Credit Exposure
- Attributes
- Countries

Integration

The following functions are available for position master data:

- *Create Position Master Data* or *Create Position Master Data by Copying*
- *Display Position Master Data* and *Change Position Master Data*
- *Position Master Data: Display Scenario Version* and *Position Master Data: Edit Scenario Version*
- *Position Master Data: Display Stressed Data*

You find these functions on the *SAP Easy Access menu* under *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Position* > *Position Master Data*.

3.1.4 Primary Object: Business Transaction

Definition

A business transaction is an independent but logically connected business event that implies a change in value or quantity, or a result. Business transactions therefore represent transaction data. A business transaction is depicted as a posting in Accounting as one whole entity, which can be reversed. In Accounting, this entity has to be identifiable.

Use

Business transactions are managed in the Source Data component and provided to the Analyzers for further processing. For example, the accounting processes read the data periodically and generate accounting documents from this data.

A business transaction generally entails several flows to several accounts. For example, when a share is bought, the quantity and the position amount are credited to a securities account in the position currency, and the position amount is debited from a settlement account in the position currency.

Structure

The *Business Transaction* primary object contains header data and business transaction items.

Integration

A *Business Transaction* primary object is linked to accounts or financial transactions/financial instruments by means of business transaction items.

You manage operational events [page 258] using business transactions.

3.1.4.1 Creating Business Transactions

Prerequisites

You have edited the *item class* and *business transaction class* in Customizing by choosing *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Transaction Data* > *Business Transactions*.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Business Transactions* > *Create Business Transaction*.
2. Select a business transaction class.

   > **Note**
   >
   > Based on the business transaction class, you can create a *Business Transaction By Reference* or a *Business Transaction By Value*. For the business transaction by value, you need to set the indicator in Customizing for the business transaction class. You can also create a *Reversal Business Transaction By Value*. For the *Reversal Business Transaction By Reference*, you use the transaction *Reverse Business Transaction* (see Reversing Business Transactions [page 156]).

3. Specify a business transaction ID.
4. Specify the source system.
5. Choose *Create*.
6. Enter the relevant data.
7. Save your entries.

   > **Note**
   >
   > When you create and reverse business transactions, the following situation can occur: The business transaction is created correctly in the source system, but it is not transferred correctly to the Source Data

Layer by the transformation layer. In this case, the transformation layer can create a reversal business transaction and then a new business transaction ID that references the original business transaction. As a result, you do not have to make any changes and you can continue to process the business transaction with the correct reference to the source system.

The system can check business transactions when they are created or supplied to see whether the referenced objects (nodes and/or master data objects) are valid at the business record date specified.

To check the business transactions when they are created or supplied, you have to select the *Object Rel.* indicator for the respective business transaction class in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Transaction Data* > *Business Transactions* > *Edit Business Transaction Class*. If you do not select the indicator, the system checks only that the reference object exists.

If a position class is assigned to a business transaction class, the system generates technical positions when supplying business transactions. These technical positions only contain information that defines positions (no descriptive characteristics or key figures). The system usually derives defining characteristics and fixed fields for positions from business transactions using identical names. You can exclude certain defining characteristics and fixed fields from this derivation when you generate technical positions by storing the characteristics in question in the assigned position class on the *Assignment of Non-Relevant Position Characteristics* screen. You can find this in Customizing for Bank Analyzer by choosing *Source Data Layer* > *Primary Objects* > *Transaction Data* > *Business Transactions* > *Edit Business Transaction Class*. Leave the field for technical position empty for characteristics stored here.

3.1.4.2 Creating Business Transactions by Copying

Prerequisites

You have edited the *item class* and *business transaction class* in Customizing under *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Transaction Data* > *Business Transactions*.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Business Transactions* > *Create Business Transaction by Copying*.
2. Select a business transaction class.
3. Assign a business transaction ID.
4. Specify the source system.
5. Select the business transaction ID and source system using input help.
6. Choose *Create by Copying*.

7. Enter the data or change the data inserted from the template.
8. Save your entries.

3.1.4.3 Reversing Business Transactions

Prerequisites

You have created at least one business transaction.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Business Transactions* > *Reverse Business Transaction*.
2. Select a business transaction class.

   > **Note**
   >
   > You can create only *Reversal Business Transactions By Reference* here. You create *Reversal Business Transactions By Value* using the business transaction Create Business Transaction (see Creating Business Transactions [page 154]).

3. Enter a business transaction ID for the reverse business transaction.
4. Specify the source system of the business transaction to be reversed.
5. Select the business transaction ID and source system using input help.
6. Choose Reverse.
7. Enter the data on the reversal business transaction.
8. Save your entries.

   > **Note**
   >
   > When you create and reverse business transactions, the following situation can occur:
   >
   > The business transaction is created correctly in the source system, but it is not transferred correctly to the Source Data Layer by the transformation layer. In this case, the transformation layer can create a reversal business transaction and then a new business transaction ID that references the original business transaction. As a result, you do not have to make any changes and you can continue to process the business transaction with the correct reference to the source system.

Result

The business transaction you selected has been reversed.

3.1.4.4 Reversing Business Transactions in Bulk

Prerequisites

You have created at least one business transaction.

Procedure

1. Execute transaction `/BA1/F2_BT_REV_BULK` to reverse multiple business transactions in single run.

   > **Note**
   >
   > As of Banking Services from SAP 9.0 FP08, the transaction only allows Reversal of Business Transactions by Value.

2. Enter the business transaction class and source system. In addition, enter one of the following data: business transaction ID, business record date or time created of the business transaction(s) to be reversed.
3. Specify suffix for reversal BT and source system for reversed BT. If you do not enter a suffix for reversal BT, the system applies the default value **`_REV`**.
4. Execute the transaction.
5. The system displays a log with results.

> **Note**
>
> When you create and reverse business transactions, the following situation can occur:
>
> The business transaction is created correctly in the source system, but it is not transferred correctly to the Source Data Layer by the transformation layer. In this case, the transformation layer can create a reversal business transaction and then a new business transaction ID that references the original business transaction. As a result, you do not have to make any changes and you can continue to process the business transaction with the correct reference to the source system.

Result

The business transaction(s) you selected has been reversed

3.1.4.5 Displaying Business Transactions

Prerequisites

You have created at least one business transaction.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Business Transaction* > *Display Business Transaction*.
2. You can select the business transactions using the following criteria:
   - Key
   - Header characteristics
   - Item characteristics
   - Account
3. Enter the selection criteria on the tab page selected.
4. Choose *Execute*.

Results

The system displays a list of business transactions. You can select individual business transactions from this list and display the detail data of the business transaction.

> **Note**
>
> - If the business transaction is an aggregation business transaction, you can display the related aggregation object and individual transactions from within the list.
> - If the business transaction relates to a transfer posting, you can display the position that was transferred.
>
> For more information, see Aggregating Business Transactions [page 307].

3.1.4.6 Changing Size Cat. of Content of Business Transaction Class

Prerequisites

You have created at least one business transaction.

Context

You use this transaction to define the size category of the dataset that is expected for the current posting process for the applicable business transaction class. The size category depends on selection criteria such as the processing period or the legal entity. If you select an applicable size category, packaging is adapted to suit the package size that you have specified. As a result, performance is improved.

> **Note**
>
> If you do not specify a size category for a business transaction class, the system uses the size category with the most data records for each selection. You can change the estimated size category for different postings at any time.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Business Transaction* > *Specify Size Category for the Content of a Business Transaction Class*.
2. Choose *New Entries*.
3. Use the input help to select a business transaction class.
4. Specify the *Size*.
5. Save your entries.

3.1.4.7 Evaluations for Data Import Buffer

Use

You can use this function to display, delete, or release for further processing, business transactions [page 153] that are in the data import buffer.

While the business transactions are in the data import buffer, they cannot be used in any other Bank Analyzer process. This means, for example, that the system cannot derive data for application events from business transactions that have an operational event [page 258] in the data import buffer.

You can run a quality check for the imported business transactions in the data import buffer before you release them. You could use the reconciliation report for positions and business transactions for this, for example.

> ⚠ Caution
>
> The functions of the data import buffer are not available for business transactions that are persisted in a flat table.

Prerequisites

When delivering the business transactions, you have entered values for the business transactions in the import parameters below in BAPI method `AddNewVersion`. This is a prerequisite if you want to use the data import buffer for business transactions.

- `BufferEnabled` (data import buffer activated: yes/no)
- `SelectionCriteria` (selection criteria for data import buffer)

Features

You can use the following functions:

- Number of entries in the data import buffer
  The system uses the specified selection criteria to calculate the number of business transactions that have the status *Not yet released*.
- Display of objects
  You can display business transactions that are not yet released based on the specified selection criteria.
- Release of objects
  You can change the status of business transactions from *Not yet released* to *Released*. These business transactions are then no longer in the data import buffer and can be used by other Bank Analyzer processes. The system uses the timestamp that is valid when the business transactions are released as the creation time. As soon as you release operational events, event management derives application events.
- Deletion of objects
  You can delete business transactions that have not yet been released.

You can display the messages for the functions used in logs, and you can delete existing logs if required.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Primary Objects* > *Business Transaction* > *Evaluations for Data Import Buffer*.

2. On the selection screen, enter the required selection criteria for the business transactions.
   You can select the business transactions using keys or using header characteristics.
3. Specify the function that you want to run for the selected business transactions, for example, *Display* or *Release*, and choose *Execute* using the quick info sysmbol.

3.1.5 Primary Object: Business Partner

SAP Business Partner for Financial Services (SAP BP-FS) is the central SAP application within the Source Data Layer in Bank Analyzer for saving and managing all the customer information.

Purpose

In Bank Analyzer, *SAP Business Partner for Financial Services* brings together all the customer data for a bank that exists across every application, and enables you to manage business partners centrally in the Source Data Layer.

The concept is designed to minimize data redundancy, and exploit the advantages of data integration. It also focuses more strongly on managing customer relationships, and new customer acquisitions.

You can process the business partner using the business partner user interface, as well as through defined interfaces. This enables you to integrate customer-specific data without having to make modifications, and makes the data release-insensitive.

The solution is developed by SAP as standard software for international banks.

You can use the enterprise services for the business object *Analytical View of Business Partner*. For more information, see the enterprise services documentation for banking services from SAP on SAP Help Portal at http://help.sap.com/bankingservices. Under *Integration & AnalyticsInformation*, open the enterprise services documentation and choose *Financial Instruments Analytical Reference Data*.

Integration

The *SAP Business Partner for Financial Services* is closely linked to other Bank Analyzer [page 8] components. This ensures continuous processing.

SAP Business Partner for Financial Services uses special tools to manage the business partner position correctly. You can use them to manage the following business transactions, for example:

- Change documents
- Document management
- Authorizations
- Data transfer

Features

The most important business partner data includes:

- Name and address of the business partner
- Supplementary data depending on the business partner category, (such as date of birth, legal form) for all application areas (such as industry keys for marketing selections or regulatory reporting)
- Relationships (for example, customer service representatives, borrower entities and units of risk, group structures)
- Business partner roles
- Legitimation data and other ID numbers
- Credit standing data/data provided by credit reference agencies/scoring results/marketing clusters
- Employment overview or fiscal year information

Business partner relationships can be linked in many different ways.

From a bank's point of view, customers have defined roles (such as *Customer*, *Issuer*, or *Guarantor*, for example). For this reason, the role is a vital component within the concept of SAP Business Partner.

3.1.5.1 Concepts

This section contains information about the basic concepts of SAP Business Partner.

3.1.5.1.1 Business Partner

Definition

Natural or legal person with whom business relationships are maintained. Typical business partners are banks, a central treasury department with which financial transactions are concluded, or a customer who has several accounts at the bank, for example.

Use

You need this business object, for example, to create a contract as a transaction with one specific business partner in *Loans Management*, which is why the business partner data must exist before the transaction can be entered. The options available for creating a business partner depend on the system settings you make here.

Structure

Business partners are divided into different **categories**, which have different attributes assigned to them. The system distinguishes between:

- Persons (for example, individuals, brokers)
- Organizations (for example, banks, companies)
- Groups (for example, societies according to the German Civil Code, people living in shared apartments)

The functions of the business partner in the contract are determined using **business partner roles**, such as *Account Holder* and *Card Holder*, as well as *Prospect* or *Processor*. You define the rights and responsibilities of the business partner by assigning the role category. A business partner can have several roles. You determine business partner roles when you assign the role category.

The data that can be processed for a business partner is divided into the following headings:

- General data
  This data is held centrally for all business partner roles. Access to the data can vary according to the role category, depending on the system settings in Customizing for the business partner.
- Company-code-dependent data (only relevant for SAP ERP)
  This is data that only applies for a particular role in the relevant company code.

You can use the field modification functions in Customizing for the business partner to define which fields are required or optional, which are displayed only, and which are suppressed.

You can link business partners using **relationships**. You also have to **authorize** the business partner for the permitted transactions.

Integration

You make settings for the business partner in Customizing for *SAP Business Partner for Financial Services*.

If an authorization object is entered for the business partner under *Authorization Group*, an authorization check is made before the business partner is processed. The standard system checks the authorization object *B_BUPA_GRP (business partner authorization groups)*.

3.1.5.1.2 Business Partner Role (BP Role)

Definition

Rights and obligations that can be taken on by a business partner in different business transactions.

Use

You can use the business partner role to carry out a business classification of a business partner.

The basis for the definition of a business partner role is a business transaction. The attributes of the BP role depend on the particular transaction involved. The data that is available for a transaction also depends on the .

Example

Transactions and Associated Roles

Transactions	BP role
Order	Ordering party
Delivery of goods	Ship-to parties

In addition to the central BP roles, there are application-specific roles. For example:

- Contract partner and installer for utility companies
- Sold-to party and ship-to party for CRM Enterprise

You can create a business partner in one or more BP roles. Central data such as name, address and bank details only has to be created once.

The BP role *General Business Partner* is automatically assigned to a business partner. Depending on the function of the business partner in question, you can also select the following BP roles:

- BP role
- BP role
- BP role

You can group roles together in a role grouping, which you can then select in the dialog.

You can use the following functions to display and edit BP roles in the dialog:

- Overview of all roles assigned to a business partner
- Deletion of a role assignment
- Time-based validity of a role assignment
- Where-used list for a role (for example, in an application)
  The where-used list allows you to check whether it is possible to delete the assignment of BP roles to business partners that form part of a business partner relationship. If the relevant role categories are defined at the level of the relationship category and form a prerequisite for creating a relationship, then the assignment cannot be deleted.

Structure

For more information about the BP roles concept, see the Implementation Guide (IMG) for the *Business Partner*.

Integration

An extension [page 168] of the BP roles to include customer-specific attributes is possible. For more information, see the .

Make the settings required for the BP role in the implementation guide (IMG) in Customizing for the *Business Partner* under *Basic Settings* *Business Partner Roles* .

Example

You are an employee in the sales department of the company Smith plc. You have made contact with Ms. Lopez at Hansen plc. You would like to send her information about your existing products. You create the following business partner master data:

- A master data record for a business partner with the name “Hansen plc.” with the BP role *Prospect*, for which you define the required validity data. You create the central data and the address.
- A master record for a business partner with the name “Ms. Lopez“ with the BP role *Contact Person*, as well as its validity. You create the central data, the address and the relationship “is contact person of” at Hansen plc.

Three months later, Hansen plc places an order. The invoice needs to be sent to the parent company, Hansen and Sons, to be settled. You create the following new business partner master data:

- You delete the *Prospect* role from Hansen plc and assign the role of *ordering party* to them. This role is valid from the day of the order. You create sales and shipping data for a specific sales area.
- A master record for a business partner with the name "Hansen and Sons" with the BP roles *Bill-To-Party* and *Payer*. You create the central data, the address, the bank details and the billing data.

3.1.5.1.2.1 BP Role: Customer

Definition

The business partner in the role of *Customer* is a person, organization, or group that is in a business relationship with your company, and is treated in this function as a business partner of the company.

Use

Business partners in the role of *Customer* exist in numerous relationships with other primary objects [page 100]. For example, they can open accounts [page 134] or enter into financial transactions [page 122] with a bank.

If you create the relationship category *Is Represented as Customer By*, you can link the customer with business partners in the role of Customer Service Representative [page 166], depending on a .

Integration

A business partner can be a customer of a company in more than one context. You do not need to create the business partner again in each case. You can create the partner in the role of *Customer* with different differentiations. If you create different differentiation criteria for this business partner, you can define the different contexts in which the partner acts as a customer. In this way the business partner can be a customer at several branches, which you can enter as differentiation criteria for the customer.

Example

The business partner Mark Adams opens a securities account in a branch of a bank in order to buy and sell securities. He has a savings account at another branch of the bank. He is entered as a business partner of the bank in the role of *Customer* and is dealt with by different agents in the respective branches.

3.1.5.1.2.2 BP Role: Customer Service Representative

Definition

The business partner in the role of *Customer Service Representative* is a natural person who represents several business partners in the role of Customer [page 165]. *Customer Service Representatives* are usually company employees.

Use

If you create the relationship category *Is Customer Service Representative for Customer*, you can link the *Customer Service Representative* with business partners in the role of *Customer*, depending on a .

Example

Carol Smith is a private customer manager in a bank. She is entered as a business partner of the bank in the role of *Customer Service Representative*. Bank customers are linked to her when you create the relationship category *Is Represented as Customer By*.

3.1.5.1.2.3 BP Role of Borrower Entity

Definition

A business partner in the role of *borrower entity* combines borrowers grouped together in borrower entities in accordance with paragraph 19 section 2 of the German Banking Act.

Use

A business partner in the role of *borrower entity* is only responsible for grouping a borrower entity. In this sense, it does not constitute an existing person, organization, or group with which transactions are made. It is a “virtual” business partner that is used as an anchor for business partners linked by relationships in borrower entities.

Integration

The real borrowers are linked as individual business partners to the business partner in the role of borrower entity by creating the relationship category "Is Member of Borrower Entity [page 171]".

For more information about editing borrower entities, see Mapping Borrower Entities [page 203].

3.1.5.1.2.4 Business Partner Roles for Basel II

Definition

Business partner role [page 163] that is used in the Basel II Solution.

Structure

The following business partner roles are relevant for Basel II:

- **Collateral Partner**
  Collateral partners provide collateral instruments for their own transactions or for transactions of other business partners.
- **Issuer**
  The issuer is the debtor in a transaction involving transferable securities. The bank does not necessarily have a direct contractual relationship with the debtor.
- **Counterparty**
  The counterparty is the direct customer of the bank, and the bank has a contractual relationship with the counterparty.

3.1.5.1.3 General Extensibility

Use

The Business Partner is designed to enable you to make extensions, as the screen structure and the screen sequences are defined in control tables. Program logic can be integrated by way of firmly defined interfaces.

Features

- Extension of the Business Partner by adding attributes
  The Business Partner can be extended to include attributes.
  If you wish to make customer-specific evaluations or enter additional information about a business partner, you can add fields to existing BP roles in an existing screen or in a separate screen.
- Extension by adding BP roles:
  An extension of BP roles is also permitted.
  Extensions should only be made by developers with ABAP/4 experience, and only if specific customer applications have been programmed.
  Adding new BP roles only makes sense if a customer-specific development is being conducted. The business partner data required for this application can comprise data supplied by SAP as well as customer-specific data.
  If no BP role-specific data is required, you should first check whether the required function cannot also be mapped using the BP role *Business partner (general) , or whether it is not enough to transfer existing business partner roles .*
  You can replace the standard titles of BP roles in Customizing with your own descriptions.
- Extension to include a check on SAP fields
  You can conduct specific checks for SAP fields. For example, you can assign the check rule that a last name must contain at least three letters and one vowel.
- Extension of the relationships by adding attributes
  Relationships can be extended by adding attributes.
  You can create new relationship categories. Examples of relationships:
  - *Is a member of (club)*
  - *Is an employee of (business partner company)*
  - *Is the son/daughter of*
  - *Is the managing director of*
  - *Is guarantor for*
  - *Is guardian of*

    You define your own relationship categories in the same way as you define your own BP roles.

> ⚠ Caution
>
> Before you create your own relationship categories, you should check carefully whether this relationship category is necessary from a business point of view, and whether it would not also suffice to rename existing business partner relationship categories.

3.1.5.1.4 Business Partner Relationship

Definition

A business partner relationship represents the business connection between two business partners.

Use

In order to create a relationship between two business partners, you have to assign a business partner relationship category to the business partner relationship. The business partner relationship category describes the characteristics of the business partner relationship.

You can assign attributes (such as a firm's address for the contact person relationship) to a relationship, which prevents data being stored redundantly.

You can limit a relationship in time by entering the start date and end date of the relationship. This means that it is possible to get an overview of the periods in which certain business partners were contact persons for a company, for example.

Integration

Make the settings required for the relationships in the implementation guide (IMG) in Customizing for the *Business Partner* under *Business Partner Relationships*.

Example

Ms. Lopez at Hansen p.l.c. is the contact person for your company, Smith p.l.c.

You create a contact person relationship with the relationship category "is contact person of" between the business partner "Lopez" having the BP category *Person* , and the business partner "Hansen" having the BP category *Organization*.

You assign the firm's address to Ms. Lopez and create more data, such as the calling and visiting hours.

3.1.5.1.4.1 Business Partner Relationship Categories

The following basic business partner relationship categories are available:

-
-

-
- Relationship Category “Is Contact Person Of”
-
-
-
-
-

3.1.5.1.4.1.1 Relationship Category Customer - Customer Service Representative

Definition

From the point of view of the business partner in the BP role customer [page 165] , this specific relationship category represents the relationship category *is represented as customer by* , and from the point of view of the business partner in the BP role customer service representative [page 166] the relationship category *is customer service representative for customer* .

The relationship category *is represented as customer by* describes the status of the business partner as a customer who is dealt with by a customer service representative in the company. The relationship category *is customer service representative for customer* describes the status of a natural person as a customer service representative responsible for attending to customers who are in a business relationship with the company.

Use

A person, organization, or group can be a customer with the company in more than one occurrence and can be assigned more than one customer service representative with the relationship category *is represented as customer by* .

A natural person can be a customer service representative in the company for a customer master and can be assigned as a customer service representative with the relationship category *is customer service representative for customer* to more than one customer.

To assign the customer - customer service representative relationship uniquely, enter the relevant for this relationship when you create the relationship in the system.

You can enter a specific time period for the customer – customer service representative relationship. This gives you an overview of the periods in which the customer was dealt with by the customer service representative in question.

Example

The business partner *Marco Adams* opens a securities account in a branch of a bank, in order to buy and sell securities. He becomes a customer of this branch. From his point of view, the private customer manager Karola

Smith from the branch has been assigned to him by way of the relationship category *is represented as customer by* . From the point of view of Karola Smith, the customer Marco Adams is assigned to her by way of the relationship category *is customer service representative for customer* . This assignment allows you to record the costs and revenue relating to the customer correctly, based on the business relationship in which they originated.

3.1.5.1.4.1.2 Relationship Category "Is Member of Borrower Entity"

Definition

Relationship category that represents the relationships of persons, groups, or organizations linked together as a borrower entity in accordance with paragraph 19 section 2 of the German Banking Act.

The relationship category *Is Member of Borrower Entity* groups:

- All companies belonging to the same company group
- Commercial partnerships and each partner assuming personal liability, as well as partnerships and each partner
- Persons and companies on whose account loans are taken out, and those who take out loans under their own name

Use

The relationship category *Is Member of Borrower Entity* is used exclusively to link persons, organizations, or groups with the head of a borrower entity. This grouping of borrowers in borrower entities plays an important role in regulatory reporting for large-scale and multi-million loans in accordance with paragraphs 13 and 14 of the German Banking Act.

Integration

The head of the borrower entity that is linked to the individual members exists in the BP role of borrower entity [page 167].

In order to edit the relationship category "Is Member of Borrower Entity" using a Business Application Programming Interface (BAPI), you can use the following methods of business object *BUSISB993 Business Partner Relationship with Shareholding Data*. These objects can be found in BAPI Explorer.

Method*	Short Description
ChangeShareholder	Change business partner relationship with shareholding data
CheckShareholderExistence	Check business partner relationship with shareholding data

CreateShareholder	Create business partner relationship with shareholding data
DeleteShareholder	Delete business partner relationship with shareholding data
GetDetail	Read business partner relationship with shareholding data

For more information about editing borrower entities, see Mapping Borrower Entities [page 203].

3.1.5.2 Functions

This section contains information about

- The general Business Partner data
- The Business Data Toolset (BDT)
- The basic functions of the Business Partner, such as data cleansing and archiving, as well as extensibility and distribution of BP data.

3.1.5.2.1 Editing Business Partner Data

Use

You can use this application to create, edit, and manage business partners and use them for integration with other functions.

You can edit business partners using the following channels:

- Dialog processing
- Business Application Programming Interface (BAPI)
- Complex interface [page 235]
- Enterprise Services
- Direct input (DI)* during the

Features

You can use the following functions:

Function	Comment
Search for business partners using different search criteria	In dialog processing, you search for business partners using the . You can also search for business partners using the quick entry function in the input help.

Function	Comment
Create business partners Display and change business partners	Depending on the settings made in Customizing and the role selected, different tab pages and data are available to edit business partners.
Make to business partner data	
Use the for business partner data	
Display business partner relationships [page 200] in the dialog in graphical form	
Display the in the change documents* for a business partner	
Manage authorizations [page 206] for business partners	
Configure and extend business partner data with the help of the	
Copy business partner data from an external system	
* of duplicates	
business partners	
business partner data	
	When you distribute business partner data, the system uses a replication process to synchronize the business partner instances. It creates copies of the data from the central business partner instance in the application instances.
* business partner data	When you decouple business partner data, one or more application instances can access the data for a central business partner instance via a business partner proxy.
	You send business partner information to inform application instances about changes made to the data in the central business partner instance.
Integrate business partner data in SAP NetWeaver Business Warehouse (BW)	For more information, see SAP Help Portal at http://help.sap.com *SAP NetWeaver* > *SAP NetWeaver Business Warehouse* > *BI Content* > *Cross-Application Components* > *Business Partner*.

Caution

*The functions marked with an asterisk are not available in analytical applications.

3.1.5.2.2 SAP Business Partner Data

Use

Frame on Screen:	Description:	What you should know:
Data for initial creation of partner	When you select the *Grouping* field, you define the number range. In conjunction with the interval, this determines whether external or internal number assignment takes place. It is not possible to change this assignment at a later date. • Internal number assignment: During data entry, the BP role field remains empty for the BP number. • External number assignment: The BP role field must be filled.	You define number ranges in accordance with the number assignment in the IMG.
	Select a grouping that requires a specific field selection In the *Business Partner Type* field.	In the IMG, you can modify the field selection for the business partner type in question and define your own business partner types according to the customer-specific requirements.
Name Fields	The selection of name fields available is defined by the business partner category. You can enter the following data: • Organization: Form of address, name, legal form, industry, legal entity • Person: Form of address, first and last name, name components (name affixes and academic titles) • Groups: Form of address, two names, partner group type (marriage, shared living arrangement)	Default entries are made in the name fields that are relevant for each particular business partner category. It is not possible to change these entries.

Frame on Screen:	Description:	What you should know:
Addresses	You can enter any number of addresses for each business partner. The standard address is always maintained on the address screen. If you want to create other addresses, you can also assign address usages [page 179] to these addresses.	You can define address types in Customizing, and store proposals for a standard address type for each transaction that is relevant for determining a BP address.
Control Data	The *business partner type* is used to create groups of business partners in combinations of your choice. The *authorization group* control function means that business partners belonging to the authorization group can only be edited by end-users with the appropriate authorization. You can define business partner numbers from a legacy or operational system in the *external partner number* field. You can enter Information on gender, marital status and other personal data for individuals. You can enter data on the foundation and liquidation of organizations.	You can define business partner types and default values for amending fields for the business partner type in Customizing.
Bank details	You can enter bank details [page 185] for a business partner.	You can use transfer programs to transfer bank master data from an external system.
Payment cards	You can enter details of for a business partner.	You can maintain payment card type, payment card category and payment card blocks in the IMG.
Notes on BP	You can enter notes on every business partner. The areas of sales, accounting and marketing are provided by SAP. You can define any other subject areas you require.	To define note types go to the menu *Tools* > *Word processing* > *Settings*. Select *Change text objects and IDs* and place the cursor on BUT000. Choose *TextID* > *Createin the menu and enter a new note type.*
Relationships	You create or edit relationships between business partners in the *Relationships* screen.	

3.1.5.2.2.1 Central Data

Definition

Groups together general information about a business partner.

Use

The central data is stored in table BPTIME_BUT000.

This includes the following data:

- Business partner category
- Business partner grouping
- Business partner name and name affixes
- Search terms
- Personal data (such as date of birth)
- Organizational data (such as the legal form of an organization)
- Industry sector
- Control parameters
- Data origin
- Bank key
- Archiving flag
- Locks

You can find the central data on various tab pages in dialog mode, depending on the business partner category and business area.

Integration

You can use the Business Data Toolset (BDT) to configure the central data to suit your requirements.

> ⚠ Caution
>
> Note that the following settings are not available for the central data for business partners in analytical applications:
>
> - Locator search
> - Tax numbers
> - Payment card
> - Business hours
> - Notes and documents

3.1.5.2.2.2 Address Data

Definition

Used to create and manage addresses for a business partner

Use

You can create one or more addresses for each business partner. If you enter only one address per business partner, this is used for all activities. If you create multiple addresses, you can define when they should be used (for example, use of a certain address as the delivery address).

The following functions are available:

- Create a standard addressif you need only one address per business partner.
  This address will be used for all activities.
- Create additional addresses (in the SAP GUI on the tab page *Address Overview*) if you need several addresses per business partner.
- You can also assign an address usage(in the SAP GUI on the tab page Address Overview).
- You can create and change a standard address using the address usage *standard address* .
- You can move an existing address to a new target address, or plan this move for the future.
  Processes that access the address number of the source address are automatically forwarded to the address number of the target address from the move date. For more information on carrying out a move, see .
- You can execute a where-used list for an address.

Integration

The provide the basic functions for managing address data.

Based on the address usage that you have established, the automatic address determination recognizes which address is used for which business activity.

You can format the data according to your requirements if you want to print address data in your application, for example, in a document. In the Business Partner dialog you can display a print preview in a dialog box for the formatted address data. For more information on print formatting, see .

You can create several for an address. This allows you to format address data and also name fields (e.g. first and last name, academic title, etc.) for all business partners in different fonts, each with its own character set.

The can be used to check whether the address data (postal code, city and street) is correct. For more information, see .

Furthermore, the BAS provide interfaces for connecting postal checks of external providers. For more information, see and .

> **Note**
>
> If you search in the locator for business partners by means of address versions, the version of the logon language is always displayed in the results list. Thus, if you search for a Cyrillic address version, for example, but are logged on in English; the English address, and not the Cyrillic address is displayed in the description for the business partner. You can then call up the relevant international versions in the business partner in question.

In the Implementation Guide (IMG) you define the activitiesand address typesyou need, and establish whether several addresses may be assigned to an address type. In addition, you can assign activities to certain address types.

Make the necessary settings in Customizing of the *Business Partner* under *Basic Settings* > *Address Determination.*

When you enter a gender-specific form of address (e.g. Mr., Mrs.) in the SAP GUI, or change the form of address, the assignment of the gender is automatically adapted on the tab page *Identification* . Make the settings for the dependency of the fields *Title* and *Gender* in Customizing under *SAP Web Application Server* > *Basis-Services* > *Address Management* > *Forms of Address and Name Components* > *Maintain Texts for Forms of Address.*

Activate the international address versions that you require in Customizing under *SAP WebApplication Server* > *Basis-Services* > *Address Management* > *International Settings* > *Activate International Address Versions.*

Example

Address Move

On the 15.04.2005 you receive the information that the business partner Julie Armstrong is moving to London on the 01.06.2005. You create Julie Armstrong's new address with the validity start date of the 01.06.2005. You also create a move with the move date of the 01.06.2005 and the new address in London. The system automatically restricts the validity of the old address to one day before the move date.

If a process accesses the old address, the system determines the data for the new address by means of the address number of the old address.

3.1.5.2.2.2.1 Standard Address

Definition

Each business partner can have more than one address [page 177]. One of these addresses is defined as the standard address. In the applications that use the business partner, you can use a different address to the standard address.

Use

The standard address is used if you do not specify another address. When you create a new business partner, the address you enter is defined as the standard address, and is valid indefinitely.

You can enter alternative standard addresses and validity periods on the *Address Overview* tab page, using the *Create* pushbutton.

For more information, see Assign Address Usage [page 179].

If the validity period you defined in the address overview for the standard address has ended without the standard address being replaced by another address, then the last address to occupy the position retains the standard address status. When you define the validity periods for standard addresses, the system requires the standard address to be filled at all times.

If the address usage you selected is not valid in the applications that use SAP Business Partner, the standard address is used.

Integration

You make settings for the use of different address to the standard address in Customizing for Business Partner.

3.1.5.2.2.2.2 Assigning Address Usage

Prerequisites

You have created several addresses in the address overview and defined the required address types in Customizing.

Procedure

1. Choose the *Address Overview* tab page in the Business Partner dialog .
   The *Address Usages* maintenance screen appears.
2. Select an address type (for example, delivery address) and choose *Create* .
3. Select the corresponding address in the dialog box, create a validity period and choose *Continue* .
   You have assigned an address to an address usage. The selected address is identified as standard usage.
   If you have assigned several addresses to an address type, the indicator *Standard* appears next to the first address entered. If you wish to define a different address as the standard usage for the address type, select the required address and choose *Standard* .
4. Create a standard address for a business partner via the address usage *standard address.*
   You can assign several standard addresses with sequential validity periods to a business partner.

> **Caution**
>
> If an address exists for a business partner, a standard address must always be assigned. If you create a new address within a period in which the standard address you created first is not valid, the system automatically defines this address as the standard address for this period.

5. Save your data.

> **Note**
>
> You can use the pushbutton *Validity* to change the validity period of an address usage .

Result

You have assigned one or more addresses to an address usage.

Example

In addition to the standard address, you require addresses for certain activities. For example, for a business partner, you want to send letters to a different address to the one used for deliveries.

You must have defined the address types *correspondence address* and d *elivery address* in Customizing.

You assign the address to which letters should be sent to the address type c *orrespondence address,* and the address to which deliveries should be sent to the address type *delivery address.*

3.1.5.2.2.3 Communication Data

Definition

Contains information about making contact with a business partner.

Use

You can enter various communication data for a business partner, for example, telephone and fax number, e-mail and other communication data, such as the pager number.

You can make one or more entries for a communication method. If you make only one entry for a communication method, this is automatically indicated as the standard entry. You can change the standard entry assignment.

In addition you can determine that an entry for a particular communication method should not be used. You can also enter remarks (for example, for preferred calling times).

You also have the option of determining the s *tandard communication method* , meaning that you can establish the method via which a business partner should be contacted.

Structure

A distinction is made between the following communication data.

- Communication data that is linked to an address (for example, a fixed network number)
  You can enter such data only in combination with an address.
- Address-independent communication data
  This allows you to create communication data (for example, for a mobile telephone number) for a business partner without having to enter a postal address.
  The country is suggested from the existing user parameter or, if this has not been entered, from the user address.

> **Note**
>
> You can preset a country key so that you do not always have to assign the country key manually for the address data and the address-independent communication data. To do this, choose *System* > *User Profile* > *Own Data* in the menu in the SAP GUI, and enter the parameter ID **LND** on the tab page *Parameters* and a valid country key as a value.

3.1.5.2.2.3.1 Conversion of Communication Data

Use

If you have created a business partner with address-independent communication data only and you then enter an address for this business partner, you have the option of converting the address-independent communication data into address-dependent communication data.

In the same way you can convert address-dependent communication data into address-independent communication data. This is necessary, for example, if several addresses have been entered for a business partner, and the mobile phone number of the business partner should apply to all the addresses.

Prerequisites

If you want to convert address-independent communication data into address-dependent communication data, you need to have created at least one address and have made one or more entries in the address-independent communication data of the required business partner.

3.1.5.2.2.4 Industry

Definition

Categorization of business partners (of the business partner category *Organization*) according to their industries.

Use

You can assign various industries to a business partner, after making the corresponding settings for these industries in Customizing.

The assignment of business partners to industries provides you with another characteristic for identifying business partners.

Conditions that apply to specific countries or industries can be represented using an industry system. The use of industry systems defined according to different criteria allows you to carry out analyses of business partners, for example.

Integration

Make the necessary settings in the Implementation Guide (IMG) in Customizing of the *Business Partner* under *Organizations* > *MaintainIndustry Systems and Industries.*

3.1.5.2.2.4.1 Assigning Industries

Use

You categorize business partners by assigning them to one or more industries.

Prerequisites

In the Customizing settings, you have defined at least one industry system, to which you have assigned several industries.

Procedure

1. *Choose the tab page Identification in the processing screen Change Organization* .
   If you have defined an industry system as the standard industry system in Customizing, this will be displayed automatically in the business partner dialog on the initial screen.
   If you have defined several industry systems, you can display them using the pushbutton *All Industry Systems* .
2. Select the required industry using the F4 help and assign it to the business partner.
   The hierarchical representation of the industry system that is defined in Customizing is displayed when you call up the input help.
   When you select an industry, the system automatically provides the description of the industry and the relevant industry system.
3. If the selected industry is the standard industry of the business partner, you should set the corresponding indicator in the dialog.
   You should always identify one industry as the *standard industry* for the standard industry system.
4. Save your entries.

Result

You have assigned a business partner to one or more industries.

3.1.5.2.2.5 Tax Number

Definition

An identification number issued by the tax authorities to taxpayers.

You may need to quote your business partners' tax numbers in business correspondence with them (such as in invoices), or when you file reports with the tax authorities. The exact requirements vary from country to country.

Use

You enter your business partners' tax numbers in the business partner master data, on the *Identification* tab. The system automatically runs checks on some tax numbers (for example, to make sure that the check digit is correct).

Integration

In Customizing for *Business Partner,* you specify which tax number categoriesyou need to be able to enter in your business partner data, by choosing *Basic Settings* > *Tax Numbers* > *Maintain Tax Number Categories* .

3.1.5.2.2.6 Identification Number

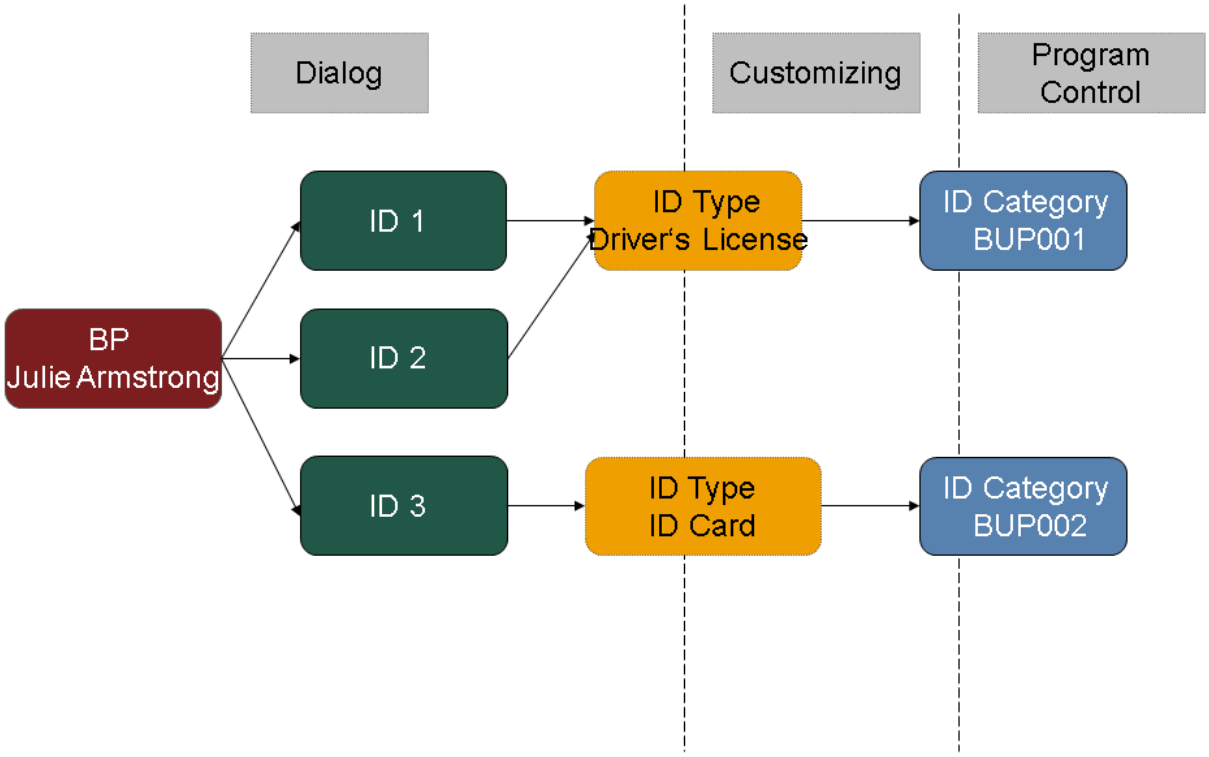
Definition

An alphanumeric key provided by external sources that can be stored as an attribute for a business partner in the system. The identification number can be used as an alternative to the business partner number to identify a business partner.

Use

In the dialog you can store as many identification numbers as you like for a business partner, to distinguish this business partner as clearly as possible You can select the relevant identification typeand assign an identification number to the business partner.

The concept of identification numbers is represented in the following graphic: Several identification numbers for various identification types, including two driver's license numbers, have been created in the dialog for the business partner Julie Armstrong.



Structure

For more information on the concept of the identification numbers, see the Implementation Guide (IMG) of the *Business Partner* .

Integration

Make these settings, as well as settings for the identification type and identification category in Customizing of the *Business Partner* under *Basic Settings* *Identification Numbers.*

You can make use of the data cleansing services of a business information provider (BIP) that assigns BIP identification numbers in the context of data comparison. For more information, see the documentation of the reports BUPA_BIP_FILE_EXPORT and BUPA_BIP_FILE_IMPORT.

Example

You work in administration and maintain master data. You create the ID card number for the identification type *ID card* for Julie Armstrong in order to identify this business partner clearly.

After doing so you can clearly determine the holder of the ID card, Julie Armstrong,by carrying out a search using the ID card number.

3.1.5.2.2.7 Bank Details

Definition

Contain all the data for the bank accounts of a business partner.

Use

You can enter any number of bank details for each business partner.

The bank and the account can be clearly identified in the system by means of the country key of the bank, the bank key and the bank account number. If the bank referred to does not already exist in the system, you should enter it in the system when you create the bank details.

The bank data contains additional information about the bank in question, for example, the address, and control data, such as the bank routing number.

Furthermore you have the possibility to define an international bank account number (IBAN) in the bank details.

You can:

- Generate an IBAN for bank details and transfer it to your bank data

  > **Note**
  >
  > You have to trigger this generation explicitly using the pushbutton *Maintain IBAN* in the column *IBAN* in the SAP GUI.
  >
  > You can recognize whether an IBAN has been defined, because the icon on the pushbutton changes color and the quick info reads *IBAN Maintained.*

- Derive bank details from an IBAN that you know of, and enter the relevant bank details ID

In addition, you can carry out a change for bank details, or plan this change for the future. Processes that access the ID of the source bank details are automatically forwarded to the ID of the target bank details from the change date. For more information on carrying out a change, see .

Integration

Read programs can be used to import bank master datafrom an external data medium to the database. For more information, see the Implementation Guide (IMG) under *Bank Directory* > *Bank Directory Data Transfer.*

You can make settings for bank data checks in Customizing of SAP NetWeaver underGeneral Settings > *Set Countries* > *Set Country-Specific Checks.*

Example

Changing Bank Details

You are informed on the 01.05.2005 that the business partner Hanson p.l.c. will change to the Deutsche Bank on the 01.08.2005. You create the new bank details of Hanson p.l.c. with the validity start date of the 01.08.2005. You also create a change with the change date of the 01.08.2005 and the new bank details. The system automatically restricts the validity of the old bank details to one day before the change date.

If a process accesses the old bank details, the system determines the data for the new bank details by means of the ID of the old bank details.

3.1.5.2.2.8 Status

Definition

Includes different characteristics for the status of a business partner (for example, the archiving flag or a lock)

Use

You can freely define and assign the user status.

To do this you define a status profilein Customizing and assign several user statuses to it.

The user statuses of the status profile assigned to the object type *Business Partner* are offered to you for selection in line with the settings you have made in Customizing.

For more information on the status profile, see the documentation of the Implementation Guide for the Business Partner under *Basic Settings* > *Status Management* .

3.1.5.2.2.8.1 Assigning User Status

Prerequisites

You have defined at least one status profile in the Customizing settings and assigned several user statuses to that profile.

Procedure

You are in the dialog in the *General Data* on the tab page *Status.*

You have to assign a status profile to the business partner before you can define user statuses for that business partner. The procedure depends on whether you have already assigned a status profile to a business partner.

Assign a status profile and user status

1. If you have not yet assigned a status profile to the business partner, choose *Assign Status Profile* .
   This brings you to the dialog box *Status Profile Selection* .
   Assign a status profile to the business partner in order to be able to define a user status for the business partner.

   > **Note**
   >
   > You can assign only one status profile to a business partner. It is not possible to cancel this assignment at a later date.

2. When you have assigned a status profile, you reach a dialog box in which the defined user statuses are displayed for selection.
   Select the required user status and copy it.
   You have now assigned a status to a business partner.

   > **Note**
   >
   > User statuses to which no status number has been assigned in the Customizing can be set at any time and can be cancelled.

On the other hand, user statuses that have status numbers cannot be deleted, but can be replaced only by other user statuses of the same numbering scheme.

Assign or change user status

1. *If you have already assigned a status profile to a business partner and want to change the user status, choose Insert Status* .
   This brings you to a dialog box in which the defined user statuses are displayed for selection.
2. Select the required user status and copy it.
   You have now assigned a status to a business partner.

3.1.5.2.2.9 Notes and Documents

Definition

User-defined texts or files (text files, graphics, tables) that can be stored for a business partner.

Use

You can create notes in different languages for every business partner, using the SAPscripttool. You can define note types with the help of text IDs. You can also define note views and assign them to BP roles.

You can also store new or existing documents. The Business Document Navigator (BDN), which you can call up in the SAP GUI via *Extras* > *Documents* , can be used for this purpose.

For more information on the BDN, see the SAP Library under *SAPNetWeaver* > *Application Platform* > *Business Services* > *Business Document Navigator* .

Integration

Make the necessary settings in the Implementation Guide (IMG) in Customizing of the *Business Partner* under *Basic Settings* > *Notes* and in *General Application Functions.*

3.1.5.2.2.10 Ratings

Definition

Object with which you can represent rating results for the business partner using established rating procedures.

Use

External service providers, government agencies, or departments within the company rate natural or legal persons for various business processes. The ratings take account of their payment history, credit standing, or other criteria.

You use this object to rate business partners according to various rating procedures, specifying a unique validity period.

You can save multiple rating results for each rating procedure. Note that you can map only one rating result for a particular rating procedure within a fixed validity period.

You can create ratings and trends for a rating procedure only if the business partner grants you permission to acquire the rating and save the results.

You can use *Dun & Bradstreet* or *SCHUFA* as external providers of rating procedures, and the *German Federal Financial Supervisory Authority (BaFin)* can also provide rating procedures as a state authority.

Structure

The ratings for the business partner consist of the following attributes:

- Rating procedure
- Rating permission
- Rating
- Trend
- Valid-from
- Valid-to
- Rated on
- Text

Integration

You make settings for the ratings in Customizing for *SAP Business Partner for Financial Services* under *Settings for Financial Services* > *General Settings* > *Ratings* > *Make Settings for Rating Procedures and Ratings* and *BAdI: Calculate Rating*.

Example

You want to check the credit standing for your customer *Build Inc.*. You obtain permission from the customer to carry out a rating and save the results using the *Moodys* rating procedure, and rate the business partner on *March 1, 2003*. The rating is intended to cover the period from *March 1, 2003* to *March 1, 2005*. The customer's credit standing is excellent, and the business partner is given the rating *Aa1*. You expect the customer's rating to remain the same in the long term too, and define the trend as *constant*.

3.1.5.2.2.11 Regulatory Reporting Data

Definition

Attributes for regulatory reporting requirements that can be stored in the business partner.

Use

Regulatory reporting data is used in the context of the banking or insurance supervisory authority to enable the revenue performance of banks, insurance companies, and financial services providers under supervision to be analyzed. The main emphasis of the supervisory regulations lies in areas such as own resources, liquidity, foreign trade, and loans business in the respective institutions. Regulatory reporting data can also be used for banking statistics, or the compilation of information about foreign trade for the German Federal Bank. It is also used to assess the tax compliance of the customers of the respective institutions.

The reports are based on the legal requirements of the German Banking Act (GBA), the German Asset Regulation, or the U.S Foreign Account Tax Compliance Act (FATCA). Organizations subject to these supervisory requirements can submit their reports to the corresponding supervisory institutions (German Federal Bank, German Federal Financial Supervisory Authority, U.S. Internal Revenue Service (IRS), and so on).

Structure

You can store attributes from the following areas of regulatory reporting in the business partner:

- Regulatory reporting data in the context of the German Banking Supervisory Body of the German Federal Financial Supervisory Authority (BaFin)
  This includes attributes such as the loan to managers, multimillion loan, borrower and borrower entity number, credit standing check in accordance with the GBA, non-resident, company relationship, German Federal Bank country code, risk group, group affiliation, monetary financial institute, GBA-relevance, large-scale loan monitoring, and crediting relief.
- Regulatory reporting data in the context of the German Insurance Supervisory Body of the German Federal Financial Supervisory Authority (BaFin)
  This includes attributes such as legal entity, paragraph 3 section 2 of the German Asset Regulation, debtor group, and paragraph 1 section 1 no. 20b of the German Asset Regulation.
- Regulatory reporting data in the context of the Austrian Financial Market Authority (FMA)
  This includes attributes such as the OeNB ident number, ident number assignment, OeNB target group, OeNB institution code, and OeNB reporting obligation.
- Regulatory reporting data in the context of tax compliance regulations (for example, the submission of reports to the IRS in the context of the U.S. Foreign Account Tax Compliance Act (FATCA))
  This includes attributes such as tax compliance ID, tax compliance type, country, region, tax compliance status with reason, validity period, rating date, customer's agreement to transfer data, and agreement date.

Integration

You make settings for regulatory reporting data in Customizing for *SAP Business Partner for Financial Services* under *Settings for Financial Services* > *General Settings* > *Regulatory Reporting Data*:

- Define OeNB Target Groups
- Define Recipients of Loans to Managers
- Define GBA Reporting Information
- Define Country Indicators in Accordance with German Federal Bank
- Tax Compliance
  - Define Tax Compliance Types
  - Assign Tax Compliance Types to Countries/Regions
  - Define Reasons for Tax Compliance Status

> **Note**
>
> If you want to use the *Tax Compliance* function, you have to activate the (`CA_FSBP_TAX_C`) business function.

3.1.5.2.2.12 Legal Data

Definition

Legal information about business partners that are categorized as *Organizations*.

Structure

You can store the following legal data in the business partner:

- Registered office, with the country, region, and city
- Balance sheet data, with the currency of the balance sheet, type of balance sheet display, and the year and amount of the most recent capital increase

3.1.5.2.2.13 Fiscal Year Information

Definition

Object belonging to the business partner where you can store figures from the financial statement and from the profit and loss statement of a company. In addition, you can store business key figures about the credit standing of a company here.

Use

You can use the information from the annual financial statements and the consolidated financial statements, as well as the company's business key figures, to gain a comprehensive insight into the business partner's economic circumstances. This can help you to make decisions about the creditworthiness of the company.

In accordance with the regulations contained in Paragraph 18 of the German Banking Act (GBA) concerning the credit standing review, banks are also obliged to request that borrowers disclose their economic circumstances, in particular by submitting annual financial statements, for loans above a certain amount.

You can edit fiscal year information for business partners belonging to the category *Organization*. You can create, display, change, and delete fiscal year information for several fiscal years. If you want to edit particular fiscal year information, you can scroll from one fiscal year to the next. Alternatively, you can go directly to selected fiscal years.

Structure

In the business partner, you can store the following fiscal year information for each fiscal year that you select:

- Dates
  This includes the start and end of the fiscal year, the date of the annual stockholders' meeting, the date and indicator of the annual financial statement, and the date of the consolidated financial statement.
- Financial statement item figures
  These include balance sheet currency, balance sheet total, equity, share capital, issued stock capital, participation certificates outstanding, capital reserve, legal retained earnings, retained earnings for own shares, retained earnings based on company bylaws, other retained earnings, subordinated liabilities, and profit and loss carried forward.
- Profit and loss account figures
  These include gross and net contributions, profit reported, annual sales, and annual net profit.
- Key figures
  These include debt ratio, return on total capital employed, debt clearance period, financing coefficient, and equity ratio.

3.1.5.2.2.14 Credit Risk

Definition

Object that you can use to map ratings for a business partner using probabilities of default, weightings, and correlation models.

Use

You can use Credit Portfolio Data Processing (CPDP) [page 1292] to rate the credit risk of a business partner. You can enter a probability of default type for each business partner for various time frames, regardless of the portfolio.

By entering weightings and a correlation factor, you can specify a borrower's dependency on particular risk factors in the correlation model. The standard system includes the weighting types **Country** and **Industry** in the correlation model, to which you can assign individual countries and industries as weighting factors. You can also define your own weighting types and weighting factors as required.

You use the default probability to assign a value to a probability of default type for a particular validity period.

You use the correlation model to assign weighting factors and weighting values to a weighting type for each correlation factor **R^2**.

Structure

The system displays the credit risk for a business partner in the following group boxes with the following attributes:

Group Box	Attributes
Default probability	• Probability of default type • Validity start date • Validity end date • Probability of default value • Key date for displaying the probability of default
Correlation model	• Correlation model • Correlation factor R^2 • Weighting type key • Weighting factor • Weighting value • Weighting type filter • Last change to correlation model data

You make settings for the credit risk in Customizing for *SAP Business Partner for Financial Services* under *Settings for Financial Services* > *General Settings* > *Credit Risk*:

- Define Correlation Model
- Define Probability of Default Types
- BAdI: Weighting Types

More Information

- Extraction Run for Business Partners [page 1298]
- Creation of Business Partner Versions [page 216]

- Display of Business Partner Versions [page 213]

3.1.5.2.2.15 Employment Overview

Definition

Object in which you can store current and past employment relationships for a business partner.

Use

This gives you an overview of the individual employment relationships of a natural person.

You can display the entire employment record in chronological order according to the start date of employment, or select specific instances of employment up to a certain key date.

You can store as many employment records for a business partner as you want in the employment overview. The validity periods of the employment records can also overlap.

Structure

The employment overview consists of the following attributes:

- Start of employment relationship
- End of employment relationship
- Occupation
- Description of occupation
- Employer
- Industry
  The industry sector of the employer in the standard industry system is used.
- Description of industry sector
- Employment status
- Description of employment status

Integration

You make the employment data settings in Customizing for *SAP Business Partner for Financial Services*. You can make employment status settings under *Settings for Financial Services* > *General Settings* > *Basic Settings* > *Natural Person* > *Define Employment Status*.

You can edit occupations under *General Settings* > *Business Partner* > *Persons* > *Occupations* > *Maintain Occupations*.

You can make industry sector settings under *General Settings* > *Business Partner* > *Organizations* > *Maintain Industry Systems and Industries*.

Only those industries included in the standard industry system are displayed in the employment overview.

Example

Your business partner *Marcus Adams* is a *bank employee* and was employed at a bank from *August 1, 1999* to *December 1, 2002*. He has been working for an insurance company since *January 1, 2003*.

3.1.5.2.2.16 Additional Data

Definition

Object that you can use to store customer-defined attributes as fields for free use (and that can be set in Customizing), free text fields, and checkboxes with their own semantics in the business partner according to customer requirements.

Use

In Bank Analyzer, use the segmentation service [page 1843] , instead of *additional data, additional information* , or the *Easy Enhancement Workbench,* as a tool for enhancing the *SAP Business Partner for Financial Services* . The segmentation service connects the Bank Analyzer system to versioning [page 209] and the Source Data Layer tools.

Structure

Additional data for the business partner consists of the following attributes:

- 11 fields for free use that can also be set in Customizing
- 8 free text fields
- 10 checkboxes for free use

Integration

You can make customer-specific settings for additional data in Customizing for *SAP Business Partner for Financial Services* under *Settings for Financial Services* > *General Settings* > *Enhancement Options* > *Additional Data* > *Define Category for Additional Data or Determine Values for Additional Data.* For more information about setting up additional data, see the documentation for the relevant IMG activities.

3.1.5.2.2.17 Additional Information

Definition

Customer-specific attributes that can be stored in the business partner in the form of categorized information typeswith corresponding customizable fields, text fields, checkboxes, amount fields, currency fields, and date fields according to user.

Use

Information types are categorized with their corresponding additional information by information categories.

You can also differentiate additional information by means of the *Differentiation* pushbutton.

In *Bank Analyzer* you should use the segmentation service [page 1843] as the enhancement tool for the *SAP Business Partner for Financial Services* , and not *Additional Data* , *Additional Information,* or the *Easy Enhancement Workbench* . In Bank Analyzer, the *segmentation service* brings about the link to the versioningfunctions and the *Source Data Layer* tools.

Structure

Additional Information consists of the following customer-specific attributes:

- Information category
- Information type
- Description of information type
- Field with F4 input help
- Description of field with F4 input help
- Checkbox
- Text field
- Amount field
- Valid from date

- Valid to date
- Additional date

Integration

You make settings for additional information in Customizing for the *SAP Business Partner for Financial Services* under *Settings for Financial Services* > *General Settings* > *Enhancement Options* > *Additional Information* > *Set Information Category.*

If you have already created information categories, when you upgrade to Release 5.0 see SAP Note *801445* for additional information. Also refer to the documentation for the relevant IMG activity.

Example

You want to store information about a business partner's leisure activities.

Choose the information category *01 Leisure* . The information category *01 Leisure* includes the information types *01 Sport, 02 Hobby, and 03 Travel* . The business partner actively participates in sports, you therefore choose information type *01 Sport* . You can create the following customer-specific additional information for the information type *01 Sport* : in the customizable field *Type of Sport* you can choose the type of sport in which the business partner participates. As the customer actively participates in this type of sport, set the indicator in the checkbox *Active.* In the text field you can store *Additional Information* about the sport for the business partner, and in the amount field *Costs* you can enter how much money the business partner spends on this sport. In the date field you can store when the business partner participates in this sport, and when you discovered that the business partner participates in this sport.

3.1.5.2.2.18 Basel II Attributes

Definition

Indicators that are used to control the analysis of companies and organizations in risk analyses for Basel II.

Structure

The structure definition of the Basel II fields is divided into attributes and methods:

Attributes

In the *SAP Business Partner for Financial Services* , you can store the following Basel II attributes for business partners categorized as *Organizations* :

- Consolidated annual sales are less than the threshold for small- and medium-sized entities
- Substitute total assets for total sales
- Treat counterparties as retail business
- Business partner is a risk-free institution
- MDB is rated as a risk-free institution
- Business partner is subject to regulation framework
- Debt instrument of low specific risk was issued
- Business partner is a promotional institution
- Include contract terms of less than 15 days
- Specialized in the interbank and government bond markets
- Special treatment

Methods

You can use the following functions to process Basel II attributes in the *SAP Business Partner for Financial Services* :

- Dialog
- Business Application Programming Interface (BAPI)
- Data transfer by means of the complex interface

Dialog

You can create and process Basel II attributes online on the *Maintain Business Partner* screen. They are displayed on the *Bank Analyzer Attributes* tab page.

Business Application Programming Interfaces

You can call the following methods using the BAPI Explorer:

Method*	Short Text
FSBaselAttrGet	Read Basel II Attributes
FSBaselAttrSet	Change Basel II Attributes

*You can execute all the methods in a test run.

Complex Interface

Function Module	Transfer Structure
BUPA_INBOUND_MAIN_SAVE	BUS_EI_EXTERN

We recommend that when you transfer data you use the complex interface, which is based on the BAPIs for the business partner.

3.1.5.2.3 Relationship Data

You can find more information on the relationship data in the following sections.

3.1.5.2.3.1 Central Relationship Data

Use

You use control tables to specify whether attributes can be maintained for a relationship category.

To date, it has been set as standard that attributes can be maintained for the contact person relationship and shareholder relationship.

You can use the following data and functions, which you can process in the dialog as follows:

Contact Person: General Data	You can specify a VIP indicator for the business partner. You can also enter data on the department, function and power of attorney of the business partner, as well as free text as a comment.
Address Data	In addition to telephone and fax numbers, and email address, you can also enter address details such as company department and name of function as user-defined text. If a company's address is maintained, you can assign it.
Address Overview	You can view the address data of the contact person in the Address Overview. You can also change the assignment of the firm's address, and assign additional firms' addresses (if, for example, the business partner has an office in several locations).
Address Print Preview	This shows you the address of the contact person in the print preview.
Business hours	You can enter the times a contact person can be visited or called. The are integrated into the relationship maintenance for this purpose.

Shareholding Data	You can specify the shareholding percentage as well as the actual amount, together with the currency. With the *Control* indicator you can state whether it is a controlling interest on the basis of a controlling agreement or key shareholding. A controlling interest is not strictly limited to one business partner. For example, if two business partners each hold an interest of 50%, both can have a controlling interest.

> **Note**
>
> If you have specified an own relationship category (see Extensibility [page 168]), you can define attribute maintenance for this relationship category.

3.1.5.2.3.2 Displaying Business Partner Relationships

Use

You can display business partner relationships in one of the following ways in the SAP GUI:

- As a list
- As a hierarchy
- As a network

Integration

You can hide the selection fields for the list display in the Implementation Guide (IMG) in Customizing of the *Business Partner* under *Business Partner Relationships* > *Basic Settings* > *Field Groupings.*

> **Note**
>
> From Release 6.20 it is possible to hide the fields by means of the IMG activity *Configure Field Attributes for Each Client* (application object BUPR). Up to Release 6.10, you can use the IMG activity *Configure Field Attributes for Each Activity* (activity '02 change'), which is also still available from 6.20.

Make the necessary settings for the layout management of the list view under *Assign Layout Groups to Relationship Categories.*

You can extend the relationship overview by adding user-defined fields with the help of a Business Add-In (BAdI).

Prerequisites

You have maintained one or more relationships for a business partner.

Features

List

In the *Overview* of the list display, you can see all the relationships of a business partner for all relationship categories. Select a particular relationship type to display the relationships of that relationship category.

From the list, you can click on the pushbutton *Change Relationship* , to access process the detailed data for a relationship.

You can restrict the number of selected relationships by means of different selection criteria and their combinations (selection fields).

> **Note**
>
> Only the first 200 relationships of a business partner are read by default, for reasons to do with the runtime. If the number of relationships exceeds the maximum number of hits, the selection fields are automatically displayed. You can expand or collapse the selection area manually.

In addition to standard communication data such as fax, telephone and e-mail that belong to the standard address, additional data, such as the standard relationship, is also displayed in the list for the contact person relationship

Other fields for the display, which you can select when you personalize the lists, are also available to you

You can adapt the list view to your specific requirements. You can use the standard functions of the SAP List Viewer(ALV) for carrying out personalization with the help of the layout management. For more information, see the user documentation for the SAP List Viewer.

> **Note**
>
> When you switch to a different business partner or a different tab page, the settings for the selection area (for example, the setting for expanding) are transferred for each tab page from the last processing transaction of the BP in question. The settings in the selection area are not transferred when you call up the transaction again.

Hierarchy

In the hierarchical display format, you can see all the relationships of a business partner belonging to each relationship type you have selected on a specified key date (1:n view). You can also select the appropriate display level in order to display a complex relationship set for a BP. If you select a high level, for example, for the contact person of a BP with the category *Organization* that you are currently processing, other organizations are displayed that also have this contact person.

From the hierarchical display, you can access and process the detailed data of the relationship by clicking on the pushbutton *Change Relationship* . The relationship is maintained from the lower-level business partner.

> **Note**
>
> Because of the hierarchical display format (1:n), it is not possible to map additional relationships for a business partner that is already displayed. For this reason, additional incoming and outgoing relationships to business partners that are already displayed are shown in the form of an icon.
>
> *To display these relationships, select the corresponding business partner, right-click on it, and select Display Relationships* . Choose *Incoming* or *Outgoing relationships.* A detail screen appears. Select the business partner for the required relationship in order to be able to maintain the detailed data.

Network

In the network display, all the relationships of a business partner of a selected relationship type on a specified *Key Date* (1:n view) are displayed graphically. The business partners are represented as nodes, and the business partner relationships are represented as lines between the nodes.

The graphical display format allows you to view all the incoming and outgoing business partner relationships as well as the relationships for additional business partners simultaneously. You can select the display level here as well, and use pushbuttons to enlarge or reduce the display. You can choose between portrait and landscape format when displaying relationships.

You can maintain both business partners and relationships from the network display. In addition, you can create relationships of the selected relationship category between any two business partners depicted in the network, by means of the pushbutton *Insert dependency* .

Overview of All the Display Formats

	Display Format:	Maintenance	Display Level (Depth)
List	**A list of all the relationships of a BP to all relationship types.**	**Maintenance of relationships possible**	**Selection not possible**
Hierarchy	Hierarchical display of all relationships of a BP of a particular relationship type (1:n display). Additional relationships with existing business partners are identified by an icon.	Maintenance of relationships possible	Selection possible
Network	Graphical display of all relationships of a BP of one relationship type (n:m display). The business partners are represented as nodes, and the business partner relationships are represented as lines between the nodes.	Maintenance of business partners and relationships possible	• Selection possible • Possibility to enlarge and reduce • Choice of vertical or horizontal format

3.1.5.2.4 Mapping Borrower Entities

Use

You can use this procedure to group together persons, groups, or organizations in borrower entities in accordance with paragraph 19 section 2 of the German Banking Act.

Borrower entities are created for the following business partners in particular:

- All companies belonging to the same company group
- Commercial partnerships and each partner assuming personal liability, as well as partnerships and each partner
- Persons and companies on whose account loans are taken out, and those who take out loans under their own name

This grouping of borrowers in borrower entities plays an important role in regulatory reporting for large-scale and multi-million loans in accordance with paragraphs 13 and 14 of the German Banking Act.

Borrower entities represent single-level hierarchies.

Prerequisites

You have made the following settings in Customizing for *SAP Business Partner for Financial Services*:

- You have made the necessary settings under *General Settings* > *Business Partner Relationships*.
- You have defined the identification categories and identification types for the borrower entity under *General Settings* > *Business Partner* > *Basic Settings* > *Identification Numbers*.

Procedure

The *Maintain Business Partner* screen appears.

Creating the Head of the Borrower Entity

1. You create a business partner with the category *Group* in the BP role of borrower entity [page 167] as the head of the borrower unit. You can use an existing business partner belonging to the *Group* category and add the role of borrower entity to this business partner, or create a "virtual" business partner that acts as an anchor for real business partners, linked by relationships.
   You can assign the group type *Borrower Entity* to this business partner on the *Control* tab page.
2. You enter the borrower entity number as the external identification number of the borrower entity in this business partner on the *Identification* tab page under the identification numbers [page 184].

Linking the Members of the Borrower Entity

- You link all the persons, organizations, or groups belonging to the borrower entity directly, as independent business partners, to the head of the borrower entity using the relationship category Is a Member of a Borrower Entity [page 171].

Persons, organizations, or groups belonging to the borrower entity can be members of several borrower entities. You can determine the percentage and the amount of the capital investment in the shareholding data for the relationships, and indicate whether a controlling relationship exists.
You create the relationship *Is Member of Borrower Entity* in addition to representing relationships from a company law perspective, as *Is Parent Company of*, or *Is Shareholder of*, for example.

- You can assign additional business partners to a borrower entity using the relationship category *Is Member of Borrower Entity*. You can remove individual members from the borrower entity either by restricting the validity period of the relationship or by deleting the relationship from the borrower entity.
- You can remove borrower entities. To do this, you restrict the validity periods of existing relationships between members and the head of the borrower entity, or delete the relationships. You can archive, but not delete, the head of the borrower entity.

Result

You have created borrower entities from persons, organizations, or groups according to various criteria, in accordance with paragraph 19 section 2 of the German Banking Act, by linking them to one another in relationships.

You can search for members of a borrower entity using the business partner search function *ID Numbers* by entering the identification type *Borrower Entity* and the identification number *Borrower Entity Number*.

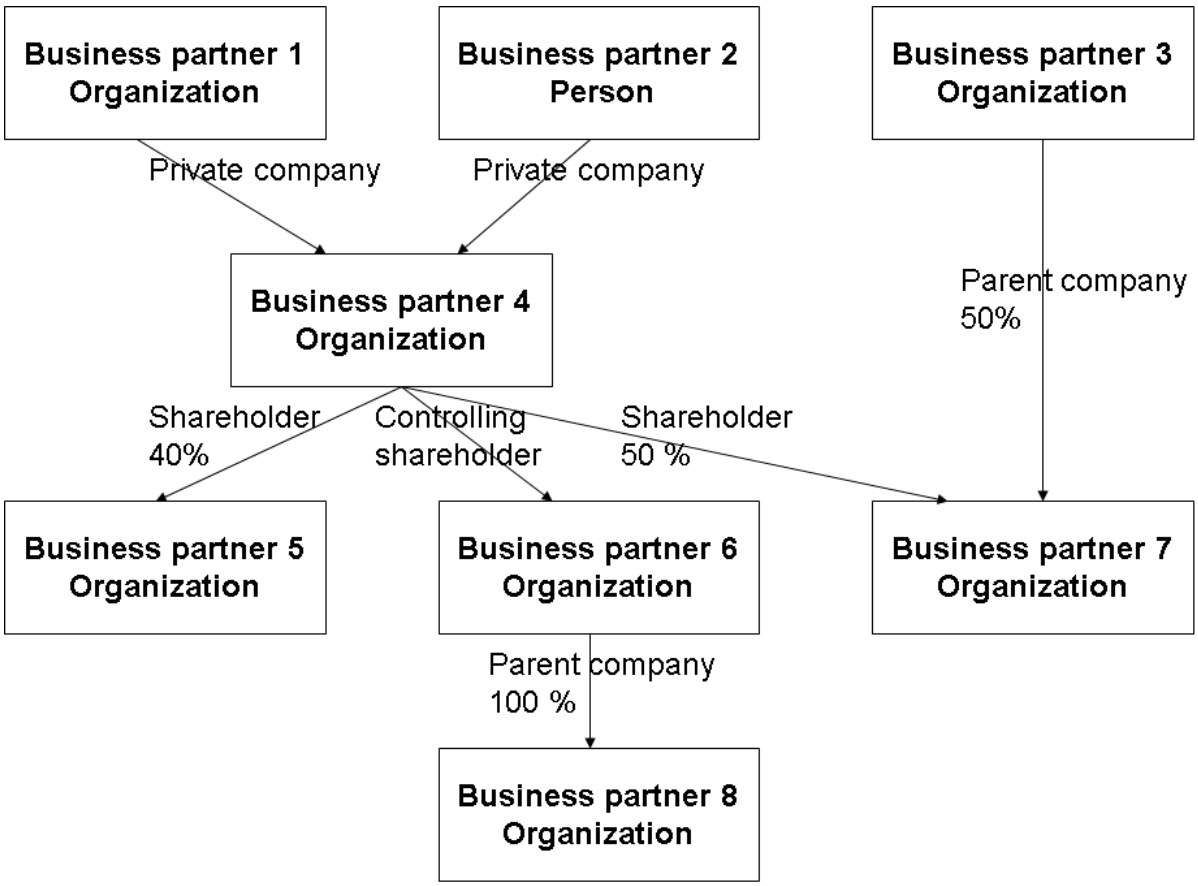
Example

See Examples of Three Borrower Entities [page 204].

3.1.5.2.4.1 Examples of Three Borrower Entities

A relationship set of eight business partners exists that leads to the formation of three borrower entities. The relationship categories *Is Member of a Partnership*, *Is Shareholder of*, and *Is Corporate Group Parent of* are used to illustrate the possible relationships between the business partners.

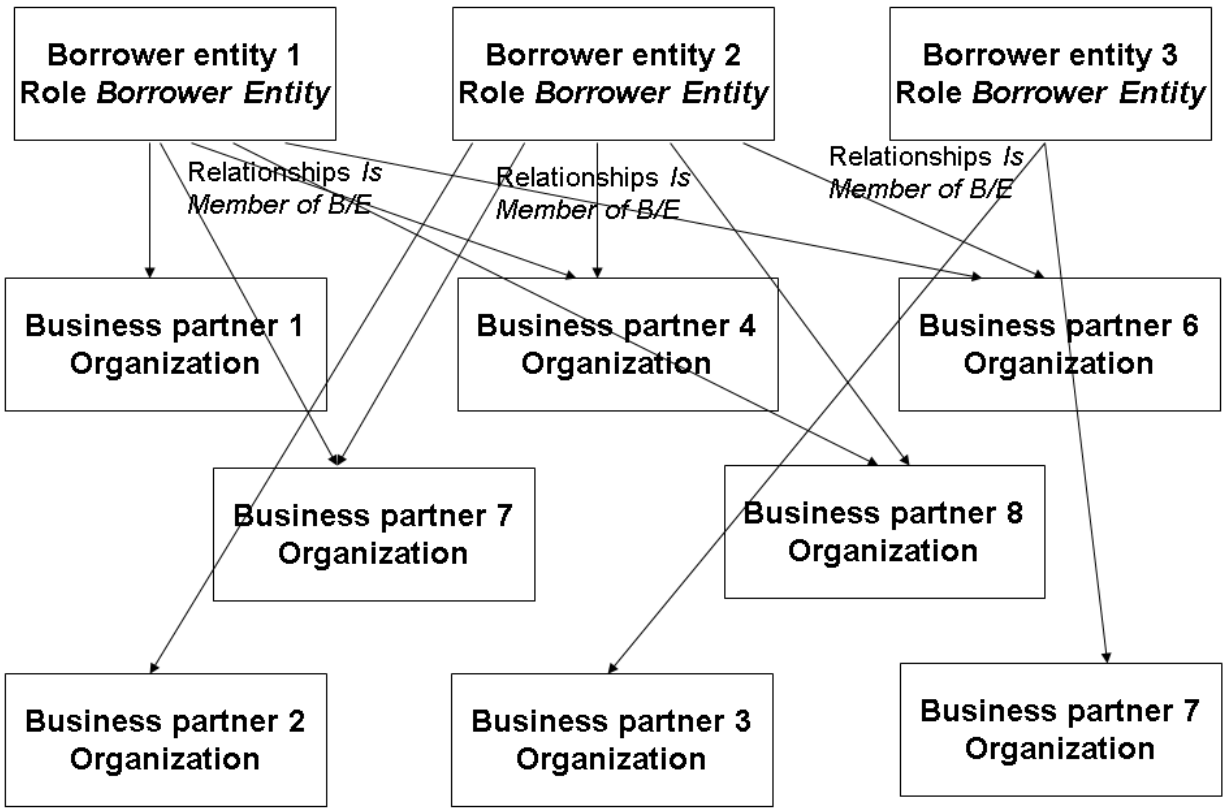
This is explained in the following figure.



The relationship set leads to the formation of the following borrower entities that exist in the business partner role of *Borrower Entity*, and are linked to the individual business partners by way of the relationship category *Is Member of Borrower Entity*:

- Borrower entity 1 comprises business partners 1, 4, 6, 7, and 8
- Borrower entity 2 comprises business partners 2, 4, 6, 7, and 8
- Borrower entity 3 comprises business partners 3, 4, and 7

This is explained in the following figure.



3.1.5.2.5 Authorization Management

Use

You can use the following authorization objects to control the authorizations for maintaining business partner data:

- Authorization objects for the business partner:
  - B_BUPA_GRP
  - B_BUPA_ATT
  - B_BUPA_FDG
  - B_BUPA_RLT
- Authorization objects for relationships:
  - B_BUPR_BZT
  - B_BUPR_FDG

In addition, you can assign an authorization group to a business partner in the dialog. The authorization group controls which users may maintain data for this business partner.

You can also define authorizations for fields and field groups using the Business Data Toolset (BDT). Depending on the settings you have made, the system carries out the relevant authorization checks.

In the SAP GUI dialog, you can display an overview of the authorizations assigned to you by pressing the *Settings* pushbutton.

For more information about authorization management, see Customizing for *Business Partner*, as well as the Developer's Handbook for the BDT under *Function Overview* > *Dialog*.

Integration

Authorization management for the business partner forms part of the SAP authorization concept.

Prerequisites

You have made the necessary settings in Customizing for the *Business Partner* under *Basic Settings* > *Authorizaton Management*.

3.1.5.2.6 Release Object FDB_20 (Business Partner)

Definition

A release object in Bank Analyzer (FS-BA) that enables the system to recognize whether a business partner version that is created or edited by a creator or processor is relevant for release. If the business partner version is subject to release, the Framework for the Principle of Dual Control (CA-GTF-TS-PDC) [page 1849] creates a work item that can be further processed only in the Business Workplace by a supervisor or the person authorized to release the object.

> **Note**
>
> In the Business Workplace, only active, released business partner versions are available for processing. Data that has not been released cannot be displayed or selected. You must use the workflow to display and change data that is currently in the release process.

Use

Customizing

You make the settings for the release object *FDB_20 (Business Partner)* and its release attributes in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *General Functions for Source Data* > *Release Procedure* in the following activities:

- *Edit Release Procedure*
- *Assign Workflow to Release Procedure*
- *Assign Release Steps to Standard Workflow Rule*

Online Processing

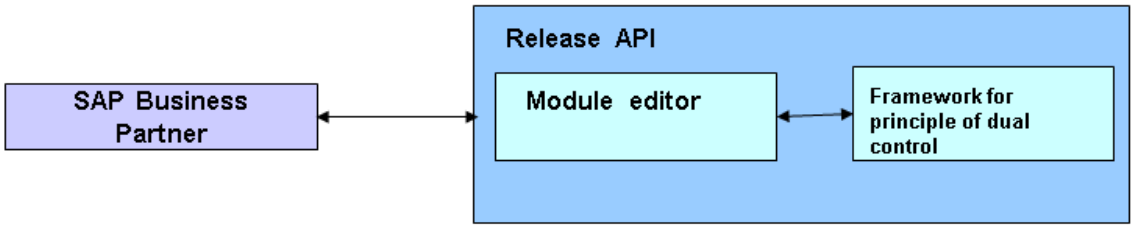
The system checks whether the business partner version that has been edited online is subject to release and generates a work item for each change that is subject to release. If a business partner version is no longer subject to release once it has been edited, the associated work item is deleted.

The Business Application Programming Interface (BAPI) and direct input is not intended to be used for the release procedure.

Structure

Release Attributes

Whether a business partner is subject to release is indirectly determined by the *Framework for the Principle of Dual Control (CA-GTF-TS-PDC)*. The module editor is used, which means that the release relevance does not have to be restricted to a particular number of fields because the transfer structure is not strictly defined.



You make the settings for creating the transfer structure in Customizing for Source Data Layer under *Source Data Services* > *Access to Source Data* > *Object Transfer Structure (OTS)* > *Edit OTS for Primary Objects* > *Edit OTS for Business Partner*.

You create an OTS with the following characteristics:

- /BA1/C11RFLG Indicator for Release Relevance FDB-PO
- /BA1/C11RUSER Release of Workflow Recipient FDB-PO

The module editor fills both characteristics with data. The data is mapped to the ABAP Dictionary structure `/BA1/F_STR_RELEASE`, which is then in turn evaluated as a transfer structure by the *Framework for the Principle of Dual Control (CA-GTF-TS-PDC)*.

You assign the OTS you have created to the primary object for the business partner [page 161] in the Customizing activity *Assign OTS to Processes*.

You make the settings in the module editor in Customizing for *Source Data Layer* under *Source Data Services* > *Access to Source Data* > *Filling of OTS with Data* > *Configuration of Module Editor*.

> **Note**
>
> For more information about how the module editor works, see the documentation for the Customizing activities for the module editor.

The module contains the conditions for release relevance and fills the OTS fields.

The release indicator controls the release check made by the *Framework for the Principle of Dual Control (CA-GTF-TS-PDC)*. If you set the *Release Relevance FDB-PO* checkbox in the transfer structure, the data is released.

Methods for the Release Object in SAP Business Workplace

You can edit the release object **FDB_20 (Business Partner)** in the Business Workplace.

- Display
  You can display the business partner version to be released (see Displaying Business Partner Versions [page 215]).
- Change
  You can change the business partner version to be released (see Changing Business Partner Versions [page 218]).
  The system checks the modified business partner, generates a new work item (if necessary), and deletes the existing work item.
- Display Change Documents
  This method works in the same way as the methods for the correction server [page 1719].
- Return
  When a note has been created, the system places the work item in the Business Workplace of the creator/processor or of the last person who edited it.
- Release
  This method in the Business Workplace changes the release status of the business partner version to *Released*. You can now edit the business partner version without using the Business Workplace.

3.1.5.2.7 Business Partner Versioning

Use

This component enables you to create business partner versions in accordance with the requirements of Source Data in Bank Analyzer.

The versioning concept in Source Data enables you to enter a precise date in the system in order to show the state of knowledge about the business partner at a precise moment in time and to correct and edit this business partner data (business partner version [page 211]) consistently.

The system ensures that the business partner data is kept up to date and consistent by automatically creating a new version each time a business partner is edited.

The business partner versions can be edited and corrected as separate instances of the business partner object. Existing business partner versions are never overwritten.

Versioning enables analysis results to be reproduced, that is, past analysis results can be reproduced identically at a later date. The results that were anticipated on a particular date for the business partner can also be reproduced (versions in the future [page 221]).

Implementation Considerations

The documentation on the *business partner versioning* component is aimed at a very specific **target group** comprising experts that use the versioning transactions to make corrections to business partner versions. The functions for editing business partner versions that are described in this documentation are not an integral part of a business partner's general editing functions, but are only tools for a restricted user group (experts).

The process of manually editing data for the primary objects using appropriate transactions is an exception and is largely restricted to making corrections.

Integration

SAP Business Partner for Financial Services [page 161] is a primary object of the Source Data Layer in Bank Analyzer. You can create versions of this object in the same way as for the other primary objects of the Source Data Layer (financial transaction, financial instrument, account, and so on).

For more information, see:

- Bank Analyzer (FS-BA) [page 8]
- Source Data Layer (FS-BA-SD) [page 100]
- Primary Object Versioning [page 103]

Features

The versioning concept in Source Data is not to be confused with time dependencies in the business partner.

The time dependency in the business partner and the change document update are incorporated into the versioning concept rather than being replaced by it.

Versioning enables business partner data, such as *Time-Dependent Address Usages* or the *Ratings* to be edited **within** a version in a time-dependent manner.

It also enables all changes made to the current version of the business partner to be entered in change documents, which means that a delta upload can still be carried out in the SAP NetWeaver Business Intelligence (BI) extraction.

The version comparison function replaces the change document function for displaying the changes made to the business partner.

Versioning provides **time-dependent** access to business partner data.

Technically speaking, versioning is achieved by maintaining the current business partner data in database tables and updating the versions in mirror tables when the business partner is created and changed.

This enables you to access business partner versions [page 211] in the past as well as future changes (in the form of versions in the future [page 221]).

Versioning is based on grouping business partner data into datasets. Versions of related data, such as business partner relationships or the business partner's addresses, are created as independent objects.

Example

Five versions exist for your business partner Karola Adams:

Business Partner Number	Name	Version	Valid From Business Record Date	Created On System Time/ Date	Cause
45698	Karola Adams	1	01.03.2003	28.02.2003	Creation of business partner
45698	Karola Adams	2	01.05.2003	05.05.2003	Change of address
45698	Karola Adams	3	01.01.2003	29.05.2003	Correction of change of address
45698	Karola Martin	4	15.10.2003	03.11.2003	New last name
45698	Karola Martin	5	01.02.2004	14.01.2004	Creation of business partner relationship

3.1.5.2.7.1 Business Partner Versions

Definition

Separate instance of the business partner that can be edited independently of other instances. The business partner version shows the state of knowledge about the business partner at a precise moment in time.

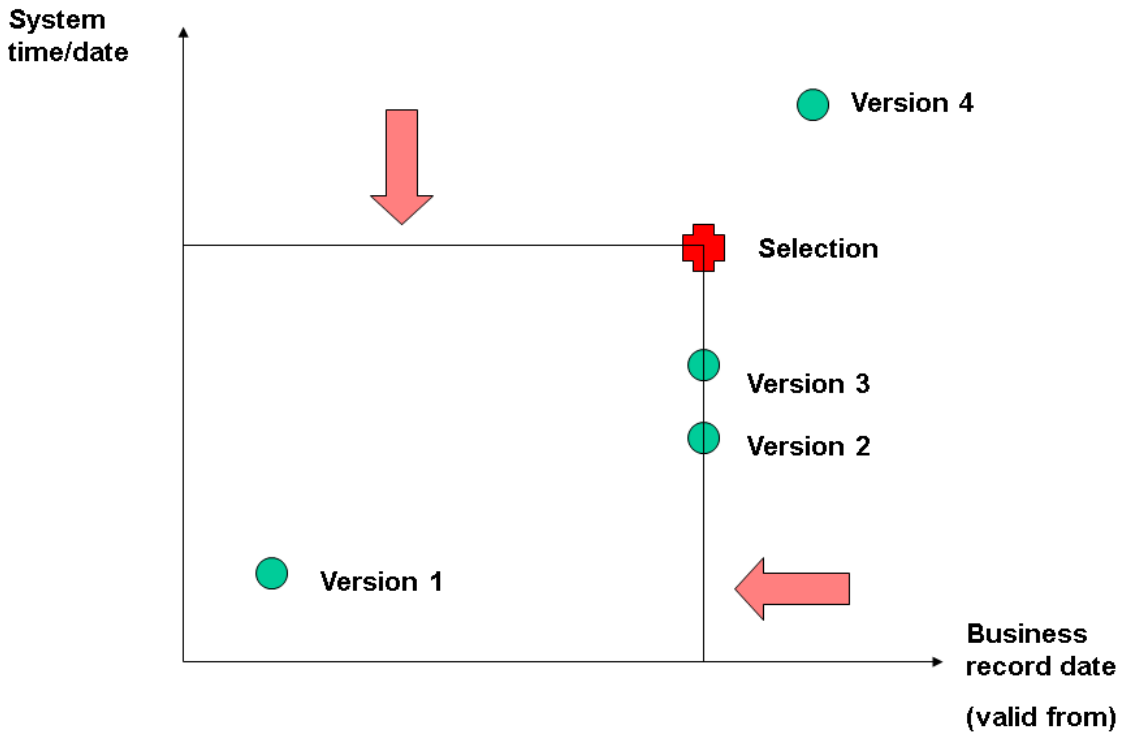
Business partner versions are identified by the following information:

- Business partner number
- Business record date
- System time/date

The versioning concept in Source Data works on the principle that a new version of a business partner is created each time the business partner in question is created and changed. This means that a version cannot be overwritten. Technically speaking, rather than “changing” or “deleting” a business partner, the system creates a new version of that business partner with changed data.

Use

Unlike with one-dimensional time dependency, a business partner version is dependent on two dates/times rather than one. The system selects the business partner version as follows:



When you select a business partner version, you specify the object ID (business partner number), the business key date for which the version is valid, and a precise date as the system time up to which corrections are to be considered for this object. The system then displays the business partner version that is valid for the key date selected and that was last corrected immediately before the required system time (Version 3 in the figure above).

The business partner in turn contains data with a separate time base, which is referred to as time dependency. Addresses are time dependent, for example. Time dependency is not the same as versioning. The time-dependent

information stored in a business partner version represents the state of knowledge about the time dependency for a specific business date.

Integration

SAP Business Partner for Financial Services [page 161] is a primary object of the Source Data Layer in Bank Analyzer. You can create versions of this object in the same way as for the other primary objects of the Source Data Layer (financial transaction, financial instrument, account, and so on).

For more information, see:

- Bank Analyzer (FS-BA) [page 8]
- Source Data (FS-BA-SD) [page 100]
- Primary Object Versioning [page 103]

Example

Five versions exist for your business partner Karola Adams:

Business Partner Number	Name	Version	Valid From Business Record Date	Created On System Time/ Date	Cause
45698	Karola Adams	1	01.03.2003	28.02.2003	Created as new business partner
45698	Karola Adams	2	01.05.2003	05.05.2003	Change of address
45698	Karola Adams	3	01.01.2003	29.05.2003	Correction of change of address
45698	Karola Martin	4	15.10.2003	03.11.2003	New last name
45698	Karola Martin	5	01.02.2004	14.01.2004	Creation of business partner relationship

3.1.5.2.7.1.1 Display of Business Partner Versions

Use

You want to display the business partner in a selected version. The version can be a selected version in the past, a version in the future as a future change, or the current valid version.

For more information, see Displaying Business Partner Versions [page 215].

Integration

SAP Business Partner for Financial Services [page 161] is a primary object of the Source Data Layer in Bank Analyzer. You can create versions of this object in the same way as for the other primary objects of the Source Data Layer (financial transaction, financial instrument, account, and so on).

For more information, see:

- Bank Analyzer (FS-BA) [page 8]
- Source Data (FS-BA-SD) [page 100]
- Primary Object Versioning [page 103]

Prerequisites

You have activated business partner versioning.

Example

Five versions have been created for your business partner Karola Adams; you can display these versions:

Business Partner Number	Name	Version	Valid from Business Record Date	Created On/ System Time or Date	Cause
45698	Karola Adams	1	01.03.2003	28.02.2003	Creation of business partner
45698	Karola Adams	2	01.05.2003	05.05.2003	Change of address
45698	Karola Adams	3	01.01.2003	29.05.2003	Correction to the change of address
45698	Karola Martin	4	15.10.2003	03.11.2003	New last name
45698	Karola Martin	5	01.02.2004	14.01.2004	Creation of business partner relationship

3.1.5.2.7.1.1.1 Displaying Business Partner Versions

Prerequisites

You have activated versioning and created the business partner versions to be displayed.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Business Partner* > *Versioning* > *Display Business Partner Version*.

   The system displays the initial screen for displaying business partner versions.
2. Select the business partner whose version you want to display.
3. Enter the business record date of the business partner version you want to display

   using the possible entries pushbutton (F4). The system proposes all of the versions that exist for this business partner.
4. Enter the system time of the version you want to display

   using the possible entries pushbutton (F4). The system proposes the system times for all the versions of a business partner that exist for a particular business record date.
5. Choose *Display*.

   Note that there are some special considerations to take into account when displaying business partner data. For more information see Points to Consider When Processing Business Partner Versions [page 220].

Example

Five versions exist for your business partner Karola Adams:

Business Partner Number	Name	Version	Valid From Business Record Date	Created On System Time/ Date	Cause
45698	Karola Adams	1	01.03.2003	28.02.2003	Creation of business partner
45698	Karola Adams	2	01.05.2003	05.05.2003	Change of address
45698	Karola Adams	3	01.01.2003	29.05.2003	Correction of change of address
45698	Karola Martin	4	15.10.2003	03.11.2003	New last name

Business Partner Number	Name	Version	Valid From Business Record Date	Created On System Time/ Date	Cause
45698	Karola Martin	5	01.02.2004	14.01.2004	Creation of business partner relationship

Enter the business partner number and the required business record date and system dates, and choose *Display*.

3.1.5.2.7.1.2 Creation of Business Partner Versions

Use

Business partner versions are created so that you can specify a particular date in order to access the state of knowledge about the business partner at a precise moment in time, and correct this business partner data consistently.

Every time you create a new business partner or make a change to a business partner version, the system creates a new version in the business partner's mirror tables and updates the current business partner version in the business partner's database tables.

You have **two options** for editing the business partner and creating business partner versions:

- To edit the business partner in the current version, choose *Business Partner* > *Versioning* > *Business Partner: Create Version*. When you save the business partner, a current business target date defined by the system is used to create the new version.
- To edit the business partner in a selected version, choose *Business Partner* > *Versioning* > *Business Partner: Change Version*. On this screen you can select the current version of the business partner or a non-current version.

Integration

SAP Business Partner for Financial Services [page 161] is a primary object of the Source Data Layer in Bank Analyzer. You can create versions of this object in the same way as for the other primary objects of the Source Data Layer (financial transaction, financial instrument, account, and so on).

For more information, see:

- Bank Analyzer (FS-BA) [page 8]
- Source Data (FS-BA-SD) [page 100]
- Primary Object Versioning [page 103]

Prerequisites

You have activated business partner versioning.

Activities

Create Business Partners

The procedure for creating a new business partner remains unchanged. The relevant data is, however, stored in the business partner's database tables and in the mirror tables as a business partner version. The new business partner version automatically becomes the current version.

Change Business Partners

At mirror table level, a business partner is never "changed". Instead , a new version of the business partner is created each time a change is made. This means that a business partner that has been changed n times has exactly n+1 versions (+1 represents the first version that was created). The new, most current version is also updated in the business partner's database tables.

You have two options for changing business partners:

- You change the current version of the business partner on the *Business Partner: Change Version* screen. You can also determine the business record date of the new version to be created.
  For more information, see Editing Business Partner Data [page 172].
- You change a selected version of the business partner on the *Business Partner: Change Version* screen.
  For more information, see: Changing a Business Partner Version [page 218].

Example

Five versions have been created for your business partner Karola Adams:

Business Partner Number	Name	Version	Valid from Business Record Date	Created On System Time	Cause
45698	Karola Adams	1	2003-03-01	2003-02-28	Creation of business partner
45698	Karola Adams	2	2003-05-01	2003-05-05	Change of address
45698	Karola Adams	3	2003-01-01	2003-05-29	Correction of change of address
45698	Karola Martin	4	2003-10-15	2003-11-03	New last name
45698	Karola Martin	5	2004-02-01	2004-01-14	Archiving of business partner

3.1.5.2.7.1.2.1 Changing Business Partner Versions

Prerequisites

You have activated versioning and created business partner versions that have to be corrected.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Business Partner* > *Versioning* > *Business Partner: Change Version*.

   The system displays the initial screen for processing business partner versions.
2. Select the business partner for which you want to correct the version.
3. Enter the business record date of the business partner version that you want to change (source version).

   Choose the input help pushbutton. The system proposes all of the versions that exist for this business partner.
4. Enter the business record date of the new version that is to be created.

   If you do not specify a target business date, the system chooses the current date as the target date. It saves the version with that (the current) date.
5. Choose *Change*.

   The system displays the editing screen for the business partner.
6. Make the necessary changes to the business partner and save your entries.

   Note that there are some special considerations to take into account when displaying business partner data. For more information see Points to Consider When Processing Business Partner Versions [page 220].

   If you make changes that are valid for the present to a business partner version that is in the past, you have to save that version and all of the versions between that version and the current version inclusive.

Results

If you change a selected business partner version, the system automatically creates a **new version** when you save the changed version. From a technical point of view, business partner data is never changed; a new business partner version is created instead.

3.1.5.2.7.1.2.2 Activating Versioning

Use

The setting for activating business partner versioning has far-reaching consequences.

Activate business partner versioning only **once** when you start up the SAP system.

You **must** avoid making or changing this setting, that is, deactivating or reactivating versioning, at a later date for the following reasons:

- If you set business partner versioning from *Inactive* to *Active* while your SAP system is in operation, the initial versionswill be missing for all the existing business partners. The business partner data will become inconsistent.
- If you set business partner versioning from *Active* to *Inactive* while the system is in operation and then reactivate versioning, inconsistencies might occur.

Prerequisites

You have already checked that versioning is switched off, and the database tables for the business partner are initial.

When you are implementing Bank Analyzer, we recommend that you activate business partner versioning only **after the initial load of the business partner data has been completed successfully** ; in other words once all the business partner data exists for all the segments in the SAP target system. This avoids an extremely large increase in the volume of data stored in the target system

For more information, plus recommendations on loading business partner data, see Data Transfer from External Systems and Source Systems [page 234] .

Procedure

1. Call up transaction *BPTIME* .
2. To activate business partner versioning, set the *Versioning Active* indicator and choose *Save* .
   The system automatically schedules report *FSBP_TIME_VERSION_HISTORIZE* to run as a job. Depending on the quantity of data, you can also schedule this report to be carried out in several parallel processes in the background. The system creates the first version of each business partner. You can schedule the report by calling the *Define Job* transaction in the *SAP Easy Access* menu under *Tools* > *CCMS* > *Background Processing* > *Define Job* .You can monitor the job by choosing *Jobs - Overview and Administration* .
3. If you also want to enable new versions of future business partner data to be created in the form of planned changes, set the *Versions in the Future Active* indicator.

You can create changes or versions that have a business date after the time they are created in the system. A version in the future [page 221] automatically becomes the currently valid version on the business date.

If you deactivate the versions in the future, you cannot create any new versions in the future and existing versions in the future can no longer become currently valid versions.

4. If you activate versioning, you also have to start the initial run. This initial report generates the necessary initial versionsof the business partner in the mirror tables. You can execute the report once as a complete run or schedule a daily job.

Result

You have activated business partner versioning. When you create and change business partners, the system creates new business partner versions in the relevant mirror tables. You can correct and edit these new versions separately. If you reactivate and deactivate versioning again, inconsistencies will arise.

3.1.5.2.7.1.3 Points to Consider When Processing Business Partner Versions

When you display and create business partner versions, the processing screen for the business partner does not contain the complete range of functions:

- When displaying or creating business partner versions, you cannot use the or the buttons for, for example, opening business partners or creating business partner categories, to select or change other business partners.
- When displaying business partner versions, you cannot switch from display mode to processing mode to make changes to these business partner versions.
- When displaying business partner versions, you cannot check the consistency of the business partner data.

In the standard system, the following functions are contained in the view of business partner data:

- In the area above the tab, you can always see the following information during processing:
  - Business partner number
  - Business partner category (as an icon)
  - Business partner role
- The mode you are working in is displayed in the title bar.
- The *Relationships* button allows you to display the business partner relationships [page 169] .
- When you edit a business partner, you can choose the *General Data* pushbutton or the pushbutton.
- You can use the pushbutton to make the settings on the business partner interface that suit your needs.

3.1.5.2.7.2 Version in the Future

Definition

This object reflects **future changes** to the business partner.

Use

You can map future changes to the business partner in the system as "versions in the future" by using an existing version (source version) as a template and creating a new version that contains the planned changes. Enter a future business record date for the new version in the future on which the future changes are to become valid.

For more information about creating versions, see: Changing a Business Partner Version [page 218].

Example

Five versions exist for your business partner Karola Adams:

Business Partner Number	Name	Version	Valid From Business Record Date	Created On/ System Time or Date	Cause
45698	Karola Adams	1	01.03.2003	28.02.2003	Creation of business partner
45698	Karola Adams	2	01.05.2003	05.05.2003	Change of address
45698	Karola Adams	3	01.01.2003	29.05.2003	Correction of change of address
45698	Karola Martin	4	15.10.2003	03.11.2003	New last name
45698	Karola Martin	5	01.02.2004	14.01.2004	Creation of business partner relationship

Both the creation of the business partner on February 28, 2003, which is not to become valid for the system until March 1, 2003, and the creation of the business partner relationship on January 14, 2004 with the business record date February 1, 2004 represent future changes in the form of business partner versions.

3.1.5.2.8 Scenario Data for Business Partners

Definition

Fictitious data for business partners. This data can be stored in the system in addition to the existing real data for the business partners, and can be used in analyses instead of real data, or in addition to real data.

You can identify scenario data using the following information:

- Business partner number
- Business record date
- System time/date
- Scenario

Use

Scenario data for business partners can be used to carry out simulations or stress tests using fictitious versions of business partners. Scenario data is not real data but fictitious data that is used for test and simulation purposes only.

It is possible to simulate or stress any of the attributes of the business partner. However, business partner ratings [page 188] or the characteristics of the segmentation service, for example, are usually the most relevant attributes for stress tests.

If you want to edit characteristics from the segmentation service in a scenario, then you have to define which characteristics from the segmentation service are to be used, and in which scenario they are to be used, in Customizing for *SAP Business Partner for Financial Services* under *Settings for Financial Services* > *Tools* > *Segmentation Service* > *Define Characteristics for Scenario*.

Scenario data can be stored, edited, displayed, and archived as *Business Partner Scenarios*.

It comprises the simulated and manipulated data from the stress test, plus the general data for the business partner from table *BUT000* and the business partner roles from table *BUT100* (which are always displayed in online mode).

Scenario data can be combined with real data to provide stress data for use in analyses. You can call this data on the *Business Partner: Display Stress Data*.

Integration

For information about scenario data in the Source Data Layer, see Scenario [page 1731].

For information about editing scenario data for business partners, see:

- Editing Business Partner Scenarios [page 223]
- Displaying Business Partner Scenarios [page 224]
- Displaying Stress Data for Business Partners [page 227]

For more information about stress data, see Retrieving Stress Data [page 1734].

The functions for business partner scenarios are part of the versioning function for the business partner. Business partner scenarios are edited in the same way as business partner versions. You can define scenarios for business partners in Customizing for *Bank Analyzer* under *Infrastructure* > *General Scenario Administration* > *Create Scenarios*. The data category to be used for the business partner is *BP (Business Partner)*.

Example

You want to use your in-house rating procedure to rate a particular business partner. However, the new rating procedure has not yet been approved, but you want to be able to see what effect the new procedure will have on the rating, and consider any possible risks. You can do this by creating the ratings as scenario data, and then using the new rating procedure to assess the business partner.

The results of this simulation are not used in the official calculation runs and analyses. Instead, they are stored as a separate data set that can be used for test purposes or in internal analyses.

3.1.5.2.8.1 Editing Business Partner Scenarios

Prerequisites

- The system administrator has activated business partner versioning.
- You have defined relevant scenarios for the business partner in Customizing for *Bank Analyzer* under *Infrastructure* > *General Scenario Administration* > *Create Scenarios*.
- If you want to edit characteristics from the segmentation service in a scenario, then you have to define which characteristics from the segmentation service are to be used, and in which scenario they are to be used, in Customizing for *SAP Business Partner for Financial Services* under *Settings for Financial Services* > *Tools* > *Segmentation Service* > *Define Characteristics for Scenario*.
- You have created the business partners whose data you want to simulate or stress test.

Context

You can edit business partner scenarios in order to create fictitious data for an exact point in time. This fictitious data is referred to as scenario data for business partners [page 222]. You use scenario data and stress data to carry out simulation runs or stress runs using fictitious versions of your business partners.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Business Partner* > *Scenario Data* > *Business Partner: Edit Scenario*.

   The system displays the initial screen for editing business partner scenarios.
2. Choose the business partner [page 162] for which you want to edit scenario data.
3. Choose the *scenario* that you want to use to simulate the fictitious situation for the business partner.
4. Enter the business record date of the business partner version to be used in the simulation (source version).

   Choose the input help pushbutton. The system proposes all of the versions that exist for this business partner.
5. Choose *Change*.

   The system displays the editing screen for the business partner.
6. Change the business partner data as required, and save your entries.

   Note that there are some special considerations to take into account when displaying business partner data. For more information see Points to Consider When Processing Business Partner Versions [page 220].

Results

You have edited the scenario data for the business partner. You can display the fictitious data for the business partner on the Business Partner: Display Scenario [page 224] screen. You can display the stress data, which comprises fictitious and real data, on the Business Partner: Display Stress Data [page 227] screen.

3.1.5.2.8.2 Displaying Business Partner Scenarios

Prerequisites

- The system administrator has activated business partner versioning.
- You have defined relevant scenarios for the business partner in Customizing for *Bank Analyzer* under *Infrastructure* > *General Scenario Administration* > *Create Scenarios*.
- You have created the business partner scenario data that you want to display.

Context

You can display fictitious scenario data for business partners [page 222], which reflects a situation that is possible or probable for the business partner.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Business Partner* > *Scenario Data* > *Business Partner: Display Scenario*.

   The system displays the initial screen for displaying business partner scenarios.
2. Choose the business partner [page 162] for which you want to display the fictitious data.
3. Choose the scenario that the system used to generate the scenario data for the business partner.
4. Choose the business record date of the fictitious business partner version.
5. Choose the technical date of the fictitious business partner version.

   Choose *Display*.

   Note that there are some special considerations to take into account when displaying business partner data. For more information see Points to Consider When Processing Business Partner Versions [page 220].

Results

The system displays the scenario data for the business partner. It displays the simulated, fictitious data from the business partner scenario, plus the general data for the business partner from table *BPTIME_BUT000* and the business partner roles from table *BPTIME_BUT100* (which are always displayed in dialog mode).

3.1.5.2.8.3 Deleting Business Partner Scenarios

Prerequisites

You have created scenario data for business partners on the Business Partner: Edit Scenario [page 223] screen.

Context

You can delete scenario data for business partners [page 222] at a specified point in time to remove fictitious data from the database. This decreases the amount of data and improves performance.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Business Partner: Delete Scenario Version*.

   The system displays the screen for deleting business partner scenarios.
2. Enter the business partner [page 162] and the scenarios that you want to delete.
3. You can restrict the scenario data for business partners to be deleted by entering the business record date.

   If you do not enter a date, the system deletes all versions of the business partner scenarios that you have selected.

   If you enter a date, then the system deletes the versions of business partner scenarios that you have selected which are older than the date that you have entered.

Results

You have deleted fictitious scenario data for the business partner according to the selection criteria that you have entered.

3.1.5.2.8.4 Generation of Stress Data for Business Partners

Use

In this function the system reads real data and scenario data for the business partner, and generates a combined version of this data, which is referred to as **stress data** .

Prerequisites

- System administration must have already activated business partner versioning.
- You have already entered the settings required for managing scenarios in the IMG activities *Create Scenarios* and *Edit Composite Scenarios* in Customizing for *Bank Analyzer* under *Infrastructure* > *General Scenario Administration*.
- You have already entered the settings required for using characteristics in scenarios in Customizing for *SAP Business Partner for Financial Services* under *Settings for Financial Services* > *Tools* > *Segmentation Service* > *Define Characteristics for Scenario.*
- You have created scenario data for business partners [page 222] in the *SAP Easy Access Menu* in *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Business Partner* > *Scenario Data* > *Business Partner: Display Scenario.*

Features

For general information about generating stress data in the Source Data Layer, see Retrieving Stress Data [page 1734] .

Scenario data and real data for the business partner is available in versioned form.For more information about versioning, see Business Partner Versioning [page 209] .

When the system generates stress data, it combines scenario data and real data for the business partner for each data set. If scenario data is available in a data set for the business partner, the system transfers this data into the stress data. You can refine the characteristics of the segmentation service in Customizing. In the IMG activity *Define Characteristics for Scenario* you can determine which characteristics can be changed using a scenario. The values of the remaining characteristics are then transferred from the real data.

This means that you can change specific data sets or characteristics of a business partner for a scenario without having to supply the rest of the data again.

Activities

The system automatically generates stress data.

You can display stress data for business partners [page 227] .

3.1.5.2.8.4.1 Displaying Stress Data for Business Partners

Prerequisites

- The system administrator has activated business partner versioning.
- You have defined relevant scenarios for the business partner in Customizing for *Bank Analyzer* under *Infrastructure* > *General Scenario Administration* > *Create Scenarios* .
- The system has generated real data and scenario data for the business partner (see Generation of Stress Data for Business Partners [page 226]).

Context

You can display the stress data of a business partner as a combination of the real data and the scenario data for business partners [page 222].

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Business Partner* > *Scenario Data* > *Business Partner: Display Stress Data*.

   The system displays the initial screen for displaying the stress data.
2. Choose the business partner [page 162] for which you want to display the stress data.
3. Choose the scenario that the system used to generate the stress data for the business partner.
4. Choose the business record date of the business partner.
5. Choose the system date/time of the business partner.
6. Choose *Display*.

   Note that there are some special considerations to take into account when displaying business partner data. For more information see Points to Consider When Processing Business Partner Versions [page 220].

Results

The system displays the stress data for the business partner. The stress data comprises real data from the source version and fictitious scenario data for the business partner.

3.1.5.2.9 Archiving

Use

You can use this function to archive business partner data and then delete it. This reduces the volume of data and improves system performance.

You can archive business partner data as follows:

- Completely using the archiving object *BABP_FULL — Analytical Business Partner (Complete)*
- Based on a key date using the archiving object *FSBP_VERSN — Archive BP Versions Based on Key Date*
- Based on a scenario using the archiving object *BABP_SCENA — Analytical Business Partner (Scenario)*

Activities

We recommend you archive business partner data using *SAP Information Lifecycle Management (SAP ILM)*. However, you can also archive business partner data using the `AR_ENGINE` transaction (Archiving Engine).

Related Information

Archiving Business Partners Completely [page 229]
Archiving Business Partner Versions Based on a Key Date [page 230]
Archiving Business Partner Scenarios [page 231]
Archiving and Destruction of Business Partner Data Using ILM [page 232]

3.1.5.2.9.1 Archiving Business Partners Completely

Use

Archiving the Primary Object Business Partner [page 161] enables you to remove all data for business partners that are no longer required from the system, without deleting this data altogether. This improves database performance and you can store the data for later evaluation. To archive business partner data completely, use the archiving object *BABP_FULL — Analytical Business Partner (Complete)*.

In the analysis phase, the system checks whether the business partner data can be archived as follows:

- The system checks whether you have set the To Be Archived indicator for a business partner for the current valid data.
- The system checks whether you still actively use the business partner data to be archived in the system, using the BAdI *FS_BUPA_ARCHIVABLE*. You can also store other checks with this BAdI.

In the write phase, the DataStore objects for the business partner are not written to the archive. In the delete phase, the DataStore objects for the business partner are deleted from analytical applications.

> **Note**
>
> Note that you cannot define or check the residence time for archiving.

Prerequisites

You have made the system settings for the archiving object *BABP_FULL* in Customizing by choosing *SAP NetWeaver* > *Application Server* > *System Administration* > *Data Archiving* > *Archiving Object-Specific Customizing*.

Procedure

1. Choose the transaction AR_ENGINE (Archiving Engine).
2. Choose the archiving scenario *BABP_FULL Analytical Business Partner (Complete)*.

> **Note**
>
> To restrict archiving to individual business partners, choose *Dynamic Selections*.

Result

You have archived the complete business partner with all data and all versions.

> **Caution**
>
> Note that you can only transfer archived business partner data from analytical applications to the active system with restrictions. This can cause inconsistencies. The later the data is transferred to the active system after being archived, the greater the risk of data inconsistencies. If you do transfer archived data to the active system, on the *SAP Easy Access* screen choose *Financial Services* > *Bank Analyzer* > *Source Data Layer* > *Tools* > *Index Tables* and the menu transaction *Regenerate Index for Business Partners* after data transfer to fill the DataStore objects.

3.1.5.2.9.2 Archiving Business Partner Versions Based on a Key Date

Prerequisites

You have made the system settings for the archiving object *FSBP_VERSN — Archive BP Versions Based on a Key Date* in Customizing by choosing *SAP NetWeaver* > *Application Server* > *System Administration* > *Data Archiving* > *Archiving Object-Specific Customizing*.

You have adjusted the residence time to suit your own requirements.

Context

Archiving historical business partner versions [page 211] enables you to remove business partner versions you no longer require from the database without deleting them altogether. You can archive the data for all business partner versions that are valid before a key date. On the key date for archiving, the system creates a new business partner version with the data that is valid on this day. The business partners are still available in the system after the archiving run.

Archiving improves database performance, and you can keep the data for later evaluation. To archive business partner versions based on a key date, use the archiving object *FSBP_VERSN — -Archive BP Versions Based on a Key Date*.

The analysis phase does not include any checks. The system archives all versions that are valid before the key date. In the write phase of archiving based on a key date, the system changes the business partner versions and creates new versions for the key date.

> **Note**
>
> The key date for archiving depends on the residence time. If, for example, you specify a residence time of one year, the key date is 365 days before the current date. The default residence time is ten years.

Procedure

1. Choose the transaction AR_ENGINE (Archiving Engine).
2. Choose the archiving scenario *FSBP_VERSN — Archive BP Versions Based on a Key Date*.
3. To specify the residence time, choose *Configuration of Archiving Scenario*.
4. To restrict archiving to individual business partners, choose *Dynamic Selections*.
5. To delete and recreate the DataStore objects for the business partner after archiving, on the *SAP Easy Access* screen, choose *Financial Services* > *Bank Analyzer* > *Source Data Layer* > *Tools* > *Index Tables* and the menu transaction *Regenerate Index for Business Partners*.

Results

You have archived all historical business partner versions for a particular key date.

> **Caution**
>
> You cannot transfer business partner versions archived based on a key date from analytical applications to the active system.

3.1.5.2.9.3 Archiving Business Partner Scenarios

Use

Archiving of scenario data for business partners [page 222] enables you to remove fictitious business partner data from the database that is no longer required and is stored in the system in addition to the real data, without deleting it altogether. Archiving improves database performance, and you can keep the data for later evaluation. To archive scenario data for the business partner, use the archiving object *BABP_SCENA — Analytical Business Partner (Scenario)*.

> **Note**
>
> Note that you cannot define or check the residence time for archiving.

Prerequisites

You have made the system settings for the archiving object *BABP_SCENA Analytical Business Partner (Scenario)* in Customizing by choosing *SAP NetWeaver* > *Application Server* > *System Administration* > *Data Archiving* > *Archiving Object-Specific Customizing*.

Procedure

1. Choose the transaction AR_ENGINE (Archiving Engine).
2. Choose the archiving scenario *BABP_SCENA Analytical Business Partner (Scenario)*.

> **Note**
>
> To restrict archiving to individual scenarios and business partners, choose *Dynamic Selections*.

Result

You have archived the scenario data for particular business partners.

> **Caution**
>
> You cannot transfer scenario-related archived business partner data from analytical applications to the active system.

3.1.5.2.9.4 Archiving and Destruction of Business Partner Data Using ILM

Use

The archiving objects `BABP_FULL`, `FSBP_VERSN`, and `BABP_SCENA` support the archiving and destruction of data using *SAP NetWeaver Information Lifecycle Management (ILM)*.

There are ILM objects with the same names for the archiving objects: `BABP_FULL`, `FSBP_VERSN`, and `BABP_SCENA`.

Prerequisites

- You have familiarized yourself with ILM, you have made ILM available in your system, and you have configured your system to use ILM.
  The following fields are available for the above ILM objects in the object category *SAP Business Suite* (`OT_FOR_BS`) to enable you to edit ILM policies (transaction `IRMPOL`):
  - **Available Time References**
    The retention period for the archiving objects `FSBP_VERSN` and `BABP_SCENA` starts with the creation date (`CREATION_DATE`). The retention period for the archiving object `BABP_FULL` starts with the date delivered using the service operation (`START_RET_DATE`).
  - **Available Condition Fields**
    - Archiving object `BABP_FULL`
      - Application name (`APPL_NAME`)
      - Application rule variant (`APPL_RULE_VARIANT`)
      - Business partner category (`BP_TYPE`)
      - Legal entity (`LEGAL_ENTITY`)
    - Archiving object `FSBP_VERSN`
      - Legal entity (`LEGAL_ENTITY`)
    - Archiving object `BABP_SCENA`
      - Legal entity (`LEGAL_ENTITY`)
      - Scenario (`SCENARIO`)
  - **Available Policy Categories**
    - `RTP`: Retention Rules
    - `RST`: Residence Rules
- You have run the report `BABP_ARCHIVING_SWITCH_AOBJ` to switch the Customizing settings for the archiving objects in transaction `AOBJ` to the ILM programs.
- You have activated the business function *ILM-Based Deletion of Business Partner Data* (`BUPA_ILM_BF`) for the archiving object `BABP_FULL`.

  > **Note**
  >
  > The business function is **not** reversible.

- You have made the required settings for the archiving object `BABP_FULL` in Customizing under *Financial Services* > *SAP Business Partner for Financial Services* > *General Settings* > *Business Partner* > *Data Protection* and, if necessary, you have implemented the available Business Add-Ins (BAdIs).

Features

Delete data from the database, archive on expiry of the residence period, and destroy on expiry of the retention period:

In the write program, select the *Archiving* checkbox under *ILM Actions*. The system calculates the residence and retention period based on the ILM rules. It deletes the data from the database on expiry of the residence period and stores the data in the ILM store. The data is destroyed on expiry of the retention period.

> **Note**
>
> You can use ILM to archive data belonging to archiving object `BABP_FULL` only if its purpose has been completed **and** the data has been blocked. The system checks in a preprocessing program whether the data is blocked.

Leave data in the database, also archive on expiry of the residence period, and destroy on expiry of the retention period:

In the write program, select the *Snapshot* checkbox under *ILM Actions*. The system calculates the residence and retention period based on the ILM rules and stores the data redundantly in the ILM store. On expiry of the retention period, the data in the database and in the ILM store is destroyed.

Leave data in the database and destroy on expiry of the retention period:

In the write program, select the *Data Destruction* checkbox under *ILM Actions*. The system calculates the retention period based on the ILM rules and destroys the data on expiry of this period.

Set legal holds for data that must not be deleted and destroyed:

You can define legal holds for the BOR object *Business Partner* (`BUS1006`). For more information, see .

More Information

3.1.5.2.10 Data Transfer from External Systems and Source Systems

Use

To process versioned business partners, transfer the data to Bank Analyzer by using the *complex interface* or the Business Application Programming Interfaces (BAPIs) for the *SAP Business Partner for Financial Services* .

If you use the BAPIs for the business partner, then switch on BAPI */BA1/BAPI_FSBP_VERSION_INIT (initialize processing of the versioned business partner*) before you use the other business partner BAPIs.

We recommend that you use the *complex interface* . The complex interface automatically processes the objects in the correct order when it calls the BAPIs, and takes into account whether the versioning BAPI *FSInitTimeVersion (initialize processing of versioned business partner) is activated.*

Integration

For detailed information about using the *complex interface* , see Complex Interface for the Business Partner [page 235] .

Business partner versioning [page 209] is not supported in direct input (DI) . As a result, the external data transfer function for importing data by means of direct input, and the related DI function modules, are not released.

3.1.5.2.10.1 Complex Interface for the Business Partner

Use

You can use the complex interface for the business partner to load and change business partners automatically into/in the target system.

You can load business partners into the target system either using an *initial load* when you implement the components, or at regular intervals after you have gone live so that you can change the data. This second option is called the *delta load* .

You can use the complex interface to load in one batch the data collected in various systems and data segments. This considerably reduces the number of objects, which improves system performance.

Features

The complex interface for the business partner can be used for the mass processing of data. It contains interfaces that can be called by the distribution function in Application Link Enabling (ALE)(IDoc), the Data Transfer Workbench, or directly by means of a function module.

Data Transfer Workbench

If you use the data transfer workbench to load business partners, you can convert the IDoc format file that you optimized or aggregated in the source system to an internal transfer structure, and then the system can process it as such. The input processing module (of the complex interface) that processes the imported IDoc has to be registered in the Data Transfer Workbench. The Data Transfer Workbench then maps the input file in IDoc format to the nested transfer structure of the input processing module.

For more information about using the Data Transfer Workbench, see .

Custom Report

If you want to use a program you created to aggregate business partner data in the SAP System during data transfer, you can directly implement the complex interface using the relevant function modules.

The complex interface contains the following function modules:

- *BUPA_INBOUND_MAIN_SAVE (IDoc Inbound Interface for Business Partner)*
- *BUPA_INBOUND_REL_SAVE (IDoc Inbound Interface for Business Partner Relationships)*

The function modules process the following table types in the complex interface:

- *BUS_EI_EXTERN_T (Complex External Interface for Business Partner (Tab.))*
- *BURS_EI_EXTERN_T (Copy of the Complex External Interface for Relationships (Tab.))*

The table types are based on the following transfer structures of the complex interface:

- *BUS_EI_EXTERN (Complex External Interface for the Business Partner)*
- *BURS_EI_EXTERN (Complex External Interface for a Relationship)*

The complex interface structure is in the form of a hierarchy, and is based on the Business Application Programming Interfaces (BAPIs)of the business partner, which are table-based.

The BAPIs make it easier for you to create and change business partners because you need to supply data to these interfaces only. The interfaces call the BAPIs for processing the business partner. The complex interface automatically processes the objects in the correct order when it calls the BAPIs.

The complex interface also makes it possible to process the business partner in *Current State* mode. This processing mode enables the system to determine the delta for the daily change service. If all business partner data is imported using the *Current State* processing mode, only the changed data is updated in the respective tables.

3.1.5.2.10.1.1 Transfer Structure for Master Data

Definition

The complex interface for the transfer of business partner master data contains transfer structure *BUS_EI_EXTERN (complex external interface for the business partner)* , which is nested deep in other components on the basis of business aspects.

Structure

ComponentLevel 1	ComponentLevel 2	Additional Information
HEADER		Contains general data and header data for the transfer of Business Partners [page 162] .
	OBJECT	Contains the object type as the internal technical key of a business object. The business object relevant for the complex interface is *BusinessPartnerFS.*

	OBJECT_INSTANCE	Contains the values that identify a business partner uniquely. A business partner is identified uniquely by means of its GUID. For more information about additional ways of identifying business partners, see SAP Note 829193.
	OBJECT_TASK	Contains the processing method that is to be executed during inbound processing and applied during outbound processing at object level. The processing methods can also be specified on the lower levels of the respective dataset. For more information, see Processing Method [page 242] .
CENTRAL_DATA		In the lower components, this contains central data for business partners, such as the Business Partner Role [page 163] , Bank Details [page 185] , and Identification Number [page 184] . This information is sorted by business criteria.
FINSERV_DATA		In the lower components, this contains business partner data for Financial Services, such as Ratings [page 188] , Regulatory Reporting Data [page 190] , and Fiscal Year Information [page 191] . This information is sorted by business criteria.

3.1.5.2.10.1.1.1 Transfer Structure for Master Data: Bank Analyzer Specifics

Definition

The complex interface for transferring business partner master data (transfer structure BUS_EI_EXTERN) contains certain components that are specific to Bank Analyzer.

Structure

Bank Analyzer-Specific Components of Structure BUS_EI_EXTERN

ComponentLevel 1	ComponentLevel 2	More Information
HEADER	OBJECT	The business object that is relevant for the complex interface in Bank Analyzer is called */BA1/B107 (SAP BP FS as part of Bank Analyzer).*
	SCENARIO	Contains general information about processing scenario data for the business partner [page 222] (scenario key and an indicator that can ensure that only scenario data for the business partner is read).
	FSBP_VERSION_DATES	Contains the date information of the business partner version [page 211] . You can use existing business partner versions as the basis for creating new versions. Existing business partner versions are the source versions and can be selected to the millisecond (ms) by specifying the timestamp of the business record dateand the system date/time. The business partner versions to be created are the target versions and you can identify them using the business record date.
BASEL		Contains the Basel II attributes [page 197] of the business partner.

ComponentLevel 1	ComponentLevel 2	More Information
SEGMENT		Contains the Bank Analyzer characteristics of the business partner for the segmentation service [page 1843] . The segmentation service is a generic service for enhancing primary objects within the Source Data Layer in a standardized manner. The segmentation service can be used to assign additional characteristics or key figures in a differentiated or undifferentiated structure to the business partner. The characteristics and key figures are available in all the BP roles to which the segmentation service is assigned. To change a characteristic in the segmentation service it is necessary to supply all characteristics for the relevant business partner anew. Characteristics that are not provided are not transferred into the current business partner version. Processing takes place in the same way as when the current state method is used.

3.1.5.2.10.1.2 Transfer Structure for Business Partner Relationships

Definition

The complex interface for the transfer of business partner relationships contains transfer structure *BURS_EI_EXTERN (complex external interface for a relationship)*, which is nested deep in other components on the basis of business aspects.

Structure

Component Level 1	Component Level 2	Additional Information
HEADER		Contains general data and header data for the transfer of Business Partner Relationships [page 169].

Component Level 1	Component Level 2	Additional Information
	OBJECT	Contains the object type as the internal technical key of a business object. The business object relevant for the complex interface is *BusinessPartnerFS*.
	OBJECT_INSTANCE	Contains the values that identify a business partner relationship uniquely. These include the identification of the business partners in the relationship, the Business Partner Relationship Category, the validity date of the relationship, and the .
	OBJECT_TASK	Contains the processing method that is to be applied at object level. The processing method can also be specified on the lower levels of the respective dataset. For more information, see Processing Method [page 242].
CENTRAL_DATA		In the lower components, this contains the Central Relationship Data [page 199] sorted by particular business criteria.
	MAIN	In the lower components, this contains central relationship data sorted by particular business criteria, such as the *Validity Period* of the business partner relationship, the Relationship Type, and the *Standard Relationship* indicator.
	ADDRESS	In the lower components, this contains the *Address Data for the Relationship*, such as the *Addresses* and the *Communication Types*. This data is sorted by particular business criteria.
	CONTACT	In the lower components, this contains the data about the contact person for the relationship, such as the *Function* and the *Department* of the contact person, or this person's *Telephone Number*.
	COMPANY_INTEREST	In the lower components, this contains shareholder data for the relationship, such as the *Amount* and the *Capital Investment as a Percentage*. This data is sorted by particular business criteria.

Component Level 1	Component Level 2	Additional Information
	BUSINESS_HOUR	In the lower components, this contains the *Hours of Business* of the relationship sorted by particular business criteria.

Integration

Before you can load business partner relationships, the business partners in these relationships must have already been transported into the target system by means of the Transfer Structure for Master Data [page 236]. Only then can a relationship be created. If the business partner master data for the related business partner has not yet been transferred, an error message is output.

Different attributes are expected, depending on the relationship category, for example the data from the COMPANY_INTEREST component is processed for shareholder relationships.

3.1.5.2.10.1.2.1 Transfer Structure for BP Relationships: Bank Analyzer Specifics

Definition

The complex interface for transferring business partner relationships (transfer structure BURS_EI_EXTERN) contains certain components that are specific to Bank Analyzer.

Structure

Bank Analyzer-Specific Components of Structure BURS_EI_EXTERN

Component Level 1	Component Level 2	More Information
HEADER	OBJECT	**Note** The business object that is relevant for the complex interface in Bank Analyzer is called /BA1/B107 (SAP FS-BP as part of Bank Analyzer).

	FSBP_VERSION_DATES	Contains the date information of the business partner version [page 211]. You can use existing versions of business partners to create new versions. Existing business partner versions are the source versions and can be selected to the millisecond (ms) by specifying the time stamp of the business record date and the system date/time. The business partner versions to be created are the target versions and you can identify them using the business record date.

3.1.5.2.10.1.3 Processing Method

Use

You enter the processing methods for the complex interface in the control segments of the transfer structures.

Control segments of transfer structures are used in the different hierarchical levels of the complex interface:

Hierarchy	Level	Component
1	Object	OBJECT_TASK
2	Table	CURRENT_STATE
3	Data record	TASK
4	Single field	DATAX

The processing methods of the complex interface define for each hierarchy level how the system treats the data that is in the complex interface and is to be transferred.

The processing method at higher levels in the hierarchy governs which processing method is used for the segments further down in the hierarchy.

For an explanation of the processing methods used in the different levels of the interface, see the example “Control Segments for Ratings”. This example takes a cross-section of the Ratings [page 188] in the transfer structure for master data

Integration

Processing methods are entered in the complex interface for the business partner in the Transfer Structure for Master Data [page 236] and the Transfer Structure for Business Partner Relationships [page 239].

For information about how the processing method selected interacts at various hierarchy levels of the complex interface, see Effects of Processing Method Selected at Object Level [page 247].

Features

Control at Object Level (Hierarchy Level 1)

At the highest level, which is object level (component HEADER-OBJECT_TASK), the control segment makes the processing methods *Insert* and *Current State* effective for the entire object.

In the inbound processing of the complex interface, the following processing methods can be used at object level:

Insert (I)

The business partner/business partner relationship is created. This processing method is, therefore, used at object level only. It creates new business partners/business partner relationships. If the business partner already exists in the system, then the system issues an error message.

The control at data record level and single field level is ignored in mode I.

Current State (C)

The business partner is completely contained in the data structure and is to be created if it does not yet exist in the system, or changed if it already exists. All the business partner/business partner relationship components that are not contained in the data structure are deleted in the system.

Control at data record level and single field level is generally not provided if you use processing mode C.

Update (U)

This processing method is used to update business partners that are already contained in the target system. If the business partner does not yet exist in the target system, the system issues an error message. You can also use this processing method to select the business partner for archiving in the target system.

When processing method U is used, the control at table level or data record level is required.

Modify (M)

When this processing method is used, a distinction has to be made between the following two cases:

- If the business partner already exists in the target system, the business partner is updated.

  In this case, the control at table level or data record level is additionally required (see Update scenario).
- If the business partner does not yet exist in the target system, then it is created. The control at data record level or single field level is ignored (see Insert scenario).

Delete (D)

This processing method is not supported at object level.

Control at Table Level (Hierarchy Level 2)

You can set the *Current State* control segment at table level. In this case, the system's behavior for the relevant table is exactly the same as it would be if the *Current State* processing had been set globally at object level.

The *Current State* mode is then effective for all data records in the relevant table (for example, for all addresses of the business partner).

Control at data record level and single field level is not provided if the CURRENT_STATE control segment is set.

Control at Data Record Level (Hierarchy Level 3)

If you use the TASK control segment at data record level, then this processing mode is effective only for one single row in the affected database table.

The system determines which data record is to be changed by means of the DATA_KEY or DATA structure. This structure contains the data record key.

In the inbound processing of the complex interface, the following processing methods can be used at data record level:

- **Insert (I)**
- **Update (U)**
- **Modify (M)**
  In mode M, the system tests whether the data record already exists. If it does, then the data record is changed as in method U. If it does not yet exist, then the data record is created as in method I.
- **Delete (D)**
  This mode deletes a single data record.

Processing mode *I* creates data records in the same way as the *Add* BAPI. Processing mode *U* changes data records in the same way as the *Change* BAPI. Processing mode D deletes data records in the same way as the *Remove* BAPI.

Control at Data Record Level for Addresses

In addition to the general processing methods, the following values for the control segments are possible for the business partner addresses:

- Standard (S) only permitted for standard addresses
- Logical key insert
- Logical key update
- Logical key delete
- Logical key modify

You can use processing mode S when you transfer a BP address. This standard mode processes an address data record in such a way that the standard address is processed. This means that an address GUID does not have to be specified in this processing method.

Logical processing methods behave in the same way as the other processing methods for control at data record level. Addresses are identified by specifying an activity in the operation field in the DATA_KEY structure rather than by specifying an address GUID.

For more information about address activities, see Address Data [page 177].

Control at Single Field Level (Hierarchy Level 4)

If you specify the processing method at single field level (DATAX component), then the processing mode is effective only for the selected field in the database table in question.

The system determines which field is to be changed by means of the DATA and DATAX components. The DATAX component contains the same number of fields as the DATA component, but it has an indicator for *change information for the relevant user data field* for the DATA component This means that you need to set the appropriate change indicator in the DATAX component for the fields of the DATA component whose data you want to change in the data transfer. The processing of control at single field level uses the same processing logic as the corresponding field in the underlying BAPIs.

The system treats control at single field level as not relevant, if you have set the processing method at table level (CURRENT_STATE component).

Example

Data Supply with the Processing Method OBJECT_TASK U [page 248]

Data Supply with the Processing Method OBJECT_TASK C [page 249]

3.1.5.2.10.1.3.1 Control Segments for Ratings

The following table, which contains example control segments for multi-value ratings in the transfer structure for master data, explains the processing methods for the various levels of the interface:

Component Level 1	Component Level 2 (Object)	Component Level 3 (Table)	Component Level 4 (Data Record/ Single Field)	Information
HEADER	*OBJECT_TASK*			Processing method for the object (I, M, U, C). The processing method is a required entry field.
FINSERV_DATA	RATINGS	*CURRENT_STATE*		Processing method for the table (C). You can select C here only if U or M is entered at object level. The processing method applies for this dataset only.

Component Level 1	Component Level 2 (Object)	Component Level 3 (Table)	Component Level 4 (Data Record/ Single Field)	Information
		RATINGS		Multi-value dataset for ratings.
			TASK	Processing method for a data record or row. If processing method C is not set at table level and data is available for transfer, then you have to enter a processing method.
			DATA_KEY	Identification of the table record for ratings.
			DATA	Ratings [page 188]
			DATAX	Processing methods at single field level: Change indicator for the related user data field. Contains information as to whether the field of the same name in the DATA component is to be changed. If you use processing method C or I, the DATAX component is of no relevance. The DATAX component contains the same number of fields as the DATA component. If a field in the DATA component is to be changed, the indicator must be specified here for the relevant field. The two structures are assigned to one another using the same name.

3.1.5.2.10.1.3.2 Effects of Processing Method Selected at Object Level

The components OBJECT_TASK, CURRENT_STATE, and TASK as well as the DATAX structure determine the processing method [page 242] for data transfer. They are described, using the ratings dataset as an example, in Control Segments for Ratings [page 245] .

The use of a processing method in the lower hierarchy levels depends on which value you have selected for the component OBJECT_TASK at object level:

Processing method under OBJECT_TASK	Effects on lower hierarchy levels
Current State (C) or Insert (I)	If you set the value C or I in the OBJECT_TASK component, this makes it clear which transfer mode is to be used. If you set the *Current State* indicator at table level, you no longer need to enter a processing method for the TASK component at data record level or to set the change indicator at single field level. The system ignores any values entered here.  We recommend that you use processing methods C and I if you want to import all business partner data (full supply).
Update (U) or Modify (M)	If you set the value U or M in the OBJECT_TASK component, this means that you have to enter further details about which method is to be carried out in the lower object levels. If you set the *Current State* indicator at table level, you have to enter a method at data record level (TASK) and/or single field level (DATAX). The following dependencies then come into effect: If the *Current State* indicator is set at table level, then you do not have to enter any more information at data record level or at single field level. The system ignores any information available at these levels. If the *Current State* indicator is not set at table level, then you have to enter which method is to be carried out at data record level (TASK). If you enter method U or M under TASK, then you also have to set the indicator at single field level (DATAX) for the values to be transferred. If you select another processing method at data record level, you do not need to enter values for the DATAX component.

3.1.5.2.10.1.3.3 Data Supply with the Processing Method OBJECT_TASK U

If you want to supply your data with processing method OBJECT_TASK U, then you have to enter processing method M, U, I, or D in the control segment TASK at data record level.

This relationship is explained using the example of business partner identification numbers:

	DATA
Business partner number	5824
Identification type	Commercial register number
Identification number	7272
Responsible institution	-
Entry date	8/20/2005
Valid from	11/21/2002
Valid to	12/31/9999
Country	Germany
Region	Hesse

	TASK U		TASK I	
	DATA	DATAX	DATA	DATAX
Business partner number	5824	-	5824	-
Identification type	Commercial register number	-	Register of associations number	-
Identification number	7272	-	6262	-
Responsible institution	-	-	-	-
Entry date	1/30/2005	X	11/21/2002	-
Valid from	11/21/2002	-	11/19/1969	-
Valid to	12/31/2020	X	12/31/9999	-
Country	-	X	Germany	-
Region	-	X	Lower Saxony	-

	TASK U* used	TASK I* used
	DATA	DATA
Business partner number	5824	5824
Identification type	Commercial register number	Register of associations number
Identification number	7272	6262
Responsible institution	-	-
Entry date	1/30/2005	11/21/2002
Valid from	11/21/2002	11/19/1969
Valid to	12/31/2020	12/31/9999
Country	-	Germany
Region	-	Lower Saxony

* The complex interface determines which TASK is required.

3.1.5.2.10.1.3.4 Data Supply with the Processing Method OBJECT_TASK C

If you want to supply your data using processing method OBJECT_TASK C, then you do not need to differentiate any further at object level.

This relationship is explained using the example of business partner identification numbers:

	DATA	DATA
Business partner number	5824	5824
Identification type	Commercial register number	Public register of cooperatives number
Identification number	7272	5555
Responsible institution	-	-
Entry date	8/20/2005	8/30/1999
Valid from	11/21/2002	1/30/1974
Valid to	12/31/9999	12/31/9999
Country	Germany	Germany

	DATA	DATA
Business partner number	5824	5824
Region	Bavaria	Hesse

	DATA	DATAX	DATA	DATAX
Business partner number	5824	-	5824	-
Identification type	Commercial register number	-	Register of associations number	-
Identification number	7272	-	6262	-
Responsible institution	-	-	-	-
Entry date	1/30/2005	-	11/21/2002	-
Valid from	11/21/2002	-	11/21/2002	-
Valid to	12/31/2020	-	12/31/9999	-
Country	-	-	Germany	-
Region	-	-	Saxony	-

	TASK U* used	TASK I* used	TASK D* used
	DATA	DATA	DATA
Business partner number	5824	5824	5824
Identification type	Commercial register number	Register of associations number	Public register of cooperatives number
Identification number	7272	6262	5555
Responsible institution	-	-	-
Entry date	1/30/2005	11/21/2002	8/30/1999
Valid from	11/21/2002	11/21/2002	1/30/1974
Valid to	12/31/2020	12/31/9999	12/31/9999
Country	-	Germany	Germany
Region	-	Saxony	Hesse

* The complex interface determines which TASK is required.

3.1.6 Primary Object: Organizational Unit

Definition

The primary object 'organizational unit' is an organizational unit that has a hierarchy. It represents the organizational structure of the organization.

You have to use the adapter for Tool Business Intelligence (Tool BI) [page 252]. The adapter for Organizational Management reads the master data, but not the hierarchy.

Use

Each organizational unit is responsible for running particular functions in the bank. Depending on how tasks are distributed, an organizational unit can be a business area, a department, or a group.

The hierarchies represent the functional structure of a bank and are required for other processes such as profitability analysis in Bank Analyzer.

Structure

Organizational units are fixed characteristics in Bank Analyzer. Characteristics are assigned a fixed role *organizational unit* (`KEY 80`). By assigning the role to an adapter in Customizing, you can determine which central repository of organizational units the system is to access. For more information, see Storage of Organizational Units and Data Access [page 251].

3.1.6.1 Storage of Organizational Units and Data Access

Use

This function allows you to store organizational units centrally, for example in Tool Business Intelligence [page 252] (Tool BI) or in Organizational Management [page 253]. You can access the data of the organizational units in different storage locations using a reference data connector.

Integration

The following DataSources can be used to extract the master data, attributes, and hierarchies of the organizational units from Organizational Management to Tool BI:

- `0TB_ORGCAT_ATTR`

Loads the master data and attributes for all organizational units that are assigned to an organizational unit category, or that are used in an organizational plan.

- `0TB_ORGCATH_TEXT`
  Loads the text for the master data.
- `0TB_ORGCATH`
  If you use Business Content, this DataSource creates a hierarchy of all organizational plans whose nodes refer to InfoObject /BA1C/CORG. As a result, extraction must occur in this InfoObject, and the InfoObject must be appropriately renamed based on the entries in BI adapter.

Prerequisites

In Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Metadata* > *Characteristics* > *Roles* > *Assign a role to an adapter*, you have defined the central storage location where the system is to access organizational units.

If you want to use profitability analysis, you have to use the BI adapter.

Features

You access the master data, attributes, and hierarchies of the organizational units using a reference data connector. This consists of the following components:

- A proxy that supplies the master data from the central data store in Bank Analyzer.
- Implementation of the role (`KEY 80`) that is assigned to fixed characteristic *organizational unit*.
  You can assign other fixed or custom characteristics that represent organizational units to this role.
- Adapters that enable access to the data sore of the organizational units in Bank Analyzer.

3.1.6.1.1 Storage in Tool BI

Use

This function allows you to store organizational units in Tool Business Intelligence [page 16] (Tool BI). You can access the data flexibly using a reference data connector.

Prerequisites

In Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Metadata* > *Characteristics* > *Roles* > *Assign a role to an adapter*, you have assigned the adapter for BI access to the role of the organizational unit. This is the default setting.

Features

You can use Tool BI to structure organizational units hierarchically and to store them. The system uses the adapter for accessing Tool BI to identify a global InfoObject for all organizational units. This InfoObject defines the master data, attributes, and the hierarchy in Tool BI.

The attribute contains master data for the organizational unit type that you can map to the predefined organizational unit categories in Customizing. This assignment allows you to map organizational units semantically for other processes such as profitability analysis. For more information, see the Implementation Guide (IMG) for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Organizational Units* > *Assign Organizational Unit Categories for Profitability Analysis*.

The organizational unit hierarchy applies to all characteristics represented by organizational units. The system creates the name of the hierarchy from a cross-client name and the source system ID that corresponds to the client of the core Bank Analyzer system. This combination allows you to use different hierarchies in different clients.

The hierarchy in Tool BI is time-dependent. This means that **one** generic hierarchy is valid for a key date. Because the hierarchy is time-dependent, you can reassign the organizational structure to a new key date. Only **one** attribute value per organizational unit is valid for a key date. You can reassign the attribute values to the organizational units for different key dates.

You can use report *Display Organizational Structure* (`/BA1/RAL_FM_ORG_SHOW`) to display the hierarchy of the organizational units.

3.1.6.1.2 Storage in Organizational Management

Use

This function allows you to store organizational units in Organizational Management. You can access the data flexibly using a reference data connector.

Integration

In Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Metadata* > *Characteristics* > *Roles* > *Assign Roles to Adapters*, you have assigned the adapter for accessing Organizational Management to the role of the organizational unit.

Features

You can store organizational units hierarchically as an organizational plan using Organizational Management. The organizational plan is based on the hierarchy of the organizational units. Bank Analyzer uses primary object 'organizational unit' to map the organizational plan.

The organizational unit types defined in Customizing for Organizational Management can be used to assign additional hierarchy levels in the organizational plan. For this purpose, each organizational unit type must be assigned to an organizational unit in the organizational plan. For more information, see Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Organizational Units* > *Edit Type of Organizational Unit for Organizational Management*.

More Information

http://help.sap.com/nw *SAP NetWeaver Plattform* > *Application Help* > *Function-Oriented View* > *Application Server* > *Application Server ABAP* > *Other Services* > *Services für Business Users* > *Organizational Management*

3.1.7 Generic Market Data

Definition

Market data with a generic key comprising the market data area and custom characteristics.

Use

Generic market data enables you to use custom market data (for example, the height of the waves in the North Sea, probability of default, and so on) for specific evaluations. This is done by categorizing generic market data into market data classes in the system. You create market data classes in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Generic Market Data* > *Edit Generic Market Data Classes*. You can assign up to five key figures and custom characteristics to each market data class.

The various Analyzers access the generic market data via a primary data source.

> **Note**
>
> Each market data class uses one primary data source.

Structure

Generic market data consists of the following fields:

- Market data area: A market data area is a component of the primary keys of all market data in Bank Analyzer, and therefore also a component of every interface that requests market data. This area is used for controlling all visibility and authorization for market data. All market data is stored with a reference to a market data area.
- Defining characteristics: Defining characteristics identify a piece of generic market data uniquely.

- Key figure: A piece of generic market data contains one key figure of the category *Percentage Rate*

Example

Before you can work through this example, you have to create a market data area in Customizing for *Banking Services* under *Financial Services* > *Foundation* > *Market Data* > *Create Market Data Area*.

To map probabilities of default (PD), you can create a market data class in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Generic Market Data* > *Edit Generic Market Data* using the following data:

- Market data class: PD01
- Key figure: PD (value of the probability of default in percent points)
- Term indicator: Activate
- Selection method: Interpolation
- Defining Characteristics:

Name of Characteristic	Type/Length	Name
VFD	CHAR 2	Valuation factor determination
PD_CLASS	CHAR 5	PD grade

On the *SAP Easy Access* screen, under *Financial Services* > *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Generic Market Data* > *Edit Generic Market Data*, and enter business record date December 1, 2002 and a creation time stamp for the following values in the above market data class:

MD Area*	VFD*	PD_CLASS*	Maturity*	Unit of Mat.*	PD
TEST	01	INT01	20	1	0,6%
TEST	01	INT01	30	1	0,8%
TEST	01	INT01	40	1	1,1%
TEST	02	INT01	20	1	0,55%
TEST	02	INT01	30	1	0,78%
...	...	...	...	...	...

Notes:

- An asterisk (*) denotes a key field.
- Unit of mat.: 1 = days, 2 = months, 3 = years

The Analyzer then makes a request for the market data class PD01 for the business record date 12/1/2002 and a corresponding timestamp with the following selection criteria:

MD Area*	VFD*	PD_CLASS*	Maturity*	Unit of Mat.*
TEST	01	INT01	37	1

Depending on the selection procedure defined for the market data class, the system returns the following values:

Selection Procedure	Result (PD)
1 Next highest maturity	1,1%
2 Next lowest maturity	0,8%
3 Interpolation	1,01%
4 Nearest neighbor	1,1%

3.1.7.1 Displaying and Editing Generic Market Data

Use

You use this transaction to display and edit custom market data.

Prerequisites

- You have created a market data area in Customizing for *Banking Services* , under *Foundation* > *Market Data* > *Create Market Data Area.*
- You have defined a market data class in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Generic Market Data* > *Edit Generic Market Data Classes* .

Procedure

Displaying Generic Market Data

1. In the SAP Easy Access menu, choose *Financial Services* > *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Generic Market Data* > *Display Generic Market Data* .
2. Select a *generic market data area* .

3. Select a *class name* .
4. Enter a *validity date* .
5. Under the additional *Defining Characteristics* option, you can define concrete values for specific characteristics.
6. Choose *Display* .
   The system displays the generic market data in a list.
7. In this list, select a market data item, and choose *Display Generic Market Data* to go to the detail screen for the item in question.

Editing Generic Market Data

1. In the SAP Easy Access menu, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Generic Market Data* > *Edit Generic Market Data* .
2. Select a *generic market data area* .
3. Select a *class name* .
4. Enter a *validity date* .
5. Select *defining characteristics* .
6. Choose *Change* .
   The system displays the generic market data in a list.
7. You now have the following options:
   - Add data to or change existing data in the generic market data item. To do this, select the item in the list, and choose *Change Generic Market Data* .
   - Create new values for the defining characteristics of the generic market data. To do this, choose *Create Generic Market Data* .
8. Save your entries.

3.1.8 Primary Object: Aggregation Object

Definition

Aggregation objects are generated in source data aggregation (see Source Data Aggregation [page 286]). They are generated in the *Aggregate Financial Transactions* process (see Aggregating Financial Transactions [page 306]). The system assigns financial transactions that have the same values as the granularity characteristics to an aggregation object. If the values change, the financial transaction is reassigned to the corresponding aggregation object.

Use

To display an aggregation object, on the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Aggregation Objects* > *Display Aggregation Objects* . Then use the external number and the system time to select the aggregation object.

> **Note**
>
> The external number is a technical GUID that is created when the system generates the aggregation object. If you do not know the GUID, you can search for it using granularity characteristics. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Aggregation Objects* > *Find Aggregation Objects*.

A list containing the attributes for the aggregation object is then displayed. From this list, you can display financial transactions that are assigned to the aggregation object.

3.1.9 Operational Event

Definition

An event that defines an occurrence in an object in an operational system. The event can refer to a change in master data or to a flow. Application event management [page 368] can derive one or more application events from the items of an operational event. Application events trigger analytical applications. Operational events are mapped using business transactions.

Use

You manage operational events as follows:

Create and display

You can create operational events as follows:

- Import as external business transaction
  - BAPI
    You can import operational events as external business transactions using the Business Application Programming Interface (BAPI) for business transactions (`AddNewVersion` method for the business object `BusinessTransaction1`). If you create operational events using this BAPI, you can first load the operational events to the data import buffer [page 159]. You do this using the appropriate BAPI parameters.
  - Enterprise service
    You can also import operational events that represent a delinquency as external business transactions using an enterprise service interface.
- Synchronous creation from master data changes
  You can derive operational events from master data changes of the **business partner**. Derivation of the operational event data does not depend on the interface used to address the business partner (UI, BAPI, enterprise services, complex RFC interface). For more information, see Derivation of Operational Events from Business Partner Changes [page 263].

You can create operational events synchronously from master data changes of a **financial transaction**. This is supported for the BAPI and enterprise service interface:

- BAPI
  You can import operational events as external business transactions using the BAPI for master data for financial transactions (`Create` and `AddNewVersion` methods for the business object `FinanceTransactionBA`). In this case, you have to enter information about the type of operational event and you can enter the master data change area and its validity data.
- Enterprise service
  You can derive operational events using service interfaces for master data for financial transactions. You must use the Integration Operational to Analytics (IOA) scenario or SAP interfaces to import data.
  In this case, the service interfaces contain the necessary information for the events (for example, business process chain type code, change indicator in the message node). The relevant event information (type of operational event, master data change area) must be transformed from this data by mapping in the Service Inbound [page 359] software layer so that it can be used in Bank Analyzer. For more information, see Processing Service Operations in Software Layers [page 355].
  You can map changes in business objects in operational systems to primary objects in Bank Analyzer using services interfaces [page 354]. For example, if the system transfers a master data change to primary object 'financial transaction', you can define whether the system is to create an operational event as well as making the changes in the financial transaction. For the creation of the operational event, you have to assign a business transaction class and an item class of the business transaction. You make this setting in Customizing for *Bank Analyzer* under *Source Data Layer* > *Interfaces to Other Applications* > *Interfaces to Account Management* > *Edit Service Operation for Target Object Financial Transaction*.

When you create operational events from master data changes to the financial transaction, you can add specific data to the operational event. For more information, see Data Enrichment of Operational Events from Financial Transaction Changes [page 260].

See also:

- Creating Business Transactions [page 154]
- Creating Business Transactions by Copying [page 155]
- Displaying Business Transactions [page 158]

Derive

If you enter your data manually, they system automatically starts to derive application events in application event management as soon as you save the business transaction with the operational event. If data is entered automatically (using BAPIs), derivation is run in the loading process provided you have released the operational event and it is not loaded to the data import buffer.

Only after **all** derivations have been successfully completed does the system save the operational event in Source Data Layer (SDL).

Delete

You can delete operational events only in the data import buffer; in other words, if they have been created, but not yet been through derivation.

Reverse

If derivation has run for the operational event and you then reverse it, the system sends the reversal of the operational event (reversal business transaction) to application event management. Application event management then reverses the derived application events.

See also: Reversing Business Transactions [page 156]

Archive

You can archive derived operational events and reversals of operational events in SDL using archiving objects [page 279]. The system archives operational events as business transactions without considering the following objects:

- The referenced master data object
- The referenced business partner
- Derived application events in application event management

Structure

Operational events are structured as follows:

- Operational event type
  You classify the operational event in the business transaction header. You define the types of operational event in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Flow Data* > *Operational Events* > *Edit Operational Event Type*.
- Reference to a business partner or object from the Bank Analyzer master data framework
  You define the business partner in the business transaction header.
  You define the master data object in the business transaction item. The account data and the external number are relevant for the master data object. This data defines the reference to the object for which a master data change or a flow is contained in the operational event.
- Relevant for operational event
  A business transaction item for which this indicator is set contains the reference fields of the master data object that the operational event refers to. You enter an account category, an account number, and an external number in the item. These values uniquely define the referenced master data object.
  The system derives application events only from items that are marked as relevant for an operational event. If business transactions refer to a flow transaction (such as a curtailment), the relevancy indicator must be set in the item that triggers the operational event. Business transaction items of operational events for master data changes must all be marked as relevant for an operational event.

3.1.9.1 Data Enrichment of Operational Events from Financial Transaction Changes

Use

Before operational events are generated from master data changes for financial transactions that are generated in sync with financial transaction versions, you can enrich the corresponding business transaction if required. This data enrichment may be required for deriving application events and for processing these in the analytical application (for example, impairment).

You have the following options:

- Determining the validity date
  The system determines the validity of master data changes from the imported financial transaction version (for example, from limits or rollovers). Several dates can be entered in the business transaction for changes to

a defined part of the master data version (master data change area). The system creates a new business transaction item for every validity change to a master data change area. The validity determines as of when a master data change is relevant for a contract.

- Refining master data change areas
  The system refines the granularity of master data change areas by replacing an imported master data change area with several other master data change areas. For an example of this process, see the implementation `/BA1/AL_F1_OE_VAL_DET_ALL_UNKN` of *BAdI: Data Enrichment of Operational Events*.

Prerequisites

- You have delivered the type of the operational event and the master data change areas with the master data version.
  If the creation of the operational event is triggered by an enterprise service, the type of the operational event and the master data change area have to be derived in the service inbound layer.
- There is no validity data for the master data change areas.
- You have made the necessary settings in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Flow Data* > *Operational Events* > *Creation of Operational Events from Master Data Changes* > *Financial Transaction*.
  For more information, see the documentation for the relevant Customizing activity.

Activities

The following table provides an overview of the system activities if an operational event is created due to a master data change in financial transactions.

Import: MDCA	Import: Validity of Master Data Change	Methods Are *Not* Assigned to OE Type and MDCA in Customizing	Methods Are Assigned to OE Type and MDCA in Customizing
No	n/a	The system creates a business transaction with a single item: • With financial transaction ID • Without master data change area • Without validity data	
Yes	Yes	The system uses the master data change areas and validity data delivered. It does not import any data.	

Import: MDCA	Import: Validity of Master Data Change	Methods Are *Not* Assigned to OE Type and MDCA in Customizing	Methods Are Assigned to OE Type and MDCA in Customizing
Yes	No	The system copies the master data change areas delivered. It does not import any data.	The Business Add-In *BAdI: Data Enrichment of Operational Events* is called before a business transaction is created. For more information, see the BAdI documentation.

Abbreviations:

MDCA = master data change area

OE type = type of operational event

> **Note**
>
> If the *Type of Operational Event* field is delivered empty, the system does not create a business transaction.

Example

Rollover for a Loan

1. The loan starts on January 1, 2008 and has a fixed interest rate period of two years.
2. On December 5, 2009, the customer comes to the bank and rolls the loan over on January 1, 2010.
3. The system generates a new financial transaction version for December 5, 2009 with the following event information:Event Type *Rollover* and Master Data Change Area *Fixed Interest Period*.
4. The system derives January 1, 2010 from the current version. To do this, it calls the BAdI implementation `/BA1/AL_F1_OE_VAL_DET_SFI`.

Interest Rate Change for a Loan

1. The loan starts on January 1, 2008 and has a fixed interest rate period of two years.
2. On January 15, 2010, the customer agrees a new interest rate that is valid from February 1, 2010.
3. The system generates a new financial transaction version for January 15, 2010 with the following event information: Event Type *Interest Change* and Master Data Change Area *Fixed Interest Period*.
4. The system derives February 1, 2010 by comparing the current version with the previous version. To do this, it calls the BAdI implementation `/BA1/AL_F1_OE_VAL_DET_FCO`.

More Information

For more information about correcting changes to a financial transaction, see Versioning of Primary Objects [page 103].

3.1.9.2 Derivation of Operational Events from Business Partner Changes

Use

You can derive operational events [page 258] from master data changes of the analytical business partner. The business partner can be addressed via different interfaces (UI, BAPI, Enterprise Services, complex RFC interface). The derivation of the operational event data is independent of the interface used.

If you have made the relevant Customizing settings (see *Prerequisites*), operational events are created from master data changes of the analytical business partner in sync with new business partner versions.

Integration

The system derives application events from the operational events and then forwards these application events to the analytical applications (to impairment processes, for example). The system derives application events only from items that are marked as relevant for an operational event. It derives one or more application events for each business transaction item for each application that is registered for the type of operational event. You can use a filter to exclude certain operational events from the derivation.

For more information, see Application Event Management [page 368].

> **Note**
>
> Error messages that occur when operational events are created from business partner changes are not returned to the business partner but are stored in the application log. To call the application log, choose transaction `SLG1` (log object: *Business Transactions*, log subobject: *Business Transaction Event*).

Prerequisites

- You have made the necessary settings in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Flow Data* > *Operational Events* > *Creation of Operational Events from Master Data Changes* > *Business Partner in Bank Analyzer*. For more information, see the documentation for the relevant Customizing activity.
- The business transaction item to which the business partner change refers is marked as relevant for an operational event.
- You have entered the source system of the business transactions on the *Bank Analyzer Attributes* tab page in the business partner.

  > **Note**
  >
  > The *Source System* field is a required entry field. If the system is unable to find a source system, it triggers an entry in the application log.

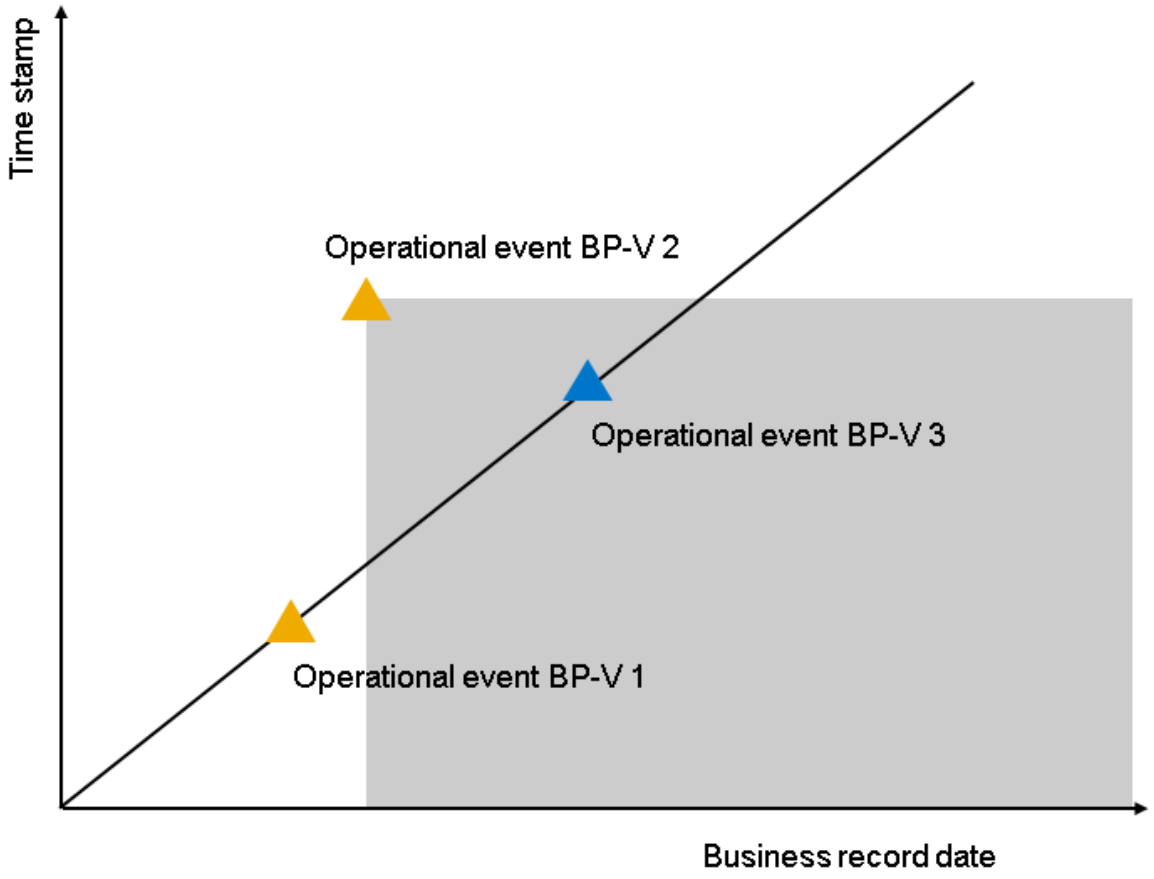
Features

Depending on your Customizing settings, the change to business partner data is mapped differently in Bank Analyzer. Changes to several business partner change areas in the same business partner version can have the following effect:

- The system creates one business transaction (an operational event) containing an item for the individual business partner change areas.
- The system creates a business transaction (operational event) for each change. Each business transaction has one item for the corresponding business partner change area.

Reversing Operational Business Partner Events

The figure below illustrates the reversal of operational events from business partner changes:



Before it saves an updated business partner version (BP-V 2), the system checks whether an operational event needs to be derived. If an operational event is derived, the system checks whether there are business partner versions with operational events that have a later business record date (in this case: BP-V 3). This may be the case if delivery of business partner version 2 was late, for example.

In this case, the system reverses all the operational events of business partner version BP-V 3 regardless of the change area before it creates business partner version 2 and its corresponding operational event. The system reverses the events only. The business partner version is kept. If you want to create events again for business partner version 3, you have to redeliver this version.

Example

Deterioration of a Rating

On March 1, a customer informs the bank that they are moving abroad on April 1, which means a deterioration in their rating from this date. On March 1, the bank enters the new rating for the business partner with a valid-from date of April 1. The operational system delivers the most up-to-date business partner version to Bank Analyzer. In Bank Analyzer, the deterioration in the rating in the Source Data Layer (SDL) triggers the creation of an operational event with the type *Business Partner Change*. The valid-from date of the new rating is entered in the item of the created business transaction.

If you have configured Customizing accordingly and the business transaction item is marked as relevant for operational events, on March 1 the system derives an application event for the impairment processes. The system can then derive events for accounting processes from the determination of the impairment attributes.

3.2 Source Data Services

You can use the following services:

- Access to Source Data [page 265]
- General Functions for Source Data [page 272]
- Methods for Source Data [page 280]
- Source Data Aggregation [page 286]

3.2.1 Access to Source Data

Use

You use this function to provide the primary object data that is stored in the Source Data Layer (SDL) to the individual Analyzers of the Bank Analyzer system.

Features

Analyzers access the source data or the system provides the Analyzers with data in the following way:

1. The system first determines a worklist, which selects the relevant objects from the SDL. The system calls InfoSets [page 1616] with appropriate filter criteria to select these objects.
   For more information, see Index Function [page 266].
2. The system then enriches individual entries from the worklist with the required information. Primary [page 1593], secondary, or generic primary [page 1617] data sources are called up to enrich this data. The system can also use derivation rules.

3.2.1.1 Index Function

Use

The index function enables you to conduct a complex data **search across the primary objects** using characteristics. This allows you to determine, for example, financial transactions using information from the business partners involved.

> **Note**
>
> Alternatively, the Bank Analyzer applications can directly access the data. To enable this, you have to specify the ID of the respective primary object as a selection criteria.

Features

For complex data selection, the system stores characteristic data from the primary objects (index) and assigned relevancy indicators in the form of index tables [page 344] (DataStore objects) in SAPNetWeaver Business Intelligence (Tool BI [page 16]).

Using InfoSets [page 1616], the DataStore objects and master data tables can be linked to one another by joins. All the characteristics of the InfoSet, including those which were not originally attributes of an individual primary object, can be used as selection criteria.

Source data is accessed and selected in two stages:

1. First the identifiers (as well as other characteristics) of the primary objects involved in the InfoSet are selected from the source data index and placed in a worklist.
2. If you require additional data for a primary object, the system can determine this data for each results record in the worklist. The system extracts this data by accessing the source data directly using a secondary data source.

   > **Note**
   >
   > The InfoSet selection is not used to read the required data completely. Creating a worklist gives you the option of preselecting or enriching data.

In general, this search across all primary objects occurs as a **key date query** using the index function. The **delta query** is an exception. For more information on this, see Key Date Query and Delta Query [page 267].

3.2.1.1.1 Key Date Query and Delta Query

Use

The *key date query* uses a business key date and a system time stamp. It considers all primary object versions that are valid on the specified business key date and that were available in the system at the time given in the system time stamp.

The *delta query* of the index function differs from the *key date query* in two ways:

- The *delta query* considers only those data records that have changed between the last delta query and the current delta query. The *key date query*, on the other hand, considers all data records.
- However, the *delta query* reads all of the new versions from these changed data records, even though the versions have different business validity dates. The *key date query* reads only the version that is valid for the key date.

The *delta query* is used in the Historical Database for generating historical data, for example (see Generation of Historical Data [page 1481]).

Example

The graphs below illustrate the differences between the *delta query* and the *key date query*. They are a two-dimensional representation of the different versions of a data record. The x-axis shows the business validity dates in ascending order (key dates) and the y-axis shows the system time stamp in ascending order. See also the references to double versioning in Example: Versioning [page 105].

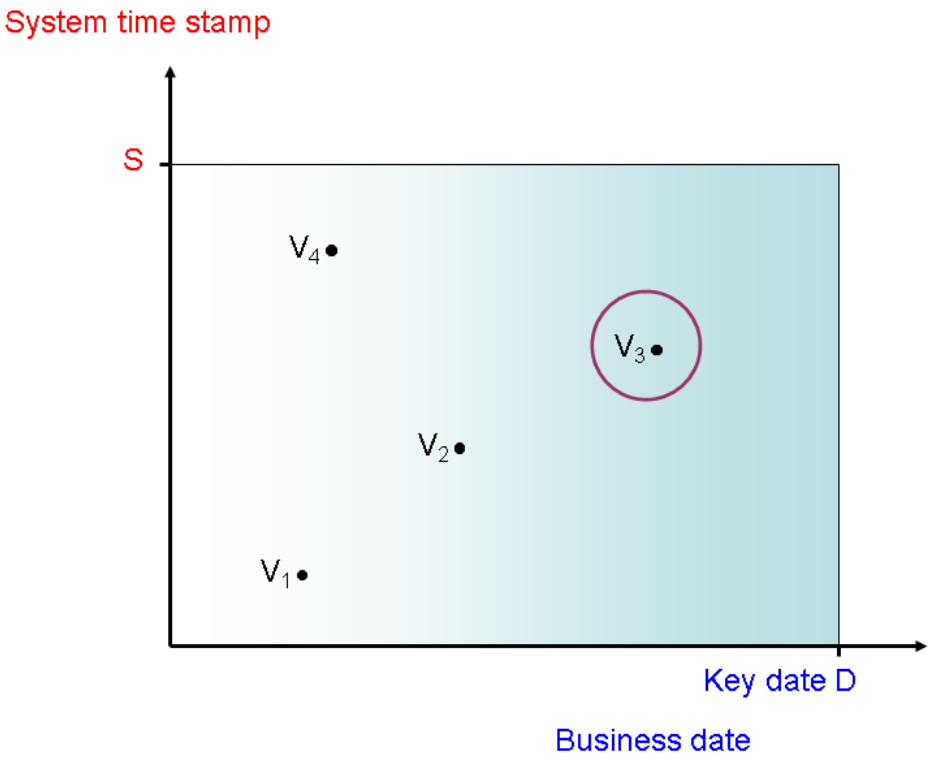


Figure 1: Key Date Query for Primary Object A

The graph shows the versions of the data record for primary object A for *key date query* D and system time stamp S. Version V3 is selected as a valid version. Version V4 is not read because it is no longer valid for the key date that is being queried.

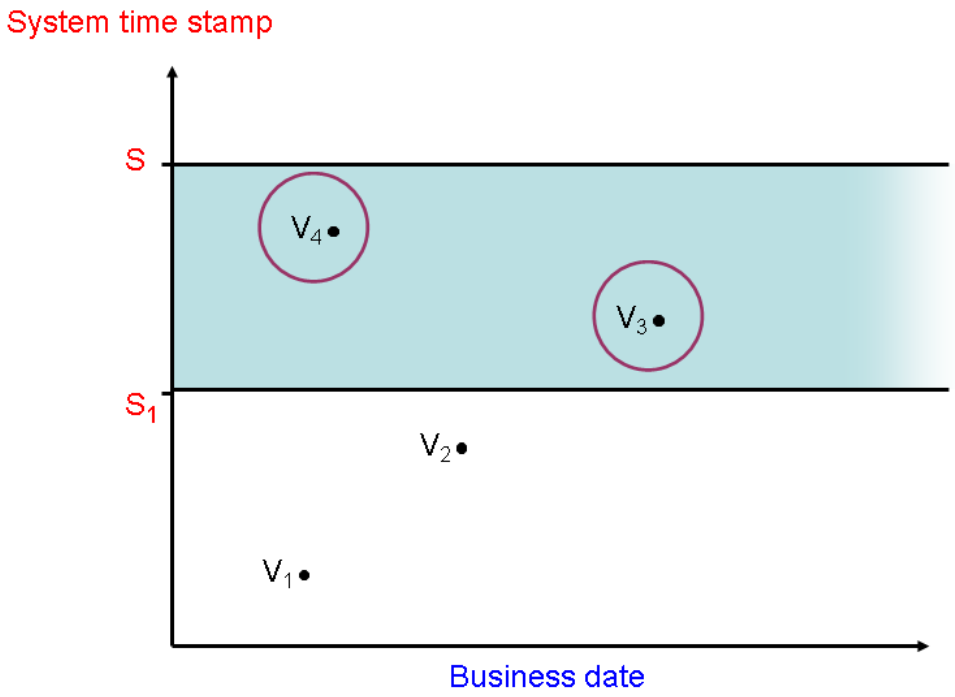


Figure 2: Delta Query for Primary Object A

This graph shows a *delta query*. In this case, versions V3 and V4 are read because they have both been created since the last query (system time stamp S1).

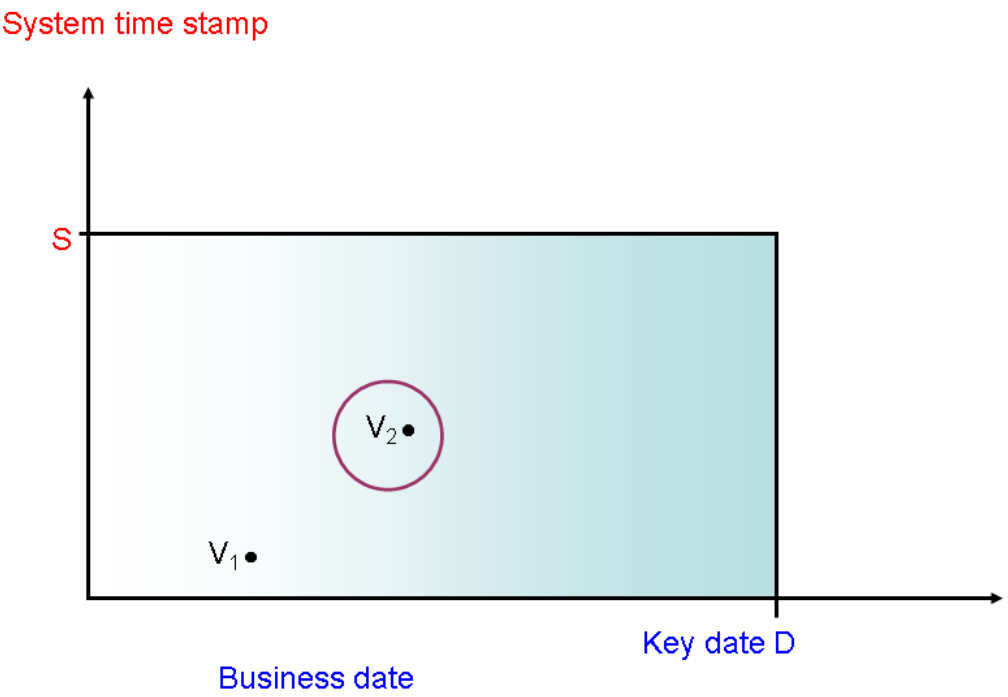


Figure 3: Key Date Query for Primary Object B

Figure 3 shows a different primary object, primary object B, and its two versions. In the *key date query* D, version V2 is read as the valid version.

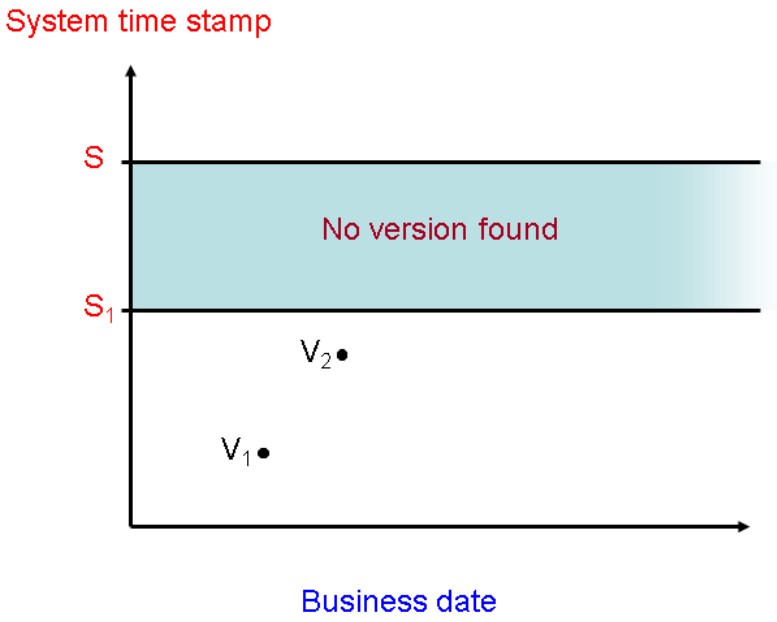


Figure 4: Delta Query for Primary Object B

This shows a *delta query* for primary object B. As there have been no changes (no new versions) since the last *delta query* (system time stamp S1), no versions are read.

Delta Query for Several DataStore Objects

In general, an InfoSet is a view of several DataStore objects. Therefore, a data record of the set of results depicts a combination of the data records of the individual DataStore objects. Let us take G/L account K and position B as examples here.

What affect does the internal composition of that combination have on visibility? Which combinations of object versions are relevant for the result of the *delta query*?

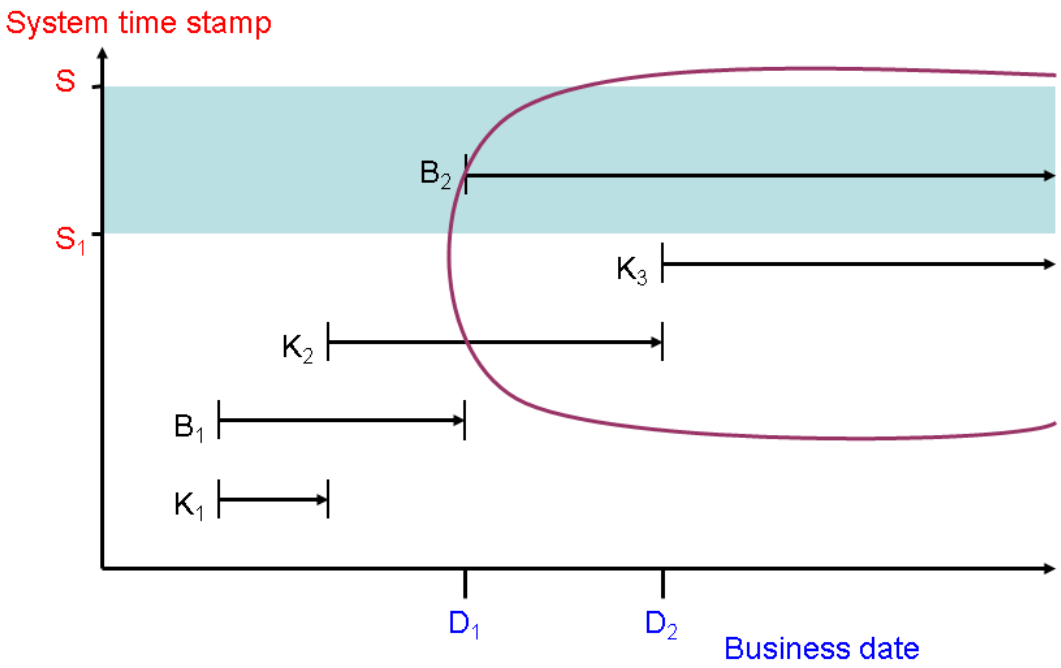


Figure 5: Delta Query for Several DataStore Objects

On the one hand only those combinations that contain at least one changed object are relevant. The example in figure 5 shows that only combinations that include version B2 fulfill this condition. On the other hand, the business validity must be considered: The *delta query* applies to the whole period of the business validity (in contrast to the *key date query*; see figures 1 and 3). However, the validity dates of the objects must overlap, as they do in the *key date query*. In the example in figure 5, the delta query does not read any combination containing K1 because the validity date of K1 does not overlap with that of B2. Thus, the only combinations that are possible are K2/B2 and K3/B2. The business validity date for these combinations results from the intersection of validity dates: D1 to D2 for K2/B2 and D2 to infinity for K3/B2.

3.2.2 General Functions for Source Data

You can use the following functions for the primary objects:

- Plausibility checks [page 273]
- Authorizations [page 276]
- Archiving [page 279]

3.2.2.1 Validations

Use

To ensure data consistency when primary objects are manually and automatically maintained, standardized SAP validation checks are run. SAP provides function modules for the module editor for this.

You can use these checks in addition to the consistency checks defined by SAP. As an extension to the validation checks in the SAP system, you can set up custom checks that can issue information messages, warnings, and error messages. You set up the customer-specific validations in Customizing for Bank Analyzer under *Source Data Layer* > *Source Data Services* > *General Functions for Source Data* > *Validations*.

Features

Validations enable you to check your actual primary object data and to run complex checks, such as method-based/cross-object checks.

You assign custom validations in the module for custom validations in Customizing for the relevant primary object. This assignment is done at different levels:

- Validations for primary objects in the Master Data Framework are assigned at the template version level.
- Validations for the primary object *position* are assigned at the position class level.
- Validations for the primary object *business transaction* are assigned at the business transaction class level.
- Validations for the primary object *business partner* are assigned at the business partner level. This enables you to assign a valid procedure.

The assignment can be especially useful when several Analyzers are being used, since they prevent the Analyzers from having a negative impact on one another.

Example

You want to enter an object. You require a field from an Analyzer, but it is empty. In this case, the system should not reject this object, since it can be processed by the other Analyzers. One solution would be to validate this field and issue a warning message that says the object has to be postprocessed for the Analyzer in question.

Note

Note that validation checks can have a negative effect on system performance regardless of whether maintenance is carried out manually or automatically. The time required cannot be estimated because it depends on the number and complexity of the validations.

3.2.2.1.1 Function Modules Shipped by SAP

In the standard SAP system, the following function modules are provided for the validation checks for primary objects:

Function Name (FUNCNAME)	Function Description
/BA1/B0_API_CXS_FPO_GET	Reads accounting data for OTS
/BA1/F1_API_CXS_ACC_GET	Reads account data for OTS
/BA1/F1_API_CXS_BAS_GET	Reads basic data for OTS
/BA1/F1_API_CXS_BPR_GET	Reads partner data for OTS
/BA1/F1_API_CXS_BPR_IT_GET	Reads business partner data for OTS (with ID type)
/BA1/F1_API_CXS_CFH_GET	Reads cash flow data for OTS
/BA1/F1_API_CXS_CHR_GET	Reads data records with characteristic values for OTS
/BA1/F1_API_CXS_CONTRACT_FILL	Provides contract/object data for derivation
/BA1/F1_API_CXS_CONTRACT_GET	Reads contract/object data from the derivation buffer
/BA1/F1_API_CXS_CONTRACT_SET	Puts contract/object data in the derivation buffer
/BA1/F1_API_CXS_CRL_GET	Reads object relationships for OTS
/BA1/F1_API_CXS_CUR_GET	Reads currency data from the derivation buffer for OTS
/BA1/F1_API_CXS_DSS_GET	Reads a data record with information about a worklist
/BA1/F1_API_CXS_HED_GET	Reads basic hedge data for OTS
/BA1/F1_API_CXS_INS_GET	Reads instrument data for OTS
/BA1/F1_API_CXS_KFG_GET	Reads key figures for OTS
/BA1/F1_API_CXS_LIM_GET	Reads general limit data for OTS
/BA1/F1_API_CXS_NODE_NO_GET	Reads contract/object data for derivation
/BA1/F1_API_CXS_OPD_GET	Reads option data for OTS
/BA1/F1_API_CXS_RAT_GET	Reads rating data for OTS
/BA1/F2_XS_BT_COMPLEX_GET	Reads business transaction data (complex key figure)
/BA1/F2_XS_FILL	Initializes the global buffer with read complex key figure

/BA1/F2_XS_HEAD_CHAR_GET	Reads header information (characteristics) of business transaction
/BA1/F2_XS_HEAD_GET	Reads header information (fixed fields) of business transaction
/BA1/F2_XS_POS_CHAR_GET	Reads item information (characteristics) of business transaction
/BA1/F2_XS_POS_GET	Reads item information (fixed fields) of business transaction
/BA1/F2_XS_POS_KFG_AMT_GET	Reads item information (key figure *amounts*) of business transaction
/BA1/F2_XS_POS_KFG_PER_GET	Reads item information (key figure *percentage*) of business transaction
/BA1/F2_XS_POS_KFG_QUA_GET	Reads item information (key figure *quantity*) of business transaction
/BA1/F2_XS_SET	Initializes global buffer with complex key figure
/BA1/F3_API_BPDFV_ID_DEVIDE	Separates the keys *business partner* and *differentiation category*
/BA1/FSBP_GET_COMPLEX_KEY_FIG	Reads the complex key figure of the business partner
/BA1/FSBP_OTS_/BA1/_SGS_GET	Derives business partner data; reads characteristics from the segmentation service for OTS
/BA1/FSBP_OTS_ADRS_GET	Reads address data of the standard address of business partner for OTS
/BA1/FSBP_OTS_BP001_GET	Reads master data of business partner (*Treasury Attribute Organization*) for OTS
/BA1/FSBP_OTS_BP1012_GET	Reads business partner ratings for OTS
/BA1/FSBP_OTS_BP1030_GET	Reads regulatory reporting data of business partner for OTS
/BA1/FSBP_OTS_BPID001_GET	Reads further partner numbers for the business partner for OTS
/BA1/FSBP_OTS_BUT000_GET	Reads header data of business partner for OTS
/BA1/FSBP_OTS_BUT020_GET	Reads addresses of business partner for OTS
/BA1/FSBP_OTS_BUT050_GET	Reads general data on business partner relationships and business partner role definitions
/BA1/FSBP_OTS_BUT0BANK_GET	Reads bank data of business partner for OTS

/BA1/FSBP_OTS_BUT0BK_GET	Reads bank details of business partner for OTS
/BA1/FSBP_OTS_BUT0ID_GET	Reads identification numbers of business partner for OTS
/BA1/FSBP_OTS_BUT0IS_GET	Reads sectors of business partner for OTS
/BA1/FSBP_OTS_BUT100_GET	Reads roles of business partner for OTS
/BA1/R2_API_CXS_CRE_GET	Reads Credit Exposure data for OTS

3.2.2.2 Authorizations

For all primary objects, authorizations are checked for both automatic and manual data maintenance.

For financial transactions, financial instruments, accounts, positions, business transactions, collateral, master agreements, hedge relationships, countries, and business partner data, a complex authorization concept is provided for manual data maintenance. This is defined by the customer and implemented using fields delivered by SAP and the customer's own characteristics.

Authorization Concept: Manual Processing

As part of the manual check, a flexible authorization check is performed at characteristic level. You can define the characteristics relevant for authorization in Customizing.

The authorization check is performed when an object version is displayed or changed during manual processing, immediately after you have entered the object ID and selected the relevant data.

If a user is not authorized to enter the values for the characteristics relevant for authorization, the system terminates manual processing for the object in question and displays an error message. When an object version is created or changed, the system also checks authorizations before data is saved.

The authorization check is performed for each activity carried out.

A cross-object authorization check can be performed to control authorizations at characteristic level. For example, users are to be allowed to display only those financial transactions for which they are named as the processor in the assigned settlement account.

Hierarchies and business partner relationships can also be evaluated as part of the authorization control process. You can also control authorizations in a time-dependent manner by stipulating that objects can be changed only three days before or after the last day of the month, for example.

In addition to the settings for authorizations for characteristic values, you can also make user-dependent field control settings during manual processing to determine for each field whether or not a user is to be authorized to display or change the value of the field in question.

3.2.2.2.1 Authorization Groups

Definition

An authorization group determines whether a user has permission to create, change, or display the primary objects Financial Transaction [page 122], Financial Instrument [page 129], and Account [page 134]. This information is stored in the user's master data. The authorization group cannot be versioned.

Structure

An authorization enables you to perform a particular activity in the SAP system, based on a set of authorization object field values.

The authorization object for editing financial transactions, financial instruments, and accounts is predefined by SAP and cannot be changed. The name assigned to this authorization object is F_BAF1_AUG.

You can create authorizations for this authorization object in Customizing.

When you create an authorization for the authorization object F_BAF1_AUG, you have to enter the appropriate values in the *Authorization Group* and *Activity* fields.

You can define a single value or a value range as values in the Authorization Group field or enter (*) to permit all values.

Users with the authorization group 001 are to be authorized to display only financial transactions, financial instruments, and accounts. The following settings are required for this purpose:

Create an authorization with the following field values:

Authorization group: 001

Activity: Display

Assign authorization to the user's master data.

Use

The authorization group is always checked when you call up the transaction for changing or displaying the primary objects Financial Transaction, Financial Instrument, and Account or when you save these primary objects.

3.2.2.2.2 Characteristic-Based Authorization Check

Use

You can use both a global authorization check and an authorization check that is based on characteristics for the source data. You can use this check (which comprises characteristics of your choice from the characteristic repository) to enable only specific user groups to process or display data. This kind of authorization check can be useful if, for example, you only want your employees to be able to process particular countries or product groups in a live system.

For more information, see Characteristic-Based Authorization Check in Bank Analyzer [page 66].

Prerequisites

To set up the characteristic-based authorization check, define settings in Customizing for Bank Analyzer, and in the roles of users. For more information, see Setting Up the Characteristic-Based Authorization Check [page 68] and Assigning Authorization Characteristic Profiles [page 278].

Features

In Bank Analyzer, characteristic-based checks are carried out as part of the following activities:

- Ad hoc reporting
- Primary data source
- Primary objects (Source Data Layer)

These characteristic-based authorization checks use the characteristics that you define for this purpose in the authorization characteristic profile in Customizing for Bank Analyzer. You also define characteristic values and value ranges in the various user roles.

3.2.2.2.2.1 Assign Profiles for Authorization Characteristics

Use

To restrict the evaluation and display of data to specific user groups, use a characteristic-based authorization check for the primary objects. To use this authorization check, you have to make settings in different parts of Customizing for Bank Analyzer.

Prerequisites

You have defined the appropriate authorization characteristic profiles in Customizing for Bank Analyzer, and you have adapted the user roles accordingly.

Procedure

First of all, create authorization characteristic profiles in Customizing: *Bank Analyzer* > *Basic Settings* > *Settings for Metadata* > *Characteristics* > *Authorizations for Characteristic Values* > *Edit Authorization Characteristic Profiles*.Next, assign the authorization characteristic profiles to the appropriate applications.

Proceed as follows:

- Ad hoc reporting
  In Customizing for Bank Analyzer, choose *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flow: Additional and Alternative Settings* > *Reporting* > *Ad Hoc Reporting* > *Configure Ad Hoc Reports*.
- Primary data source
  In Customizing for Bank Analyzer, choose *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Assign Authorization Characteristic Profiles to Primary Data Sources*.
- Primary objects
  In Customizing for Bank Analyzer, choose *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flow: Additional and Alternative Settings* > *Data Export* > *Export of Cash Flows* > *Assign Profile of Authorization Characteristic to Processing Area.*.

3.2.2.3 Archiving of Primary Objects

Use

The purpose of archiving is to remove data that is no longer required from the database without permanently deleting it. This improves the performance of your database. You still have read access to the archived data.

You can archive the following primary objects:

- Object data
- Position data
- Business transaction data
- Business partners
- Market data

Related Information

Data Archiving and Data Deletion in Bank Analyzer [page 1754]
Archiving and Destroying Position Data [page 1775]
Archiving and Destroying Business Transaction Data [page 1777]
Archiving Market Data [page 1814]

3.2.3 Methods for Source Data

You can use the following methods for the primary objects:

- Plan/Actual Comparison Method [page 280]
- Determination of Original Term Method [page 285]

3.2.3.1 Plan/Actual Comparison Method

Use

You can use this function to compare plan data and actual data for selected financial transactions, and to correct the difference within a Customizable range of tolerance.

> **Note**
>
> You can run the plan/actual comparison for the primary object 'financial transaction' only.

Prerequisites

In Customizing for *Bank Analyzer*, you have edited the Customizing activities under *Source Data Layer* > *Source Data Services* > *Methods for Source Data* > *Plan/Actual Comparison*.

In Customizing, you can enter an absolute amount-based tolerance, for example. If during the comparison the system determines a difference between the planned balance and the actual balance that is less than or equal to the tolerance amount, it does **not** create an adjustment flow.

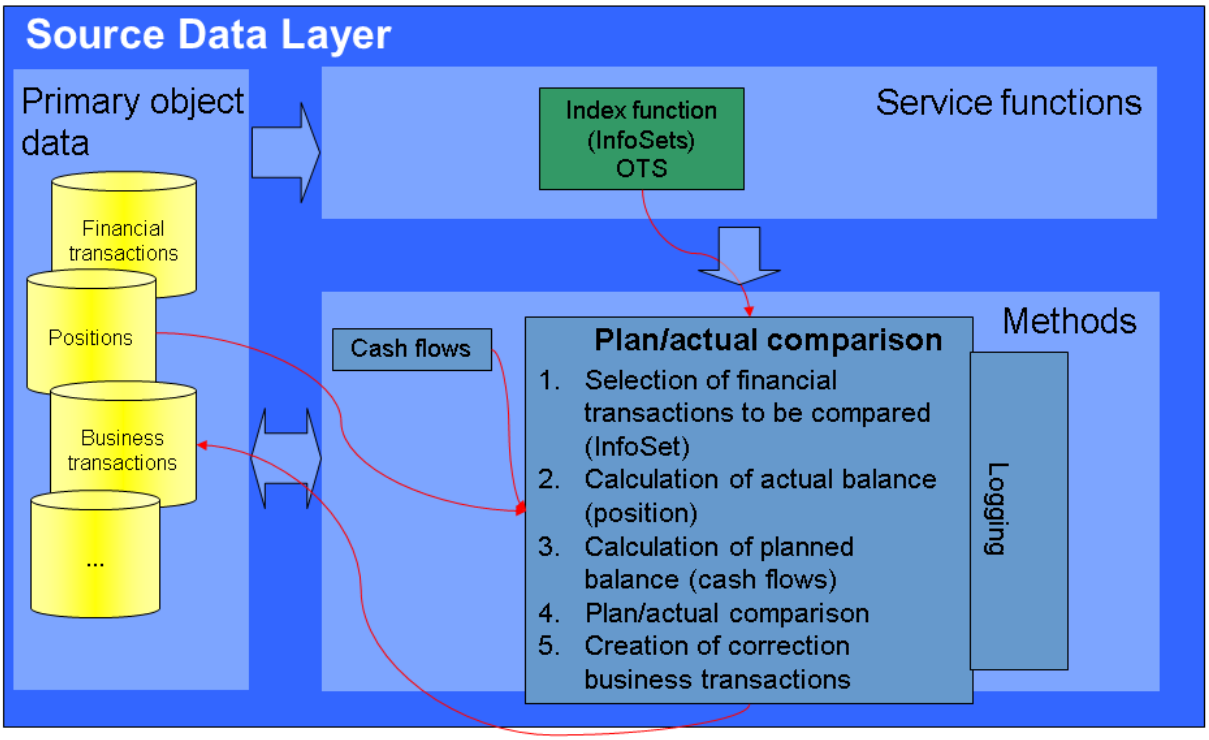
You can use Business Add-In (BAdI) *Exit Planned/Actual Comparison* (`/BA1/BADI_FP_CPA`) to transfer different tolerance amounts for each currency in different source systems.

Features

You carry out the plan/actual comparison for a business key date for a version of the plan data and a specific key figure of the associated actual data. The data is selected in line with the versioning concept. If there is a difference between the plan and actual data, the system balances it by creating a correction business transaction in Source Data Layer.

The procedure for comparing a financial transaction is:

1. Determine the relevant financial transaction via an InfoSet on the basis of its characteristics for the key date of the comparison.
2. Determine the associated position (actual balance) via the financial transaction ID, and read the actual balance for the business date specified for the comparison.
3. Read the existing adjustment flows (business transactions) for the business key date of the plan data. Determine the planned balance from the cash flow data for the version of the financial transaction, and the existing adjustment flows for the key date of the comparison.
4. Determine the difference between the planned and actual balance. Call up a customer enhancement for a custom adjustment of the difference between the planned and current balance.
   You create the customer enhancement by implementing `/BA1/BADI_FP_CPA`.
5. Generate an adjustment flow to correct the difference between the planned and actual balances for the business key date of the planned balance with the business key date of the comparison as the payment day.



3.2.3.1.1 Example: Plan/Actual Comparison Method

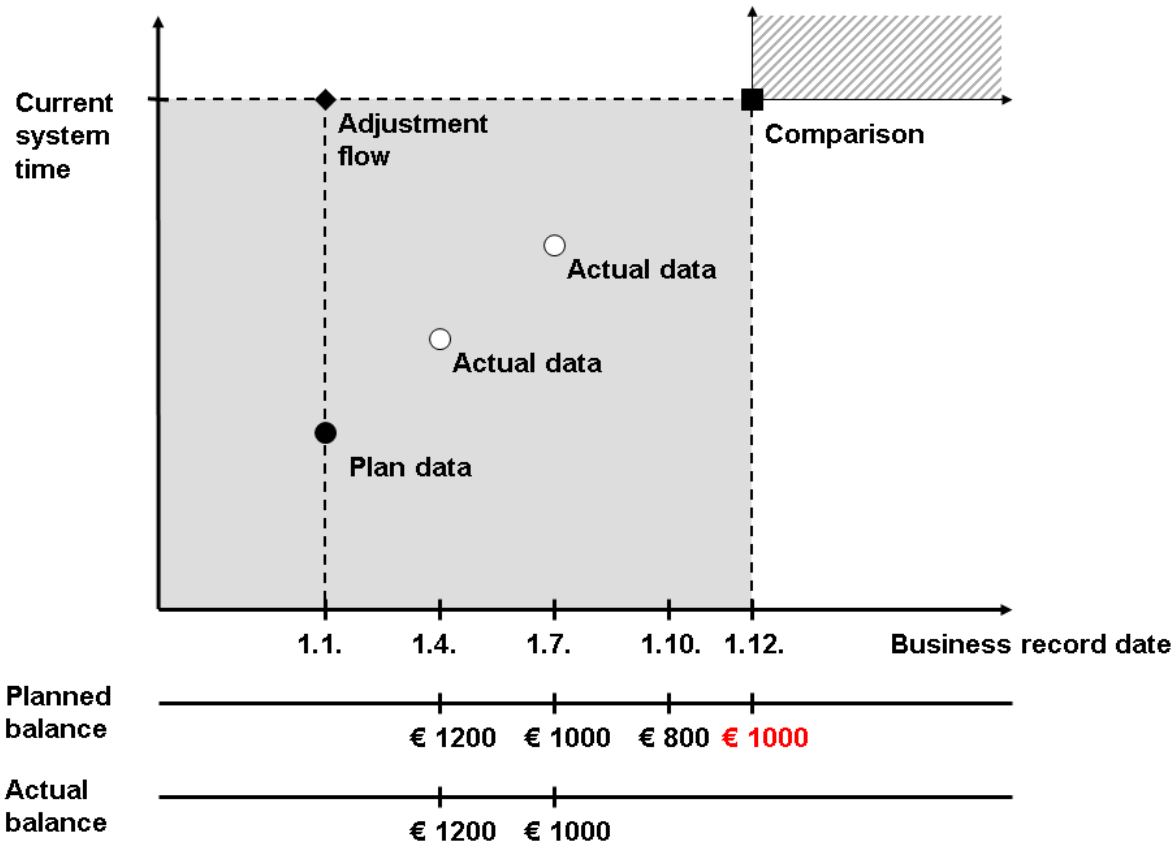
A plan/actual comparison is carried out for a financial transaction on the business record date 12/01/2002 and for the current system date/time. The figure shows the versions of the plan and actual data for this financial transaction that are contained in Source Data Layer (SDL).

The financial transaction was entered in the system on the business record date 01/01/2002. For the period considered here, the financial transaction conditions result in the following cash flow:

Payment Date	Flow	Amount	Currency	Plus/Minus Sign
04/01/2002	Disbursement	1200	EUR	+
07/01/2002	Repayment	200	EUR	-
10/01/2002	Repayment	200	EUR	-

In order to determine the planned balance, all the cash flows for a given date are totaled, taking the plus/minus sign into account. The planned balance depicted below, up to and including 10/01/2002, when the planned balance is €800, can be obtained from the cash flow data for the financial transaction. At the same time, two actual balances were created in the system on 04/01/2002 and on 07/01/2002. An additional change was supposed to be made to the balance on 10/01/2002, but this did not happen.

A plan/actual comparison is now carried out on 12/01/2002. The actual data that was last entered in SDL is read from the comparison date onwards. The version from 07/01/2002 with a current balance of €1,000 is found. A difference of €200 is determined as a result of comparing the two balances. An adjustment flow is then entered for this amount in the form of a business transaction with an appropriate business transaction class. The business transaction is created for the business record date 01/01/2002 and with the payment day 12/01/2002. This means that the planned and current balances now match for the key date of the comparison.



If you now select the financial transaction anywhere in the shaded area, the plan data that the system displays matches the actual data.

If the plan or actual data is subsequently adjusted by importing a new version, the comparison has to be repeated to ensure that the balances match.

3.2.3.1.2 Carrying Out a Plan/Actual Comparison

Prerequisites

- You have made the settings required in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Methods for Source Data* > *Plan/Actual Comparison*.
- You have transferred data to the Source Data Layer.

Context

Once data has been transferred to Source Data Layer (SDL) you can carry out a plan/actual comparison. You use this method for the analyses in the Analyzers.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Methods for Source Data* > *Plan/Actual Comparison*.
2. Enter the following data:
   - *Key date* for which the system is to run the comparison.
   - *InfoSet* that is to be used to search for the relevant financial transactions.
     Depending on the InfoSet you have entered, you can specify the characteristic values the system is to use to get the data for the financial transactions it compares.
   - An *InfoSet profile* is only valid for a specific InfoSet and contains all the characteristics for which you have to enter selection criteria when you make a selection and, therefore, when you create a worklist.
     You can define InfoSet profiles for InfoSets in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Index Function* > *InfoSets* > *Edit Additional Settings for InfoSets*.
   - *Position class* of the actual data that the system uses for the comparison.
   - Position *key figure* that contains the actual balance to be compared.
   - *Cash flow generation type* that the system uses to create the cash flow.
     The system then calculates the planned balance based on the cash flow. The cash flow generation type has to consider the adjustment business transactions for the financial transaction.
   - *Simulation Run* indicator: This indicator is set by default. During the simulation run, the system only calculates and logs the difference between planned data and actual data. It does not create a correction business transaction.
   - *Detail Log* indicator: This indicator is set by default. The system displays all messages in the log.
   - *Package size for parallel processing*: The default value is `1`. If you run the plan/actual comparison as a background job, it is processed as a parallel job, so you have to specify the package size.
3. Choose *Execute* or the `F9` function key. If you press `F9`, you execute the transaction in the background.

   > **Note**
   >
   > If you want to compare a large number of financial transactions, we recommend you start the job as a background job by pressing `F9`. This maintains system performance.

   The system collects the results of the comparison in a log. After the system has completed the comparison, you can display the messages for each financial transaction.

3.2.3.1.3 Displaying Logs for the Plan/Actual Comparison

Prerequisites

You have carried out a plan/actual comparison. You start this transaction on the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Methods for Source Data* > *Plan/Actual Comparison*.

Context

You use this transaction to display logs for the plan/actual comparison. You can enter dates and the user name as selection criteria.

Procedure

1. To display the logs, on the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Methods for Source Data* > *Display Logs for Plan/Actual Comparison*.
2. Enter a time period or a user.
3. Choose *Execute*.

   The required logs are displayed in a list.

3.2.3.2 Determination of Original Term Method

Use

You use this function along with the contract start and end dates to calculate in days the original term of the financial transaction or financial instrument. Methods for calculating interest and working day regulations are not taken into consideration here.

Integration

The 'determination of original term' method can be used across all Bank Analyzer components.

Prerequisites

- You have defined one or more key figure categories of the classification *Quantity* in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Master Data* > *Settings for Attributes* > *Key Figures* > *Edit Key Figure Categories*.
- You have activated the attributes of the template required in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Master Data* > *Templates* > *Edit Template Version*.

Activities

You can use function module `/BA1/FP_OBJECT_TERM_DETERMINE` to calculate the original term. Alternatively, you can use a key figure value from a source system. Select a key figure of the category *Quantity* that has the unit *Days* for the respective financial transaction or financial instrument.

You assign a key figure category for the 'determination of original term' method at template level. You assign key figure categories of key figure class "quantity" to templates of template category "financial transaction" and "financial instrument" in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Methods for Source Data* > *Determination of Original Term* > *Assign Key Figure Categories for the Determination of the Original Term*.

Example

Calculation of the Original Term

Contract Start Date	Contract End Date	Original Term
01/01/2005	01/01/2005	0
01/01/2005	01/02/2005	1
02/20/2005	03/05/2005	14
01/01/2005	01/15/2006	379

3.2.4 Source Data Aggregation

Use

This function generates aggregated objects from single objects in the Source Data Layer (SDL) and in the Results Data Layer (RDL) that together form an abstract aggregation object. This reduces the volume of input data for subsequent processes, such as accounting. This reduction in input data improves system performance.

Source data aggregation supports the accounting scenarios of Accounting for Financial Instruments (AFI) as a subledger scenario. You can use these to summarize mass products whose valuation in accounting does not require a cash flow (such as checking account, saving accounts, or credit card accounts).

Integration

You execute the source data aggregation processes before the AFI processes.

Prerequisites

You have edited the source data aggregation settings in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Source Data Aggregation*.

> **Note**
>
> Source data aggregation assumes that the business transactions and accrual results for a financial transaction come from the same system as the financial transaction. The system uses the source system as the granularity characteristic and as part of the granularity of the process. Make sure that the flow data comes from the same source system as the master data. If your data is delivered by services, you can change the source system entered in Bank Analyzer. You make the required settings in Customizing for *Bank Analyzer* under *Source Data Layer* > *Interfaces to Other Applications* > *Interfaces to Account Management*.

Features

You can aggregate financial transactions, business transactions, accrual results and expected loss amounts. When you do so, the system assigns a single object to an abstract aggregation object:

Single Object	Abstract Aggregation Object
Financial transaction	Aggregation Object
Business transaction	Aggregation business transaction
Accrual result	Aggregation accrual result
Expected loss (EL)	Aggregation expected loss

The following functions are also available:

- Error Correction
  - Reverse Aggregation Business Transactions [page 324]
  - Reverse Aggregation Runs [page 325]

- Statistics [page 328]
- Archiving [page 328]

More Information

- Metadata for Source Data Aggregation [page 290]
- Processes for Source Data Aggregation [page 298]
- Example: Source Data Aggregation [page 288]
- For information about using source data aggregation in Accounting, see Use of Source Data Aggregation [page 962].
- For information about source data aggregation when you implement the functions for impairment at a later point in time, see Initialization of Impairment [page 973].

3.2.4.1 Example: Source Data Aggregation

In 2008, a German car manufacturer sets up an Internet bank. On January 6, 2009 George Baker opens his first account at this bank. His salary is paid into the account on January 30, 2009. In February, George Baker moves to Sydney, Australia. He informs his bank of his new address on February 5, 2009. Initially, the currency of the account does not change because the salary is still paid by the German company.

To represent this information, the bank uses the following granularity characteristics:

- Living at home (Germany) or living abroad (other countries)
- Product, in this case checking account (CA) and balance
- Currency
- Asset (A) or liability (L)

After January 6, 2009 the system contains the following business partner:

Business Record Date	Business Partner	Country
2009–01–06	George Baker	Living at home (Germany)
2009–02–05	George Baker	Living abroad

The following financial transactions are assigned to the business partner:

Business Record Date	Financial Transaction	Product	Currency	Business Partner
2009–01–06	CA_GEORGE_BAKER	CA	EUR	GEORGE_BAKER

On January 30, 2009 the first salary is paid into the account. The following business transaction is recorded in the system:

Business Date	Business Transaction	Financial Transaction	Amount
2009–01–30	BT1:	CA_GEORGE_BAKER	EUR -3000

The system contains the following position for this financial transaction:

Business Record Date	Financial Transaction	Amount	Asset/Liability Item
2009–01–30	CA_GEORGE_BAKER	EUR -3000	L

Since the system only aggregates financial transactions that contain a position, it generates the first aggregation object.

Business Record Date	Financial Transaction	Aggregation Object	Product	Currency	Country	Asset/Liability Item
2009–01–30	CA_GEORGE_BAKER	AO1	CA	EUR	Living at home (Germany)	L

Business transactions with financial transactions that contan the same granularity characteristics are aggregated in an aggregation business transaction and assigned to the aggregation object `AO1`.

Business Record Date	Aggregation business transaction	Aggregation object	Amount
2009–01–30	ABT1	AO1	EUR -3000

On February 5, 2009, George Baker moves to Sydney. His business partner data is changed accordingly.

Business Record Date	Business Partner	Country
2009–01–06	GEORGE_BAKER	Living at home (Germany)
2009–02–05	GEORGE_BAKER	Living abroad

The *country* granularity characteristic changes. The system assigns his account to a new aggregation object:

Business Record Date	Financial transaction	Aggregation object	Product	Currency	Country	Asset/liability item
2009–01–30	CA_GEORGE_BAKER	AO1	CA	EUR	Living at home (Germany)	L
2009–02–05	CA_GEORGE_BAKER	AO2	CA	EUR	Living abroad	L

Since the position amount has to be credited to the checking account that has granularity characteristic *living abroad*, the system creates two reassignment transactions. One reassignment business transaction deducts the acquisition value of the checking account from the old aggregation object (`AO1`). The second one posts the acquisition value of the checking account to a new aggregation object (`AO2`):

Business Record Date	Aggregation business transaction	Aggregation object	Amount
2009–01–30	ABT1	AO1	EUR -3000
2009–02–05	NBT1	AO1	EUR 3000
2009–02–05	NBT2	AO2	EUR -3000

3.2.4.2 Metadata for Source Data Aggregation

Definition

Source data aggregation is based on a series of Customizing entities that you can use to control the aggregation process. The possible settings, such as possible granularity characteristics, depend on the SAP system entities to which you have assigned the Customizing entity:

Customizing Entity	System Entity
Aggregation view [page 291]	Aggregation view category (such as *DEFAULT*)
Subcomponent view [page 292]	Subcomponent view category (such as *financial transactions*)
Aggregation type [page 297]	Aggregation category (such as *accrual results*)

You make the settings for the metadata for source data aggregation in the following Customizing activities under *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* :

- *Edit Subcomponent Views*
- *Edit Aggregation Views*

Use

Using this metadata, source data aggregation generates the following abstract aggregation objects:

Abstract Aggregation Object	System Activity
Aggregation object	The system assigns all financial transactions with the same combination of characteristic values to the same aggregation object provided you have not excluded the financial transaction from aggregation in exception rules. **Example of an exception rule:** Checking accounts with a position over EUR 100,000 must be evaluated individually in the accounting system and are excluded from aggregation.
Aggregation business transaction	The system aggregates all business transactions whose related financial transactions are assigned to the same aggregation object and assigns them to an aggregation business transaction.
Aggregation accrual result	The system aggregates all accrual results whose related financial transactions are assigned to the same aggregation object and assigns them to an aggregation accrual result.
Aggregation expected loss	The system aggregates all expected loss amounts whose related financial transactions are assigned to the same aggregation object and assigns them to an aggregation expected loss.

After source data has been aggregated for the first time, the system differentiates between the following object categories at **single transaction level**:

- Non-aggregated objects, that is, individual objects that source data aggregation has *not* assigned to an abstract aggregation object
- Aggregated objects, that is, individual objects that have been assigned to an abstract aggregation object

3.2.4.2.1 Aggregation View

Definition

Central Customizing entity for source data aggregation. It groups the settings for controlling the aggregation of financial transactions, business transactions, or accrual results. The aggregation view provides you with a consistent overview of the business data in the Source Data Layer (SDL) and the Results Data Layer (RDL), that is to say, the non-aggregated objects and abstract aggregation objects:

When you create an aggregation view, you assign it to an aggregation view category provided by SAP.

You use the aggregation view to activate or deactivate the enhancement of the source data aggregation. For more information, see Processes for Source Data Aggregation [page 298].

Structure

Assigned Subcomponent Views

On the basis of the subcomponent views [page 292] assigned to it, the aggregation view defines which objects and which instances for the objects the system aggregates, how it performs source data aggregation, and where it saves the results.

Reassignment Information

Reassignment information controls the adjustments required when a financial transaction is reassigned. If the system assigns an already aggregated financial transaction to a different aggregation object or displays it as a single transaction again, it generates reassignment business transactions. These reassignment business transactions transfer the position components of the financial transaction that are relevant for accounting processes at the time of reassignment.

Non-Relevant Source Systems

Source systems that do not contain any objects to be aggregated can be ignored during source data aggregation.

Integration

You assign at least one subcomponent view to an aggregation view in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Edit Aggregation Views*.

If you aggregate source data, you have to assign only one aggregation view to each results data area in the Results Data Layer. If you do not assign an aggregation view to a results data area, you work automatically with non-aggregated data.

Every accounting system is assigned to one results data area only. The accounting system obtains the results data area from the accounting system and the aggregation view from this. This indicates which aggregation view is used by the accounting processes. You can assign the same aggregation view to different results data areas and, therefore, accounting systems.

> **Note**
>
> You can edit an aggregation view only after it has been *deactivated*.
>
> You can use an aggregation view only after it has been *activated*.

3.2.4.2.2 Subcomponent View

Definition

Customizing entity for source data aggregation. It defines the procedure and the criteria that the system uses to aggregate objects in a Bank Analyzer subcomponent. When you create a subcomponent view, you assign it to a

subcomponent view category provided by SAP, which controls the possible Customizing settings for the subcomponent view:

	Subcomponent View Category: Financial Transactions	Subcomponent View Category: Business Transactions	Subcomponent View Category: Accrual Results	Subcomponent View Category Expected Loss Amount
Aggregation source and target	Yes	Yes	Yes	Yes
Granularity of the aggregation	Yes	Yes	Yes	Yes (predefined by the system)
Mapping granularity characteristics	Yes	Yes	No	No
Deriving the aggregation type	Yes	No	No	No
Selection criteria	Yes	No	Yes	Yes

Structure

Subcomponent View Category: Financial Transactions

- Aggregation source and target
  You enter the position class from which you want the system to read the positions for the financial transactions.
  You can also view the automatically generated primary data sources for aggregation objects and granularity characteristics once you have activated the subcomponent view:
  - The primary data source for aggregation objects is used as the interface for single record access.
  - The primary data source for granularity characteristics is used as the interface for worklist generation. The system reads the granularity characteristics from the aggregation index tables.

  The system saves the aggregation results in an aggregation object whose template and template version depend on the derived aggregation type.
- Granularity of the aggregation
  You can define granularity characteristics at the level of the subcomponent view and the aggregation type. The system looks at a financial transaction's granularity characteristics (such as object currency, domestic/foreign, legal entity) and assigns it to the aggregation object with the same combination of characteristics.

  > **Note**
  >
  > Source data aggregation does not support changes to the settlement account in the financial transaction. The settlement account in the business transaction must match the settlement account in the related financial transaction. It is possible to use the same settlement account for all financial transactions.

- Mapping granularity characteristics

By mapping a granularity characteristic, you define the source from which the system reads the value of this characteristic. The values of the granularity characteristics for financial transactions can be derived from the financial transaction itself or from the position or business partner assigned to the financial transaction.
The mapping of granularity characteristics is partly predefined by the system using mapping categories. You can also define custom mappings that apply to the current subcomponent view only or to a subcomponent view category.

- Deriving the aggregation type
  You can assign one or several aggregation types to a subcomponent view. The system derives the aggregation type on the basis of the values of the financial transaction's granularity characteristics.
  For more information, see Aggregation Type [page 297].
- Selection criteria
  Some processes in source data aggregation create your worklist from financial transactions in the Source Data Layer (SDL). You use the selection criteria for production control and sales product to determine which products are to be aggregated (checking accounts and savings accounts, for example). The worklist of financial transactions is limited to these products.
  If you aggregate source data, you can only execute product changes that no longer include a financial transaction (the transaction is evaluated individually). The system intercepts product changes that would result in a subsequent aggregation (such as the conversion of a time deposit into a savings account); this means that it continues to evaluate the transactions at single transaction level.

  > **Note**
  >
  > You can still exclude financial transactions for products that are to be aggregated from aggregation at aggregation type level using the following exception rule: If the system does not find an aggregation type during the derivation of the aggregation type from the granularity characteristics of the financial transaction, it processes the individual financial transaction.
  >
  > For more information, see Aggregation Type [page 297].

Subcomponent View Category: Business Transactions

- Aggregation source and target
  The business transactions from the source subcomponent are included in source data aggregation. The source subcomponent is a non-aggregated business transaction class. If you would like to aggregate business transactions from more than one business transaction class, you have to create a separate subcomponent view for each business transaction class.
  The target subcomponent is an aggregation business transaction class in which the system saves the results of the source data aggregation.

  > **Note**
  >
  > When the system aggregates business transactions, it includes only those business transaction classes for which you have created subcomponent views. System performance is optimized if only the business transactions for relevant financial transactions are processed. Therefore, you have to separate business transactions into relevant and nonrelevant financial transactions using different business transaction classes. You also have to ensure that you create only subcomponent views for the business transaction classes in relevant financial transactions.
- Granularity of the aggregation
  You can define granularity characteristics at the level of the subcomponent view and the aggregation type. The system looks at a business transaction's granularity characteristics and assigns it to the aggregation business transaction with the same combination of characteristics.

- Mapping granularity characteristics
  By mapping a granularity characteristic, you define the source from which the system reads the value of this characteristic. The values of granularity characteristics for business transactions can be read from a characteristic field or a fixed field of the business transaction.
  The mapping of granularity characteristics is partly predefined by the system using mapping categories. You can also define custom mappings that apply to the current subcomponent view only or to a subcomponent view category.

Subcomponent View Category: Accrual Results

- Aggregation source and target
  You define the source results data area and the source result type in the Results Data Layer from which you want the system to read the accrual results. You enter the aggregation target in the aggregation type (see Aggregation Type [page 297]).
- Granularity of the aggregation
  You can select granularity characteristics at the level of the subcomponent view and the aggregation type. The system looks at the accrual results' granularity characteristics and assigns them to the aggregation accrual result with the same combination of characteristics.

  > **Note**
  >
  > You can define additional granularity characteristics for accrual results only at aggregation type level because the characteristics depend on the target result type. You define the target result type for each aggregation type.

- Mapping granularity characteristics
  The mapping of result type characteristics to granularity charactersitics is not required for the subcomponent view category *accrual results*. It is assumed that the values of the granularity characteristics are determined from the same fixed characteristics as those in the source result type. The system maps the characteristics automatically based on the identical names.
- Selection criteria
  Some processes for the aggregation of accrual results create your worklist based on the Results Data Layer (RDL). You can use selection criteria for production control, sales product, accrual environment and accrual item type to limit the worklist to accrual results for products that are aggregated. This reduces the number of objects that are selected and so reduces the runtime of this process.
  The prerequisite for this is that the corresponding fields in the accrual result are filled correctly. If you use production control and the sales product for selection criteria, for example, the fields must be filled during the import of accrual results by Enterprise Services by implementing the BAdI *RDL Inbound After Mapping of PI Structure to RDL Data Model*. The selection criterion must match the selection criterion for the subcomponent view for financial transactions.
  If you do not specify any selection critieria, the processes select all of the accrual results but only consider the results for products that are aggregated.

Subcomponent View Category Expected Loss Amount

- Aggregation source and target
  You define the source results data area and the source result type in the Results Data Layer from which you want the system to read the expected loss amounts to be aggregated. You enter the target for the aggregation in the aggreagtion type (see aggregation type [page 297]). For expected loss amounts the source and target results data area and the source and target results type have to be identical.
- Granularity of the aggregation

You can select granularity characteristics at the level of the subcomponent view and the aggregation type. The system looks at the expected loss amounts' granularity characteristics and assigns them to the aggregation expected loss with the same combination of characteristic values.

> **Note**
>
> For expected losses, all granularity characteristics are defined in the system. You cannot define any additional granularity characteristics in Customizing because they are not needed in accounting.

- Mapping granularity characteristics
  The mapping of result type characteristics to granularity charactersitics is not required for the subcomponent view category *expected loss amount*. It is assumed that the values of the granularity characteristics are determined from the same fixed characteristics as those in the source result type. The system maps the characteristics automatically based on the identical names.
- Selection criteria
  Some processes for the aggregation of expected loss amounts create your worklist based on the RDL. You can use selection criteria for production control and sales product to limit the worklist to results for products that are aggregated. This reduces the number of objects that are selected and so reduces the runtime of this process.
  The prerequisite for this is that the corresponding fields in the result are filled correctly. The *Calculate Percentage Expected Loss* process fills these fields. The selection criterion must match the selection criterion for the subcomponent view for financial transactions.
  If you do not specify any selection critieria, the processes select all of the results but only consider the results for products that are aggregated.

Integration

- You assign one or more subcomponent views to an aggregation view in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Edit Aggregation Views*. However, you can only assign one subcomponent view and one aggregation view to each other for each combination of component and subcomponent within Bank Analyzer:
  - You assign an aggregation view with just one subcomponent view in the *Financial Transactions* subcomponent view category and one in the *Accrual Results* subcomponent view category.
  - You assign an aggregation view with one subcomponent view in the *Business Transactions* subcomponent view category for each business transaction class to be aggregated.
  - You assign a subcomponent view for the subcomponent view catetgory *Expected Loss Amount* to an aggregation view for every results data area containing expected loss amounts that are to be aggregated. Several results data areas are used if you conduct accounting according to different accounting standards using several accounting systems.
- You can define the mapping of granularity characteristics at subcomponent view category level in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Mapping Settings for Subcomponent View* > *Edit Global Mapping*.
- You can enter your own derivations for certain mapping types in the *Financial Transactions* subcomponent view category in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Source Data Integration* > *Mapping Settings for Subcomponent View* > *Module Editor*.

- You can define derivations for use in the *Edit Subcomponent Views* Customizing activity in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Derivations for Subcomponent View* >.

> **Note**
>
> You can edit a subcomponent view only after it has been *deactivated*.
>
> You can use a subcomponent view only after it has been *activated*.

3.2.4.2.3 Aggregation Type

Definition

Customizing entity that controls the procedure for source data aggregation. Every aggregated object is uniquely assigned to an aggregation type. The aggregation type defines the granularity and the target of the aggregation. You assign every aggregation type to an aggregation category and save specific information depending on the subcomponent view category.

Structure

Subcomponent View Category: Financial Transactions

In the case of financial transactions, you specify a template, template version, and a template type under which the system stores the aggregation object. The system assumes that the characteristics of the template type are identical to the granularity characteristics.

You can create one or more aggregation types for each subcomponent view in the *financial transactions* subcomponent view category. The system derives the aggregation type for a financial transaction on the basis of the values of the granularity characteristics.

> **Note**
>
> If you have only assigned one aggregation type to a subcomponent view, you do not have to save a derivation of the aggregation type.

To exclude a financial transaction from source data aggregation after deriving the aggregation type, save exception rules in the aggregation type. To do so, define derivation rules for the aggregation type. When the system derives the value *initial* from the granularity characteristics, the financial transaction remains single.

Subcomponent View Category: Business Transactions

In the case of business transactions, you assign an aggregation type to every possible item class in the source business transaction class. For each aggregation type, you specify a non-aggregated source item class and an aggregated target item class. The system applies the aggregation type of the corresponding item class to a business transaction. You also specify the key figures to be aggregated and the function that the system uses to aggregate these key figures.

Subcomponent View Category: Accrual Results

In the case of accrual results, you assign one aggregation type to each subcomponent view. The system saves the aggregation accrual results in the target results data area and in the target result type of the Results Data Layer that you save in the aggregation type. You also specify the key figures to be aggregated and the function that the system uses to aggregate these key figures.

Subcomponent View Category Expected Loss Amount

In the case of expected loss amounts, you assign one aggregation type to each subcomponent view. The system saves the aggregation expected loss amounts in the target results data area and in the target result type of the Results Data Layer that you save in the aggregation type. You also specify the key figures to be aggregated and the function that the system uses to aggregate these key figures.

3.2.4.3 Processes for Source Data Aggregation

The following processes enable you to use the functions for source data aggregation:

- Processes for Source Data Aggregation without Enhancement [page 298]
- Processes for Source Data Aggregation with Enhancement [page 310]

In the Customizing for the aggregation view you use the *Activate Enhancements to Source Data Aggregation* field to determine which process chain you want to use. For more information, see the field documentation.

The activation of the enhancements is required in the following cases:

- You are implementing source data aggregation for the first time.
- You are already using the source data aggregation without enhancement and would also like to implement the impairment solution from Bank Analyzer. For more information, see Initialization of Impairment [page 973].
- The activation of the enhancements is mandatory for new customers.

The enhancements enable you to also aggregate expected loss amounts, which are stored at individual transaction level in the Results Data Layer (RDL), and derive granularity characteristics from operational events.

3.2.4.3.1 Processes for Source Data Aggregation without Enhancement

Use

The processes for source data aggregation (SDA) compress data for further processing. For more information about the available process chains and their activation, see Processes for Source Data Aggregation [page 298].

Prerequisites

- You have imported the data from your source system to Bank Analyzer. Changes to the granularity characteristics for source data aggregation have been updated successfully in the aggregation indices for the financial transaction, the business partner and the operative events.
- You then created a source data time stamp for a process partition. The processes for source data aggregation for the same process partition read data changes or new data from a time stamp interval of which the upper limit is the time stamp created. For more information, see Set the SDL Time Stamp [page 367].

Process

During source data aggregation a series of mandatory and optional processes need to be carried out in sequence. The system checks whether a previous mandatory processes has been completed successfully when the next process is started.

> **Note**
>
> You can deactivate this check for test purposes in systems that are not live by selecting the *Suppress Check of Process Sequence* checkbox on the selection screen for the process. Note, however, that this can lead to inconsistencies and incorrect results.

The processes for source data aggregation differentiate between the following process errors:

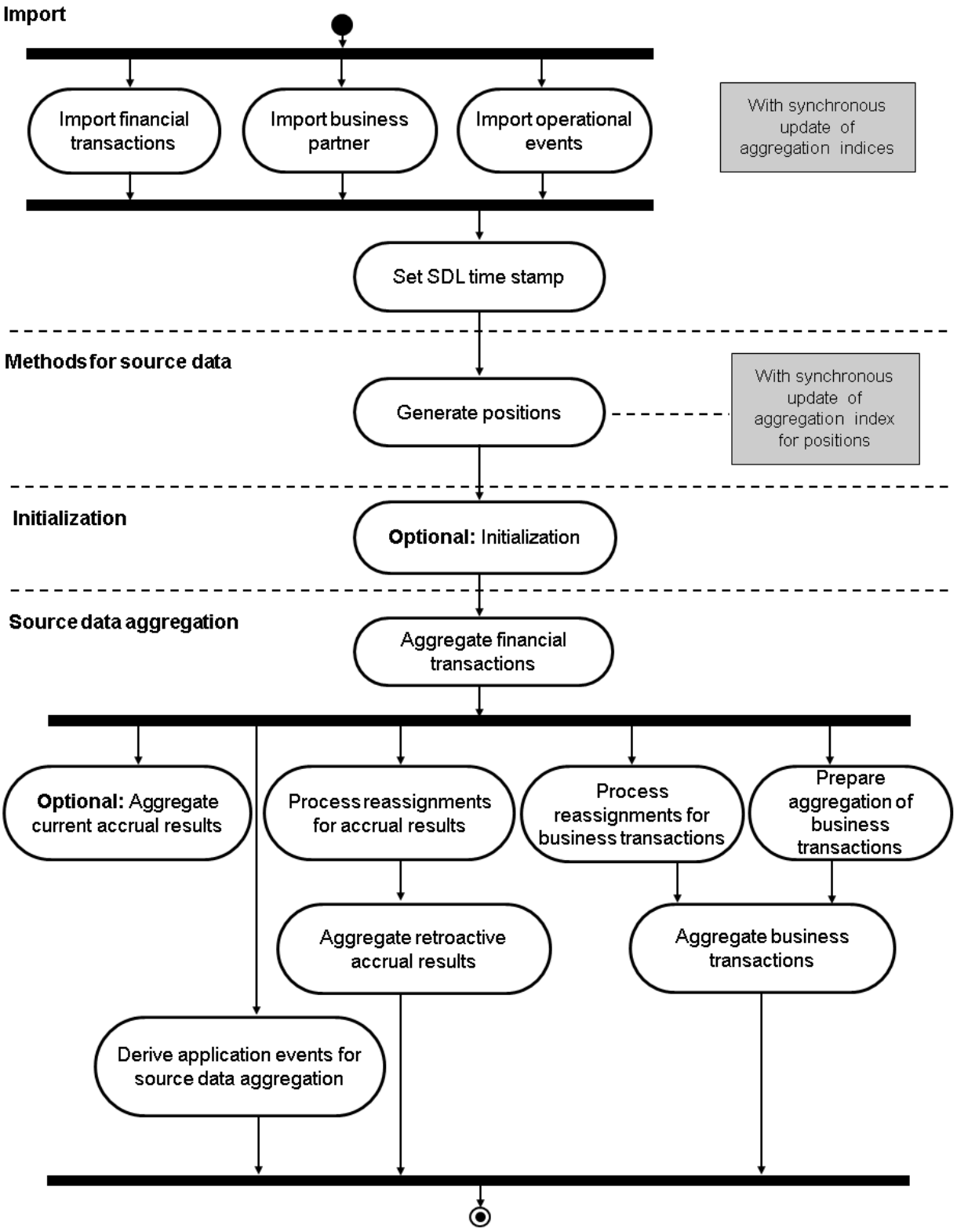
Basic errors

Basic errors can be resolved subsequently and do not interrupt the process chain. In the case of basic errors, the run status in the monitor for the Calculation and Valuation Process Manager (CVPM) is *Successful* (green traffic light). The errors are listed in the application log. A basic error can occur, for example, if the currency of an imported business transaction does not match the currency of the position. In this case, reverse the business transaction and import the transaction with the correct currency.

Serious errors

Serious errors must be resolved immediately, before the process chain can continue. The run status of processes that have ended with serious errors is not displayed as *Successful* in the CVPM process monitor (yellow or red traffic light). In these cases, system checks prevent the subsequent processes from being executed. The errors are displayed in the application log for the run. You must remove the cause of the error before you are restart the process. If you then start the process again with the same parameter values, the process selects the data packages that were not processed and continues with the processing. These errors can be the termination of one or more processes in parallel processing or the resetting of incorrect data packages by the application, for example.

The figure below shows the sequence and dependencies of the process:



The parameters *Aggregation View*, *Legal Entity*, *Source System* and *Process Partition* are mandatory for all process runs for source data aggregation. You can execute the same process in parallel for different legal entities and source systems.

Import

The processes for source data aggregation select the relevant new and changed data in the Source Data Layer (SDL) with an entry time stamp that lies within a time stamp interval. The upper limit of the interval is the selection time stamp for source data from the last successful run of the *Set SDL Time Stamp* process for the same process partition. The lower limit is calculated from the last successful run of the corresponding processes for source data aggregation with the same values for the aggregation view, legal entity, and source system.

If you do **not** select the *Time Stamp Calculation* checkbox, you can enter the lower limit of the interval manually. This can be necessary if you are implementing the source data aggregation in a live system at later stage. If the lower limit is 0 during the first position generation, for example, when the position balances are created by initialization business transactions, only these transactions are considered and not the existing business transactions from the past.

> **Note**
>
> Import accrual results in the RDL before you run the *Set SDL Time Stamp* process. This ensures that the data is processed correctly in the source data aggregation.

You need to set a SDL time stamp every time the process chain is started after a part of the data has been imported. The process partition selected identifies this part of the data. Every combination of legal entity and source system usually corresponds to a separate process partition.

Methods for Source Data

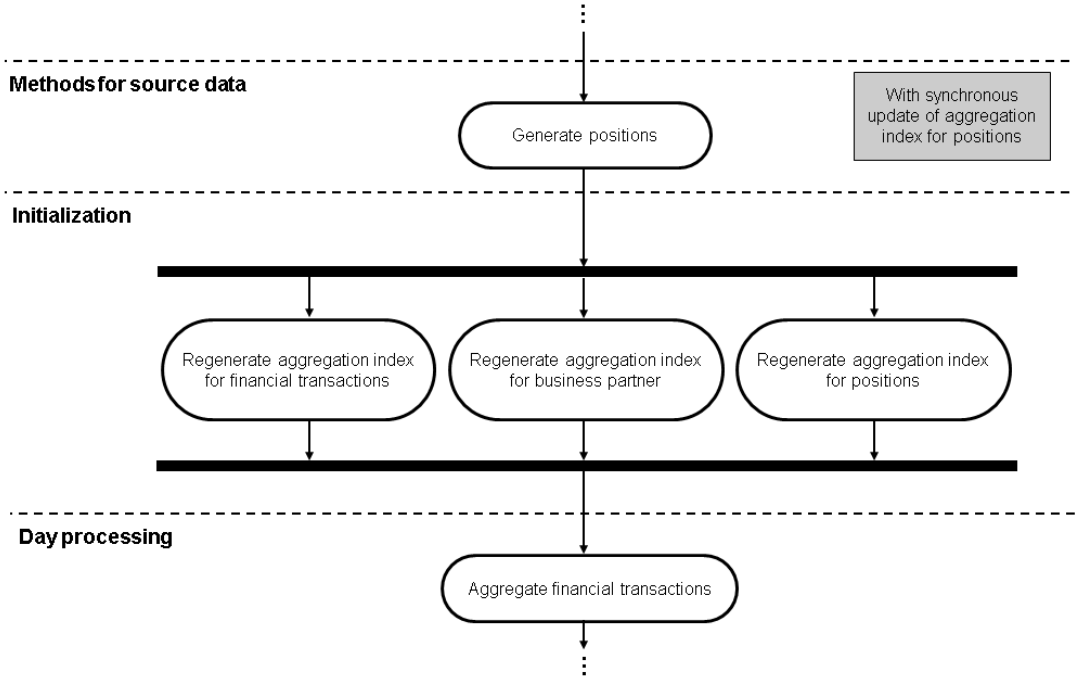
You generate or update positions for the position class specified from new imported business transactions. For more information, see Generate Positions [page 304].

Initialization

In the following cases you need to regenerate aggregation indices for granularity characteristics for aggregation:

- Later implemenation of source data aggregation in a live system
- Implementation of additional granularity characteristics or changes to derivation of exisiting granularity characteristics in Customizing

The following figure shows the sequence and dependencies of the initialization processes:



For more information about the initialization processes, see Regenerating Aggregation Indices [page 303].

Aggregation

1. Aggregate the financial transactions [page 306]
2. Aggregate the business transactions [page 307]
   - Prepare for the aggregation of business transactions
   - Process the reassignments for business transactions
   - Aggregate the business transactions
3. Aggregate the accrual results [page 308]
   - Aggregate the current accrual results (optional)
   - Process the reassignments for accrual results
   - Aggregate retroactive accrual results
4. Derive application events [page 310]

Error Correction

You can reverse individual aggregation business transactions or all aggregation business transactions that have been generated within a run. For more information, see Reverse Aggregation Business Transactions [page 324] and Reverse Aggregation Runs [page 325].

More Information

Calculation and Valuation Process Manager (CVPM) [page 1622]

3.2.4.3.1.1 Regenerating Aggregation Indices

Use

Aggregation indices are updated when financial transactions, business partners and operational events are imported and when positions are generated. If you implement source data aggregation at a subsequent stage, or change settings in Customizing, these indices have to be regenerated.

Process

On the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Initialization* and start the following processes:

- *Regenerate Aggregation Index for Financial Transactions*
- *Regenerate Aggregation Index for Business Partners*
- *Regenerate Aggregation Index for Positions*

These processes determine the granularity characteristics from data in the Source Data Layer (SDL). However, an external import of operational events is required to regenerate the aggregation index for operational events.

> **Note**
>
> If you are already using an impairment solution in your operational system and implement the source data aggregation at a later stage, import the late payment events again in order to regenerate the aggregation index for operational events. You also need to include all of the granularity characteristics (such as *Delayed/Late Payment*, *Business Partner Rating*) that are required for the scenario used in the aggregation.
>
> You can create a new type of operational event for initialization events for the regeneration of the index table, from which no application events should be derived for impairment processes and accounting, in the Customizing activity *Edit Operational Event Type*. You also make settings in the Customizing activity *Derive Application Events from Operational Events*. This type of operational event is only used to generate the aggregation index.

Result

At the end of these processes the system has updated the information it needs to generate worklists for the aggregation of financial transactions.

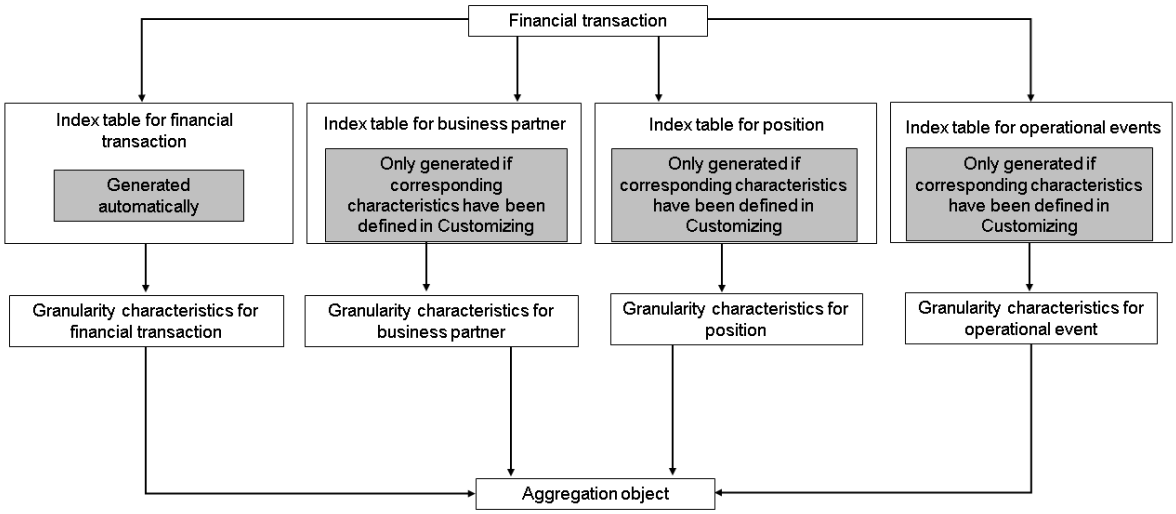
Reversal

If a business transaction that maps an operational event is reversed, the system creates a correction entry in the index table. The reversed entry is not deleted.

Background

In each aggregation view (see Aggregation Views [page 291]), which financial transactions are assigned to which aggregation objects depends on the defining characteristics in Customizing. The granularity characteristics of a

financial transaction depend on the financial transactions, business partners, positions, and operational events that belong to it. There is an index table for each aggregation view for these types of granularity characteristic. This table is created by the system when the aggregation view that has the aggregation type *Financial Transaction* is activated. The content of the index tables is used to create the worklists for the aggregation of financial transactions. The situation described above is illustrated in the following figure:



3.2.4.3.1.2 Generating Positions

Use

This process generates and updates positions for the position class specified from new imported business transactions in the Source Data Layer (SDL).

Prerequisites

- You have entered rules to define how positions are generated in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Methods for Sources Data* > *Position Generation*.
- You have imported the business transactions from the relevant source systems to Bank Analyzer and have set a SDL time stamp for the process partition. For more information, see Set the SDL Time Stamp [page 367].

Process

1. To start the process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Methods for Source Data* > *Generate Positions*. You restrict the run according to position class, legal entity, source system, and process partition.
2. The process selects all of the newly imported business transactions for the relevant business transaction classes for the process partition selected.
3. The system calculates the position by mapping the key fields and defining characteristics to the fields in the business transaction. The position date of a business transaction is a date field in the business transaction header. This date field is defined in Customizing for the position class. A business transaction for this date generates a new position version that has the same validity date.
   In Customizing, you define which key figures for the item classes contribute to which key figures in the position classes. By assigning item types to position key figures you define whether a business transaction item is included in the calculation of changes to the position key figures, and if so, whether it is included as a negative or positive value.

> **Note**
>
> This process does not fill the descriptive characteristics of positions.

This process has two particular features:

- A reversal business transaction is reversed by value. This means that the signs of the key figures in the original business transaction are reversed and added to the position.
- If there are retroactive business transactions, the system updates all subsequent positions.
- If there are business transactions that are in the future, the system creates positions for future dates.
- The business transactions are processed in packages. During parallel processing, the packages are distributed over the processes. Lock errors can occur if business transactions for the same position are processed in different packages. To process the business transaction, the system requests several times that the affected position is locked. If the lock cannot be set, the system saves the business transaction temporarily. In this case, the package is still regarded as having been processed without errors. The system processes the lock errors stored temporarily in the last sequential step of the process.

Result

The system creates new positions for all the items in the selected business transactions, and updates the existing positions.

Error Handling

You find information about the position generation runs for the `/BA1/RF2_BUILD_POSITIONS` process on the *SAP Easy Access* screen under *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. The process *Generate Positions* differentiates between basic and serious errors. For more information, see Processes for Source Data Aggregation without Enhancement [page 298] or Processes for Source Data Aggregation with Enhancement [page 310].

If the process terminates before all packages have been processed without errors or before all lock errors could be processed in the last sequential step of the process, you need to restart the run:

- Single-step restart
  You can manually restart the process for the data that has not been processed from the CVPM process monitor by choosing the *Information on the Restart Points* button. This processes the business transactions that were reset due to **lock errors**. For more information, see Capability to Restart Parallel Processes [page 1632] in the documentation for the Calculation and Valuation Process Manager (CVPM).
- Two-step restart
  You continue the processing of unresolved **lock errors** and of reset **packages** by triggering another generation of positions using the same values for the process parameters. Before you do so, check the application log and correct the serious errors. Only then is is possible to process the data packages that were reset. You can schedule an automatic process restart.

More Information

For more information about the common principles for the processes for position generation and source data aggregation, see Processes for Source Data Aggregation without Enhancement [page 298] or Processes for Source Data Aggregation with Enhancement [page 310].

3.2.4.3.1.3 Aggregating Financial Transactions

Use

You use this process to aggregate financial transactions, such as accounts, to create aggregation objects.

Process

At the start of the process, the system reads the changed data in the index tables and uses this data to create a worklist. You can start this process on the *SAP Easy Access* screen under *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing with Enhancement* > *Aggregate Financial Transactions* or *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing without Enhancement* > *Aggregate Financial Transactions*.

Each entry in the worklist contains the following data:

- Financial transaction ID
- The values of the granularity characteristics
- Validity date, time stamp, and other administrative data

The system uses this data to do the following:

- The system derives the aggregation type from the values, and assigns the financial transaction to one of the granularity characteristics and to an aggregation object for the aggregation type in question. If there is no aggregation object, the system creates one.
- If the system cannot derive the aggregation type, it does not assign the financial transaction to an aggregation object, and handles the transaction individually in the next processes.

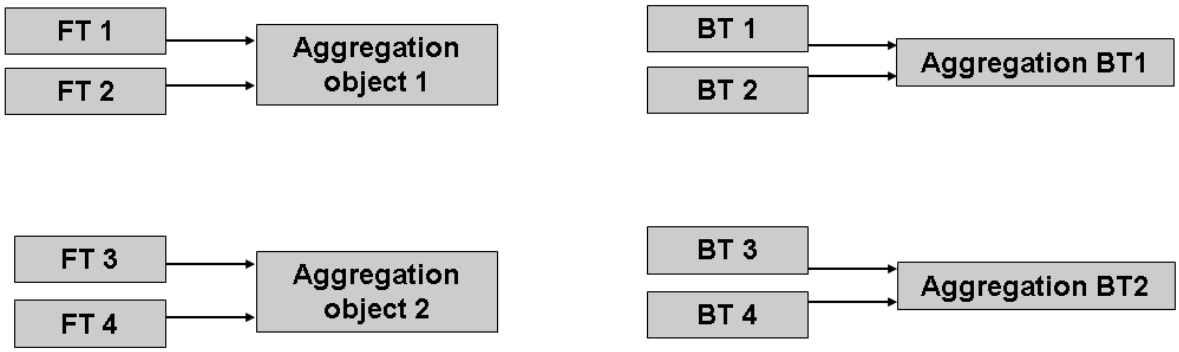
Result

At the end of this process, the financial transactions are assigned to aggregation objects, and the system has created new aggregation objects.

3.2.4.3.1.4 Aggregating Business Transactions

Use

In this step in the process chain, the system aggregates business transaction items to create financial transactions that have the same aggregation object. These are referred to as aggregation business transactions (see Metadata for Source Data Aggregation [page 290]). The diagram below explains the relationship between these objects.



Process

You start the following processes:

- **Prepare for the aggregation of business transactions**
  The system prepares the new business transactions for the business transaction classes that are to be aggregated. To start this process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing with Enhancement* > *Prepare Business Transactions for Aggregation* or *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing without Enhancement* > *Prepare Business Transactions for Aggregation*.

This process has two particular features:

- Reversal by value: Data is reversed as a reversal by value. This means that the item type of a business transaction is replaced by an offsetting item type. You assign offsetting item types to item types in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Assign Offsetting Item Types to Item Types*.
- Retroactive business transactions: When the assignments of financial transactions change after the position-relevant date of the business transaction, the system creates two reassignment business transactions for the time point in question. These business transactions transfer the position components of the individual transaction from the old aggregation object to the new aggregation object.
- Business transactions that are in the future: The system does not processes these business transactions until the position-relevant date of the business transaction.
- Business transactions that relate to financial transactions that are not aggregated: If a business transaction relates to a financial transaction that is not to be aggregated, the business transaction is not aggregated either.

- **Process reassignments for business transactions**
  In this process the system reassigns financial transactions. When the financial transactions are reassigned, the system creates two internal business transactions and prepares them for the aggregation process. These reassignment business transactions transfer the position components of the individual transaction from the old aggregation object to the new aggregation object. The system also creates reassignment business transactions if the reassignments are back-dated to a date before the position-relevant date, and relate to aggregated business transactions. For more information, see Example: Source Data Aggregation [page 288].
  To start this process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing with Enhancement* > *Process Reassignments for Business Transactions* or *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing without Enhancement* > *Process Reassignments for Business Transactions*.
- **Aggregate business transactions**
  In this process, the system reads the data from the previous processes and aggregates the business transactions. To start this process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing with Enhancement* > *Aggregate Business Transactions* or *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing without Enhancement* > *Aggregate Business Transactions*.

Result

At the end of this process the system contains the aggregation business transactions it needs to process the data further.

3.2.4.3.1.5 Aggregating Accrual Results

Use

In this part of the process chain, the system aggregates the accrual results for financial transactions that have the same aggregation object. The objects created in this way are referred to as aggregation accrual results (see Metadata for Source Data Aggregation [page 290]).

Process

You start the following processes:

- Aggregate current accrual results
  This process selects new accrual results for key dates between the date of the last successful aggregation of all accrual results and the current date. The system derives the current date from the date in the SDL time stamp. The system aggregates the accrual results for financial transactions that are assigned to the same aggregation object. To start the process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Periodic Processing without Enhancement* > *Aggregate Current Accrual Results*.

  > **Note**
  >
  > This process is an optional process in the chain of periodic processes. If you want to start a key date valuation, however, you must run this process. This is normally the case at month-end. It may also be the case that you want to run the key date valuation daily.

- Process the reassignments for accrual results
  This function processes all new reassignments for financial transactions that have dates that are before or on the date on which the last *Aggregate Current Accrual Results* process was started. For each assignment, the accruals that are valid in this date range are deducted from the aggregation accrual results of the previous aggregation object, and added to the current aggregation object.
  To start the process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing without Enhancement* > *Process Reassignments for Accrual Results*.
- Aggregate retroactive accrual results
  This function processes accrual results for key dates that are before or on the date on which the most recent *Aggregate Current Accrual Results* process was run. For each result the system subtracts the value of the previous version from the current value and adds the difference to the aggregation accrual result for the relevant aggregation object.
  To start the process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing without Enhancement* > *Aggregate Retroactive Accrual Results*.

Restart If Errors

As part of the processes for processing reassignments for accrual results and for aggregating retroactive accrual results you can restart a terminated run with a new run group, meaning that the system reprocesses all the data. To do this, select the *Generate New Run Group* checkbox on the selection screen for the relevant process.

However, the system cannot restart a terminated run with a new run group if the storage step had been reached when the previous run terminated and some results had already been saved. A new run group cannot be generated in this case because this would lead to inconsistent results. Instead, restart the process by running it again with the same parameters.

Result

At the end of this process the system contains the aggregation accrual results it needs to process the data further.

3.2.4.3.1.6 Deriving Application Events

Use

The system derives application events for each application in Accounting and each results data area from an operational event. If source data is not aggregated, application events are derived when operational events are imported. If you aggregate source data, application events are derived in this process for results data areas, which are assigned to an aggregation view.

Process

The process is as follows:

- The system creates a worklist that contains the newly imported operational events (business transactions that contain a value for the type of operational event).
- The operational events for financial transactions that are assigned to an aggregation object are filtered.
- The remaining operational events are transferred to application event management. The system can derive one or multiple application events for each operational event. You define how it derives application events in Customizing for *Bank Analyzer* under *Processes and Methods* > *Application Event Management*.

To start the process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Periodic Processing* > *Derive Application Events for Source Data Aggregation*.

Result

At the end of this process the system contains the application events it needs to process the data further.

3.2.4.3.2 Processes for Source Data Aggregation with Enhancement

Use

The processes for source data aggregation (SDA) compress data for further processing. For more information about the available process chains and their activation, see Processes for Source Data Aggregation [page 298].

The processes for source data aggregation are scheduled in part daily and in part periodically:

- Day processing
  To ensure that the general ledger is updated daily with operational changes, you can run certain source data aggregation processes once a day. You can run these processes in different frequencies but they must be run with the same frequency with which external business transactions are posted.
- Periodic processing

These processes are run in the following cases:

- Accrual results are provided for a valuation key date, expected loss amounts are calculated, or both. Regularly scheduled SDA processes aggregate these key date results so that the key date valuation can access the results for each aggregation object.
  You enter an RDL time stamp for the legal entity and aggregate retroactive key date results for each relevant subcomponent view. You then aggregate the current key date results.
- You have already corrected calculated key date results in the Results Data Layer (RDL) and want to adjust the corresponding results for each aggregation object. In this case, you have to run the key date valuation again for the adjustments.
  You enter an RDL time stamp for the legal entity and aggregate retroactive key date results for each relevant subcomponent view.

Prerequisites

Day Processing

- You have imported the data from your source system to Bank Analyzer. Changes to the granularity characteristics for source data aggregation have been updated successfully in the aggregation indices for the financial transaction, the business partner and the operative events.
- You then created a source data time stamp for a process partition. The processes for source data aggregation for the same process partition read data changes or new data from a time stamp interval of which the upper limit is the time stamp created. For more information, see Set the SDL Time Stamp [page 367].

Periodic Processing

- The daily processes for source data aggregation have run without errors. Specifically, SDA day processing has been completed for all relevant source systems of a legal entity.
- You have have set an SDL time stamp for the process partition of the selected legal entity.
- If you are calculating impairments, you have used the following periodic processes outside of source data aggregation to calculate expected losses and the free line for the key date and the legal entity and you have stored these in the RDL:
  - Determination of the Free Line for Financial Products [page 388]
  - Calculation of the Percentage Expected Loss [page 970]
- If you want to include accrual results, you have delivered these to the RDL for the key date and the legal entity.

Process

During source data aggregation a series of mandatory and optional processes need to be carried out in sequence. The system checks whether a previous mandatory processes has been completed successfully when the next process is started.

> **Note**
>
> The following applies to day processing: You can deactivate this check for test purposes in systems that are not live by selecting the *Suppress Check of Process Sequence* checkbox on the selection screen for the process. Note, however, that this can lead to inconsistencies and incorrect results.

The processes for source data aggregation differentiate between the following process errors:

Basic errors

Basic errors can be resolved subsequently and do not interrupt the process chain. In the case of basic errors, the run status in the monitor for the Calculation and Valuation Process Manager (CVPM) is *Successful* (green traffic light). The errors are listed in the application log. A basic error can occur, for example, if the currency of an imported business transaction does not match the currency of the position. In this case, reverse the business transaction and import the transaction with the correct currency.

Serious errors

Serious errors must be resolved immediately, before the process chain can continue. The run status of processes that have ended with serious errors is not displayed as *Successful* in the CVPM process monitor (yellow or red traffic light). In these cases, system checks prevent the subsequent processes from being executed. The errors are displayed in the application log for the run. You must remove the cause of the error before you are restart the process. If you then start the process again with the same parameter values, the process selects the data packages that were not processed and continues with the processing. These errors can be the termination of one or more processes in parallel processing or the resetting of incorrect data packages by the application, for example.

Import

The processes for source data aggregation select the relevant new and changed data in the Source Data Layer (SDL) with an entry time stamp that lies within a time stamp interval. The upper limit of the interval is the selection time stamp for source data from the last successful run of the *Set SDL Time Stamp* process for the same process partition. The lower limit is calculated from the last successful run of the corresponding processes for source data aggregation with the same values for the aggregation view, legal entity, and source system.

If you do **not** select the *Time Stamp Calculation* checkbox, you can enter the lower limit of the interval manually. This can be necessary if you are implementing the source data aggregation in a live system at later stage. If the lower limit is 0 during the first position generation, for example, when the position balances are created by initialization business transactions, only these transactions are considered and not the existing business transactions from the past.

> **Note**
>
> Import accrual results in the RDL before you run the *Set RDL Read Time Stamp for Source Data Aggregation* process. This ensures that the data is processed correctly in the source data aggregation.

The following applies to day processing: You need to set an SDL time stamp every time the process chain is started after a part of the data has been imported. The process partition selected identifies this part of the data. Every combination of legal entity and source system usually corresponds to a separate process partition.

> **Note**
>
> The *Aggregate Current Key Date Results* and *Aggregate Retroactive Key Date Results* processes are exceptions. They select the relevant new and changed data in the Results Data Layer (RDL) with an entry time stamp that lies within a time stamp interval. The upper limit of the interval is the selection time stamp for source data from the last successful run of the *Set RDL Read Time Stamp for Source Data Aggregation* process for the same legal entity. The lower limit is calculated from the last successful run of the corresponding processes for source data aggregation with the same values for the aggregation view, legal entity, and source system.

Methods for Source Data

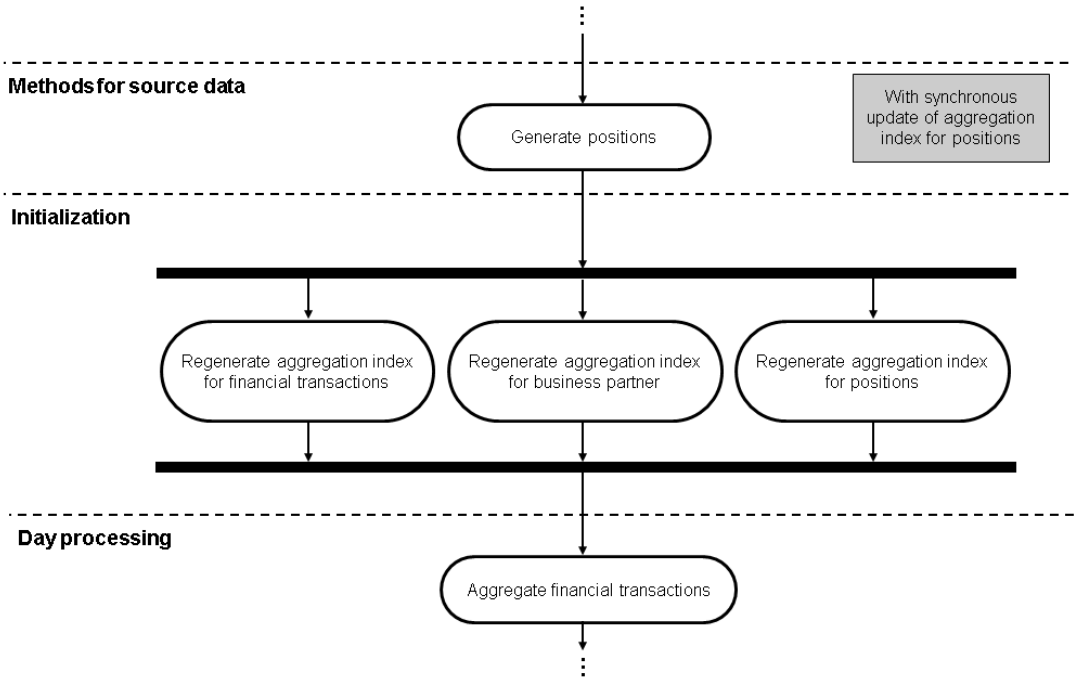
You generate or update positions for the position class specified from new imported business transactions. For more information, see Generate Positions [page 304].

Initialization

In the following cases you need to regenerate aggregation indices for granularity characteristics for aggregation:

- Later implemenation of source data aggregation in a live system
- Implementation of additional granularity characteristics or changes to derivation of exisiting granularity characteristics in Customizing

The following figure shows the sequence and dependencies of the initialization processes:



Sequence and Dependencies of the Initialization Processes

For more information about the initialization processes, see Regenerating Aggregation Indices [page 303].

Day Processing with Enhancement

Day processing can be executed in parallel for different legal entities and source systems:

1. Aggregate the financial transactions [page 306]
2. Aggregate the business transactions [page 307]:
   - Prepare for the aggregation of business transactions
   - Process the reassignments for business transactions
   - Aggregate the business transactions
3. Process reassignments for key date results
   For more information, see Aggregate Key Date Results [page 322].

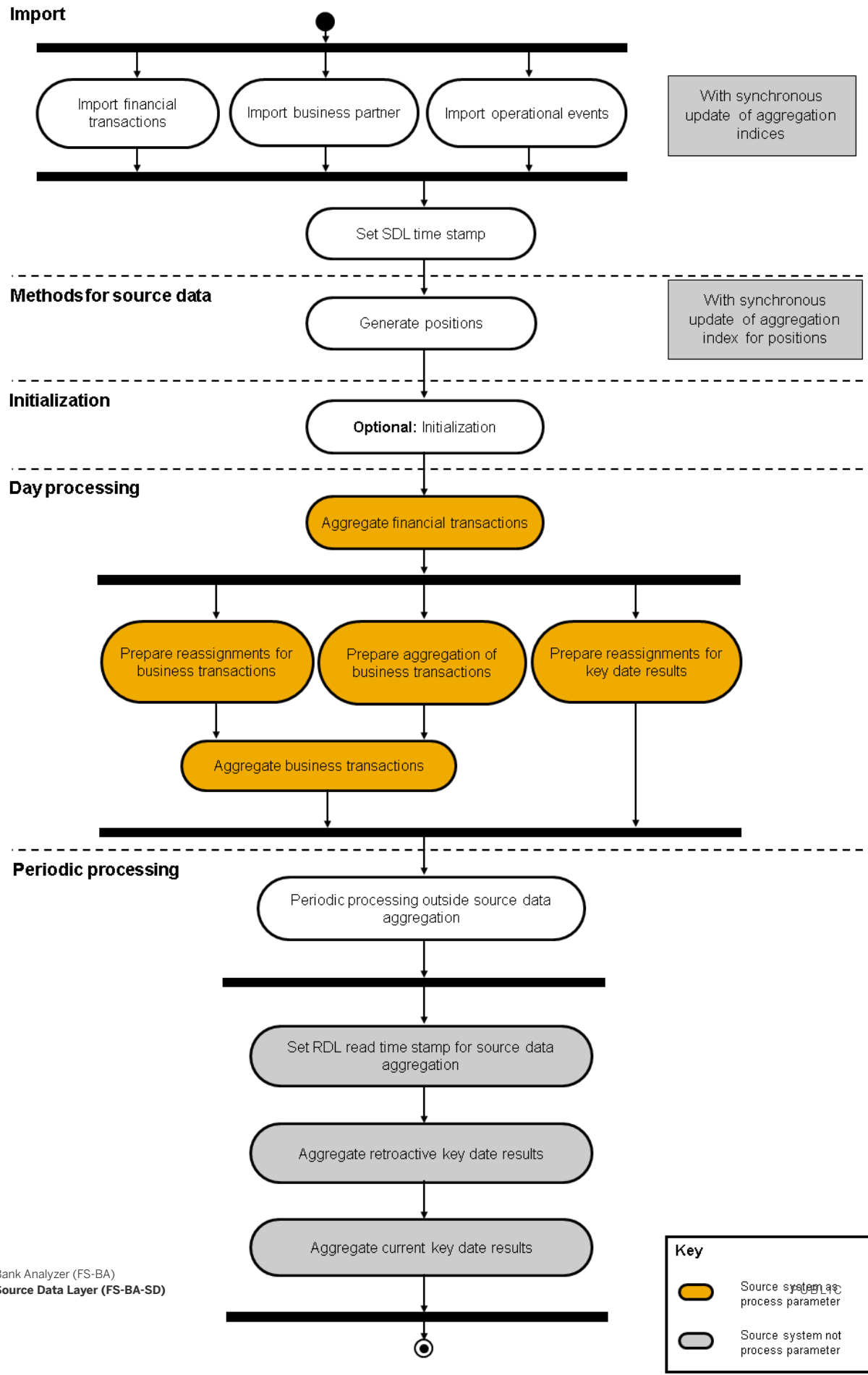
Periodic Processing with Enhancement

Periodic processing can be executed in parallel for different legal entities:

1. Set RDL read time stamp for source data aggregation
   You only complete this process if you want to execute one of the following periodic processes. If you have set the RDL time stamp, the retroactive aggregation of key date results is then mandatory.
2. Aggregate retroactive key date results
3. Aggregate current key date results

You need to execute all three processes if you want to calculate key date results at the end of a period. For more information, see Aggregate Key Date Results [page 322].

The sequence described above and the process dependencies are illustrated in the following figure:



Error Correction

You can reverse individual aggregation business transactions or all aggregation business transactions that have been generated within a run. For more information, see Reverse Aggregation Business Transactions [page 324] and Reverse Aggregation Runs [page 325].

More Information

Calculation and Valuation Process Manager (CVPM) [page 1622]

3.2.4.3.2.1 Regenerating Aggregation Indices

Use

Aggregation indices are updated when financial transactions, business partners and operational events are imported and when positions are generated. If you implement source data aggregation at a subsequent stage, or change settings in Customizing, these indices have to be regenerated.

Process

On the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Initialization* and start the following processes:

- *Regenerate Aggregation Index for Financial Transactions*
- *Regenerate Aggregation Index for Business Partners*
- *Regenerate Aggregation Index for Positions*

These processes determine the granularity characteristics from data in the Source Data Layer (SDL). However, an external import of operational events is required to regenerate the aggregation index for operational events.

> **Note**
>
> If you are already using an impairment solution in your operational system and implement the source data aggregation at a later stage, import the late payment events again in order to regenerate the aggregation index for operational events. You also need to include all of the granularity characteristics (such as *Delayed/Late Payment*, *Business Partner Rating*) that are required for the scenario used in the aggregation.
>
> You can create a new type of operational event for initialization events for the regeneration of the index table, from which no application events should be derived for impairment processes and accounting, in the Customizing activity *Edit Operational Event Type*. You also make settings in the Customizing activity *Derive Application Events from Operational Events*. This type of operational event is only used to generate the aggregation index.

Result

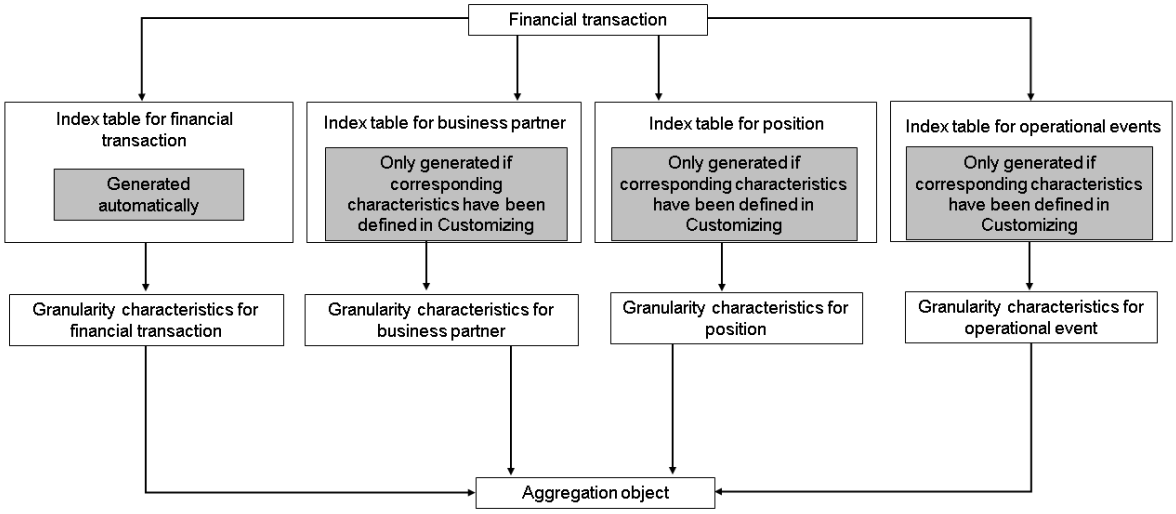
At the end of these processes the system has updated the information it needs to generate worklists for the aggregation of financial transactions.

Reversal

If a business transaction that maps an operational event is reversed, the system creates a correction entry in the index table. The reversed entry is not deleted.

Background

In each aggregation view (see Aggregation Views [page 291]), which financial transactions are assigned to which aggregation objects depends on the defining characteristics in Customizing. The granularity characteristics of a financial transaction depend on the financial transactions, business partners, positions, and operational events that belong to it. There is an index table for each aggregation view for these types of granularity characteristic. This table is created by the system when the aggregation view that has the aggregation type *Financial Transaction* is activated. The content of the index tables is used to create the worklists for the aggregation of financial transactions. The situation described above is illustrated in the following figure:



3.2.4.3.2.2 Generating Positions

Use

This process generates and updates positions for the position class specified from new imported business transactions in the Source Data Layer (SDL).

Prerequisites

- You have entered rules to define how positions are generated in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Methods for Sources Data* > *Position Generation*.
- You have imported the business transactions from the relevant source systems to Bank Analyzer and have set a SDL time stamp for the process partition. For more information, see Set the SDL Time Stamp [page 367].

Process

1. To start the process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Methods for Source Data* > *Generate Positions*. You restrict the run according to position class, legal entity, source system, and process partition.
2. The process selects all of the newly imported business transactions for the relevant business transaction classes for the process partition selected.
3. The system calculates the position by mapping the key fields and defining characteristics to the fields in the business transaction. The position date of a business transaction is a date field in the business transaction header. This date field is defined in Customizing for the position class. A business transaction for this date generates a new position version that has the same validity date.
   In Customizing, you define which key figures for the item classes contribute to which key figures in the position classes. By assigning item types to position key figures you define whether a business transaction item is included in the calculation of changes to the position key figures, and if so, whether it is included as a negative or positive value.

> **Note**
>
> This process does not fill the descriptive characteristics of positions.

This process has two particular features:

- A reversal business transaction is reversed by value. This means that the signs of the key figures in the original business transaction are reversed and added to the position.
- If there are retroactive business transactions, the system updates all subsequent positions.
- If there are business transactions that are in the future, the system creates positions for future dates.
- The business transactions are processed in packages. During parallel processing, the packages are distributed over the processes. Lock errors can occur if business transactions for the same position are processed in different packages. To process the business transaction, the system requests several times that the affected position is locked. If the lock cannot be set, the system saves the business transaction temporarily. In this case, the package is still regarded as having been processed without errors. The system processes the lock errors stored temporarily in the last sequential step of the process.

Result

The system creates new positions for all the items in the selected business transactions, and updates the existing positions.

Error Handling

You find information about the position generation runs for the `/BA1/RF2_BUILD_POSITIONS` process on the *SAP Easy Access* screen under *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. The process *Generate Positions* differentiates between basic and serious errors. For more information, see Processes for Source Data Aggregation without Enhancement [page 298] or Processes for Source Data Aggregation with Enhancement [page 310].

If the process terminates before all packages have been processed without errors or before all lock errors could be processed in the last sequential step of the process, you need to restart the run:

- Single-step restart
  You can manually restart the process for the data that has not been processed from the CVPM process monitor by choosing the *Information on the Restart Points* button. This processes the business transactions that were reset due to **lock errors**. For more information, see Capability to Restart Parallel Processes [page 1632] in the documentation for the Calculation and Valuation Process Manager (CVPM).
- Two-step restart
  You continue the processing of unresolved **lock errors** and of reset **packages** by triggering another generation of positions using the same values for the process parameters. Before you do so, check the application log and correct the serious errors. Only then is is possible to process the data packages that were reset. You can schedule an automatic process restart.

More Information

For more information about the common principles for the processes for position generation and source data aggregation, see Processes for Source Data Aggregation without Enhancement [page 298] or Processes for Source Data Aggregation with Enhancement [page 310].

3.2.4.3.2.3 Aggregating Financial Transactions

Use

You use this process to aggregate financial transactions, such as accounts, to create aggregation objects.

Process

At the start of the process, the system reads the changed data in the index tables and uses this data to create a worklist. You can start this process on the *SAP Easy Access* screen under *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing with Enhancement* > *Aggregate Financial Transactions* or *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing without Enhancement* > *Aggregate Financial Transactions*.

Each entry in the worklist contains the following data:

- Financial transaction ID
- The values of the granularity characteristics
- Validity date, time stamp, and other administrative data

The system uses this data to do the following:

- The system derives the aggregation type from the values, and assigns the financial transaction to one of the granularity characteristics and to an aggregation object for the aggregation type in question. If there is no aggregation object, the system creates one.
- If the system cannot derive the aggregation type, it does not assign the financial transaction to an aggregation object, and handles the transaction individually in the next processes.

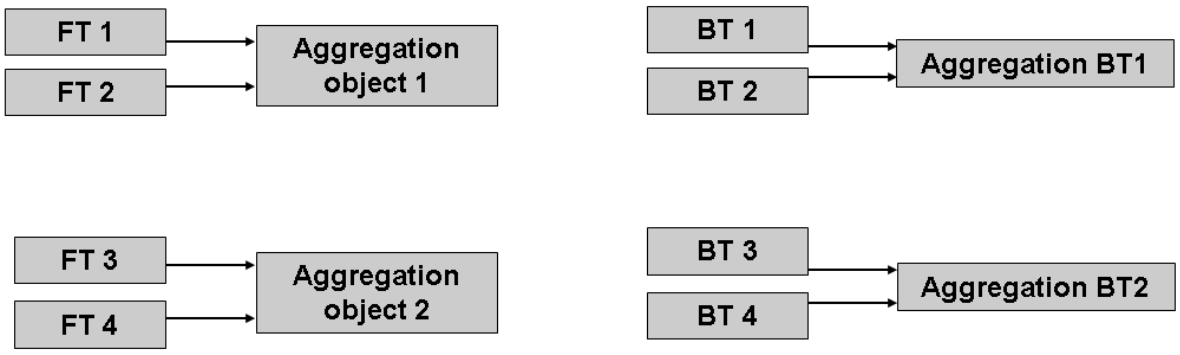
Result

At the end of this process, the financial transactions are assigned to aggregation objects, and the system has created new aggregation objects.

3.2.4.3.2.4 Aggregating Business Transactions

Use

In this step in the process chain, the system aggregates business transaction items to create financial transactions that have the same aggregation object. These are referred to as aggregation business transactions (see Metadata for Source Data Aggregation [page 290]). The diagram below explains the relationship between these objects.



Process

You start the following processes:

- **Prepare for the aggregation of business transactions**
  The system prepares the new business transactions for the business transaction classes that are to be aggregated. To start this process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data*

Layer > *Source Data Services* > *Source Data Aggregation* > *Day Processing with Enhancement* > *Prepare Business Transactions for Aggregation* or *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing without Enhancement* > *Prepare Business Transactions for Aggregation*.

This process has two particular features:

- Reversal by value: Data is reversed as a reversal by value. This means that the item type of a business transaction is replaced by an offsetting item type. You assign offsetting item types to item types in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Assign Offsetting Item Types to Item Types*.
- Retroactive business transactions: When the assignments of financial transactions change after the position-relevant date of the business transaction, the system creates two reassignment business transactions for the time point in question. These business transactions transfer the position components of the individual transaction from the old aggregation object to the new aggregation object.
- Business transactions that are in the future: The system does not processes these business transactions until the position-relevant date of the business transaction.
- Business transactions that relate to financial transactions that are not aggregated: If a business transaction relates to a financial transaction that is not to be aggregated, the business transaction is not aggregated either.

- **Process reassignments for business transactions**
  In this process the system reassigns financial transactions. When the financial transactions are reassigned, the system creates two internal business transactions and prepares them for the aggregation process. These reassignment business transactions transfer the position components of the individual transaction from the old aggregation object to the new aggregation object. The system also creates reassignment business transactions if the reassignments are back-dated to a date before the position-relevant date, and relate to aggregated business transactions. For more information, see Example: Source Data Aggregation [page 288].

  To start this process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing with Enhancement* > *Process Reassignments for Business Transactions* or *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing without Enhancement* > *Process Reassignments for Business Transactions*.

- **Aggregate business transactions**
  In this process, the system reads the data from the previous processes and aggregates the business transactions. To start this process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing with Enhancement* > *Aggregate Business Transactions* or *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing without Enhancement* > *Aggregate Business Transactions*.

Result

At the end of this process the system contains the aggregation business transactions it needs to process the data further.

3.2.4.3.2.5 Aggregate Key Date Results

Use

In this part of the process chain, the system aggregates the accrual results or expected loss amounts for financial transactions that have the same aggregation object. The objects created in this way are referred to as aggregation accrual results and aggregation expected loss (see Metadata for Source Data Aggregation [page 290]).

Process

Day processing with enhancement

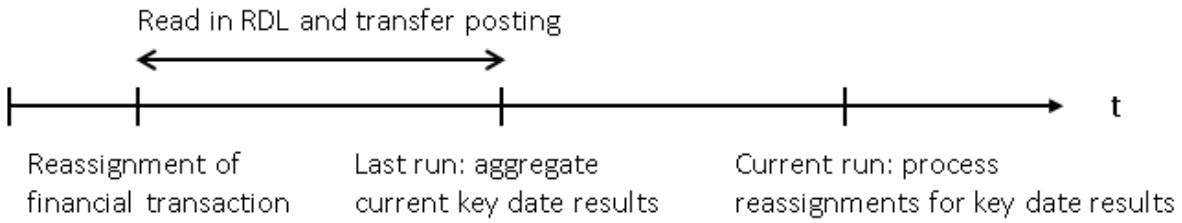
The system executes the processes for day processing for each legal entity and source system.

Process Reassignments for Key Date Results

This function processes accrual results or expected loss amounts for financial transactions in which the assignment to an aggregation object has changed. A reassignment is triggered when the sales product of a financial transaction is changed, for example.

The process selects all reassignments of financial transactions with a date that is before or on the date on which the *Aggregate Key Date Results* process was last run, in other words, retroactive assignment changes that were entered after the last aggregation of current key date results was completed. For these reassignments the system deducts the RDL results from the aggregation accrual results or the aggregation expected loss from the previous aggregation object and adds them to the aggregation object currently assigned. The assignment of a financial transaction depends on the key date of the reassignment, therefore the system only transfer posts the key date results on or after the day of the reassignment. Key date results before the key date of the reassignment remain in the previous aggregation object.

The situation described above is illustrated in the following figure:



To start the process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Day Processing with Enhancement* > *Process Reassignments for Key Date Results*. The subcomponent you select determines whether the system reassigns accrual results or expected loss amounts during a run. If you have defined subcomponent views for expected loss amounts and accrual results in Customizing, execute this processes once for each subcomponent.

Periodic processing with enhancement:

The system executes this process for each legal entity. The prerequisite for this is that accrual results have already been imported or calculated and the processes *Determine Free Line for Financial Products* and *Calculate Percentage Expected Loss*. For more information, see Determination of the Free Line for Financial Products [page 388] and Calculation of Percentage Expected Loss [page 970].

- **Set RDL Read Time Stamp for Source Data Aggregation**
  This process is only required if you want to execute one of the following two processes. You enter a RDL time stamp for the selection of expected loss amounts and accrual results to be aggregated in the processes *Aggregate Current Key Date Results* and *Aggregate Retroactive Key Date Results*.
  To start the process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Periodic Processing with Enhancement* > *Set RDL Read Time Stamp for Source Data Aggregation*.
- **Aggregate Retroactive Key Date Results**
  This function processes accrual results or expected loss amounts for key dates that are before or on the date on which the most recent *Aggregate Current Accrual Results* process was run. For each result the system deducts the value of the previous version from the current value and adds the difference to the aggregation accrual result or aggregation expected loss amount for the relevant aggregation object.
  To start the process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Periodic Processing with Enhancement* > *Aggregate Retroactive Key Date Results*. The subcomponent you select determines whether the system aggregates accrual results or expected loss amounts during a run. If you have defined subcomponent views for expected loss amounts and accrual results in Customizing, execute this process once for each subcomponent.

  > **Note**
  >
  > If the *Set RDL Read Time Stamp for Source Data Aggregation* process has been executed, you need to run the aggregation of retroactive key date results for the relevant subcomponent views before you can aggregate the current key date results or reassign key date results.

- **Aggregate Current Key Date Results**
  This process selects new accrual results and expected loss amounts for key dates between the date of the last successful aggregation of all key date results and the current date. The system derives the current date from the date in the RDL time stamp. The system aggregates the key date results for financial transactions that are assigned to the same aggregation object. The process also aggregates accrual results and expected loss amounts at the same time.
  To start the process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Periodic Processing with Enhancement* > *Aggregate Current Key Date Results*.

  > **Note**
  >
  > This process is an optional process within periodic processing. If you want to start a key date valuation for a new key date, however, you must run this process. This is normally the case at month-end. It may also be the case that you want to run the key date valuation daily.

Restart If Errors

As part of the processes for processing reassignments for key date results and for aggregating retroactive key date results you can restart a terminated run with a new run group, meaning that the system reprocesses all the data. To do this, select the *Generate New Run Group* checkbox on the selection screen for the relevant process.

However, the system cannot restart a terminated run with a new run group if the storage step had been reached when the previous run terminated and some results had already been saved. A new run group cannot be generated in this case because this would lead to inconsistent results. Instead, restart the process by running it again with the same parameters.

Result

At the end of this process the system contains the aggregation accrual results and aggregation expected loss amounts it needs to continue processing the data.

3.2.4.3.3 Reverse Aggregation Business Transactions

Use

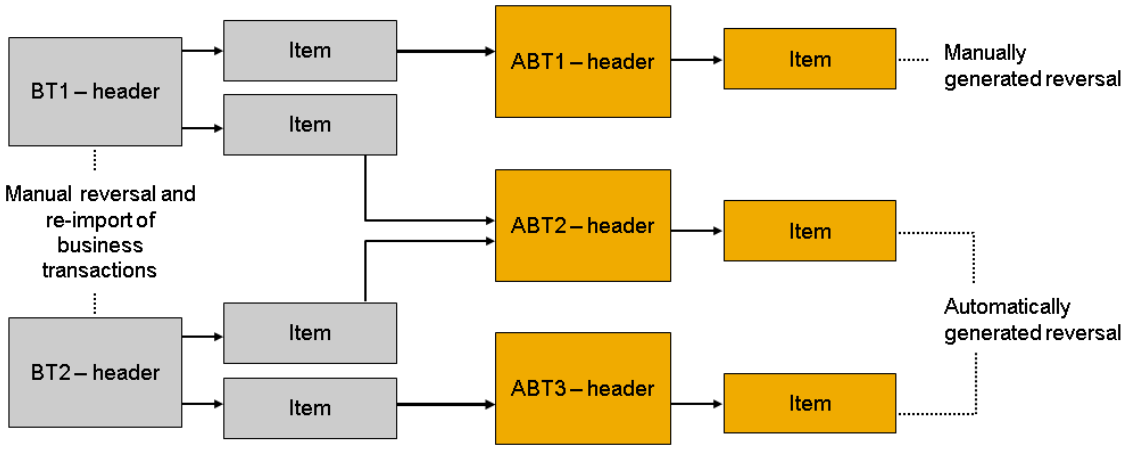
You can reverse individual aggregation business transactions that have been generated as part of a run within the *Aggregate Business Transactions* process. This source data aggregation function enhances the error correction strategy for business transactions in the Source Data Layer (SDL) without aggregation. For more information about this error correction strategy, see Error During Posting of External Business Transactions (Subledger Scenario) [page 784].

> ⚠ Caution
>
> The reversal function cannot be used in aggregation business transactions that were created by reassignments.

Process

1. You call this process on the *SAP Easy Access* screen under *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Error Correction* > *Reverse Aggregation Business Transactions*.
2. Use the business transaction ID and the source system to select the aggregation business transaction that you want to reverse and choose .
   If the items of a single business transaction belong to different financial transactions and therefore to different aggregation business transactions, the system finds and displays all aggregation business transactions connected to the selected aggregation business transaction. It also displays all SDL business transactions that are entered in these aggregation business transactions.

The following figure illustrates a group of aggregation business transactions and SDL business transactions that are to be reversed:



Reversal of an Aggregation Business Transaction

3. Reverse the aggregation business transactions by choosing .
   The system reverses the aggregation business transactions by reference and registers the single business transactions that have to be reversed manually.
4. Reverse the displayed single business transactions manually. For more information, see Reversing Business Transactions [page 156].
   The next time you call the reversal transaction for this aggregation business transaction, the system will not display the reversed single business transactions. As the system first registers which business transactions are to be reversed, this means that the *Prepare Business Transactions for Aggregation* and *Process Reassignments for Business Transactions* processes do not process the reversals for single business transactions.
5. Import the single business transactions again.
   The position generation function processes both reversals and new imports, meaning that it always reflects the current status.

More Information

For more information about reversing an entire run for the *Aggregate Business Transactions* process, see Reverse Aggregation Runs [page 325].

3.2.4.3.4 Reverse Aggregation Runs

Use

You can reverse all aggregation business transactions by reference that have been generated as part of a run within the *Aggregate Business Transactions* process. This enables you to easily and systematically correct more serious errors by reverse and restarting the relevant process runs. This source data aggregation function enhances the

error correction strategy for business transactions in the Source Data Layer (SDL) without aggregation. For more information about this error correction strategy, see Error During Posting of External Business Transactions (Subledger Scenario) [page 784].

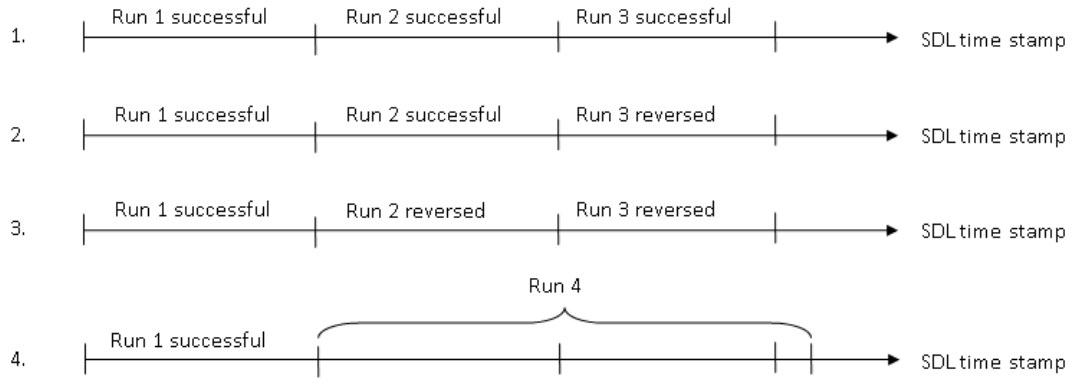
> **Note**
>
> The system processes financial transactions for relevant source data aggregation products individually if it cannot find an aggregation type (exception rule for aggregation). Business transaction for these financial transactions are not aggregated in source data aggregation and are processed individually in the accounting processes.
>
> The reversal process also reverses these business transactions by reference so that all the postings caused by the run are reset in accounting. However, the reversal is generated with the target business transaction class of the subcomponent view for the business transaction class of the imported business transaction. Moreover, the imported business transaction is not set to "reversed". This means that the system can make a distinction between a reversal caused by a reversal process and an external reversal delivered to the SDL.
>
> The next time you run the *Prepare Business Transactions for Aggregation* process the system resets the reversal of these business transactions. This does not ensure that all the processing errors in accounting have been corrected for these business transactions. If errors occur in accounting when you process these business transactions, you must also follow the error correction strategy detailed above for SDL business transactions without aggregation.

You may need to use this reversal process in the following cases:

- Errors have occurred in a large number of objects and you do not want to correct these individually. Instead, you want to reset and restart the entire run.
- Errors have occurred in the accounting processes that can only be resolved by the reversal of aggregation business transactions (such as incorrect postings caused by Customizing errors; program errors in the coding of accounting processes).

You have to schedule the reversal process several times if you want to reverse one aggregation run after another for multiple system time intervals. The reversal process is first run at the last system time interval in which the business transactions were aggregated without errors. The next runs reverse all previous runs in reverse chronological order. In a subsequent run of the *Aggregate Business Transactions* process, the system re-processes

the imported business transactions of the reversed system time intervals as well as the interval from the end of the first reversed run up until the current SDL time stamp. The figure below illustrates this procedure:



Reversal and Restart of Aggregation Runs

Prerequisites

- You have configured the `/BA1/RFG_BT_MASS_REVERSAL` – *Reverse an Aggregation Run for Business Transactions* process in Customizing for the Calculation and Valuation Process Manager (CVPM). For more information, see Calculation and Valuation Process Manager (CVPM) [page 1622].
- The source data aggregation and accounting processes have been executed in their entirety.

Process

1. You call this process on the *SAP Easy Access* screen under *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Error Correction* > *Reverse Aggregation Runs*:
   - Restrict the run according to aggregation view and legal entity.
   - Select the source system of the aggregation business transactions.

   According to how you have configured the step sequence for the CVPM process, corresponding activities run in the system.
2. Run the reversal run as many times as required to reset all incorrect runs.
3. Correct the cause of the error.
4. Restart the source data aggregation and accounting processes. After the reversal has been executed without errors, you can re-process the initial data in the Source Data Layer that lead to the aggregation business transactions.

3.2.4.4 Statistics

Use

You can use the statistics function to measure the summarization of data after it has been aggregated. This means that you can evaluate data before and after aggregation. The function is available for the following objects:

- Financial transactions
- Business transactions

Activities

You can use selection options to define the basic set of financial transactions and business transactions that are counted in a statistic scenario in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Statistics* > *Edit Statistic Scenarios*.

To count the raw data for statistics, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Statistics* > *Count Raw Data for Statistics*. This function counts financial transactions, aggregation objects, business transactions, and aggregation business transactions that are assigned to a statistic scenario and an aggregation view. Once you have defined an interval, the statistic results are stored individually for each interval.

> **Note**
>
> You execute the data count in the background.

You can then display the data results. To do this, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Statistics* > *Display Statistics*. The results is displayed according to the statistic scenario and the aggregation view in the form of a list.

3.2.4.5 Archiving

Use

At the time of Customizing, the system generates for the following areas of source data aggregation separate archiving objects that can be archived using the Archiving Engine:

- Aggregation index for financial transactions (FT)
- Aggregation index for business partners (BP)
- Aggregation index for positions
- Assignment table

The archiving process of the Archiving Engine has the following phases:

- Analysis: The system determines the archiving status of an object to be archived. The following checks are run here:
- Write: The system writes the objects that can be archived from the database into the archive.
- Delete: The objects that were written by the system to the archive are deleted from the database.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Results Data Layer* > *Archiving* > *Archiving Engine*.
2. Select an archiving scenario.

   > **Note**
   >
   > The following scenarios are available for source data aggregation:
   >
   > - Type 1 <name of subcomponent view><letter, where A = Assignment Table, B = BP Index, F = FT Index, P = Position Index> area 1SDA client <number> result
   > - Type 1 <name of subcomponent view><letter, where A = Assignment Table, B = BP Index, F = FT Index, P = Position Index> area 1SDA client <number> version
   >
   > The number of the scenarios depends on the generated objects entered in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Source Data Aggregation* > *Edit Subcomponent Views*.
3. On the *Phases* tab page, select whether you want to archive, reload, or display data in the archive.
4. On the *Addit. Data* tab page, select the maximum number of objects to be processed by the Archiving Engine, and determine the number of objects that are to be contained in the packages that are to be created.
5. On the *Mode* tab page, select the *Parallel* checkbox so that the system can process the generated archiving runs of an archiving step in parallel.
6. Choose *Execute* to start the archiving run.

3.3 General Access to Corrections

Definition

This function allows you to make corrections to primary objects via the analysis object.

An analysis object is not a physical data object in its own right. An analysis object comprises all of the functions in which a business object is made available for an Analyzer. An analysis object contains the following objects:

- The versions of a simple or structured primary object
- DataStore data from the relevant versions of the primary object

- Relevancy indicators

This data is managed using a common access key.

Use

In Bank Analyzer, relevancy indicators are maintained for the following primary object categories: financial transaction, financial instrument, account, position, business partner, country, and physical asset. Relevancy indicators enable the Analyzers to retrieve the data they require from the primary objects. A relevancy indicator always relates to exactly one primary object. Each primary object is assigned a separate relevancy indicator for each Analyzer. The link is established by a reference to the relevant primary object.

When Analyzers access InfoSets to read data, the relevancy indicator can restrict the hits for the initial data selected by the Analyzers. Restricting the initial data that is read optimizes the processing of the Analyzer when data is selected. This is possible only if certain primary objects are not relevant for individual Analyzers and you can set the relevancy indicator to *Not Relevant*.

Integration

Relevancy indicators can be set as follows:

- Relevancy indicators can be created and changed automatically using derivation strategies. You can define derivation strategies in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Index Function* > *Derivation for DataStore Updating*.
- Below the primary objects, there are BAPIs for making automatic changes.
- You can change relevancy indicators manually by choosing Edit Relevancy Indicators [page 331].

3.3.1 Making General Corrections Using the Analysis Object

Context

You can only make corrections to primary objects in the Source Data Layer. This transaction makes it easier for you to access the appropriate Bank Analyzer component, because you only need to know the InfoSet for the primary object you want to correct. To make the actual correction, you can navigate directly to the appropriate transaction for changing the primary object in question.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *General Correction Screen* > *Call General Initial Screen for Corrections via Analysis Object*.
2. To select the objects you wish to edit, you can make the following entries:
   - InfoSet
   - Key date
   - Date and time up to which changes to the object are considered
   - Analyzer for which the primary objects are relevant

   To make a more detailed selection, choose *Characteristics* and select an object by its detailed characteristic values.
3. The results of the selection are primary objects that correspond to the selection criteria. To make corrections, select the relevant field. This takes you to the transaction for changing the primary object in question.

3.3.2 Editing Relevancy Indicators

Context

You use the *Edit Relevancy Indicator* transaction to change relevancy indicators manually.

Relevancy indicators are provided for the following components:

- Balance Analyzer
- Profit Analyzer
- Strategy Analyzer
- Credit Exposure
- Counterparty Risk
- Limit Manager
- Historical Database
- Regulatory Reporting

In line with the Bank Analyzer versioning concept, a new version is created automatically when a relevancy indicator is changed. Existing object versions are never overwritten or deleted and can still be read even after a new version has been created.

> **Note**
>
> Two date fields the business record date and the system time are available in the system for displaying versions.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *General Correction Screen* > *Edit Relevancy Indicator*.
2. Enter a key date. The key date is the business validity date for which you wish to select the version of the relevancy indicator.
3. Enter an identifier for the relevant object.
   - Financial transactions, financial instruments, accounts, countries, and physical assets are identified uniquely by their external number.
   - Business partners are identified by the business partner number.
   - If you want to select a position, a separate selection screen is displayed where you can enter the exact selection criteria.
4. You have the following options:
   1. *Create Analysis Object*
      The system creates a new version of the analysis object. You can select *Relevant* or *Not Relevant*. The initial value is *Not Created*.
      If the component in question has not been implemented and derivation rules have not been defined for the relevancy indicator, or if a value has not been entered manually, the initial value remains *Not Created*. However, you cannot manually specify the value *Not Created*. Save your entries.
      Before you save your data, you can use the following functions on this screen:

Action	Function
Multiple Correction	You can make the same changes to several versions of the relevancy indicators.
Versions	You can display a list of all the versions of the relevancy indicators saved for the object. To add the *Valid to* and *Changed on* columns to the list, choose *Change Layout*. **Note** Note that the *Changed On* date of the most recent version is December 31, 9999, which means that the version has not yet been changed. The *Valid To* date is December 31, 9999, which means the validity of the version is not limited.

Action	Function
Index	You can display all index data for all versions of the primary object. Each row of the list represents one version. To add the *Valid to* and *Changed on* columns to the list, choose *Change Layout*. **Note** Note that the Changed On date of the most recent version is December 31, 9999, which means that the version has not yet been changed. The *Valid To* date is December 31, 9999, which means the validity of the version is not limited.
Main Object	You can navigate to the display of the relevant object.

2. *Display Analysis Object*
   You can display the version of an analysis object [page 329] that is valid on the specified key date.
   You can use the following functions on this screen:

Action	Function
Versions	You can display a list of all the versions of the relevancy indicators saved for the object. To add the *Valid to* and *Changed on* columns to the list, choose *Change Layout*. **Note** Note that the Changed On date of the most recent version is December 31, 9999, which means that the version has not yet been changed. The *Valid To* date is December 31, 9999, which means the validity of the version is not limited.
Note	You can display the correction notes for the analysis object.

Action	Function
Index	You can display all index data for all versions of the main object. Each row of the list represents one version. To add the *Valid to* and *Changed on* columns to the list, choose *Change Layout*. **Note** Note that the *Changed On* date of the most recent version is December 31, 9999, which means that the version has not yet been changed. The *Valid To* date is December 31, 9999, which means the validity of the version is not limited.
Main Object	You can navigate to the display of the relevant object.

3. *Main Object*
   The display mode is displayed for the object in question.
4. *Main Object*
   The change mode is displayed for the object in question. You can enter changes directly.
5. Balance Analyzer
   The Balance Analyzer correction transaction is called. You can enter line items in the usual way.
6. Profit Analyzer
   The screen for entering line items in Profit Analyzer is called. You can enter line items in the usual way.

3.4 Tools

The tools for the Source Data Layer comprise the following functions:

- Comparing Positions and Business Transactions [page 335]
- Position Master Data Check [page 336]
- Master Data and Master Data Hierarchies [page 337]
- Segment Hierarchies [page 342]
- Index Tables [page 344]

3.4.1 Comparing Positions and Business Transactions

Prerequisites

In Customizing, you have worked through the Customizing activities in the section *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Transaction Data* > *Comparison of Positions and Business Transactions*.

Context

The report for comparing positions and business transactions takes the changes that have been made to positions in the Source Data Layer (SDL) over a specific period of time, and compares them with the position changes that have resulted from business transactions. This enables you to identify discrepancies between positions and business transactions.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Transaction Data* > *Compare Positions and Business Transactions*.
2. Enter a reconciliation type, a period, a key date and, if applicable, a time.
3. In order to limit the number of positions selected, specify *account characteristics*, *template category*,*external number*,*parent template*, *child template*, *node number*, or *defining characteristics*as selection criteria.
4. If necessary, select a layout for displaying the results list.

   > **Note**
   >
   > If you want to select a layout, you must already have run a comparison for a reconciliation type, and created a layout.

5. If you wish to save the log for this run, activate the appropriate indicator.
6. Choose *Execute*. The system displays the log for the run.
7. You can display any results in a results list. To do so, choose *Back* (F3).
8. In order to identify possible reasons for any discrepancies that exist, you can click the position header in the results list, and display the initial position, final position, or the business transactions involved.

3.4.2 Position Master Data Check

Use

You use this transaction to recognize and repair any inconsistencies in the environment of positions and position master data.

Examples of such inconsistencies are the existence of position master data without an associated position, or a position that makes reference to position master data that is no longer in the Source Data Layer (SDL).

For basic information about position master data, see Primary Object: Position [page 142].

Integration

When you manually create position master data that does not have an associated position, this position master data might – as a result of incorrect Customizing - still be created and updated in the DataStore objects. However, the system cannot save the technical position that is generated here because the DataStore update of the position is terminated.

Features

You can check whether positions exist for position master data. To do so, proceed as follows:

1. Select the *Check Position Master Data* checkbox. Note: Limit the selection to specific position master data, if necessary.
2. Choose *Execute*.Note: If inconsistencies exist, the system returns the external numbers of the relevant position master data. By selecting a position master data object and calling the *Select Position* function (Shift + F1), you can either assign the position master data to an existing position or create a new technical position that refers to the master data.

You can check whether the positions refer to master data that no longer exists in the SDL. To do so, proceed as follows:

1. Select the *Check Positions* checkbox. Note: Limit the selection to one position class if necessary. If you do not, the system will check all positions.
2. Choose *Execute*. Note: If a position refers to a master data object that is no longer in the SDL, but which is in the archive, the system marks this position in yellow. You do not need to correct this. However, you might experience problems when you try to display position master data for this position, because the system does not read any data from the archive. Note: If, for other reasons, the reference but not the master data still exists, you can delete this reference (Shift + F2). Afterwards you can recreate the master data for the position.

3.4.3 Master Data and Master Data Hierarchies

You manage the characteristics for the whole of Bank Analyzer using the characteristic repository [page 20]. You can store master data for characteristics there. Characteristic master data is application data that is managed in the characteristic repository, for example characteristic values, attribute values, and texts.

Reading the Master Data for the Various Characteristic Categories

- **Custom Characteristics**
  Master data is created for the InfoObject in the Tool BI system.
- **Local Characteristics**
  Master data is read from the check tables of the relevant data element. The master data in the check tables is either shipped by SAP or customer-defined.
- **Fixed Characteristics**
  Master data is read from the check tables of the relevant data element. The master data in the check tables is either shipped by SAP or customer-defined.

Master Data Hierarchies

Characteristics with master data can have master data hierarchies. You can create master data hierarchies for master data in the Tool BI system only. They are available in the characteristic repository. The characteristic repository also provides function modules that can read information about master data hierarchies. You can transport master data hierarchies to your Reporting BI system, and thus make this data available for reporting purposes.

Versioning Master Data Hierarchies

If a master data hierarchy is used, the system always reads its current status from the Tool BI system, and writes it (and the corresponding time stamp) to the core Bank Analyzer system. In this way, you can read older versions of master data hierarchies from the core Bank Analyzer system.

Replication of Texts for Characteristic Master Data in Several Languages

You can replicate the texts for characteristic master data irrespective of your logon language in SAP NetWeaver Business Intelligence (BI). You can do this for several languages.

- You can select a language in the master data monitor [page 338]. This language is then used for the comparison of the master data texts with those in the BI system, as well as for the replication of master data texts from the monitor.

- You can specify the languages used to replicate the texts of the master data in the replicate master data [page 339] report for fixed and local characteristics.
- You can specify the languages used for the texts of the master data that are replicated in the BI system in the report for the copying BI master data. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Tools* > *Transport and Client Copy*, and then choose *Copy BI Master Data*.

3.4.3.1 Calling the Master Data Monitor

Prerequisites

You have run the initial configuration in Customizing by choosing *Bank Analyzer* > *Basic Settings* > *Global Settings* > *Run Initial Configuration*. In this activity, you registered fixed characteristics in the core Bank Analyzer system.

You have created and registered local characteristics in Customizing by choosing *Bank Analyzer* > *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Register Characteristics*.

Context

The Master Data Monitor compares the master data and master data texts of the registered fixed and local characteristics with those of the corresponding InfoObjects in the Reporting BI system, and displays any differences that are detected.

During this comparison, the system first of all attempts to write the master data to the secondary InfoObject. If the system is unable to find the secondary InfoObject, you can decide whether or not the master data should be written to the primary InfoObject.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Master Data and Hierarchies* > *Call Master Data Monitor*.
2. To compare the master data for local characteristics, specify the logical system of the Reporting BI system, and flag the *Local Characteristics* indicator. To compare the master data for fixed characteristics, set the *Fixed Characteristics* indicator. If necessary, set the *Check Master Data Texts* indicator. If you set this indicator, the system checks the texts as well. If there are any differences, the system displays a warning message.
3. Choose *Execute*. The master data monitor appears.
4. A red symbol indicates that the master data for a characteristic is inconsistent. In the characteristic monitor, you can navigate to update the master data. To do so, choose *Update Master Data*.

If you set the indicator in step 2 for texts to be checked, inconsistent texts are indicated by a yellow symbol.

3.4.3.2 Replicating Master Data

Prerequisites

You have run the initial configuration in Customizing by choosing *Bank Analyzer* > *Basic Settings* > *Global Settings* > *Run Initial Configuration*. In this activity, you registered fixed characteristics in the core Bank Analyzer system.

You have created and registered local characteristics in Customizing by choosing *Bank Analyzer* > *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Register Characteristics*.

Context

You use this transaction to call up a report that compares the master data of fixed and local characteristics with the master data of InfoObjects in the Reporting BI system. During this comparison, the system first of all attempts to write the master data to the secondary InfoObject. If the system is unable to find the secondary InfoObject, the master data is written to the primary InfoObject.

You can call up logs for master data replication at any time.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Master Data and Hierarchies* > *Replicate Master Data*.

   Specify the logical Reporting BI system, and start the report. The system runs the report in the background. Confirmation of this appears in a message at the bottom of the screen.
2. To display the various steps involved in the report, press F3. A screen appears with the following information:
   - Processing number of the replication run
   - Processing status
   - Start date of processing
   - Start time of processing
   - User

3.4.3.3 Checking Master Data Hierarchies for Changes

Prerequisites

- In Customizing for BI, you have created InfoObjects with master data hierarchies.
- You have registered characteristics in Customizing by choosing *Bank Analyzer* > *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Register Characteristics*.
- In Customizing, you have recorded the user of the master data by choosing *Bank Analyzer* > *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Master Data Hierarchy* > *Assign Recipient Groups to Master Data Hierarchies*.

Context

You use this transaction to compare the current status of master data hierarchies in the Tool BI system with the lasts status used (see Versioning of Master Data Hierarchies [page 337]). If there are any differences, the system notifies the user and displays details about the changes.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Master Data and Hierarchies* > *Check Master Data Hierarchies for Changes*.
2. Set the *Execute Version Creation* indicator, and choose *Execute*.

   A list of changed master data hierarchies appears. The system notifies users in accordance with the setting made in Customizing.

   > **Note**
   >
   > When using the notification function, we recommend you schedule the report to run periodically in the background.

3.4.3.4 Deleting Versions of Master Data Hierarchies

Prerequisites

- In Customizing for BI, you have created InfoObjects with master data hierarchies.
- You have registered characteristics in Customizing by choosing *Bank Analyzer* > *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Register Characteristics*.

Context

In this transaction, you can delete versions of master data hierarchies [page 337] from the core Bank Analyzer system whose time stamp is before the date and time entered.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Master Data and Hierarchies* > *Delete Versions of Master Data Hierarchies*.
2. Enter the date for which the master data hierarchies are to be deleted.
3. Choose *Execute*.

   The master data hierarchies you selected have been deleted. The system displays a list of the versions of master data hierarchies that have been deleted.

3.4.3.5 Restoring Master Data Hierarchies

Prerequisites

- In Customizing for the Tool BI system, you have created InfoObjects with version-dependent master data hierarchies.
- You have registered characteristics in Customizing by choosing *Bank Analyzer* > *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Register Characteristics*.
- The system has read the master data hierarchies from the Tool BI system and written them to the buffer in the core Bank Analyzer system.

Context

You use this transaction to call up older versions of master data hierarchies from the buffer in the core Bank Analyzer system, and write them to the Tool BI system.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Master Data and Hierarchies* > *Restore Master Data Hierarchy*.
2. Select the characteristic, hierarchy, hierarchy version if necessary, and validity date of the master data hierarchy that you wish to restore in the Tool BI system.
3. Select the date and time at which the master data hierarchy was stored in the buffer.
4. Enter the hierarchy version under which you want to write the master data hierarchy to the Tool BI system.
5. Choose *Execute*. The master data hierarchy is written to the specified hierarchy version in the Tool BI system.

3.4.4 Segment Hierarchies

Use

Segment hierarchies are used by the Analyzers to organize existing results data hierarchically. Analyzers need segment hierarchies for the following purposes:

- Processing
  The Analyzers use segment hierarchies as the basis for calculating non-cumulative key figures. Using Analyzer-specific procedures, these non-cumulative key figures can be determined at the nodes of the segment hierarchies, where results data for individual data records is output.
- Account assignment
  The Analyzers can use segment hierarchies to store the resulting non-cumulative key figures at the appropriate hierarchy nodes.
- Reporting
  Reporting results are output to the nodes of the segment hierarchy.

Features

Segment hierarchies are classified by hierarchy types. Each segment hierarchy is assigned to one hierarchy type. A hierarchy type is a collection of the characteristics registered in Bank Analyzer by the characteristic repository. When you define a segment hierarchy, you must use the characteristics contained in the corresponding hierarchy type. Since the end nodes in a segment hierarchy contain all the values for all characteristics in the hierarchy type, when you select a hierarchy type you simultaneously define the level of detail displayed for the results data in the Analyzers.

You **define** a segment hierarchy by selecting the characteristics that make up the hierarchy, and by specifying the sequence of characteristics for aggregation. You can define segment hierarchies in Customizing for Bank Analyzer under *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Segment Hierarchies*.

Before you can use segment hierarchies in the Reporting BI system, you have to generate the relevant InfoObjects in the Tool BI system. One InfoObject is created for each hierarchy type. InfoObjects can be transported from the Tool BI system to the Reporting BI system. The nodes of the segment hierarchies can be created as master data for this InfoObject, the tree structure of the segment hierarchies can be created as master data hierarchies in reporting BI.

> **Note**
>
> This master data and the master data hierarchies make it technically possible to use segment hierarchies in BI. They are **not** equivalent to the master data and master data hierarchies in the Bank Analyzer characteristics.

3.4.4.1 Extracting Segment Hierarchies

Prerequisites

- You have worked through the Customizing activities in the section *Bank Analyzer* > *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Segment Hierarchies*.
- The segment hierarchies have been set up by the individual Analyzers.

Context

You use this transaction to extract segment hierarchies, and write them to the Reporting BI system.

This report writes the segment hierarchies of a hierarchy type as master data to InfoObject `/BA1/HSGH...`, and the segment IDs as master data to InfoObject `/BA1/HHTP...`.

The outcome of the extraction is saved to a log that you can display at any time.

Procedure

1. On the SAP Easy Access screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Master Data and Hierarchies* > *Extract Segment Hierarchy*.
2. Select a hierarchy type and a segment hierarchy.
3. You have the following options:

- To display all the logs for replicating master data and hierarchies to the BI system, select *Display Logs*.
- To delete logs from a specific period, select *Delete Logs*.
- To start replicating the hierarchy nodes automatically as master data to the BI system, select *Replicate Master Data*. The log is displayed.
- To start replicating the segment hierarchy automatically to BI, select Replicate Hierarchies. The log is displayed.

> **Note**
>
> Alternative to *Replicate Hierarchies:*
>
> To write segment hierarchies as master data hierarchies to the Reporting BI system, choose *Start InfoPackage*. Enter the source system in the Reporting BI, and the InfoSource. If several InfoPackages exist for the InfoSource, enter a description of the InfoPackage.

4. Choose *Execute*.

3.4.5 Index Tables

Definition

See index table.

Use

Index tables (DataStores) allow the Analyzers to use global selection criteria to search across all primary objects for relevant data.

When you create and save a primary object, the system automatically writes the contents of the primary object to the DataStore. The process for doing this is as follows:

Once the primary object is saved, the data fields of this primary object are mapped from the core Bank Analyzer system to the characteristics of the object transfer structures (OTS) for the index. The central derivation tool is used for this purpose. An OTS is assigned to DataStore objects by using characteristics with the same names. In this way, the index OTSs are used to fill and update entries in the index tables. Key figures cannot be used in DataStore objects.

Separate index tables are kept for all primary objects.

Structure

A DataStore consists of a key part and the actual data part. The key part contains the source system ID (scenario with source system ID), a key identifying the appropriate primary object, and date fields for mapping the version. In addition to date fields for versioning, the data part contains filter characteristics, which can be used to search for data from primary objects.

See also Index Function [page 266].

3.4.5.1 Calling Report for Comparing Primary Objects with Index Tables

Use

This report compares data from the DataStore objects in the Tool BI system with data from the primary objects. This helps make sure that the data in the DataStore objects is consistent.

Activities

1. On the SAP Easy Access screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Index Tables* > *Call Report for Reconciliation of Primary Objects and Index Tables*.
2. Enter a time interval. This refers to the time at which a particular version was created.
3. Select one or more primary objects.
4. In *Numbers ... to*, enter a number interval for the primary object(s).
5. Choose *Execute*.
   If any differences are detected, the system displays these in an overview list. In this way, the system shows you whether any versions of the primary object data is missing in the DataStore objects, or vice versa.

   > **Note**
   >
   > If a DataStore version exists but there is no corresponding version in the Source Data Layer (SDL), the following report has to be run to delete the DataStore version: /BA1/RF3_IDX_REORG
   >
   > If an SDL version exists but no corresponding DataStore version, data has to be regenerated. See:
   >
   > - Regenerating Index Tables for Master Data Objects [page 347]
   > - Regenerating Index Tables for Positions [page 348]
   > - Regenerating Index Tables for Business Partners [page 348]
   > - Regenerating Index Tables for Organizational Units [page 349]

3.4.5.2 Initializing Archived Position Information

Use

You can use this transaction to restore archived position information for position items that you need to reorganize archived object versions (see Reorganizing Archived Object Versions [page 346]). You need to do this if you previously partially archived positions and now want to reorganize the corresponding entries in the index tables.

> **Note**

- You do not need to call this transaction for any positions that you have archived as of banking services from SAP 7.0. This is because the system updates the position information for reorganizing archived object versions automatically when archiving.
- Position information is only restored if a corresponding entry still exists in the index table.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Index Tables* > *Initialize Archived Position Information*.
2. Specify the package size to be used to generate the worklist.
3. Use the *Position Class*, *Account Category*, and *Account Number* fields to select the position information that you want to initialize, and choose *Execute*.

Result

The system displays the number of selected position items.

3.4.5.3 Reorganizing Archived Object Versions

Use

You can use this transaction to reorganize index table entries for partially archived object versions that have entries in the index tables for master data, positions, and relevancy indicators. During the reorganization, the system selects archived object versions that have not yet been reorganized and adds them to the backup table in Customizing. In Customizing for *Bank Analyzer*, choose *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Index Function* > *DataStore Objects* > *Register OTS/DataStore Objects*. The system then deletes the data from the index table. This reduces the volume of data and enables faster access to the index tables.

> **Note**

When you reorganize relevancy indicators, you specify the same values as those you specify when you archive the corresponding object versions. By doing this, you ensure that the entries for the relevancy indicators for object versions are kept consistent.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Index Tables* > *Reorganize Archived Object Versions*.

2. Specify the package size to be used to generate the worklist.
3. Enter the following selection parameters:
    - For master data: template category and external number
    - For positions: position class, account category, and account number
    - For relevancy indicators: business record date, pro rata, or corrected versions only
4. Choose *Execute*.

Result

The system displays a list of the number of selected and reorganized objects.

3.4.5.4 Regenerating Index Tables for Master Data Objects

Context

You use this transaction to regenerate the relevant index table for financial transactions, financial instruments, accounts, countries, hedging relationships, and physical assets.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Index Tables* > *Regenerate Index for Master Data Objects*.
2. You have the following options:
    1. To display logs of previous processing activities, choose *Display Logs*. Then select *Execute*.
    2. To delete logs of previous processing activities, select *Delete Logs*. Then select *Execute*.
    3. To generate a new index table for a template category, choose *DataStore Update*. Enter a *system date* and *system time*. All versions created after this point in time are selected. If necessary, restrict the range of *external object numbers*. Then select *Execute*. The system displays a log of the processing activity.
    If you want to update the DataStore for the external object numbers specified, make sure that the system date specified is not older than the objects you want to process. This means that the system updates all of the DataStore objects whose system date is a date between the system date that you the user specify, and the highest possible system date.

3.4.5.5 Regenerating Index Tables for Positions

Context

You use this transaction to regenerate the index table for the relevant position class.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Index Tables* > *Regenerate Index for Positions*.
2. Enter a range for *Position Class*, *Account Number*, and *Account Category*.
3. You now have the following options in the *Processing* section of the screen:
   1. To generate a new index table for the position data you have specified, choose *Reconstruct*.
   2. To regenerate index entries for all positions that do not have an entry in the DataStore object of their position class, choose *Index unprocessed positions*.
   3. To regenerate relevancy indicators only and not index entries, choose *Change Relevancy Indic. Only*.
   4. To generate new index tables for the primary objects reloaded from the archive, choose *Reloaded from Archive*.
4. Choose *Execute*.

3.4.5.6 Regenerating Index Tables for Business Partners

Context

You use this transaction to regenerate the index tables for business partners/business partner differentiation types.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Index Tables* > *Regenerate Index for Business Partners*.
2. You have the following options:

1. To generate a new entry in the index table for the selected business partners, choose *Update DataStore*. Enter a range of business partner numbers.
2. To display logs of previous processing activities, choose *Display Logs*.
3. To delete logs of previous processing activities, choose *Delete Logs*.

3. Choose *Execute*.

3.4.5.7 Regenerating Index Tables for Organizational Units

Context

You use this transaction to generate the index table for organizational units.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Index Tables* > *Regenerate Index for Organizational Units*.
2. Enter a *validity date*, and a value range for *ID of Organizational Unit*.
3. Choose *Execute*.

   The system displays a log of the processing activity.

3.4.5.8 Trace for Index Update

Use

This function can be used as required to monitor and analyze the way in which methods for updating indexes work.

> **Note**
>
> In order to use the trace tool, the coding concept for updating indexes must be recognized in debugging mode. For this reason, we recommend the trace tool be used by system administrators.

Features

The trace tool has two basic processing modes. In trace mode, information is recorded and in debugging mode, the recorded information is used to analyze the process in the ABAP Debugger.

- **Trace mode**
  A trace is activated for a particular user and time in accordance with a table entry. If the trace is active for a user, the system logs the following data each time the relevant functions are called up:
  - Information about the execution time and processing context
  - Call parameters
  - Depiction of the data in the database for the relevant objects before the function is executed
  - Depiction of the data that is written to the database

  In this process, a logical unit of work (LUW) is treated as a trace unit. If the check functions are called up several times within an LUW, the system logs the details for each separate call. When an LUW is saved using the save function or DB save function, the trace unit is flagged in the information.
- **Debugging mode**
  The user interface for the trace tool provides users with information about all active trace frames and the traces recorded for them. When a trace is selected, the LUW recorded is processed in the ABAP Debugger. The functions are supplied with the necessary data from the saved parameters and the database information. At the end of the LUW, the data is not updated in the database.

Activities

1. On the SAP Easy Access screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Index Tables* > *Edit Trace for Index Update*.
2. To create a trace frame, enter a user and an activation period. Choose *Activate*.

   > **Note**
   >
   > The activation period must not start in the past. To refresh the activation period, choose *Proposal*.
3. In the *Trace frame*, select a user. The traces for that user are displayed.
4. You have the following options:
   - To display the trace parameters, choose *Parameter*.
   - To debug the trace, choose *Debug*.
   - To use the Debugger to display the data that has been updated, choose *Data*.

3.4.5.9 Reorganizing the DataStore

Context

You use this transaction to delete data from the relevant DataStore object if, for example, the log for an Analyzer process displays primary objects that the system is unable to find in the Source Data Layer (SDL).

> **Note**
>
> Only ever use this report for primary objects which have versions only in the DataStore object, but not in the SDL. If a version of primary object exists in the SDL, you have to use the report for regenerating master data objects, positions, or business partners.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Index Tables* > *Reorganize DataStore*.
2. Select a primary object, and enter a number or number range for it.
3. In the *Process Control* box, you will notice that the *Test Run* indicator has been automatically activated. If you want to reorganize the DataStore for real, deactivate this indicator.

   > **Note**
   >
   > If the *Test Run* indicator is activated, the system does not delete any data. If you reorganize the DataStore for real, then any data that has no counterpart in the SDL will be deleted from the index.

4. Choose *Execute*. The system displays a list of the reorganized inconsistent object versions in the DataStore.

3.4.5.10 Editing Index Update Logs

Use

Indexes are updated to the DataStore before the corresponding data is posted to the SDL. This means that in rare cases (where the process terminates, for example), you might have a situation where data that is in the index is not written to the SDL. You use this transaction to check the consistency of the index data in the core Bank Analyzer system and the Tool BI system.

> **Note**
>
> This problem only arises if the Tool BI system is accessed via Remote Function Calls (RFCs) from the core Bank Analyzer system for the purpose of updating indexes. In other words, this problem will not occur if the DataStore objects are accessed locally. If you use a local Tool BI system, no logs are written.

Features

- A log records the index update process, when the Tool BI system is accessed via RFC. This log contains the keys of all the primary objects that were processed, as well as an indicator showing whether the process was completed without any errors.
  You cannot deactivate this log function.
- You do not need to take any further action for index update runs whose logs have an *Error-free Run* indicator. You should, however, delete these logs on a daily basis (as a batch job).
- You need to regenerate the index for primary objects whose logs do **not** have an *Error-Free* indicator. You can do this directly from the log function. If you regenerate the index successfully for such a log, it is flagged as *Regenerate OK*. You can delete this log later since it is now error-free.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Index Tables* > *Edit Logs for Index Update*.
2. Specify the period you want to check. Logs from this period will be checked. You can restrict your selection even further by, for example, specifying a particular user or transaction code as selection criteria. To restrict your selection further, you can also enter any of the following values for the *Logs: Error-Free Runs* and *Logs with Error-Free Regeneration* parameters:
   - "X" = Only logs that are error-free
   - "-" = Only logs that contain errors
   - " "= Undetermined
3. In the *Processing Types* box, specify whether you wish to select, delete, regenerate, or reconcile logs.
4. Choose *Execute*. The system displays a list of the logs that were selected/processed.
5. In you selected the processing type *Selection of Logs*, you can execute the following functions directly from this list:
   - You can display the *log data* for a log.
   - You can *delete error-free logs*.
   - You can *delete a selected log*.
   - You can *regenerate* the index for a selected log.
   - You can *reconcile* the SDL with the index for the selected log.

3.4.5.11 Specify Size Category for Content of DataStore Objects

Prerequisites

You have made default settings in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Index Function* > *DataStore Objects* > *Register OTS/DataStore Objects*.

Context

You use this transaction to estimate the number of packages for the parallel reading (not sequential) of InfoSets. The estimated number of packages determines the size of the package, in other words, the number of data records per package. If you select an applicable size category, packaging is adapted to suit the package size that you have specified. As a result, performance is improved.

> **Note**
>
> This size category comprises the complete dataset of all packages for the selection of the parallel processing currently being used. The size category does not represent the number of data records in the DataStore object.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Tools* > *Index Tables* > *Specify Size Category for Content of DataStore Objects*.
2. Assign an appropriate size category to the respective components, subcomponents, and DataStore objects. You have the following selection options:

Size Category	Description
To 10,000	Up to 10,000 data records per selection
To 500,000	Up to 500,000 data records per selection
From 500,001	Over 500,000 data records per selection

> **Note**
>
> If you do not enter a size category, the system takes *From 500,001* as a standard value. You can change the estimated size category in the application menu.

1. Save your entries.

3.5 Connection of Operational Systems to Bank Analyzer

Use

You can use this component to asynchronously connect operational source systems to a system (such as Bank Analyzer), where you analyze the data. This component is based on enterprise service-oriented architecture (enterprise SOA). Financial source data is stored in Bank Analyzer. The event-oriented and message-based interface can be used to integrate SAP systems and non-SAP systems. It can integrate different versions and different programming languages (such as ABAP or Java).

This integration scenario supports the "checking account", "deposit account'", and "loan" financial transactions, and the "payment", "settlement date", and "accruals and deferrals" business transactions.

Implementation Considerations

You can make the required settings in Customizing for *Bank Analyzer* under *Source Data Layer* > *Interfaces to Other Applications* > *Interfaces to Account Management*.

Features

SAP implements this application-to application process (A2A process) using SAP NetWeaver Exchange Infrastructure (SAP XI) as part of SAP NetWeaver.

The complex, generic, and flexible data model used for objects in Source Data Layer (SDL) of Bank Analyzer requires a multi-level mapping process. For this reason operational source systems are connected to Bank Analyzer across multiple steps, software layers, and data structures.

More Information

For more information about XI structure mapping, see XI Structure Mapping [page 1735].

If you use Account Management (FS-AM) as your account-managing system, you can find more information under .

Some financial products, for example current accounts and savings accounts, result in a high processing volume. This can have an impact on performance. Therefore, you can improve performance by aggregating these products before the analytical process. For this reason, you are able to aggregate business transactions, financial transactions, and accrual results in the Source Data Layer before further processing takes place. For more information, see Source Data Aggregation [page 286].

You can use enterprise services for the following business objects:

- Analytical View of Loan Contract
- Analytical View of Operational Financial Instrument Flow Transaction
- Analytical View of Current Account Contract
- Analytical View of Deposit Account Contract
- Financial Instrument Position Accrual

For more information, see http://help.sap.com ▶ *SAP for Industries* ▶ *SAP for Banking* ▶ *banking services from SAP* ▶ *Enterprise Services Documentation* ▶.

3.5.1 Processing the Service Operation in Software Layers

Use

You can use this process to import data from operational feeder systems to Source Data Layer (SDL) in Bank Analyzer. A conversion process is used to fill the data of the objects into different structures and to pass it through different software layers.

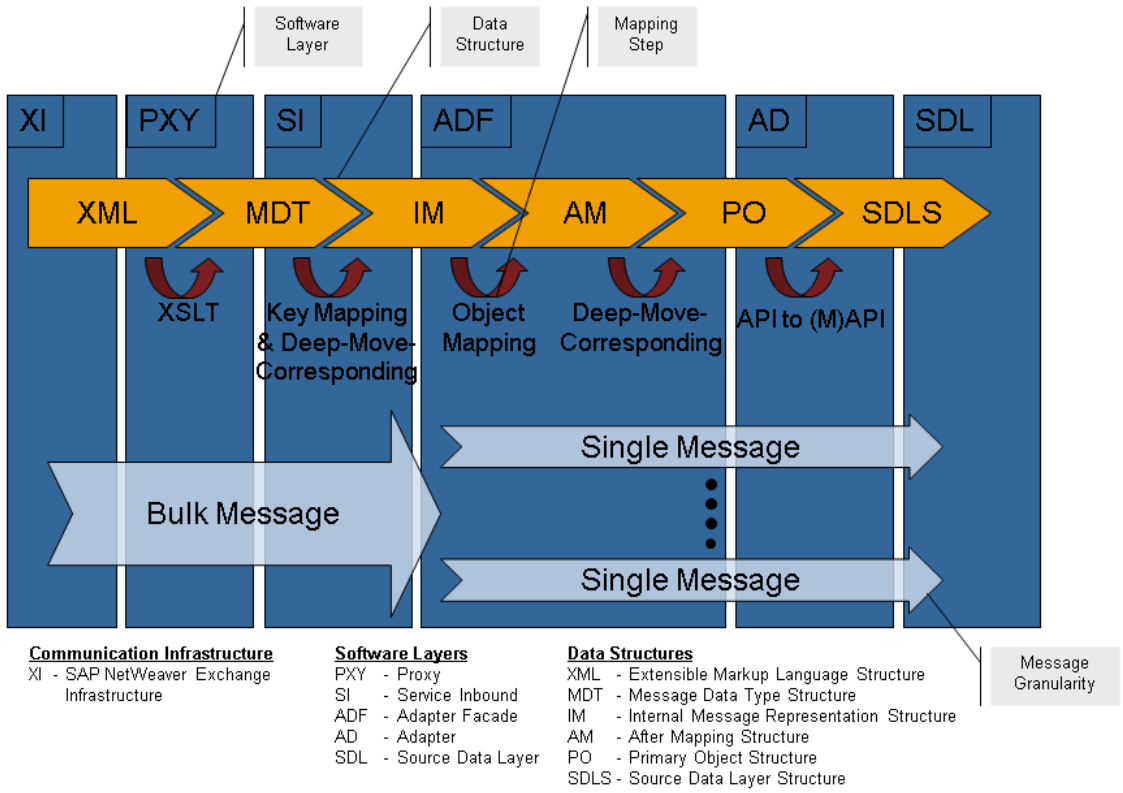
Here, an analytical service operation defines a concrete activity in the system (for example, 'Create Current Account'). This analytical service operation represents a service operation from the point of view of the analytical application. It is part of an analytical service interface that groups together the analytical service operations. An analytical business object implements analytical service interfaces by using the analytical service operations that are grouped there.

> **Note**
>
> You can display all available analytical service interfaces and their corresponding analytical business objects in maintenance view `/BA1/VFB_ABO_SI`. You can display the analytical service operations available for the analytical service interfaces in maintenance view `/BA1/VFB_ABO_SO`.

Process

Operational source systems are connected to Bank Analyzer across multiple steps, software layers, and data structures.

The complete conversion process runs through the following layers one by one:



Overview of Data Flow

1. SAP NetWeaver Exchange Infrastructure (SAP XI)
   SAP XI links cross-system business processes.
2. Proxy
   Proxies implement the message exchange in the proxy layer.
3. Service Inbound Layer (SI) [page 359]
4. Adapter Facade [page 360]
5. Adapter [page 361]
6. Source Data Layer (SDL) [page 100]
   SDL is the target of the complete mapping process. It manages master data and flow data for financial products. SDL mirrors the data of the operational source systems from an analytical point of view.

The objects receive different data structures [page 357] in each of these layers:

- Extensible Markup Language Structure (XML Structure)
- Message Data Type Structure (MDT Structure)
- Internal Message Representation Structure (IM Structure)
- After Mapping Structure (AM Structure)
- Primary Object Structure (PO Structure)
- Source Data Layer Structure (SDL Structure)

The messages are transferred through the different software layers to the adapter facade in one bulk message. From the adapter facade onwards, the system processes each single message separately.

3.5.1.1 Data Structures

Concept

The sender system delivers the messages in an XML data structure that is gradually converted in different mapping processes to ABAP dictionary structures that are compliant with Source Data Layer (SDL). The following dictionary structures are involved in the conversion process of a message:

Message Data Type Structure (MDT Structure)

The services available are defined in Enterprise Services Repository (ESR). XML is used to call the services and deliver data. In the proxy software layer, the system uses Extensible Stylesheet Language Transformation (XSLT) to convert this XML data to corresponding data types that are defined in the ABAP Dictionary. These generated ABAP Dictionary representations of the ESR service definitions are called message data types (MDT).

Internal Message Representation Structure (IM Structure)

The IM structure represents the relevant content of the message. The content of the IM structure fields has already been converted from the format of the global data types (GDT) to ABAP format. In the GDT conversion, the system converts the special field formats of the message (GDT format) to ABAP format. For example, the system converts the field format of the timestamp from `STRING` to `DATE/TIME`.

Example

The following are examples of the IM structures that are used:

- `/BA1/F8_STR_IM_LOAN` for creating and changing loans
- `/BA1/F8_STR_IM_CURACC` for creating and changing current accounts
- `/BA1/F2_STR_IM_ACCOUNT_ITEM` for creating and reversing payment items
- `/BA1/F2_STR_IM_SETTLEMENT` for creating and reversing settlement data

Each IM structure has a **key part** and a **data part** of the message. The system fills the key part in the service inbound layer [page 359] during key mapping. It contains all Bank Analyzer key fields, such as the external object ID, the business record date, and the source system. It also contains other object reference keys as they are stored in SDL.

The data part of the IM structure contains the data for the object mapping process that is subsequently run in the adapter facade [page 360].

You can extend IM structures in the ABAP Dictionary using append structures.

After Mapping Structure (AM Structure)

The system fills AM structures during the object mapping process using IM structure data and the defined mapping rules. They contain all of the fields required for creating the primary object in SDL.

Example

The following are examples of the AM structures that are used:

- `/BA1/F8_STR_AM_LOAN`
- `/BA1/F8_STR_AM_CURACC`
- `/BA1/F2_STR_AM_ACCITEM`
- `/BA1/F2_STR_AM_SETTLEMENT`

Each AM structure contains a **data part** and a **part with additional settings**. The data part is a semantic representation of the primary objects (such as loan, current account, and business transaction), but it is already in the data format of Bank Analyzer. The data model of the AM structure depends on the Customizing settings you have made: For the financial transaction, it is based on the template you have defined, and for the business transaction, it is based on the business transaction class used.

You can extend AM structures of the financial transaction in the ABAP Dictionary using append structures.

The part with additional settings contains controlling fields that the system fills during the object mapping process, and can consider during subsequent processing. All AM structures use the same substructure for this (`/BA1/FP_STR_INBOUND_SETTINGS`). This substucture contains general properties that are relevant for the import of data. This includes the following parameters:

- *Test run*
  The system imports the data in the test run, but it does not post the data.
- Ignore processing of an object (*ignore object*)
  If you want to suppress the processing of an object because, for example, irrelevant objects will be imported, you have to set this indicator. This allows the system to filter objects and to thereby improve system performance. As no further processing occurs, Error and Conflict Handler [page 362] ignores this behavior.

Primary Object Structure (PO Structure)

The PO structures contain all nodes and service modules of the financial transaction, or all possible fields of a business transaction that are used in the AM structures.

In contrast to the AM structures that represent each product or flow in a separate semantic structure, the PO structures map a primary object in SDL.

Example

The following are examples of the PO structures that are used:

- `/BA1/F8_STR_PO_FT` for the financial transaction
- `/BA1/F2_STR_PO_BT` for the business transaction

You can extend PO structures of the financial transaction in the ABAP Dictionary using append structures.

Source Data Layer Structure (SDL Structure)

This structure represents the existing SDL format of the function modules of the Main Application Programming Interfaces (MAPI) and the Application Programming Interfaces (API).

More Information

Enhancement of Structures [page 363]

3.5.1.2 Service Inbound Layer

Use

The 'service inbound' software layer (SI) receives messages that are transferred, for example, using SAP NetWeaver Exchange Infrastructure (XI). In this software layer, the system converts the fields of the key data and reference data for the associated object using key mapping. In addition, the system converts the global data types (GDT conversion) of the message data and then transfers this data to the next software layer. Therefore, the service inbound layer is the internal representation of the services.

Features

In the service inbound layer, the system creates a deep ABAP structure, which is called the internal message structure (IM structure) from the message data type structure (MDT structure) of the inbound message. The system has previously converted the message from XML format in the proxy layer.

There are no fields in Source Data Layer (SDL) that correspond to the key data and reference data that are in MDT format in the message. Key mapping maps this data to keys and references that can be clearly recognized by SDL. GDT conversion converts special data types of the message (for example, timestamps) to the internal ABAP format in the service inbound layer.

If you want to use your own key mapping and not that predefined in the system, you can assign your own mapping strategy in Customizing for the analytical service operations. For more information, see the Implementation Guide (IMG) of *Bank Analyzer* under *Source Data Layer* > *Interfaces to Other Applications* > *Interfaces to Account Management* > *Assign Mapping Tools to Service Operations*.

The proxy layer delivers the content of the message to the generated proxy class in the service inbound layer in MDT format. The service inbound software layer consists of one generated proxy class per analytical service interface. The generated proxy class contains different methods that implement the analytical service operations of the interface. These methods also run a validation and the GDT conversions, as well as calling key mapping. The service inbound layer then transfers the converted structure to the adapter facade [page 360].

If fundamental errors are found (for example, if the external ID is missing), the service inbound layer rejects the complete bulk message. This is the case even if the error is in one object only, in other words in a single message. For more information, see Handling Errors and Conflicts [page 362].

3.5.1.3 Adapter Facade

Use

The 'adapter facade' software layer receives the bulk message including the key and reference fields, and the data part of the message as an internal message structure (IM structure). The system splits the bulk message into single messages to further process it.

Each object type has its own method in the adapter facade. These methods are called from the service inbound layer [page 359]. For example, the financial transactions 'loan' and 'current account' have their own methods, as do the business transactions 'payments' and 'settlement data'.

Prerequisites

You can use the Module Editor delivered by SAP for object mapping, or you can implement your own mapping tool. In the latter case, you can develop the content of the object mapping process yourself in this tool.

If you use the Module Editor delivered by SAP as a mapping tool, you have made the relevant settings in Customizing for *Bank Analyzer* under *Source Data Layer* > *Interfaces to Other Applications* > *Interfaces to Account Management* > *Mapping Tool* > *Module Editor*. You can create your own functions and assign them to the modules, and you can define the object mapping process in individual business objects and assign them to the Module Editor as a mapping tool.

If you want to use your own mapping tool, you have made the relevant settings in Customizing for *Bank Analyzer* under *Source Data Layer* > *Interfaces to Other Applications* > *Interfaces to Account Management* > *Mapping Tool* > *Register Mapping Tools* and *BAdI: Implementation of an Alternative Mapping Tool*.

Features

In the adapter facade, the system calls a sequence of special methods (API sequence) for each single message:

- Target object: Financial transaction
  1. Existence check
  2. Call of object mapping
  3. Filling of the primary object structure (PO structure) from the after mapping structure (AM structure) for transfer to adapter
  4. Call of adapter
- Target object: Business transaction
  1. Decision about posting type: Preliminary posting, or final posting
  2. Decision about use of suitable methods in the adapter facade: Create, or reverse
  3. Call of object mapping
  4. Filling of PO structure from AM structure for transfer to adapter
  5. Call of adapter

Object Mapping

Object mapping converts the object from the cross-system view (IM structure) to the view of the analytical system (AM structure). Write-protected constant structures stand for business transactions (for example, `/BA1/F2_STR_OM_CONSTANTS_BT`) and for financial transactions (for example, `/BA1/F8_STR_OM_CONSTANTS_FT`) in object mapping. The constants structure for financial transactions contains information about the template, the template version, and about the node number, in other words the information about how the Source Data Layer (SDL) is to store a financial transaction.

In the Module Editor, conversion is supported by specialized "mini-transformation APIs", and by the call of modules from the derivation tool for value mapping.

> **Note**
>
> The basic settings for master data are globally applicable and cannot be overwritten in object mapping.

Deep-Move-Corresponding

The 'deep-move-corresponding' tool transports data from AM structures to PO structures. One function of the tool that corresponds to the `MOVE-CORRESPONDING` ABAP command can transfer the data of a source structure to a target structure field by field. In addition, 'deep-move-corresponding' transfers table types line by line using `MOVE-CORRESPONDING` and can convert the data of a flat structure to a table entry.

If an enhanced structure [page 363] has a prefix before the name of a component, the 'deep-move-corresponding' tool ignores this prefix and can, despite the different names, merge the relevant fields of the source and target structure.

3.5.1.4 Adapter

Use

The 'adapter' software layer receives primary object structures (PO structures) that are imported by the adapter facade [page 360]. In this software layer, the system converts PO structures to Source Data Layer structures (SDL structures).

Features

Each primary object (financial transaction and business transaction) has an adapter.

- In the adapter for financial transactions, the system converts the complex PO structures to flat parameters of the SDL MAPIs that already exist (SDL transformation).
  In this adapter the system also fills the MAPI parameters that control the creation of an operational event [page 258]. When master data is changed, and provided the appropriate Customizing settings have been made, the system simultaneously creates an operational event in the form of a business transaction in this adapter.
- In the adapter for business transactions, the system calls SDL Application Programming Interfaces (APIs) and converts the complex PO structure to complex parameters of these SDL APIs.

The system does not automatically convert enhancements in PO format (PO structure data) to MAPI format and vice versa. You can use a Business Add-In to convert enhancements in PO structures. For more information, see Enhancement of Structures [page 363].

All messages rejected by the MAPI are collected and categorized by Error and Conflict Handler [page 362]. Every MAPI error message can be seen in Error and Conflict Handler.

3.5.1.5 Handling Errors and Conflicts

Use

This process allows you to collect and classify errors and conflicts that occur when the system transfers messages and converts data formats. This simplifies the correction process. In asynchronous communication, the recipient of the messages handles the errors and conflicts because the sender does not receive confirmation of the transfer of the messages.

Errors and conflict are handled differently based on where the errors occur. You can use Integration Engine to monitor the status of inbound messages and, after you have corrected them, reschedule them for processing. Error and Conflict Handler groups all errors and conflicts that occur in the subsequent software layers according to error symptoms. You can then analyze the incorrect data and either process it manually, or trigger automatic processing.

Prerequisites

You have made the following Customizing settings:

- *Cross-Application Components* > *General Application Functions* > *Postprocessing Office*
- *Bank Analyzer* > *Source Data Layer* > *Interfaces to Other Applications* > *Interfaces to Account Management* > *Handling of Errors and Conflicts*
- *Bank Analyzer* > *Infrastructure* > *Settings for XI Services* > *Error and Conflict Handler*

Process

1. If errors occur during the conversion of the XML format of messages (Extensible Stylesheet Language Transformation, XSLT) in the proxy and in the key mapping process in the service inbound layer [page 359], the system reports these to Integration Engine.
   You can use transaction `SXMB_MONI` (*Integration Engine – Monitoring*) to monitor the processed messages and to correct the errors.
2. If errors occur in Source Data Layer (SDL) or in the object mapping process, the system collects these errors in the adapter facade [page 360] and then categorizes the errors.
3. Error categorization uses hierarchical derivation trees of all messages in the message table to derive a main error message and its error category. Based on this main message, Error and Conflict Handler can further process the error messages.

4. The system transfers the error messages to Error and Conflict Handler via the service inbound layer.
5. Error and Conflict Handler decides on a resolution strategy for the errors and conflicts and uses the to store all data relevant for postprocessing in a postprocessing order.
6. You call the *Postprocessing Office* to manually correct incorrect data or to trigger further processing automatically.

   On the *SAP Easy Access screen*, choose *Cross-Application Components* > *Postprocessing Office*. In the *Edit Postprocessing Order* transaction (`/SAPPO/PPO2`), you can select error messages by error category and edit them, for example.

3.5.1.6 Enhancement of Structures

Use

You can use this function to extend the data model of the objects that are involved in the integration process. Reasons for extending the data model include:

- You want to use fields in the operational system for valuations or accounting processes in the analytical system. These fields are not in the messages that transfer the data to the analytical system.
- There are fields that exist in the operational system and in the corresponding messages, but that are not considered relevant in the analytical system. You want to use these fields in the analytical system.
- The data model of the operational system has itself been extended. The analytical system now has to consider these extensions.

You can use this function to extend all of the follow-on functions of messages that have been extended. You extend data structures in Bank Analyzer using append structures.

> **Note**
>
> The hierarchy of the primary object created is defined for a particular business object. You cannot change or extend this hierarchy.

Integration

If you extend the primary object structure delivered for the 'financial transaction' primary object by service modules using an append structure, you implement Business Add-In (BAdI) *Field Transport Between SDL and PO:Representation of Financial Trans.* (`/BA1/BADI_F8_FT_SDL_TRANSFORM`). The BAdI converts the extended primary object structure to the Source Data Layer structure (SDL structure).

For the creation of the operational event [page 258], you can, in addition, transfer other characteristics in the business transaction header and business transaction item to the Main Application Programming Interface (MAPI) of the SDL.

For more information, see the Implementation Guide (IMG) for *Bank Analyzer* under *Source Data Layer* > *Interfaces to Other Applications* > *Interfaces to Account Management* > *Enhancement Concept* > *BAdI: Field Transport Between SDL and PO:Representation of Financial Trans.*.

The 'deep-move-corresponding' tool also automatically transports extended data structures during the conversion of message data type structures (MDT structures) to internal message structures (IM structures), and of after mapping structures (AM structures) to primary object structures (PO structures). For more information about the 'deep-move-corresponding' tool, see the *Features* section in Adapter Facade [page 360].

Features

You extend **IM structures** in the same way as you extend messages, using append structures and prefix `ZZ`.

However, if a message is extended, you do not necessarily always have to extend the AM structures or the PO structures. If the target field is a characteristic or a key field, you can easily add this field in Customizing for object mapping.

You can extend **AM structures** as follows:

- You can add fields for existing service modules. If you add fields, this does not affect the PO structure because the PO structure is the set of the corresponding AM structures.
  The following example demonstrates a case where you have to extend the AM structure at field level:
  In Customizing for the template, you activate a field in a service module that is supported (for example, a field in the basic data) and this field does not exist in the AM structure. An append structure is used to make the extension. The field name is made up of `ZZ` and the API field name of the Source Data Layer (SDL). The data type has to be identical to the data type of the API field.

  ### Example

  You want to add fields `NOTICE_PERIOD` (*notice period*) and `PERIOD_UNIT` (*Time Unit for Notice Period*) of service module *Basic Data* to the root node of the current account.

  1. In Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Master Data* > *Templates* > *Edit Template Version*, activate the fields in template `S40CURRACC`.
  2. Create an append structure for AM structure `/BA1/F8_STR_AM_CURACC_ROOT_BAS`.
  3. Add field `ZZNOTICE_PERIOD` and data element `/BA1/F1_DTE_NOTICE_PERIOD`.
  4. Add field `ZZPERIOD_UNIT` and data element `/BA1/F1_DTE_PERIOD_UNIT`.
  5. Add the appropriate entries for the derivation of these fields in object mapping for the current account.

- You can add complete service modules. You then have to extend the **PO structure** as appropriate.
  The following example shows a case where you have to extend the AM structure and the PO structure at the level of the service module:
  In Customizing for templates, you activate fields (for example, rating data for loans) in a service module that is not yet supported by the system. In this case, you extend the complete structure that is represented by the service module data, or the table type. The field name is made up of `ZZ` and the three-character API field name for the service module in SDL. The data type has to be identical to the API data type of complex key figure `/BA1/F1_STR_CONTR_40` for the corresponding service module.

  ### Example

  You want to add the fields of the *Ratings* service module to the root node of the loan:

  1. In Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Master Data* > *Templates* > *Edit Template Version*, activate the fields in template `S40CASHFLOW`.

2. Create an append structure for AM structure `/BA1/F8_STR_AM_LOAN_ROOT`.
3. Add field `ZZRAT` and component type `/BA1/F1_TAB_CONTR_RAT_DATA`.
4. Check whether PO structure `/BA1/F8_STR_PO_FT` already contains the service module `RAT`.
   If this is this case, you do not need to extend the PO structure. If this is not the case, add append structure field `ZZRAT` to the PO structure as described above.

4 Processes & Methods (FS-BA-PM)

You can use this component to carry out all financial and risk calculations for Bank Analyzer. Unlike Methods, Processes combine the selection, checking, and processing of data into one step.

The system generates the calculation results using either original data from the Source Data Layer (SDL) [page 100] or existing results data. Existing results data comes from either source systems or previous calculation steps. The system then stores data that has been completely valuated in the Results Data Layer (RDL) [page 1364].

General Calculation and Valuation Methods (FS-BA-PM-GM)

General calculation and valuation functions provide you with various methods for upstream processing. Various Bank Analyzer components can use the results data from this method.

Determination of Net Present Values and Calculation Bases (FS-BA-PM-EIC)

You use this process to calculate net present values and other key figures that you can use as input for calculating funding costs and standard costs. For example, the component calculates funding results, standard cost rates, and the effective capital over time.

Accounting for Financial Products (FS-BA-PM-AFP)

Accounting Processes

Accounting processes comprise business transaction processing and financial position management in Accounting for the subledger scenario.

Impairment Processes (FS-BA-PM-IP)

Impairment processes provide functions for measuring and recognizing the impairment of financial transactions and financial instruments in accordance with IFRS.

Hedge Processes (FS-BA-PM-HP)

Hedge processes provide various functions for IAS and Basel II. In particular, you can use these service functions for key date valuations and hedge accounting.

Credit Risk (FS-BA-PM-CR)

Credit risk provides up-to-date control instruments for the simulation, planning, and analysis of the overall bank with its different levels. Risk management reflects the reporting obligations imposed by the banking supervisory authorities.

4.1 Set SDL Time Stamp

Use

The system uses both business times and a technical time stamp to ensure that data is read consistently. If you specify an SDL time stamp for processes, the system reads only the data with a time stamp that is before or identical to the SDL time stamp specified. This ensures that the system does not access new data versions when rereading data.

You can set the SDL time stamp for different processes, including accounting processes and cost accounting processes.

The SDL time stamp is not relevant for the processes in .

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

The function for setting the SDL time stamp uses the process partition as a generic entity of the CVPM to process financial products at a desired granularity level with the same time stamp. The process partition links processes (such as the SDL time stamp) with leading processes carried out by the system with CVPM. The system always uses the name that you defined. With the process partition model, you can also use the same time stamp for different legal entities.

Activities

1. On the *SAP Easy Access* screen, choose *Processes and Methods* > *Set SDL Timestamp*.
2. On the selection screen, make the following entries:
   - Process Partition (required entry field)
     You can make any entry. Note that the accounting processes require a time stamp at the granularity level of the legal entity. You therefore enter the corresponding legal entity in the *Process Partition* field. This also applies for the period-based calculation of internal costs.
   - Reference Partition (optional field)
     When you carry out adapter processes and want to refer to the time stamp of a preceding process, enter the process partition of the preceding run as the reference partition.
   - Date of the system time for source data and market data (optional)
     Enter a time stamp. If you do not enter anything, the system enters the time at which the function is called as the time stamp. This means you can also schedule this process in the Schedule Manager. For example, the system might then automatically set the time stamp after the end of an aggregation process.
   - Technical Settings
     You can call up filter variants or specify that the SDL time stamp is to be set in a test run.
3. Choose  .

> **Note**
>
> To display the relevant SDL time stamp from the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the *Set Time Stamp* process (`/BA1/RB0_CLOSE_SDL`), and display the application log for the required run. This log contains the SDL time stamp entry.

Example

For more information about the significance of the SDL time stamp for versioning SDL data and a corresponding example, see Primary Object Versioning [page 103] and Example: Versioning [page 105]. The system time referred to there is determined by setting an SDL time stamp.

4.2 Application Event Management

Use

This component connects the Source Data Layer (SDL) that supplies operational events [page 258], and the Results Data Layer (RDL) that imports analytical events, with processes that are managed using application events [page 369]. Application event management generates application events from operational and analytical events, saves the application events in the database, and provides them to requesting applications. An example of an application is the determination of net present values and calculation bases.

The monitor for application events [page 370] provides you with an overview of the existing application events and their properties. In addition, it allows you to change the delivery status and to reverse application events.

Implementation Considerations

In Customizing for *Bank Analyzer* under *Processes and Methods* > *Application Event Management*, you have executed the following Customizing activities:

- *Define Application Event Types*
- *Edit Modules for Application Events*
  This is where you can restrict the derivation of application events to specific business transactions, product types, or source systems, for example.

  > **Note**
  >
  > This Customizing activity is optional.
- *Derive Application Events from Operational Events*
- *Derive Application Events from Analytical Events*
- *Define Derivation Relevance of Application Events*
- *Edit Generation Information for Application Events*

4.2.1 Application Event

Definition

An event that is derived from an operational event [page 258] or an analytical event [page 689] so that it can be saved in Application Event Management [page 368] and processed by certain applications.

Typical examples of application events are the issuing of a loan (commitment costing), a rollover, or a cash flow disturbance.

Use

You can display and reverse existing application events in the monitor for application events [page 370], and also change the delivery status manually.

The system derives separate application events for each application, meaning that for each application there can be one or more application events for one operational event or one analytical event. For example, issuing a loan is relevant for both accounting and profit center accounting, which is why separate application events are derived for the corresponding components.

The system derives application events from operational events only from the items of the business transaction that are marked as relevant for an operational event (*OE-Relevant*)

The application events are named differently in accordance with the application that they are processed by:

- Accounting event
  These events are processed by the Accounting Processes [page 693] application.
- Impairment event
  These events are processed by the Impairment Event Processing [page 966] application.
  When it processes impairment events, this application creates new analytical events (for example, change of the impairment status) from which application event management derives application events for the accounting processes.

Structure

Application events contain either references to the business transaction used to map the corresponding operational event or references to an analytical event in the RDL. Amongst other things, they contain the following fields that are relevant for further processing steps:

- Fields that are specific to the application event
  - Application
    Process that processes application events, for example the impairment and calculation of internal costs for periods.
  - Application event type
    Groups together application events that relate to the same type of operational event and that are processed in the same way.
  - Application event category
    Defines how application events are processed.
- Fields from the operational event
  - Business transaction ID
  - External number
  - Source system
- Fields from the analytical event
  - ID of an analytical event
- Fields with status information about the application event
  - Status of reversal
  - Status of activation
  - Delivery status

In addition, further administrative information is provided, for example the creation time stamp.

4.2.2 Monitor for Application Events

Use

This function allows you to display and reverse application events [page 369], and to change their status. This enables you to follow the processing of application events and take action if required. For example, you can define

manually which application events you do not want to be delivered to applications requesting them, or which application events you want to be redelivered. This means that you can prevent or repeat the processing of individual application events when errors occur.

The monitor for application events is part of Application Event Management [page 368].

Prerequisites

If an event affects several legal entities, the monitor displays the break-down in legal entities. You have made the settings for this in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Break Down Business Partner into Financial Transactions & Fin. Instrument*. For more information, see the documentation for this Customizing activity.

Example

If a business partner rating changes, the system generates the operational event *Business Partner Change* and derives an application event from this. The *Process Impairment Events* process determines which financial transactions from which legal entities belong to the changed business partner.

Features

The monitor for application events displays the properties of an application event.

Change the Status

The system automatically sets the status for delivery, reversal, and activation of application events. In addition, you have the following options for changing the status manually:

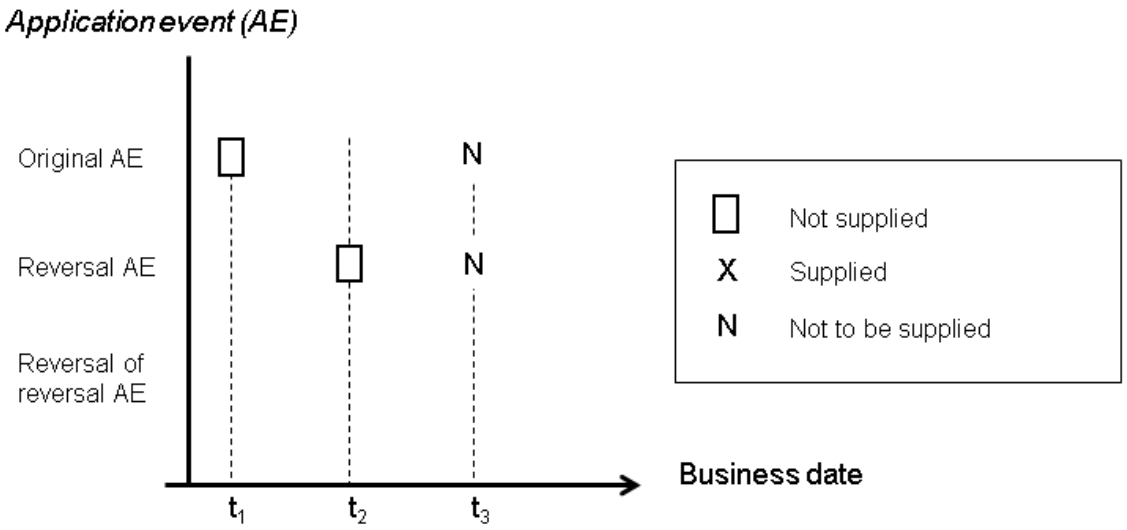
Function	Result
	The system sets the status of the application event to *reversed* and generates a reversal application event.
	The system sets the status to *supplied*.
	The system sets the status to *not supplied*.

Reverse

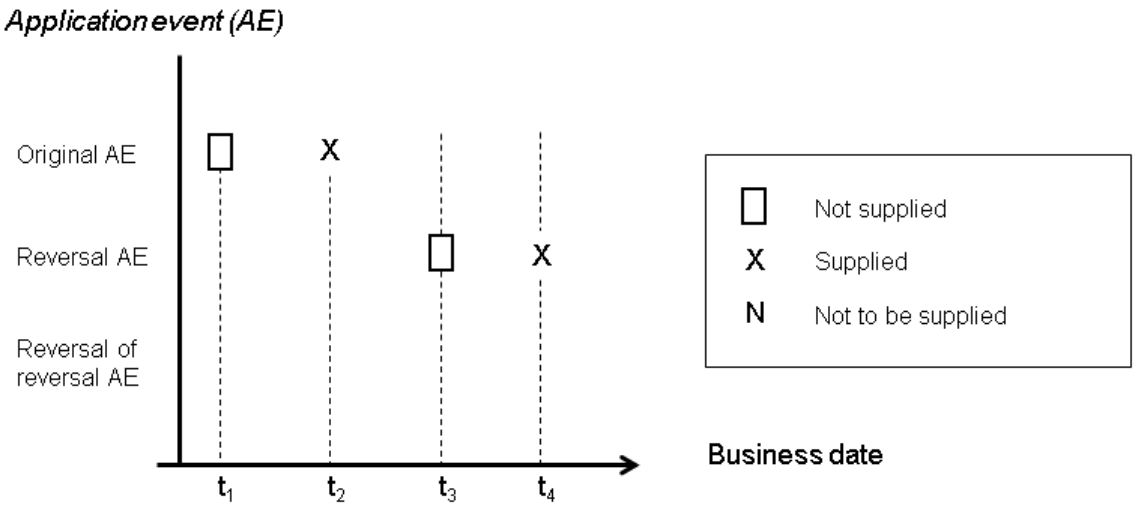
When you reverse an application event, the system sets the status to *Reversed* and generates a further application event as a reversal.

Certain applications, such as the accounting processes for calculation of internal costs for periods and determination of net present values and calculation bases, do not process application events with their reversal if

they are **not yet supplied**. The original application event and the reversal application event are both assigned the delivery status `N` (*Not to Be Supplied*). The situation described above is illustrated in the following figure:

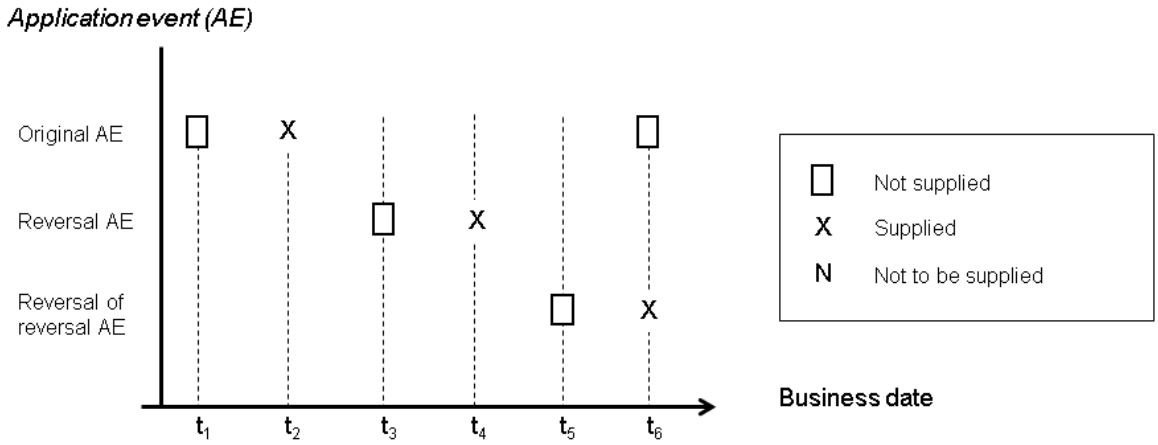


If the applications for the application event being reversed have already been processed, you must also retrieve and process the reversal application event. Both application events are assigned the delivery status `X` (*Supplied*). The situation described above is illustrated in the following figure:



You can reverse a reversal application event, which results in the original application event no longer counting as reversed. If the first reversal application event has already been processed, the system sets the original application

event back to *Not Supplied*. After it has been processed, the second reversal application event is assigned the status X (*Supplied*). The situation described above is illustrated in the following figure:



If the first reversal application event has not yet been processed, the system sets both reversal application events to N (*Not to Be Supplied)*.

You cannot reverse the second reversal application event. You can only execute a third reversal by reversing the original application event again.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Application Event Management* > *Manage Application Events*.
2. On the selection screen that is shown, you can define which application events you want the system to display. For example, you can search for application events of a particular business transaction ID.
3. Choose .
   In the monitor for application events, the system displays a table with the selected application events.
4. You can display and reverse application events, and change their delivery status.

Example

An application event has been supplied to an application, therefore its delivery status is set to *supplied*. However, errors occurred during processing in the application. As a result, you want to redeliver the application event. Select the application event and choose *Redeliver*. This prompts the system to set the delivery status to *Not Supplied* so that the calling system can reprocess the application event.

4.2.3 Regeneration of Application Events

Use

This function allows you to re-trigger the derivation of application events [page 369] from an operational event [page 258] that has already been derived. A new business transaction with operational event is not created in the process, rather the derivation takes place from the same operational event.

You require this function if the derivation of application events from operational events is incorrect and you change the settings for this in Customizing. You use this function to reverse the application events that have already been derived and generate new application events from the same operational event.

Prerequisites

You have configured the derivation of application events from operational events in Customizing for *Bank Analyzer* under *Processes and Methods* > *Application Event Management*.

Application events are derived from the operational event.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Application Event Management* > *Regenerate Application Events*.
2. Choose the business transactions with operational event for which you want to trigger the derivation of application events again and choose .
   You can select the business transactions via the external key or via header characteristics.
3. The Source Data Layer (SDL) transforms the business transactions into a complex key figure and sends them to Application Event Management via an interface.
4. The system reverses the derived application events of the selected business transactions in Application Event Management.
   The business transaction itself remains unchanged.
5. Using the current derivation rules, the system rederives the operational events mapped via the selected business transactions.
6. Application Event Management saves the resulting application events.
   You can then display the application events in the monitor for application events [page 370].

Example

You have created a business transaction with operational event OE1 and derived application events AE1 and AE2 from OE1. AE1 is available to application W, AE2 to application X.

You change the derivation rules in Customizing and then regenerate the application events for OE1.

Application Event Management reverses the application events AE1 and AE2 and their entries in the worklists, and derives the new application events AE3 and AE4 from OE1 using the new derivation rules. These new application events are now available to applications Y and Z.

4.2.4 Processing of Application Events

Use

This function generates a worklist from the active application events [page 369] for further calculations in the respective application. You can choose the application events on a selection screen using process-specific parameters (for example, the legal entity) and make technical specifications, for instance that you want to execute processing as a test run. You then generate a worklist from the chosen application events. A log displays messages so that you can correct any errors that occurred during processing.

Integration

The application events are derived from operational events [page 258] or from analytical events.

Prerequisites

You have checked the status of the application events in the monitor for application events [page 370].

If you use the accounting processes, you have specified how the system processes accounting events when secondary business transactions are updated and, if necessary, you have assigned characteristics for transfer postings for financial positions to an accounting system. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Application Events* > *Accounting Events*.

If you use the determination of net present values and calculation bases, you have defined the calculation steps for application event categories in Customizing for *Bank Analyzer* under *Processes and Methods* > *Determination of Net Present Values and Calculation Bases* > *Define Calculation Steps*.

If you use the calculation of internal costs for periods, you have assigned event processing categories to the application event types in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Application Events* > *Cost Accounting Events* > *Assign Event Processing Categories to Application Event Types*.

Activities

Individual processing exists for each of the application events of the calling applications:

- Accounting events
  On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Automatic Processing* > *Accounting Processes* > *Process Accounting Events*. For more information, see Processing of Accounting Events [page 769].
- Events for net present values and calculation bases
  From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Determination of Net Present Values and Calculation Bases* > *Process Events for Net Present Values and Calculation Bases*.
  Cost accounting events
  On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Automatic Processing* > *Costing Processes* > *Process Costing Events*.
- Impairment events
  On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Impairment Processes* > *Process Impairment Events*. For more information, see Impairment Event Processing [page 966].

4.3 General Calculation and Valuation Methods (FS-BA-PM-GM)

Use

This component contains methods that are used by various Bank Analyzer applications and that are used to process the data for financial transactions, financial instruments, and other primary objects before other analyses can be started.

Features

Bundling Service

Some Bank Analyzer processes do not analyze just individual primary objects but entire bundles of objects that are linked to each other by different relationships. You can use the bundling service to create bundles, regardless of how the objects are linked together technically. You can also adapt the transformation of bundles to the requirements of the applications.

Account Pooling

Account Pooling nets account balances that, for contractual or legal reasons, have to be grouped into one entity. The balances of the contracts in a pooling group are netted off on a defined (main) account.

Free Line for Financial Products

The free line describes the undrawn part of a limit (for example, external credit line for accounts, commitment capital for loans, global limit for traditional off-balance-sheet transactions). You can use the transaction *Determination of the Free Line for Financial Products* to calculate the free line for individual financial transactions. The determination of the free line for financial products is required to calculate the risk provision for the off-balance-sheet part of a receivable under IFRS 9, for example.

Free Line for Facilities

The free line for facilities describes the part of a facility that is not used. The facility itself can be available to one or more business partners.

When you determine the free line for facilities, you can calculate it within a hierarchical grouping of related facilities and a credit commitment is then compared with the (cumulated) drawings. Since risk factors can differ between various hierarchy levels of the facility group, it is important in terms of Basel II to calculate the free line for each commitment in the facility group.

Collateral Distribution

By distributing collateral you can map a complex object, which contains one or more contracts and collateral items, to a number of effective individual contracts.

Determination of Default

You can use Determination of Default to derive a default event as defined in Basel II.

Common Functions

Common Functions comprises the common run administration for the following Bank Analyzer components, and information about postprocessing, archiving, and stress tests:

- Account Pooling
- Determination of the Free Line
- Collateral distribution
- Determination of Default

Cash Flow Generation

A cash flow is comprised of a set of flows, which together depict an individual flow of money (such as disbursement, interest, or repayment). You can use cash flow generation to generate cash flows at the level of an individual node of a financial transaction or a financial instrument. The system generates a cash flow for each node of a financial transaction or financial instrument from a particular business view.

Cash Flow Refinement

The data in the Source Data Layer (SDL) [page 100] is stored in a generic structure that you define in Customizing. To measure financial transactions and financial instruments, the system maps the SDL templates to the valuation structure. This structure cannot be changed by Customizing settings.

The Cash Flow Refinement component contains functions that the system uses to generate the valuation structure and to enable other components to recognize this structure. There are also cash flow refinement methods that you can use to change financial transactions and financial instruments at runtime after they have been selected from the Source Data Layer in order to prepare them for analysis.

Present Value Calculation

You can use the Fair Value Server to calculate the full fair value, hedge fair value, amortized cost,hedge amortized cost, and the effective capital over time. These key figures are required for hedge accounting in the Accounting component, for instance. The Fair Value Server calls the Price Calculator in order to calculate the present value.

You can use the Price Calculator to calculate the present values and other key figures for financial transactions and financial instruments. Key figures include the accrued interest, clean price, and durations, and the Greeks for options: delta, gamma, theta, and rho.

Others

- Fair Value Production Process [page 536]
- Maturity Grouping [page 543]
- Accrual Methods [page 656]
- Impairment Server [page 661]
- Balance Offsetting [page 672]

4.3.1 Bundling Service

Use

There are certain application processes of the Analyzers (the calculation process in Credit Exposure, for example) that do not analyze individual primary objects, but analyze a complete bundle of objects that are linked together by different relationships. These relationships might be object relationships, or hierarchical relationships in the form of structured templates.

> **Example**
>
> Exposures that are securitized by collaterals that are based on different physical assets are one example of a bundle that has various levels.

You use the bundling service to create bundles that are subsequently analyzed in different business contexts. The technical relationships between the objects does not affect the creation of bundles.

Integration

To improve system performance, parallel processing is used during the creation of bundles. However, to ensure that bundles are not processed twice because objects of one bundle are in different runs in parallel, a persistence layer is used (bundle database). The system first saves the bundles that are created in parallel in the persistence layer and then permits the parallel processing of these bundles.

Prerequisites

You have made the settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Bundling Service:*

- Edit Node Types

- Edit Link Types
- Edit Bundle Types

By doing so, you have defined the types of relationships that the system considers in a bundling process, the objects that are read, and the sequence in which the system reads these objects.

> **Note**
>
> You classify the bundle type by means of the bundle category. You can choose between the bundle categories *Master Data Objects and Master Data Subobjects* (00) and *Creation of Bundles Without Deleting Relationships* (01). You can select *Bundle Category* 01 for a bundle type only if you plan to use the creation of bundles as a process step, and you are sure that no object relationships exist that are relevant for the process. The selection of this bundle category improves system performance because there is no search for relationships and the bundles can be directly created using one object each.

Features

The system calls the bundling service from within the application using a selected bundle type and an input list of objects. Based on the Customizing settings for this bundle type, the procedure is as follows:

1. The system identifies all of the objects on the input list.
2. The system identifies all of the objects that have a direct or an indirect relationship to the objects on the input list.
3. The system creates bundles of objects that are related to each other (Bundle Category 00) Note: When Bundle Category 01 is used, the system creates only simple bundles from the input list, and the process ends there.
4. The system provides the bundles that have been determined in a standardized form (bundle ID, table of the objects, table of the links between the objects) and stores them in the bundle database.

The system then further processes the bundles of the bundle database in the application processes of the Analyzers.

> **Note**
>
> You can undo the generation of bundle data on the *SAP Easy Access* screen by choosing *Processes and Methods* > *General Calculation and Valuation Methods* > *Bundling Service* > *Delete Bundle Data*. The run ID that you have to enter in this report refers to the respective evaluation run of the Analyzer where the bundling occurred. You should schedule regular deletions of the bundle data.

You can test the settings you have made in the bundling service at any time. In addition, you can display the results and the intermediate steps of test runs for bundling. To do this, in Customizing for *Bank Analyzer* choose *Processes and Methods* > *General Calculation and Valuation Methods* > *Bundling Service* > *Test Creation of Bundles*.

You can transform bundles to adapt them to the requirements of other application processes. For more information, see Bundle Transformation [page 380].

4.3.1.1 Bundle Transformation

Use

Bundle transformation adjusts bundles in the bundling service to the conditions of the application in question. Application processes in the Analyzers that use the bundling service can process bundles of objects only if the structures of these objects are standard. This applies to the following applications:

- *Credit Exposure Calculation*
- *Determination of the free line for facilities*
- *Account Pooling*
- *Free Line*
- *Collateral Distribution*

Convert bundles to the standard form before you process them further.

Prerequisites

You have made the settings required in Customizing for Bank Analyzer under *Processes and Methods* > *General Calculation and Valuation Methods* > *Bundling Service*.

- Edit Transformation Rules
- BAdI: Implementation for Bundle Transformation

To use the bundle transformation function in Credit Exposure Calculation, if required you need to have made the settings in Customizing for Bank Analyzer under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *General Settings* > *Selection* > *Settings for Bundle Transformation* > *Assign Hierarchical Link Types to Data Read Modules*.

Features

The system calls bundle transformation from the application in question and does the following:

1. Reads all the bundles in the bundling service
2. Derives the transformation rule
   The system uses the link types and the bundle type in the bundle to find the transformation rule. There may be more than one transformation rule for a bundle, which means that different transformations can be applied to a bundle in turn. However, you can create only one transformation rule for a given combination of trigger link type and bundle type.

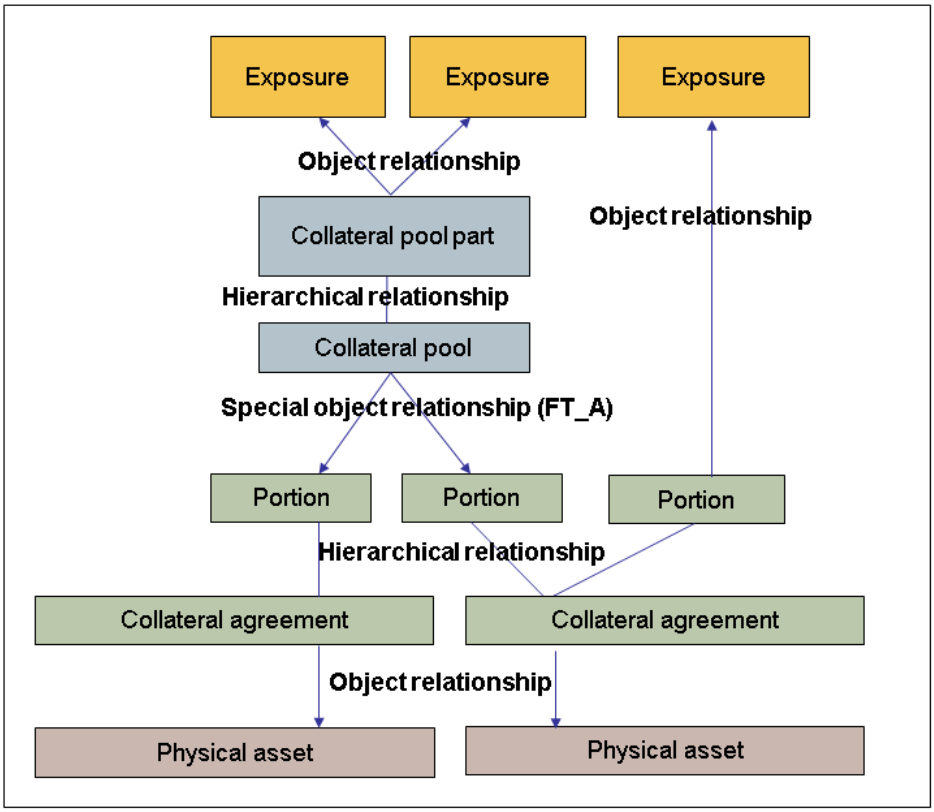
> **Note**
>
> The assignment of trigger link types and bundle types to a specific transformation rule is *optional*. However, we recommend that you assign a trigger link type because the system would otherwise apply the transformation rule to all bundles that have the bundle type in question.

3. Transforms the bundles
   The system calls the BAdI implementation for bundle transformation that you assigned to the transformation rule in Customizing. For example, you can use a BAdI implementation to remove unwanted objects and links from the structure of the bundle, add new links between the remaining objects, and calculate new values for the attributes of link types (nominal value, weight, AB order, BA order).
4. Processes the transformed bundles
   The system then processes the transformed bundles.

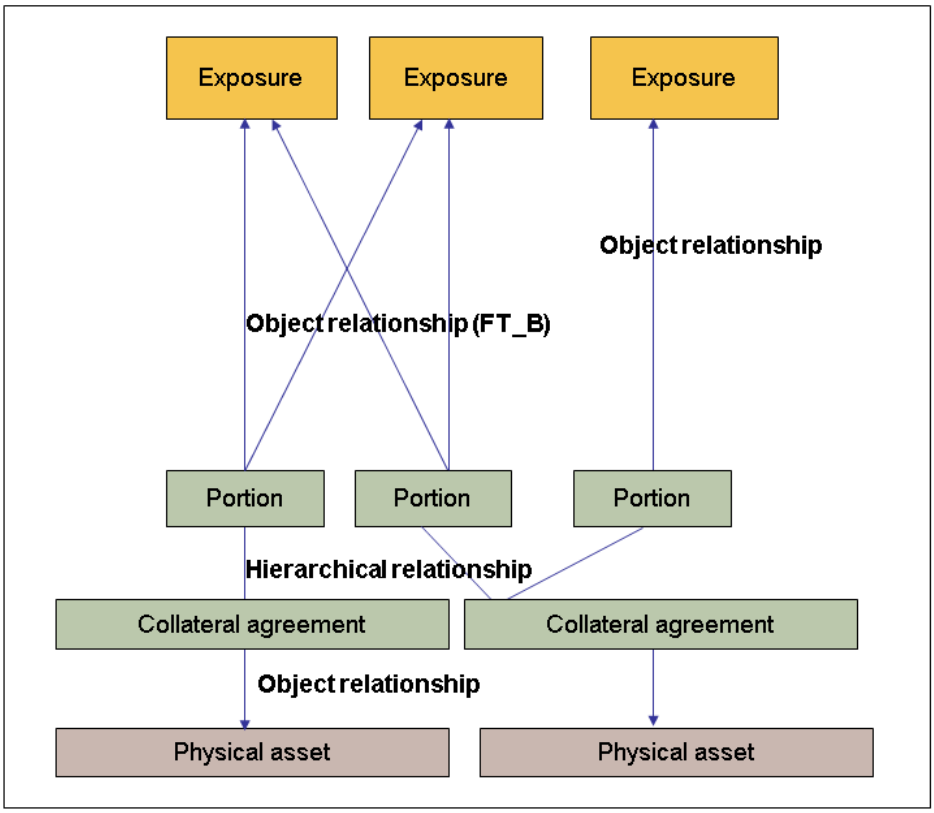
Example

You can model collateral pools as financial transactions in the Source Data Layer. Collateral pools contain multiple collateral agreements and certain object relationships. Exposures are related to the collateral pool, and do not have a direct relationship to the collateral agreements. The credit exposure calculation run expects the bundles to have a standard structure that enables them to be broken down into exposures and elementary collateral instruments (ESIs). An elementary collateral instrument contains a collateral agreement, a portion, and, optionally, a physical asset.

The example shows a transformation rule that has the trigger link type FT_A, the bundle type CREDIT_EXPOSURE, and an implementation of the BAdI for bundle transformation. The system uses link type FT_A to determine whether the bundle contains a collateral pool. If the bundle type is CREDIT_EXPOSURE, the system uses the custom BAdI implementation to transform the bundle. This implementation removes the collateral pool and all its parts from the structure of the bundle and creates new links between the individual exposures and portions. These links have the link type FT_B. The system also recalculates the attributes.



Bundle before transformation



Bundle after transformation

4.3.2 Account Pooling

Use

Account Pooling nets account balances that, for contractual or legal reasons, have to be grouped into one entity. The system nets payables and receivables of the contracts for the pooling group on a (main) account that you have defined.

You can start the run for account pooling for a run type you specify, or for all run types. The system uses a selection ID for the account pooling function where the run type is not specified. In Account Pooling for a specific run type you can use multiple master parameters for data selection in one run. You define this in Customizing by assigning multiple selection IDs to selection groups, which you then assign to a run type.

Prerequisites

You have made the settings required in the following Customizing activities:

- *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Selection*

- *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Account Pooling* >
  - *Define Market Data Area and Exchange Rate Category*
  - *Edit Pooling Type*
  - *Edit Grouping Key*
  - *Assign Key Figures for Pooling*
  - *Secondary Data Source for Account Pooling* > *Define Module for Account Pooling*
- *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Common Settings for Result Database* or *Common Settings for Results Data Layer*

For more information, see the documentation for the relevant sections in Customizing.

Features

The system nets the balances for a pooling group. You model pooling groups using the object relationships of the primary objects. You can sort the pooling group into legal entities (such as the client or company code) using the grouping key. The pooling type then defines which balances are to be used in account pooling. The system assigns the pooling key figure that is to be calculated to the combination of grouping key and pooling type.

The system stores the account pooling results in the Results Data Layer (RDL) [page 1364] or in the Result Database (RDB) [page 1823], and they can be accessed from there and used in other calculations. You define where results are stored for each client individually. You do so in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Storage of Results* > *Define Results Storage*. You can display the results in Run Administration [page 433].

> **Note**
>
> Although online detail reporting [page 418] uses the same processes as Account Pooling, the online detail reporting function it is purely a test function. You can use it to display additional interim results, which helps you to understand each calculation step by looking at a small amount of data. Note that the results of online detail reporting are not stored in the RDL or the RDB.

Account Pooling for Utilizations

When the system pools account balances, it distinguishes between utilizations that are assets and those that are liabilities (assets minus liabilities).

Account Pooling for Specific Provisions and (Optional) Calculation Components

For specific provisions, and for up to 15 additional optional calculation components, the system ignores whether the amounts are assets or liabilities when it pools the accounts.

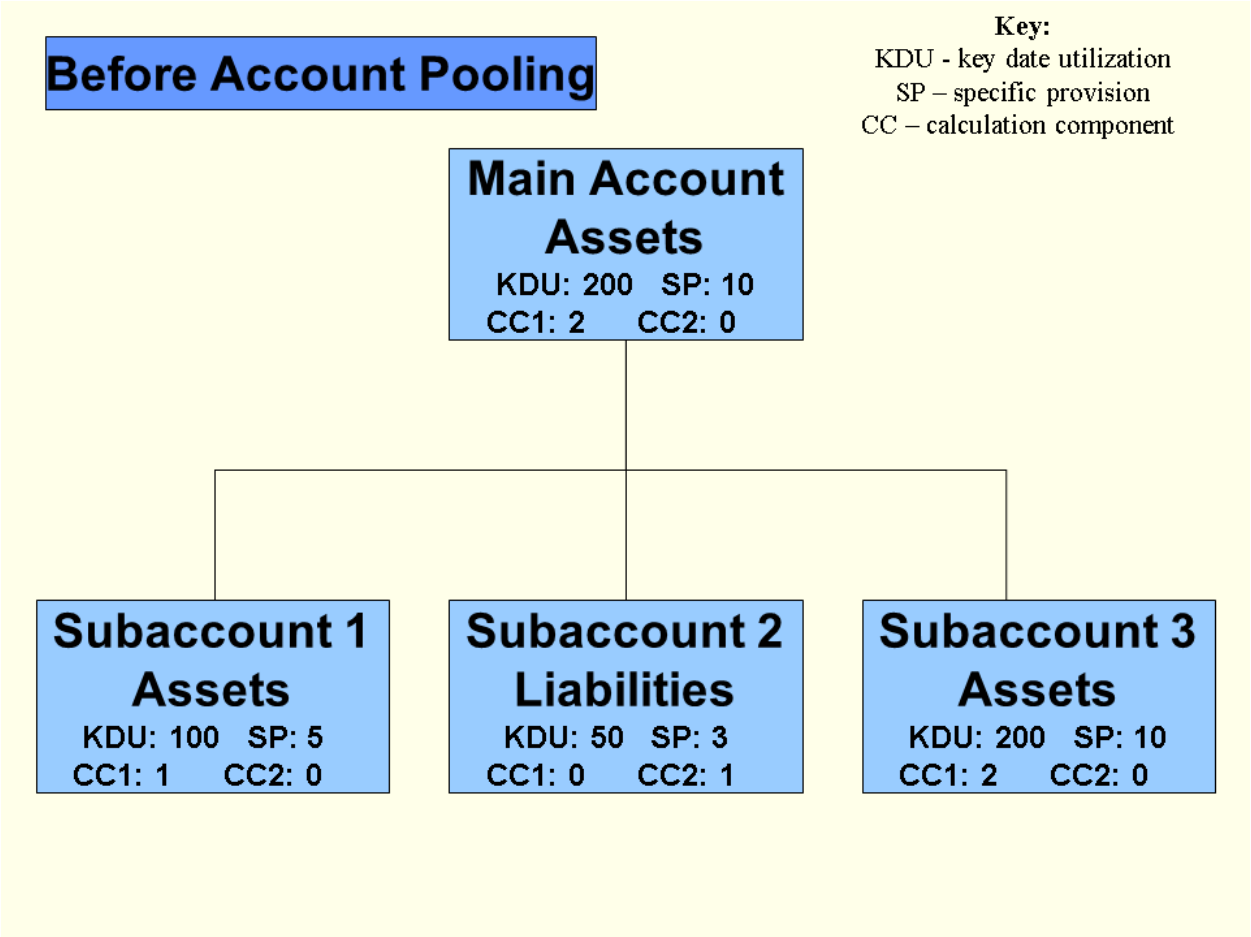
Activities

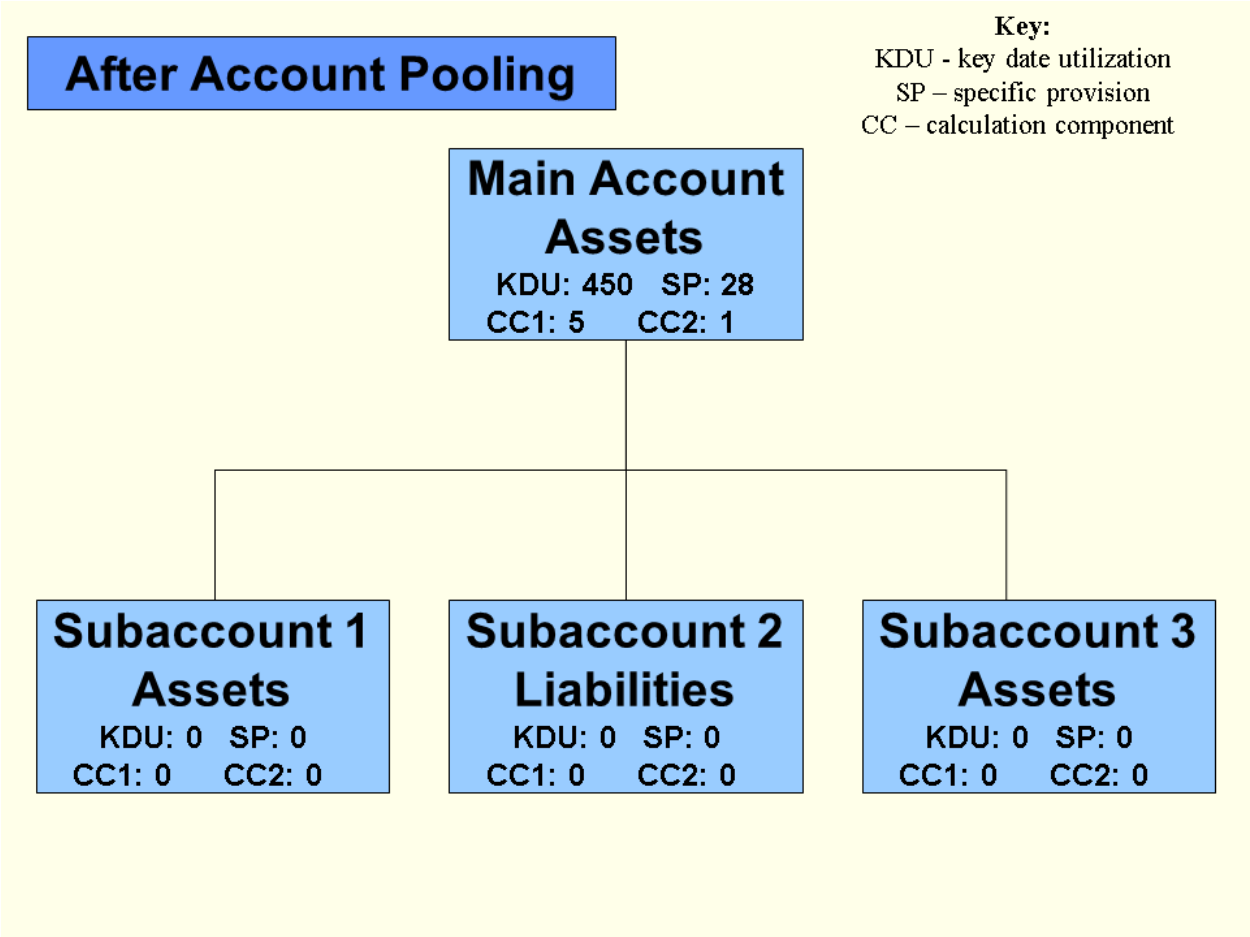
To pool account balances without specifying a run type, see a. below. To pool account balances for a run type, see b. below.

1. On the *SAP Easy Access* screen, choose:
    1. *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling* > *Account Pooling*.
    2. *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling* > *Account Pooling for a Run Type*.
2. Enter the following data:
    1. In the *Cust. ID WL Generatn w.InfoSet* field enter the master parameters for data selection. You edit the master parameters for data selection in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Selection* > *Edit Master Parameters for InfoSet Selection*.
    2. In the *Run Type* field specify the run type if you are using the facility distribution function for a run type. You enter the settings required in Customizing for Bank Analyzer under:
        - *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Common Settings for Result Database* > *Enter Additional Settings for the RDB Parameters for a Run Type*.
        - *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Common Settings for Results Data Layer* > *Edit RDL Parameters for a Run Type*.
3. Enter a business record date. The default value is today's date.
4. If required, change the time stamp of the access to the data in the Source Data Layer (SDL) and Results Data Layer (RDL). The default values are today's date and the current system time.
5. Enter a filter variant ID to define how messages are stored and displayed in the run administration function. You define filter variants in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages*.
6. Choose *Execute*.

Example

In this example, the optional calculation component CC1 has been assigned the key figure *Credit Interest* and the calculation component CC2 has been assigned the key figure *Debit Interest*. You make this assignment in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Account Pooling* > *Assign Key Figures for Pooling*.


Before Account Pooling
Key:
KDU - key date utilization
SP – specific provision
CC – calculation component
Main Account
Assets
KDU: 200 SP: 10
CC1: 2 CC2: 0
Subaccount 1
Assets
KDU: 100 SP: 5
CC1: 1 CC2: 0
Subaccount 2
Liabilities
KDU: 50 SP: 3
CC1: 0 CC2: 1
Subaccount 3
Assets
KDU: 200 SP: 10
CC1: 2 CC2: 0



4.3.2.1 Prototype: CVPM Run for Account Pooling

Use

You can use this function to schedule the run for account pooling or the stress run for account pooling as a process and have it run automatically by the system.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. The same prerequisites apply to starting the CVPM run as apply for the account pooling runs [page 383].
2. You have also made the necessary settings for CVPM in Customizing under *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager*.
3. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling* > *Prototype: CVPM Run for Account Pooling* or *Prototype: CVPM Stress Run for Account Pooling*.
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Run Type
     Specify the run type.
   - Combined Scenario
     Specify the combined scenario. This is required only if you start the CVPM run for a stress run.
   - Process Partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. The system enters the time at which you called this transaction as the default value.
   - Technical Settings
     You can use a filter variant or start the CVPM run as a test run.
3. Choose .

> **Note**
>
> To display the CVPM run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.3.3 Determination of the Free Line for Financial Products

Use

This key date process calculates the free line for financial products. The system first evaluates payment transactions and uses these to calculate the actual utilization. The system then compares a credit commitment (for example, external line for a checking account, commitment capital for a loan or global limit for a traditional off-balance-sheet transaction) with the utilization. The resulting difference is the free line.

In this process individual transactions are always processed, in contrast to the Determination of the Free Line for Facilities [page 395], which is applied to a hierarchically structured grouping of facilities.

The determination of the free line for financial products is configured in the Calculation and Valuation Process Manager (CVPM) [page 1622]. You can schedule the process as a periodic background job.

Prerequisites

The prerequisites for the free line calculator [page 391] apply.

Process

1. **Set SDL time stamp**
   Set an SDL time stamp. For more information, see Set the SDL Time Stamp [page 367]. This process is the leading process for determining the free line for financial products.
2. **Execute run**
   To call transaction `/BA1/RU_FREELINE`, from the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Free Line for Financial Products* > *Determine Free Line for Financial Products*:
   - You restrict the run according to legal entity.
   - You can use the *Recalculate Key Date* checkbox to specify whether you want the system to recalculate the results for the current key date or to use restart points:
     - If you do **not** select the checkbox, the system uses utilizations calculated in previous runs as restart points for the current key date. Performance with this setting is better.
     - If you select the checkbox, the system only uses the utilizations calculated for *other* key dates as restart points. This setting is required, for example, if existing results for the key date of the current run contain errors. If you update the results of a key date with a new run, existing results of later key dates become invalid. You therefore have to execute the runs for later key dates again, and select the *Recalculate Key Date* checkbox.

     The process transfers the setting for this field to the free line calculator.

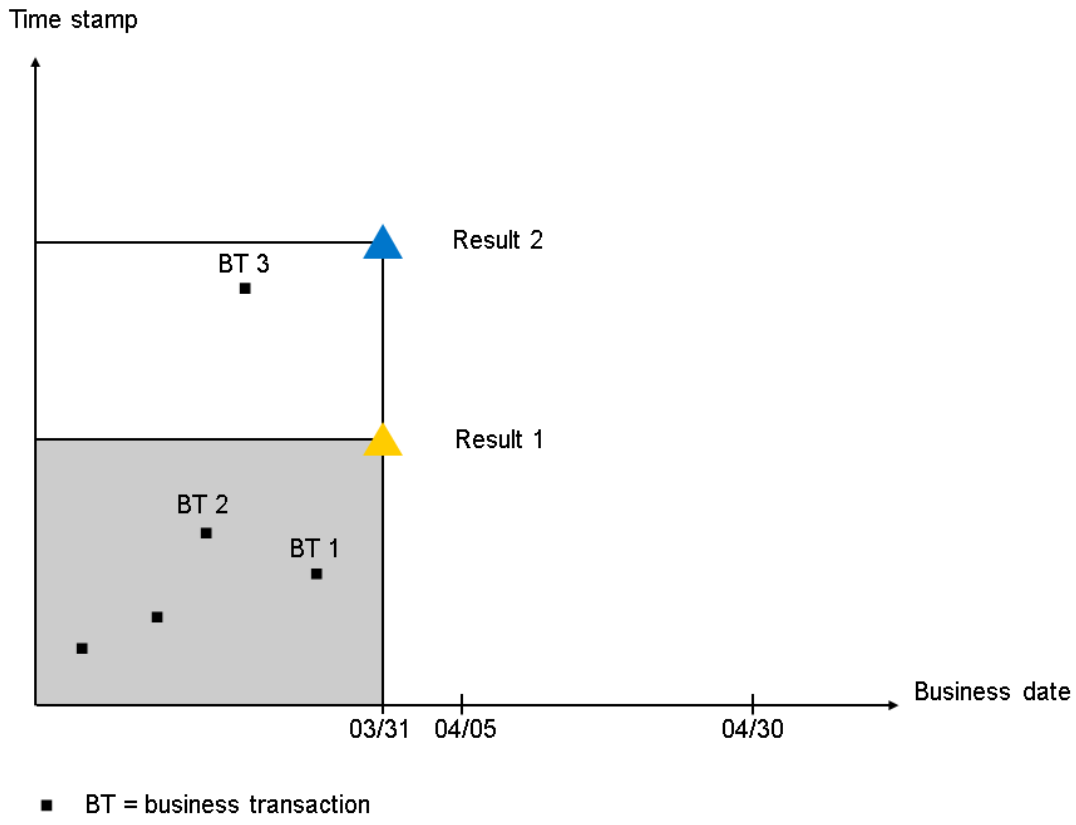
Settings for CVPM Fixed Step Sequence

You configure the CVPM fixed step sequences in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Fixed Step Sequences for Analytical Processes*.

First enter the results data area that you want to use to store the results. The fixed step sequence comprises the following steps:

1. **Create worklist**
   Using the CVPM step sequence you have selected which option the system uses to create the worklist. Ensure that the worklist consists of financial transactions. You can use an InfoSet from the Source Data Layer (SDL) as data source, for example. You can use selection criteria to restrict the worklist to financial transactions with specific properties.
2. **Import data**

This process calls the free line calculator. This calculates the free line for every financial transaction in the worklist. The figure below explains which business transactions from sequential runs are considered on the *same* key date:



Calls to the free line calculator on the same key date

When the calculator is first run on March 31, the system selects all business transactions that have been recorded in the system up to and including this key date and for the first time stamp (BT 1 and BT 2). These flows are aggregated to a utilization (result run 1).
If the *Recalculate Key Date* checkbox on the selection screen has not been selected, the system uses the result of the first run as the restart point for the second run on the same day. In this case, the second run only processes those business transactions that have been imported after the first time stamp (BT 3). All business transactions up to the current key date are processed again without the use of restart points.

3. **Results display**
   Results are only displayed in the test run.
4. **Storage**
   Use a results type that is assigned to the results category `HKFRL` to save the results. Entries must be made in the key fields *Financial Transaction ID* and *Key Date* for this results category. The amount for the free line is saved in the Results Data Layer (RDL) for each financial transaction. The results for the result key figures *Free Line* and *Utilization* are saved in object currency. The utilization is saved as a restart point so that it can be used again in subsequent runs.

Result

The system uses the results to calculate the risk provision for the off-balance-sheet part of a receivable:

- The functions in **Calculation Management** in Accounting use the results to calculate the off-balance-sheet risk provision.
- If you are using **Source Data Aggregation**, the *Calculate Percentage Expected Loss* process uses the results to determine expected loss amounts for financial transactions that are assigned to an aggregation object.

You can also display the free line directly in the **balance sheet notes**.

> **Note**
>
> You can also call the results using Enterprise Services `FIC_ManageFinancialInstrumentPositionAccountingBalanceKeyFigureIn`.

4.3.3.1 Free Line Calculator

Use

Various applications use the free line calculator to calculate the free line for financial transactions. The CVPM process `/BA1/RAL_RU_FREELINE` *Determination of the Free Line for Financial Products* calls the free line calculator in the data enrichment step, for example.

Integration

The following applications use the free line calculator:

- Determination of the Free Line for Financial Products [page 388]
- Accounting for Financial Products [page 678]
  Accounting uses the free line calculator to determine the off-balance-sheet risk provision.

Prerequisites

You have made the required settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Free Line for Financial Products*:

You have assigned exactly one method for determining the free line for financial products to a production control. The *Determination of the Free Line for Financial Products* process calls the assigned method for each financial transaction to determine the free line. This method controls which service module [page 118] of the financial transaction is used to determine the limit.

- Key figures
  If you assign this service module to the method, the system uses only the key figure category `NMVAL` *Nominal Amount*.
- Limit data
  If you assign this service module to the method, the system uses only the limit category `EXT` *External Credit Line*.

> **Note**
>
> The key figures and limit data must be available in object currency.

In Customizing, you have also assigned business transaction classes and transaction types to the method. These define which flows the system includes when it determines the utilization. By selecting the relevant transaction types, you can control the determination of the utilization in the following way, for example:

- For a **checking account**, the calculator uses all the business transactions to determine the account balance.
- For a **non-revolving loan**, the calculator determines the total of the disbursed amounts.
- For a **revolving loan**, the calculator determines all the flows that have an effect on receivables (for example, disbursed amounts, repayments, and capitalized interest).

Features

The system reads the limit of a financial product from the master data in the Source Data Layer (SDL) (see *Prerequisites*). It determines the utilization by reading the business transactions defined in Customizing from the SDL, and using the utilizations stored in the Results Data Layer (RDL) as restart points. The restart points are end-of-day values that the *Determine Free Line for Financial Products* process has saved in the RDL in previous runs.

The free line is the difference between the limit and utilization. The system ensures that the free line is always positive and that it never exceeds the limit.

> **Note**
>
> Liability items are regarded as negative utilization. If there is a credit balance, the utilization would be zero. However, in this case the system posts a negative utilization so that the key figure can be used as the restart point. Due to this, you cannot derive from the utilization whether a position is an asset or a liability.

- For a credit balance (negative “utilization”), the following applies: free line = limit
- If 0 utilization < limit: free line = limit utilization
- If limit utilization: free line = 0

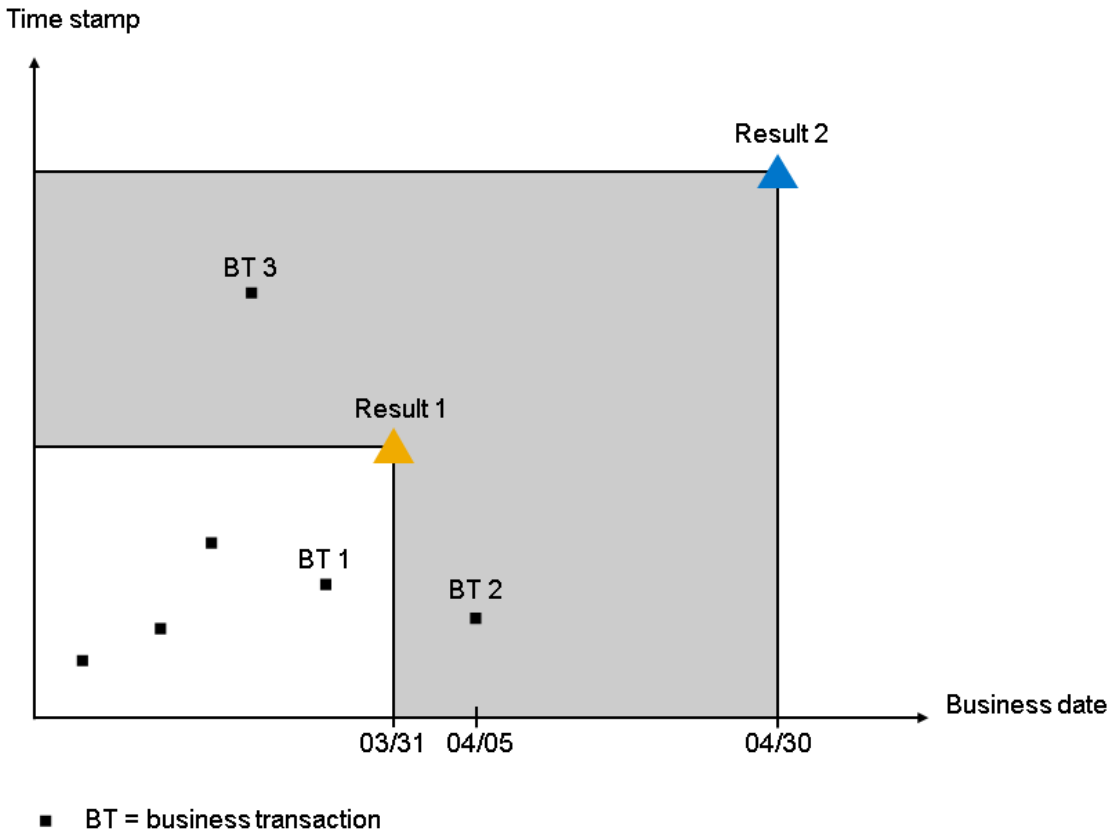
In the following case, the system does not determine a free line and utilization: A financial transaction has a status to which the status category `006` (object expired) has been assigned in Customizing for *Bank Analyzer* under ▶ *Source Data Layer* > *Settings for Attributes* > *Basic Data* > *Edit Status* ◀. This setting allows you to no longer display the free line and to set the off-balance-sheet risk provision to **0**. You can use this after you cancel a loan contract, for example.

> **Note**
>
> The free line calculator can calculate the free line at the start of the day. If you have set the corresponding indicator, the calculator reads the transactions with the key date but includes only business transactions and restart points before the key date.
>
> Start-of-day values for the free line are used by the calculation method `ME47` (Pro Rata Clearing of Fair Value Adjustment for Free Line) in Calculation Management, for example. You can calculate start-of-day values if the *Update Secondary Business Transactions* process calls the free line calculator on the fly, but not if the call is triggered by the *Determine Free Line for Financial Products* process.

Activities

The free line calculator checks whether there are already utilizations for the financial transaction in the Results Data Layer that can be used as restart points. If these kinds of results exist, the system only searches for new business transactions for the financial transaction from the time stamp of the last RDL result onwards. It offsets these business transactions with the utilization from the last results data record.

The figure below explains which business transactions are included on *different* key dates when the free line calculator is called consecutively.



Calls to the free line calculator on different key dates

When the calculator is first called on March 31, the system selects all business transactions that have been recorded in the system up to and including this key date and for the selected time stamp (BT 1). These flows are aggregated to a utilization (result 1) and saved in the RDL. If you trigger the calculation of the free line again on April 30, this utilization is used as a restart point. When the calculator is called the second time, it also processes business transactions that were created on a later key date from a business perspective (BT 2). Furthermore, it processes business transactions that belong to the period before March 31 from a business perspective but that were imported at a later date, for example in the case of a correction (BT 3).

Example

Checking account with a negative utilization (credit)

Limit = 3000

Utilization = - 2000

Free line = limit = 3000

Checking account with partial utilization

Limit = 3000

Utilization = 2000

Free line = limit - utilization = 1000

Checking account with a tolerated overdraft

If the bank tolerates an overdraft within certain limits, the utilization can be greater than the external credit limit:

Limit = 3000

Utilization = 3100

Free line = 0

4.3.4 Determination of the Free Line for Facilities

Use

The free line of credit (undrawn amount) for facilities is calculated for a hierarchical grouping of related transactions (facility group), and a credit commitment is then compared with the (cumulated) drawings.

Under Basel II, the disclosure of the free line of credit for each commitment in the facility group is important, as risk factors (maturity, business partner rating) can vary across the levels of the facility group. The free line is calculated in the same way for all approaches.

You can start the run for calculating the free line for a run type you specify, or for all run types. The system uses a selection ID for the calculation of the free line where the run type is not specified. In Determination of the Free Line for a specific run type you can use multiple master parameters for InfoSet selection in one run. You define this in Customizing by assigning multiple selection IDs to selection groups, which you then assign to a run type.

Prerequisites

You have made the following required settings in Customizing for *Bank Analyzer*:

- *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Selection*
- *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Common Settings for Free Line and Collateral Distribution*:
  - *Edit Field Assignment for Source Data Layer*
  - *Edit Field Assignment for Results Data Layer or Result Database*
- *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Common Settings for Result Database* or *Common Settings for Results Data Layer*

For more information, see the documentation for the relevant Customizing activity.

Features

The determination of the free line for facilities can involve facilities, multi-borrower facilities, or individual credit lines. The process is different in each case.

The system stores the results of the calculation of the free line in the Results Data Layer (RDL) [page 1364] or in the Result Database (RDB) [page 1823], and they can be accessed from there and used in other calculations. You define where results are stored for each client individually. You do so in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Storage of Results* > *Define Results Storage*. You can display the results in Run Administration [page 433].

> **Note**
>
> Although online detail reporting [page 418] uses the same processes as Determination of the Free Line, the online detail reporting function it is purely a test function. You can use it to display additional interim results, which helps you to understand each calculation step by looking at a small amount of data. Note that the results of online detail reporting are not stored in the RDL or the RDB.

Activities

To calculate the free line for facilities, do the following:

1. On the *SAP Easy Access* screen, choose:
   1. *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Free Line for Facilities* > *Determine the Free Line*.
   2. *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Free Line for Facilities* > *Calculate the Free Line for a Run Type*.
2. Enter the following data:
   1. In the *Cust. ID WL Generatn w.InfoSet* field enter the master parameters for data selection. You edit the master parameters for data selection in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Selection* > *Edit Master Parameters for InfoSet Selection*.
   2. In the *Run Type* field specify the run type if you are using the facility distribution function for a run type. Make the required settings in Customizing for *Bank Analyzer*:
      - *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Common Settings for Result Database* > *Enter Additional Settings for the RDB Parameters for a Run Type*.
      - *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Common Settings for Results Data Layer* > *Edit RDL Parameters for a Run Type*.
3. Enter a business record date. The default value is today's date.
4. If required, change the time stamp of the access to the data in the Source Data Layer (SDL) and Results Data Layer (RDL). The default values are today's date and the current system time.

5. Enter a filter variant ID to define how messages are stored and displayed in the run administration function. You define filter variants in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages*.
6. Choose .

Example

For each financial transaction, the free line is the difference between the generalized external commitment (GXC), and the generalized drawing (GD).

Free line for facilities = max {GXC - GD; 0}

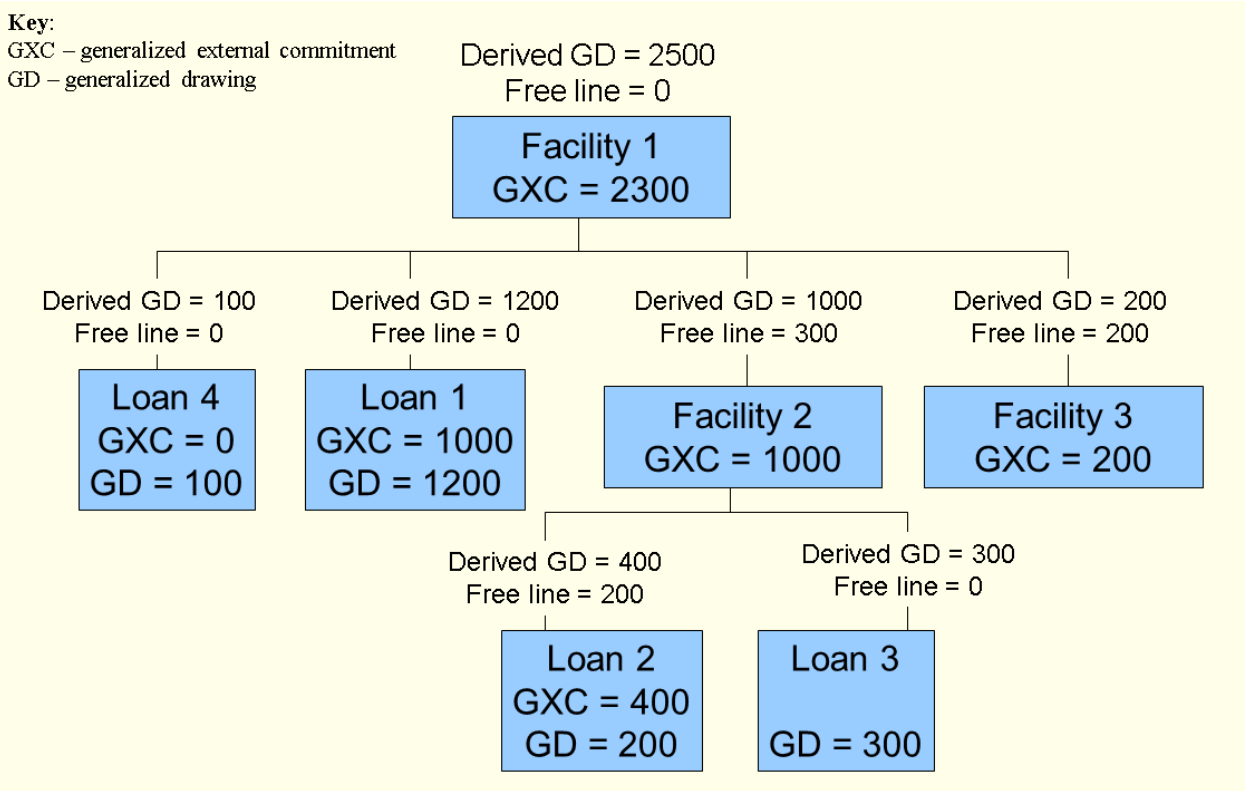
The generalized drawing on facilities is calculated from the total of the derived generalized drawing on the related loans.

GD = Ój GDderived (j)

The derived generalized drawing on a particular loan is calculated from the maximum of the generalized external commitment and the generalized drawing.

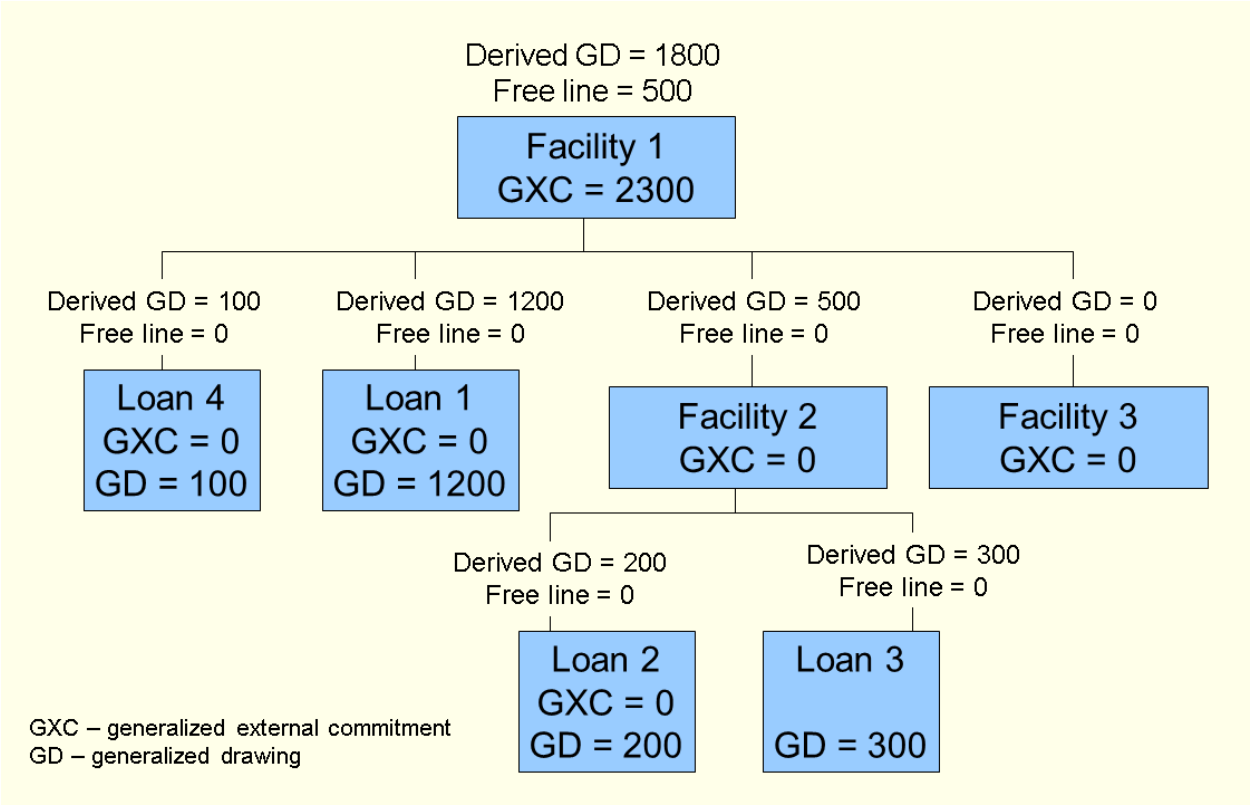
GDderived = Max {GXC, GD}

The determination of the free line for facilities can involve facilities, multi-borrower facilities, or individual credit lines. The process is different in each case. The following is the result for a special facility group:



By defining the data enrichment process suitably, you can display a free line for facilities at the level of the uppermost facility as well. This value is the difference between the generalized external commitment of the

uppermost facility, and all the generalized drawings of the levels below the facility. To be able to display this value, you must set all the generalized external commitments on the lower levels to 0. This gives the following result:



4.3.4.1 Prototype: CVPM Run for the Free Line for Facilities

Use

You can use this function to schedule the run for the free line or the stress run for the free line for facilities as a process and have it run automatically by the system.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. The same prerequisites apply to starting the CVPM run as apply for the runs for determining the free line for facilities [page 395].
2. You have also made the necessary settings in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager*.
3. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Free Line for Facilities* > *Prototype: CVPM Run for Free Line* or *Prototype: CVPM Stress Run for the Free Line*.
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Run Type
     Specify the run type.
   - Combined Scenario
     Specify the combined scenario. This is required only if you start the CVPM run for a stress run.
   - Process Partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. The system enters the time at which you called this transaction as the default value.
   - Technical Settings
     You can use a filter variant or start the CVPM run as a test run.
3. Choose  .

> **Note**
>
> To display the CVPM run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.3.4.2 Distribution of Facilities

Use

This process maps a complex object containing a facility and multiple subcontracts to a number of (effective) individual contracts. This procedure is required if subsequent processes can handle only (effective) individual contracts (single record processing).

Cash flows can be generated only if there are contracts with distributed facilities, for instance. This means that if you want to generate cash flows, you have to run facility distribution before doing so.

Facility distribution distributes the external credit line of a facility to the assigned subcontracts. The following distribution categories are provided:

- No distribution
- Equal distribution (the external credit line is divided by the number of contracts)
- Pro rata distribution by factors you specify
- Custom distribution (by means of a user exit)

The system then deducts the distributed credit lines from the utilization and calculates the free line for facilities or the overdrawn amount for each contract.

You can start the run for facility distribution for a run type you specify, or for all run types. The system uses a selection ID for the facility distribution function where the run type is not specified. In Facility Distribution for a specific run type you can use multiple master parameters for data selection in one run. You define this in Customizing by assigning multiple selection IDs to selection groups, which you then assign to a run type.

Prerequisites

You have made the required settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution*. For more information, see the documentation for the relevant Customizing activity.

Features

Facility distribution can involve facilities, multi-borrower facilities, or individual credit lines. The process is different in each case.

The facility distribution program results in a contract with distributed facilities. You can display the results of facility distribution in run administration [page 433].

The system stores the results of the facility distribution process in the Results Data Layer (RDL) [page 1364] or in the Result Database (RDB) [page 1823], and they can be accessed from there and used in other calculations. You define where results are stored for each client individually. You do so in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Storage of Results* > *Define Results Storage*. You can display the results in Run Administration [page 433].

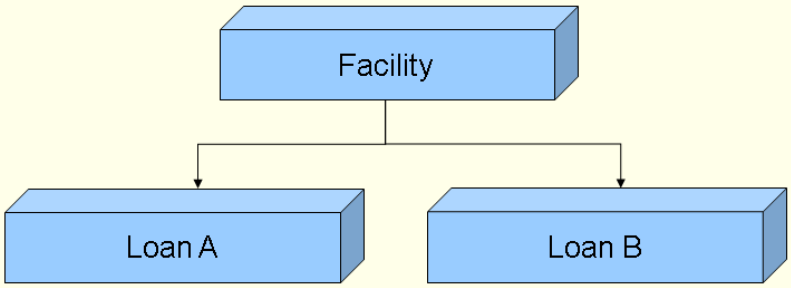
Activities

To distribute facilities without specifying a run type, see a. below. To distribute facilities for a run type, see b. below.

1. On the *SAP Easy Access* screen, choose:
    1. *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Free Line for Facilities* > *Facility Distribution*.
    2. *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Free Line for Facilities* > *Facility Distribution for a Run Type*.
2. Enter the following data:
    1. In the *Cust. ID WL Generatn w.InfoSet* field enter the master parameters for data selection. You edit the master parameters for data selection in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Selection* > *Edit Master Parameters for InfoSet Selection*.
    2. In the *Run Type* field specify the run type if you are using the facility distribution function for a run type. Make the required settings in Customizing for *Bank Analyzer*:
        - *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Common Settings for Result Database* > *Enter Additional Settings for the RDB Parameters for a Run Type*.
        - *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Common Settings for Results Data Layer* > *Edit RDL Parameters for a Run Type*.
3. Enter a business record date. The default value is today's date.
4. If required, change the time stamp of the access to the data in the Source Data Layer (SDL) and Results Data Layer (RDL). The default values are today's date and the current system time.
5. Enter a filter variant ID to define how messages are stored and displayed in the run administration function. You define filter variants in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages*.
6. Choose  .

Example

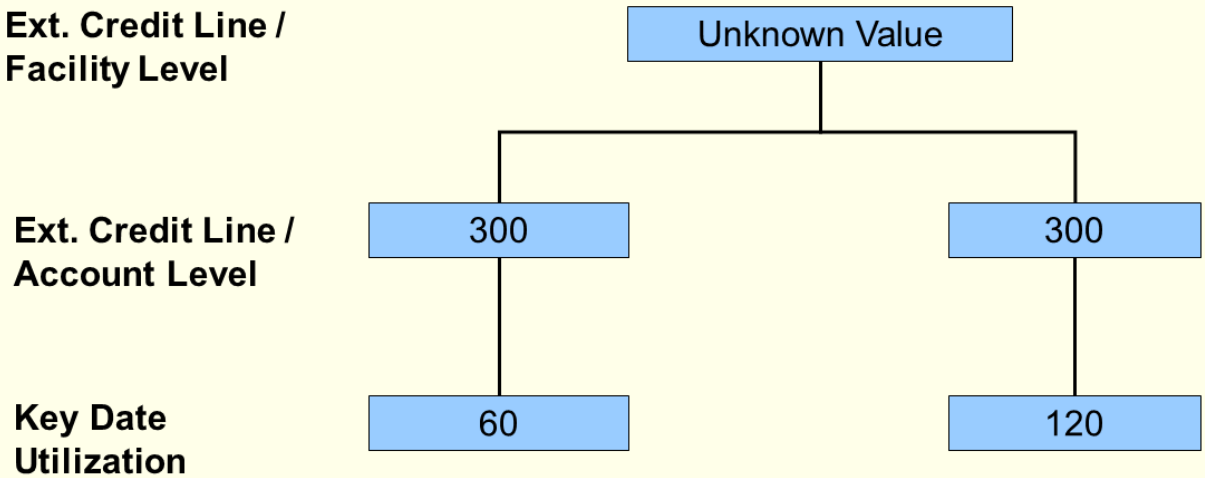
- **Facility is mapped as a financial transaction**
- **Drawings are mapped as financial transactions (loans) with a settlement account and a position**



Σ External commitment at facility level
./. Σ Utilization of loan
= Free line at facility level

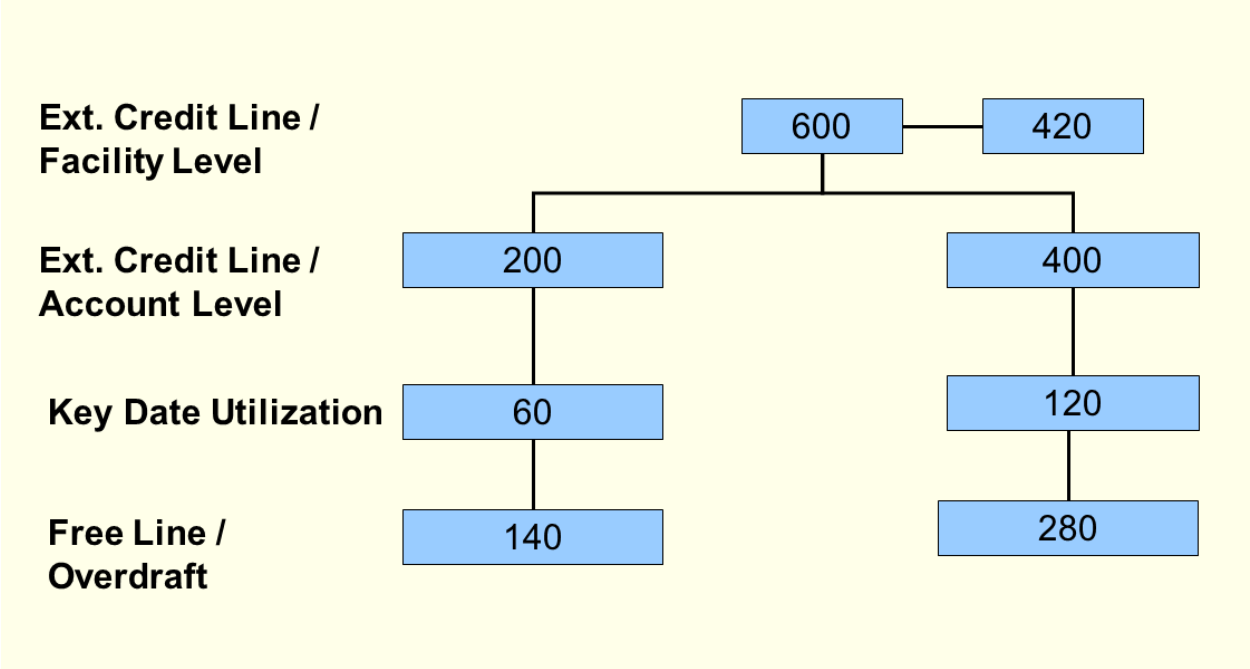
Before facilities are distributed:

Before a facility is distributed, the system calculates the external credit line, which is the contractually agreed line of credit on an account for each business partner on a key date, and the key date utilization.



After facilities have been distributed:

The amount of the external credit line at facility level is distributed in relation to the key date utilization that was calculated in the previous step. In this way, each business partner is allocated its own external credit line at account level, and the respective key date utilization is subtracted from this. This results in the free line for facilities or the overdraft on the credit line. The total of the free lines or the overdraft on the credit line is calculated by adding together the respective free lines or overdrafts.



4.3.4.2.1 Prototype: CVPM Run for Facility Distribution

Use

You can use this function to schedule the run for facility distribution or the stress run for facility distribution as a process and have it run automatically by the system.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. The same prerequisites apply to starting the CVPM run as apply for the facility distribution [page 399] runs.
2. You have also made the necessary settings in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager*.
3. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Free Line for Facilities* > *Prototype: CVPM Run for Facility Distribution* or *Prototype: CVPM Stress Run for Facility Distribution*.
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Run Type
     Specify the run type.
   - Combined Scenario
     Specify the combined scenario. This is required only if you start the CVPM run for a stress run.
   - Process Partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. The system enters the time at which you called this transaction as the default value.
   - Technical Settings
     You can use a filter variant or start the CVPM run as a test run.
3. Choose  .

> **Note**
>
> To display the CVPM run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.3.5 Collateral Distribution

Use

This process maps a complex object, which contains one or more contracts and collateral items, to a number of (effective) individual contracts. This procedure is required if subsequent processes can handle only (effective) individual contracts (single record processing).

You can start the run for collateral distribution for a run type you specify, or for all run types. The system uses a selection ID for the collateral distribution function where the run type is not specified. In the function for distributing collateral for a specific run type you can use multiple master parameters for data selection in one run. You define this in Customizing by assigning multiple selection IDs to selection groups, which you then assign to a run type.

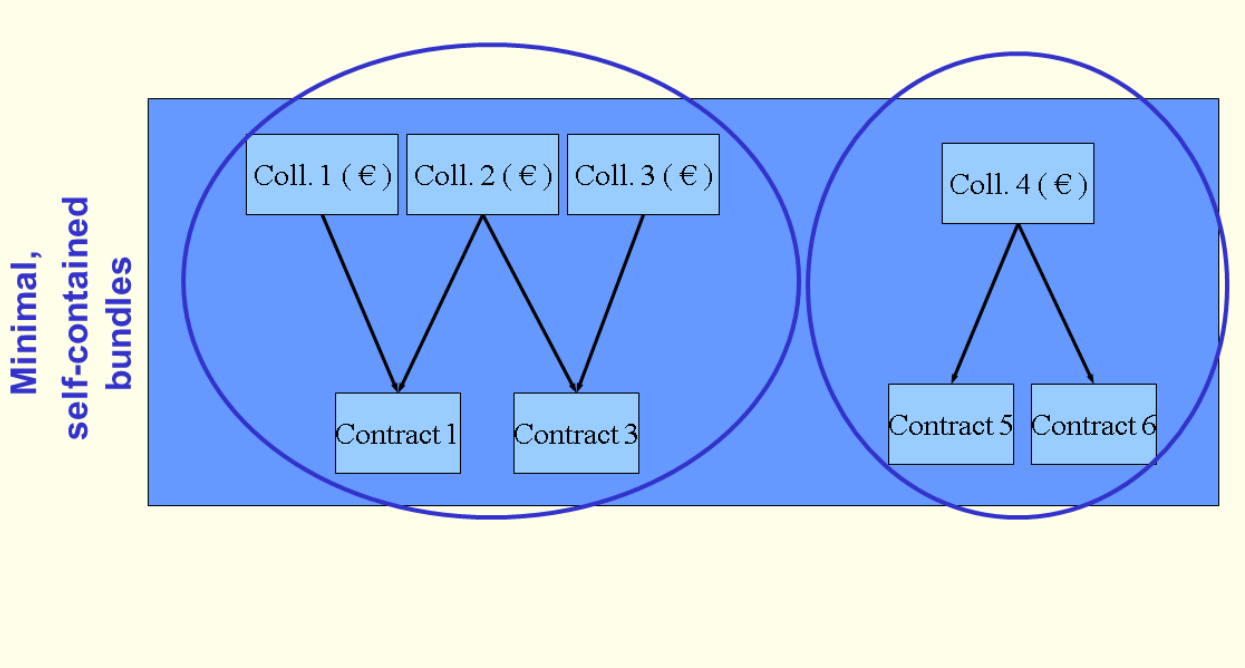
Prerequisites

You have entered the Customizing settings required under *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* >. For more information, see the relevant sections in the Implementation Guide (IMG).

> **Note**
>
> If pooled positions are to be used as the basis for collateral distribution, you need to have already run the account pooling function before you can start collateral distribution.
>
> If distributed external commitments are to be used as the basis for distributing collateral, you need to have distributed facilities or determined the free line for facilities before you can start collateral distribution.

Features

When collateral is distributed, the system groups the collateral and relevant transactions into minimal self-contained bundles.



Amount-based collateral that secures one transaction only, and percentage-based collateral are regarded as “trivial” in terms of their distribution. However, amount-based collateral that secures more than one transaction has to be distributed. For example, it can be distributed using an algorithm that you define by implementing a user exit.

The distribution of amount-based collateral for multiple transactions is not trivial. For this reason, amount-based collateral that secures only one transaction and percentage-based collateral are referred to as “trivial” bundles, while amount-based collateral for multiple transactions is not trivial.

The collateral distribution function returns contracts to which distributed collateral has been assigned. The results object contains additional information about the contract defined in the Source Data Layer (SDL) [page 100]. This is a data record that can be identified by an object ID. The result object contains the identifying key and the value of the distributed collateral as key figures.

The system stores the results of the collateral distribution process in the Results Data Layer (RDL) [page 1364] or in the Result Database (RDB) [page 1823], and they can be accessed from there and used in other calculations. You define where results are stored for each client individually. You do so in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Storage of Results* > *Define Results Storage*. You can display the results in Run Administration [page 433].

Activities

To distribute collateral without specifying a run type, see a. below. To distribute collateral for a run type, see b. below:

1. On the *SAP Easy Access* screen, choose:
    1. *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Collateral Distribution* > *Collateral Distribution*.
    2. *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Collateral Distribution* > *Collateral Distribution for a Run Type*.
2. Enter the following data:
    1. In the *Cust. ID WL Generatn w.InfoSet* field enter the master parameters for data selection. You edit the master parameters for InfoSet selection in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Selection* > *Edit Master Parameters for InfoSet Selection*.
    2. In the *Run Type* field specify the run type if you are using the facility distribution function for a run type. You enter the settings required in Customizing for Bank Analyzer under:
        - *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Common Settings for Result Database* > *Enter Additional Settings for the RDB Parameters for a Run Type*.
        - *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Account Pooling, Facility Distribution, and Collateral Distribution* > *Common Settings for Results Data Layer* > *Edit RDL Parameters for a Run Type*.
3. Enter a business record date. The default value is today's date.
4. If required, change the time stamp of the access to the data in the Source Data Layer (SDL) and Results Data Layer (RDL). The default values are today’s date and the current system time.

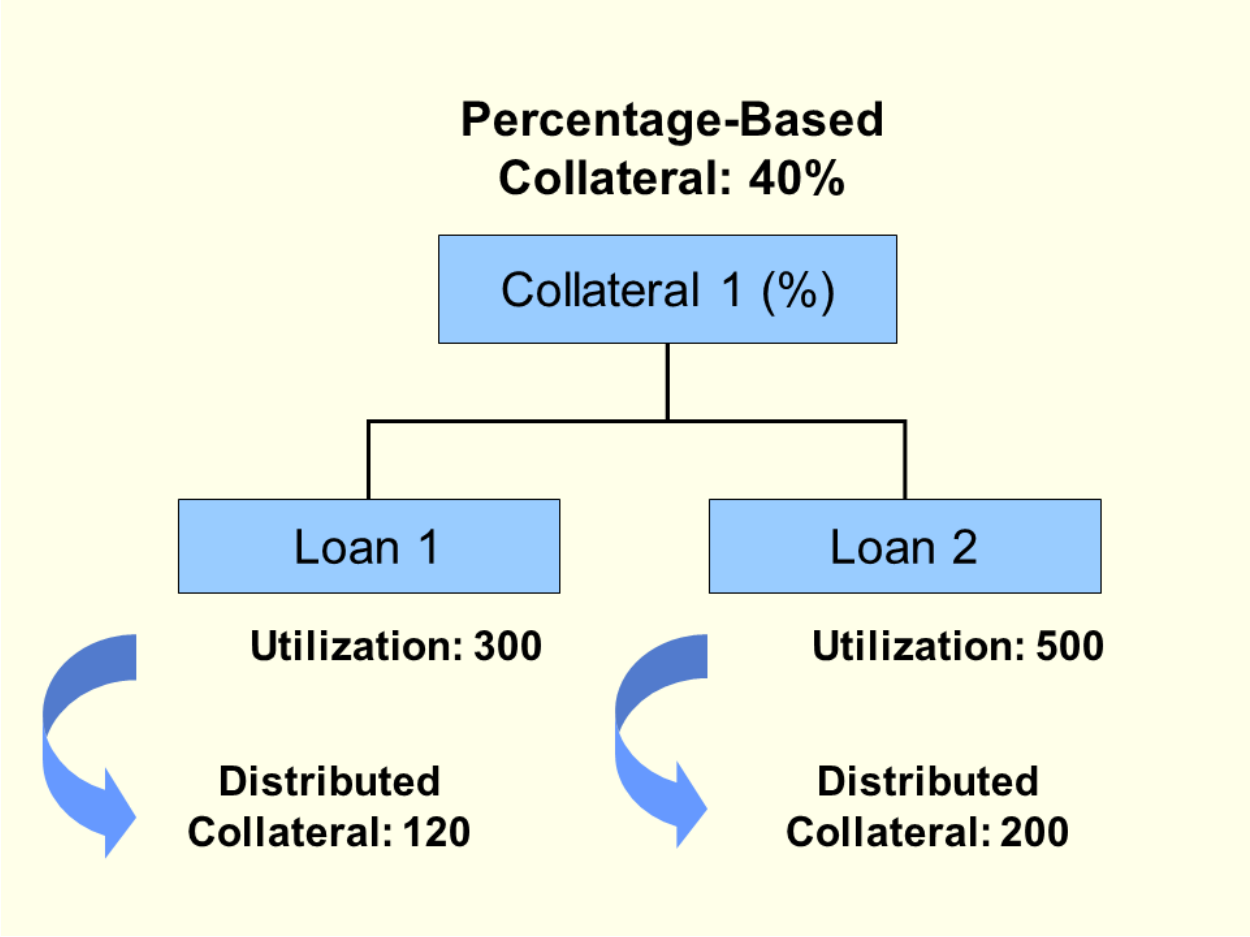
5. Enter a filter variant ID to define how messages are stored and displayed in the run administration function. You define filter variants in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages*.

Choose *Execute*.

Example

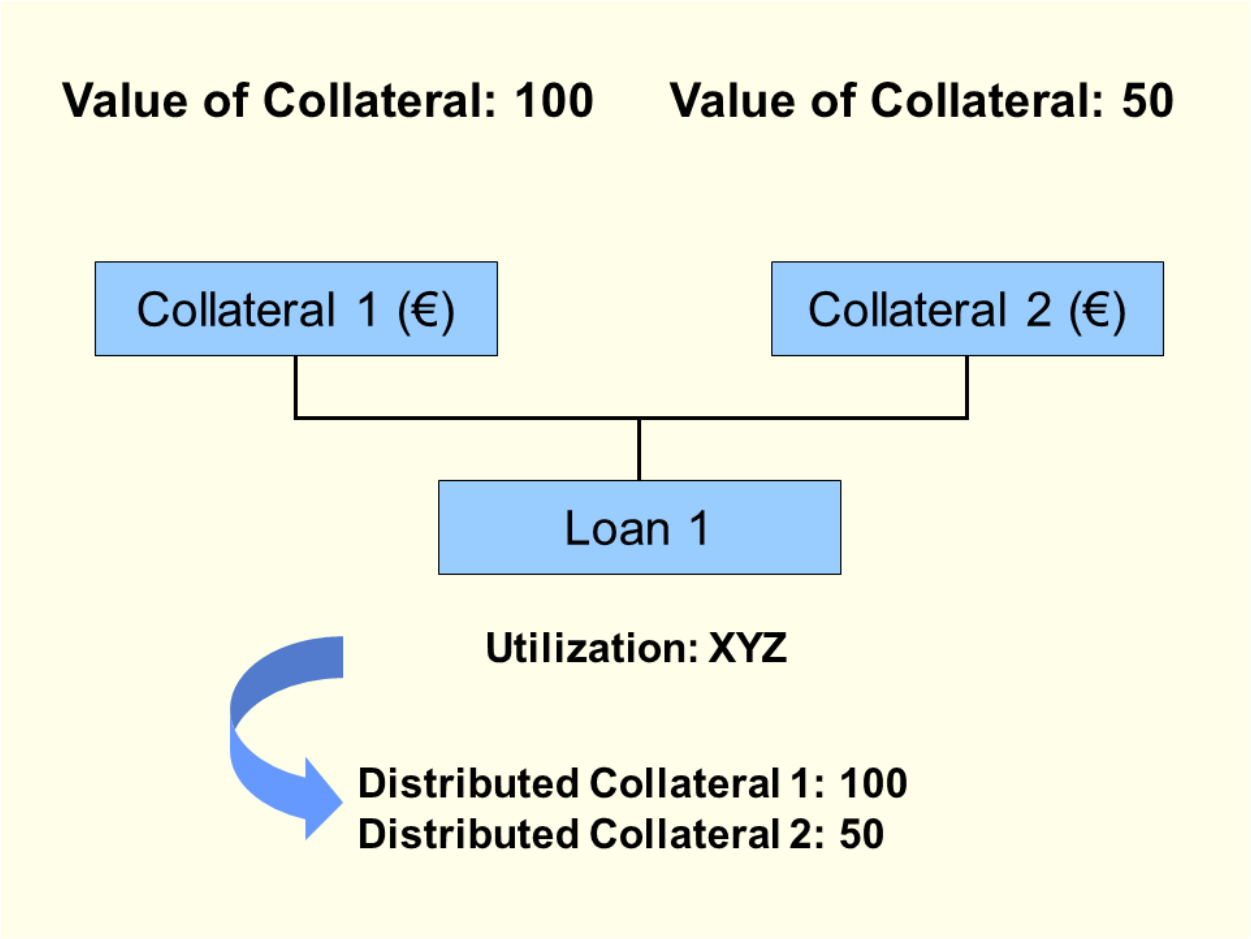
Example of the Distribution of Percentage-Based Collateral:

Percentage collateral of 40% is distributed across two loans. In this case, the amount of the collateral for loan 1 and loan 2 is 40% of the utilization respectively (trivial case).



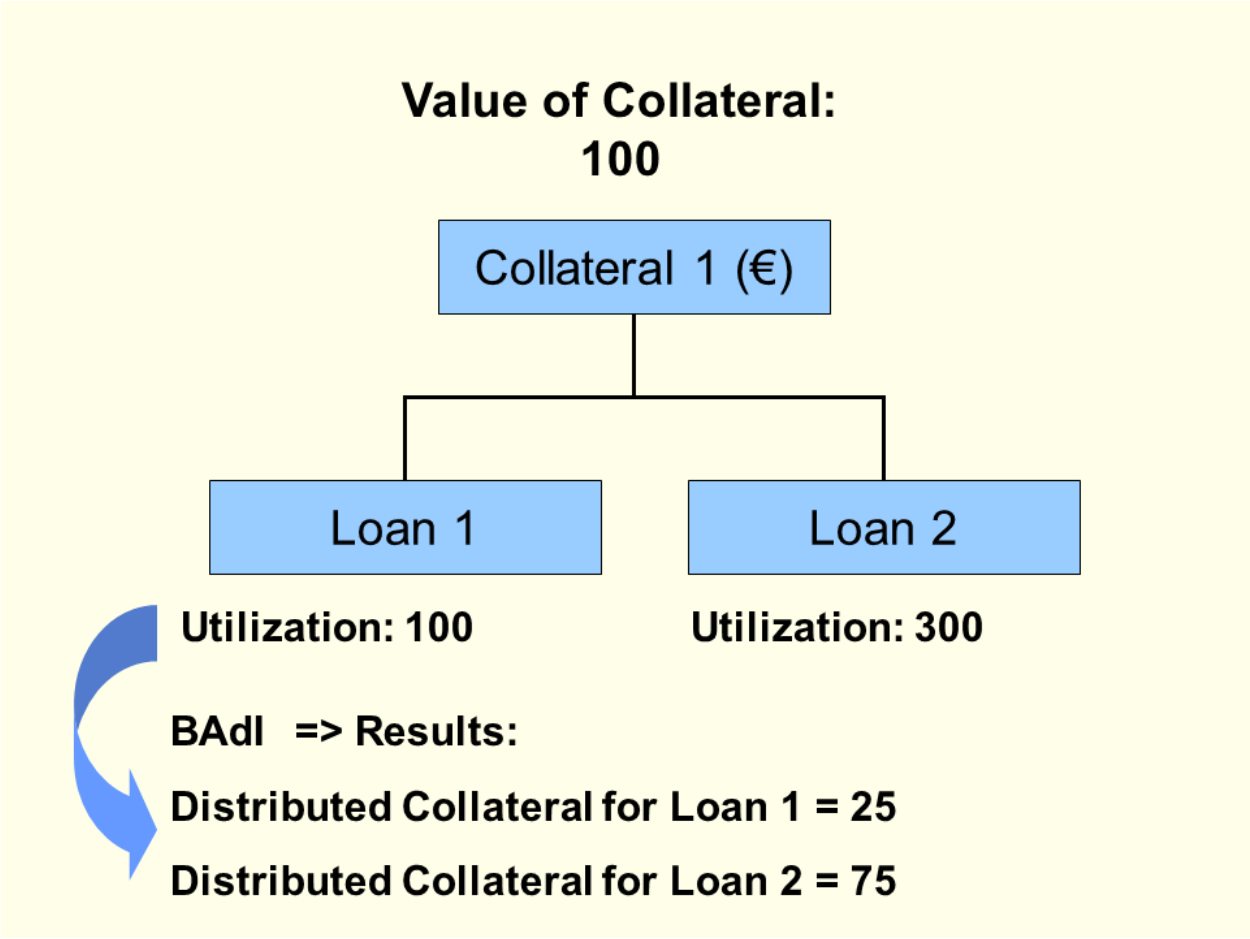
Example of the Distribution of Amount-Based Collateral to One Loan:

Two amount-based collateral items are assigned to one loan. In this case, the utilization and the values of collateral 1 and 2 are not dependent (trivial case).



Example of the Distribution of Amount-Based Collateral to Two Loans:

The amount-based collateral is distributed to loan 1 and loan 2 (non-trivial case). The BAdI (Business Add-In) **/BA1/DE_PRORATE_SAP**, which is predefined in the system, is used to distribute the collateral proportionally.



You can use BAdI **/BA1/DE_SPLX** if you want to define how amount-based multi-transaction collateral is distributed. This BAdI is empty, which means you add your own coding to define your own distribution process.

4.3.5.1 Prototype: CVPM Run for Collateral Distribution

Use

You can use this function to schedule the run for collateral distribution or the stress run for collateral distribution as a process and have it run automatically by the system.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. The same prerequisites apply to starting the CVPM run as apply for the collateral distribution [page 404] run.
2. You have also made the necessary settings for CVPM in Customizing under *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager*.
3. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Collateral Distribution* > *Prototype: CVPM Run for Collateral Distribution* or *Prototype: CVPM Stress Run for Collateral Distribution*.
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Run Type
     Specify the run type.
   - Combined Scenario
     Specify the combined scenario. This is required only if you start the CVPM run for a stress run.
   - Process Partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. The system enters the time at which you called this transaction as the default value.
   - Technical Settings
     You can use a filter variant or start the CVPM run as a test run.
3. Choose  .

> **Note**
>
> To display the CVPM run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.3.6 Determination of Default (FS-BA-PM-GM-DD)

Determination of Default derives a default event as defined in Basel II. In addition to the actual determination of default [page 411], you can change the data manually [page 413]. Stress tests [page 426] can also be carried out in the Determination of Default component.

4.3.6.1 Determination of Default

Use

In the run for determining default events, the system derives a default in accordance with the definition given in the Basel II Accord. It then posts the results in the form of defaulted objects (for example, financial transactions or business partners) to the Result Database (RDB) [page 1823]. The objects can be objects that have defaulted directly, or whose default is derived, for example when a related object is in default.

Prerequisites

Make sure that the settings are complete in Customizing for Determination of Default under *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Determination of Default*. For more information, see the Customizing documentation for the relevant activities.

Activities

1. On the SAP Easy Access screen, choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Determination of Default* > *Determine Default*. The system displays the *Determination of Default* screen.
2. Make the following entries in the Identification of Default Layer area of the screen:
   - In the *ID of Default Data* field, enter the ID of the data layer. By entering the layer ID, you specify the target for the data.
   - If required, enter the ID of the data target. You need to specify the ID of the data target only if you want to use distributed data targets. For more information see Customizing for Determination of Default under *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Determination of Default* > *General Settings for Data Selection* > *Edit Distributed Data Targets*.
   - If you want to start a *delta run* that is to consider only a subset of the relevant characteristic values in order to supplement the results already stored, for example, then set the appropriate indicator. The checkbox is set by default.
   - The *Use Key Date as Base Date* checkbox is set by default. Deselect the indicator if you want to use a specific date for the data records that are to be stored in the Result Database (RDB).
   - In the *Key Date* field specify the name of the key date of access to the data source. Today's date is used as the default value for this field. For more information about the key date, see Key Date Queries and Delta Queries [page 267].
   - In the *System Time of Worklist* field, enter the system date and time from which, and through to which, data changes are to be taken into account. If a delta query can be used with the primary data source in question, you can start the query for the time period you require by entering values for both time points. For more information about the primary data sources, see Servers for General Primary Data Sources [page 1597].

- In the *System Time of Enrichment* field enter the time point that the system is to use to access the primary data sources of the enrichment that are not in the same data read group as the primary data sources accessed by the *system time of the worklist*.

3. Make the following settings in the *Data Collection* data group. Use the *Group ID of Selection* and the *Selection ID* fields to define the data source and the processing steps:
   - Group ID of the selection
   - Selection ID for primary and secondary data sources
   - You can use *Selection Criteria* to define which data is to be selected, for example, by specifying the business partner. We recommend that you use selection criteria, particularly for test runs.
   - Once you have entered values for all the required entry fields, by choosing *Display Details*, you can display all the relevant Customizing settings in a dialog box. Choose the respective tab page:
     - *Characteristics and Key Figures*
     - *Selection*
     - *Selection Options: BAdI* (selection options for worklist generation and BAdI calls)
     - *Data Target Information* (custom fields that contain additional information, which you have stored in Customizing for the default layer).
4. If required, make the following settings in the *Technical Settings* area of the screen:
   - Set the *Parallel Processing* checkbox to speed up processing. It is advisable to use the parallel processing function, particularly if you are processing a large volume of data.
   - Set the *Test Run* indicator to simulate the run for determining default events. The system does not write the results of the test run to the database. They are simply displayed as a list. If required, you can enter a particular layout that is to be used when the system displays the results data.
   - Enter a filter variant ID to define how messages are stored and displayed in the run administration function. You define filter variants in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages*.
5. In the Free Selection Characteristics data group, you can enter more characteristics from the characteristic repository. The system transfers these characteristics through to the secondary data source in the module editor. You can also use a BAdI call to define that when the system generates the worklist, the free selection characteristics are to be used in addition to those you selected by means of the selection criteria. For more information see Customizing for the Determination of Default under *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Determination of Default* > *General Settings for Data Selection* > *BAdIs for Selection* > *BAdI: Selection Options for Worklist Definition*..
   The following data categories are available for the free selection characteristics:
   - In the *Date 1* and *Date 2* fields, you can choose characteristics in date format.
   - In the *Time 1* and *Time 2* fields, you can choose characteristics in time format.
   - In the *Field 1* and *Field 2* fields, you can choose any characteristics.

   Then choose *Enter* to display the names of the characteristics that you have selected.

> **Note**
>
> Unlike the selection criteria, which you add by using the *Selection Criteria* pushbutton, you can derive the data for the free selection characteristics dynamically within the variants.

4.3.6.2 Manual Changes to Default Data

Use

You use this function to change default data manually.

Prerequisites

In Customizing for *Bank Analyzer*, you need to have entered the settings in the *Default Layers* section. For more information, see Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Determination of Default*.

Activities

Selection

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Determination of Default* > *Change Data Manually*.
   The system displays the *Default Data* screen, in which you have the following options:
   In the left-hand part of the screen, the system displays an overview of all the RDB runs. To display the data for a particular RDB run, expand the respective node. To display the default determination runs for a particular RDB run, expand the RDB run.
2. To display the run data in the right-hand part of the screen, double click the required run in the left-hand part of the screen.
   If data exists for the entry selected, the system displays a dialog box in which you can choose a time period, or enter characteristics as selection criteria (for example, you can define that data is displayed for a particular business partner only).

Action

The overview list contains the following pushbuttons:

Pushbutton	Description
 	Expands or collapses the nodes in the left-hand part of the screen (*hierarchy*)
	Updates the display of the runs in the left-hand part of the screen (*hierarchy*)
	Displays or hides the description of the nodes in the left-hand part of the screen (*hierarchy*)

Pushbutton	Description
	Displays the data that can be changed for the run selected
	Displays the changes that have already been made
	Displays the logs of the selected run in the bottom right-hand part of the screen

The system displays the maximum number of hits in the upper window in the right side of the screen. The default value for the maximum number of hits is 500. To change the maximum number of hits, click the number that is displayed and overwrite it with the number you require.

In the overview list in the right-hand part of the screen, you can use the following functions for default determination runs:

Pushbutton	Description
 	Enables you to delete multiple rows at the same time: • Choose the *Multiple Selection Mode* pushbutton. You can now select and delete multiple rows at the same time. • Choose the *Single Selection Mode* pushbutton to return to the original view, which contains the other pushbuttons.
	Enables you to change the data for a row: 1. Select the row in which you want to change data. 2. Choose the pushbutton. 3. The system displays a dialog box in which you change the data as required. It is not possible to change the data in certain key fields, such as *Financial Transaction ID*, *Base Date for Default Determination Run*. 4. Choose  . The changes are written to the database.
	Enables you to change the data in a column: 1. Select the column in which you want to change data. 2. Choose the pushbutton. 3. The system displays a dialog box in which you can enter the new value for all the records selected. 4. Choose  . The changes are written to the database.

	1. Choose the pushbutton. 2. In the dialog box that is then displayed, enter the required data. 3. Choose  . The new data record is written to the database.
	1. Select the row that you want to copy. 2. Choose the pushbutton. 3. The system displays a dialog box in which you enter the data as required. The values of the row that you want to copy are displayed here as default values. This means that you need only to change the values as required, rather than entering new data. 4. Choose  . The new data record is written to the database.
	1. Select the row that you want to delete. 2. Choose the pushbutton. 3. In the dialog box that is then displayed, confirm that the row is to be deleted The data record selected is deleted from the database.
	1. Select the row for which you want to display the transaction data that is stored in the Source Data Layer (SDL). 2. Choose the pushbutton. The system displays the transaction data in a new window.

Result

You have changed the default data manually, and stored the changes on the database.

If you have set up a release procedure (principle of dual control [page 1849]), the changes do not take effect immediately. They are released as part of the workflow. Until the changed data is released, the system continues to use the old data record in all applications. This data record cannot be changed again until the first change has been released.

You define the release procedure in Customizing for *Determination of Default* under *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Determination of Default* > *Configure Default Determination* > *Release Procedure for Default Layers*.

4.3.6.3 Tools

The deletion report *Delete Default Stress Data* is available as a tool in the Determination of Default component. For more information about the deletion report, choose with the quick info *Program Documentation*.

> ⚠ Caution
>
> The deletion report does not contain a function for checking whether the data you want to delete is still being used. Before you start any of the report, make sure that the data you want to delete is not being used anywhere else.

4.3.7 Common Functions

Common functions comprise a common run administration for the following Bank Analyzer components:

- Account Pooling
- Free Line for Facilities (facility distribution)
- Collateral Distribution
- Determination of Default

In addition, they provide you with an overview of the following functions:

- Postprocessing
- Online Detail Reporting
- Archiving
- Stress Tests

4.3.7.1 Postprocessing

Use

In the account pooling function and the function for the determination of the free line for facilities, it may be the case that the initial data taken from the Source Data Layer (SDL) [page 100] and used in these functions was incorrect. If so, you need to correct the data in the SDL and recalculate the transaction in question. You can use the postprocessing function to do this without having to start the entire run again. The system deletes the results data stored in the Results Data Layer (RDL) [page 1364] or Result Database (RDB) [page 1823] and replaces it with the new results data.

You can also include additional transactions for a run that was already completed. This enables you to add the data for important transactions not included in those selected in the original run to the results already stored in the RDB.

> ℹ Note
>
> Note that there is a difference between postprocessing a run and restarting a run. You can use the postprocessing function to correct a complete run by changing just a subset of the data or transactions. If you restart the run, the system continues calculating a (mass) run that terminated due to technical reasons.

Prerequisites

To be able to postprocess a run, it must have the status *Run Completed Successfully* (End). Runs that are completed (`Cmtd`) or that are flagged for deletion (`Del.`) cannot be postprocessed.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods*
   - *Account Pooling* > *Postprocess Account Pooling*
   - *Free Line for Facilities* > *Postprocess Determination of the Free Line*
2. In the *Selection Criteria for Run* data group, enter the following data:
   1. By entering the relevant combination of external run ID, key date, run category, and run version, you define which run you want to postprocess.
   2. Specify the time stamp for the start of the time period for which data is to be selected for postprocessing. If you choose *Run Admin*, the system uses the timestamp of the last change made the run specified above.
   3. If required, change the time stamp of the access to the data in the Source Data Layer (SDL) and Results Data Layer (RDL). The default values are today's date and the current system time.
   4. If required, set the *Delta Selection* indicator. The delta selection function selects from the SDL only those transactions that were changed between the start time stamp and the SDL time stamp in the SDL. Note that the results for data that has been deleted in the meantime are not changed.
      If you have not set the indicator, the postprocessing function selects the data for the start time point from the Source Data Layer (SDL), and sets the corresponding results in the Results Data Layer (RDL) to *invalid* or deletes them in the Result Database (RDB). The data is selected again and calculated for the SDL time stamp. The new results are stored in the RDL or RDB. This also includes any deleted objects.
      For more information, see Key Date Query and Delta Query [page 267].
3. In the *Settings for Data Selection* data group, enter the InfoSet. The InfoSet defines which data is to be selected and hence enables you to override the Customizing settings for the selection group ID stored in the run type. Once you have selected an InfoSet, choose `Enter`. You can then choose the *Selection Characteristics* pushbutton to enter additional selection criteria for the InfoSet specified.
4. Choose  to start the postprocessing run.

The system stores the changed data in the RDL or RDB. The run then has the status *Run in Postprocessing* (`InPP`).
You can display the results in run administration. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Common Functions* > *Common Run Administration* > *Overview of General Method Runs*.

4.3.7.2 Online Detail Reporting

Use

Online detail reporting enables you to check account pooling and the determination of the free line for facilities by analyzing a smaller set of data online. Looking at a smaller volume of data makes it easier to understand the calculation process. The online run uses the Customizing for the run types but with the following differences:

- In online detail reporting, you can use your own selection criteria to define which transactions are to be analyzed, rather than having to use the criteria stored in the Customizing for selection groups, as is the case for runs scheduled as batch jobs. These selection criteria are completely independent of those stored in Customizing for the selection ID. For example, you can restrict the selection of data to that for particular transactions.
- Results are calculated for one bundle at a time. Once a bundle has been processed by the bundling transformation function, the system displays the transformed bundle in the calculation screen.
- Since the analysis is based on a small volume of data, you do not need to use parallel processing here.
- Unlike the logs for account pooling runs or the logs for determining the free line for facilities, online detail reporting provides you with detailed interim results.
- The results of online detail reporting are displayed only, and are not stored in the Results Data Layer (RDL) [page 1364] or Result Database (RDB) [page 1823].

Prerequisites

You have entered the settings required in Customizing for *Bank Analyzer* under ▶ *Processes and Methods* ▶ *General Calculation and Valuation Methods* ▶ *Account Pooling, Facility Distribution, and Collateral Distribution* ▶. The settings under *Common Settings for Results Data Layer* and *Common Settings for Result Database* are not required for the online detail reporting functions.

Procedure

Selection

1. On the *SAP Easy Access* screen, choose ▶ *Bank Analyzer* ▶ *Processes and Methods* ▶ *General Calculation and Valuation Methods* ▶
   - ▶ *Account Pooling* ▶ *Online Detail Reporting for Account Pooling* ▶
   - ▶ *Free Line for Facilities* ▶ *Online Detail Reporting for Determination of the Free Line* ▶
2. Enter a business record date. The default value is today's date.
3. If required, change the time stamp for access to the data of the Source Data Layer (SDL) and the RDL. The default values are today's date and the current system time.
4. If required, specify a filter variant ID for message storage in the application log. You define filter variants in Customizing for *Bank Analyzer* under ▶ *Basic Settings* ▶ *Settings for Storing Messages* ▶ *Edit Filter Settings for Messages* ▶.

5. If you want to start the online detail reporting for a stress test [page 426] you must also enter the ID of the relevant scenario.
6. In the *Settings for Data Selection* data group, enter the InfoSet. The InfoSet defines which data is to be selected. and hence enables you to override the Customizing settings for the selection group ID stored in the run type.
   Once you have selected an InfoSet, choose *Enter*. You can then choose the *Selection Characteristics* pushbutton to enter additional selection criteria for the InfoSet specified.

   > **Note**
   >
   > Before you execute the run, you can display the Customizing settings relevant for the selection by choosing *Customizing*.

7. Choose to start data selection.

   > **Note**
   >
   > Online detail reporting uses internal variants so that you can call up the analyses again at a later point in time. The system automatically saves the parameters that were used. Note, however, that the internal variants are client-dependent.
   >
   > The system displays an overview of the selected data.

Action

In the overview list of the selected data, you have the following options:

Action	Use
Application Log	Displays the application log. The application log contains data only if you have started at least one calculation run.
Fair Value Server Log	Displays the detail log of the valuations in the Fair Value Server (FVS).
Customizing	Displays the relevant Customizing settings.
Display SDL Objects	Displays the data of the SDL object, such as the basic data of the financial instrument.
Double-click an object (for example, bundle or contract) in the dialog structure	Starts the calculation for the selected object.
Context menu (right mouse button), for example on the trans-action ID in the dialog structure	Displays the data of the SDL object, such as the basic data of the financial instrument.
Display Eligible Collateral Only *Display All Collateral* (top right-hand part of the screen)	If you switch the display mode from *Display Eligible Collateral Only* to *Display All Collateral*, you can display the relationships of all financial transactions on the tab pages, and not just those for eligible collateral.

Action	Use
Tab pages in the results screen	The tab pages in the results screen are based on the approach used, and contain interim results.

You can also use the display function in online detail reporting to display interim results for an account pooling run or for a run to determine the free line for facilities. Proceed as follows:

1. In the overview of runs [page 434] choose the detail view for a run.
2. Choose *RDL Viewer*.
3. In the overview of results objects select a row, and then choose *Branch to Online Detail Reporting*.

> **Note**
>
> You can use this function only if you store results in the Results Data Layer (RDL) [page 1364].

4.3.7.3 Archiving

Use

You can use the archiving tool to delete obsolete General Calculation and Valuation Methods data from the system. For more information about the archiving tool, see .

Features

You can use the archiving functions for the following General Methods:

- Archive Data from Account Pooling [page 420]
- Archive Data from Facility Distribution [page 420]
- Archive Data from the Free Line for Facilities [page 420]
- Archive Data from Collateral Distribution [page 420]
- Archiving of Default Data [page 422]

4.3.7.3.1 Archiving of Data From General Calculation and Valuation Methods

You can use the following archiving objects to archive data for general calculation and valuation methods and destroy it as part of SAP Information Lifecycle Management (ILM):

- `BA1_R4_071` *(RDB: Archiving for Pooling)*
- `BA1_R4_072` *(RDB: Archiving for Facility Distribution)*

- BA1_R4_073 *(RDB: Archiving for Collateral Distribution)*
- BA1_R4_074 *(RDB: Archiving for Distribution of Free Line)*

> **Note**
>
> Use archiving objects BA1_R4_068*(RDB: Archiving of GM Data - Default Data)* andBA1_R4_069*(RDB: Archiving of GM Data - Default Stress Data)* for data relating to the determination of default. For more information, see *Related Links*.

For these ILM-enabled archiving objects, you can execute the following programs using Archive Administration (transaction SARA):

- /BA1/DE_ARCHIVE_WRITE to write the archive files
- /BA1/DE_ARCHIVE_DELETE to delete data from the operational system
- /BA1/R4_ARCHIVE_RELOAD to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction SARA), enter the archiving object, and choose Database Tables.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.
   The *Archive Administration: Initial Screen* appears.
2. Choose the required object name. For example, BA1_R4_071 *(RDB: Archiving for Pooling)*.
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* fields to select the data to be archived (for example, pooling data).
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.
    The system displays a list of data (for example, pooling data). In this list, you can select individual items (for example, pooling data) that you want to archive.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]
Archiving and Destroying Default Data [page 422]

4.3.7.3.2 Archiving and Destroying Default Data

You can archive default data using the following archiving objects, and destroy it as part of SAP Information Lifecycle Management (ILM):

- `BA1_R4_068` *(RDB: Archiving of GM Data - Default Data)*
- `BA1_R4_069` *(RDB: Archiving of GM Data - Default Stress Data)*

> **Note**
>
> Note that you cannot archive individual default determination runs. The archiving function always archives complete RDB runs, each of which usually consists of multiple default determination runs or stress runs in Determination of Default.

You can execute the following write, delete, and reload programs for these ILM-enabled archiving objects in archiving administration (transaction `SARA`):

Archiving Object	Programs

`BA1_R4_068`	`/BA1/RA_ARCHIVING_WRITE_DEF`
	`/BA1/RA_ARCHIVING_DELETE_DEF`
	`/BA1/RA_ARCHIVING_RELOAD_DEF`

`BA1_R4_069`	`/BA1/RA_ARCHIVING_WRITE_SCEN`
	`/BA1/RA_ARCHIVING_DELETE_SCEN`
	`/BA1/RA_ARCHIVING_RELOAD_SCEN`

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction `SARA`), enter the name of the archiving object, and choose *Database Tables*.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*. The *Archive Administration: Initial Screen* appears.
2. Choose the required object name. For example, `BA1_R4_068` *(RDB: Archiving of GM Data - Default Data)*.
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* fields to select the data to be archived (for example, GM default data).
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.

11. Choose *Execute*.
    The system displays a list of data (for example, GM default data). In this list, you can select individual items (for example, GM default data) that you want to archive.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

4.3.7.3.3 Control of Archiving for Default Data

Use

This function is supplementary to the archiving report for Determination of Default. You can use this function to manage the relevant RDB runs, to flag them for archiving, and to monitor their status, for example.

Features

Default data and default stress data is archived for entire RDB runs. One RDB run contains multiple default determination runs. You define how many default determination runs an RDB run contains by specifying the RDB run validity in months. You do this in the Customizing for the Determination of Default under *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Determination of Default* > *Configure Default Determination* > *Configure Default Layers*.

> **Note**
>
> For performance reasons, we recommend that you use as large a time period as possible for the validity of the runs in months (for example, 24 months). However, using the smallest possible time period (for example, 3 months) gives greater flexibility. Therefore, you need to find a balance between the need for greater flexibility and for optimal system performance.

You can also change the validity of an RDB run in archiving control. However, make sure that you choose a time period that at least contains the times of the default determination runs that the RDB run already contains. You can also create this run version manually in archiving control.

If you have already started archiving for a particular time period, and then want to start a default determination run within this time period, the system creates a new version of the run for the RDB run.

Activities

On the SAP Easy Access screen, choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Determination of Default* > *Archiving* > *Control of Archiving for Default Data*.

The Control of Archiving screen appears. You have the following options:

- In the left-hand part of the screen, under the *Default Data* node, the system displays an overview of all the RDB runs. The runs are listed in chronological order. Under the *Default Data (Scenario Area)* node, the system displays an overview of the RDB stress runs in Determination of Default. To display the data for a particular RDB run, expand the respective node. To display the default determination runs for a particular RDB run, expand the RDB runs. In the RDB run, you can monitor the status of the run by means of the following symbols:
  Status Display

Status Display	Comments
Not Archived	The RDB run has not yet been archived. You can flag the run for archiving. To do so, select the run and choose the push-button *Flag for Archiving/Deregister*. By choosing the push-button again, you can reset the flag.
	The RDB run has been flagged for archiving.
	The RDB run has been archived. This means that archive files were generated. You can now delete the run from the database.
	The RDB run has been deleted from the database. You can read that data in the archive, or reload the data if required.
	The RDB run has been reloaded. This means that the database contains the data once again. **Note** You can delete the data for reloaded runs again.
	The RDB run is write-protected. This status can be set for the following reasons: ○ The RDB run is being archived. ○ The archiving of the RDB run was terminated due to an error. Restart archiving.

- In the right-hand side of the screen, choose the *RDB Data* tab page to display information about the RDB run. If required, you can change the validity dates of the run. On the *Run Data* tab page, you can display information about the default determination run.

- Depending on which level you have selected in the left-hand part of the screen, the system displays the following pushbuttons:
  Pushbuttons

Pushbutton	Comments	Display
	Expands or collapses the nodes in the left-hand part of the screen	General display
	Updates the display of the status of the RDB runs	General display
	Displays or hides the description in the left-hand part of the screen	General display
	Sets an RDB run to the status *Selected for Archiving*. If you choose this push-button for an RDB run that is already selected for archiving, this resets the selection. In other words, the run is de-registered for archiving.	Display of the RDB run
	Creates a new RDB run	Display of the RDB run
	Switches to change mode so that you can change the validity dates for an RDB run	Display of the RDB run
	Flags a default determination run as a complete run (the run contains a version for all relevant characteristic values) or as a delta run (the run contains only a subset of all the relevant characteristic values)	Display of the determination of default run or the default stress run
	Within a default layer, for example, this function displays the time periods that are not covered by an RDB run	Display of the RDB run

For information about the archiving process, see the document Archiving of Default Data [page 422].

4.3.7.4 Stress Tests

Use

To meet the requirements of Basel II, banks that use the IRB approach must have sound stress test methods that enable them to assess their total capital adequacy. The banks are responsible for configuring the stress tests. The tests must be both meaningful and conservative.

Stress tests are also of interest to banks that use the standardized approach if, for example, migration analyses are to be carried out for external ratings. Banks that use the standardized approach are also required to assess their capital adequacy. Stress tests can be used for this purpose.

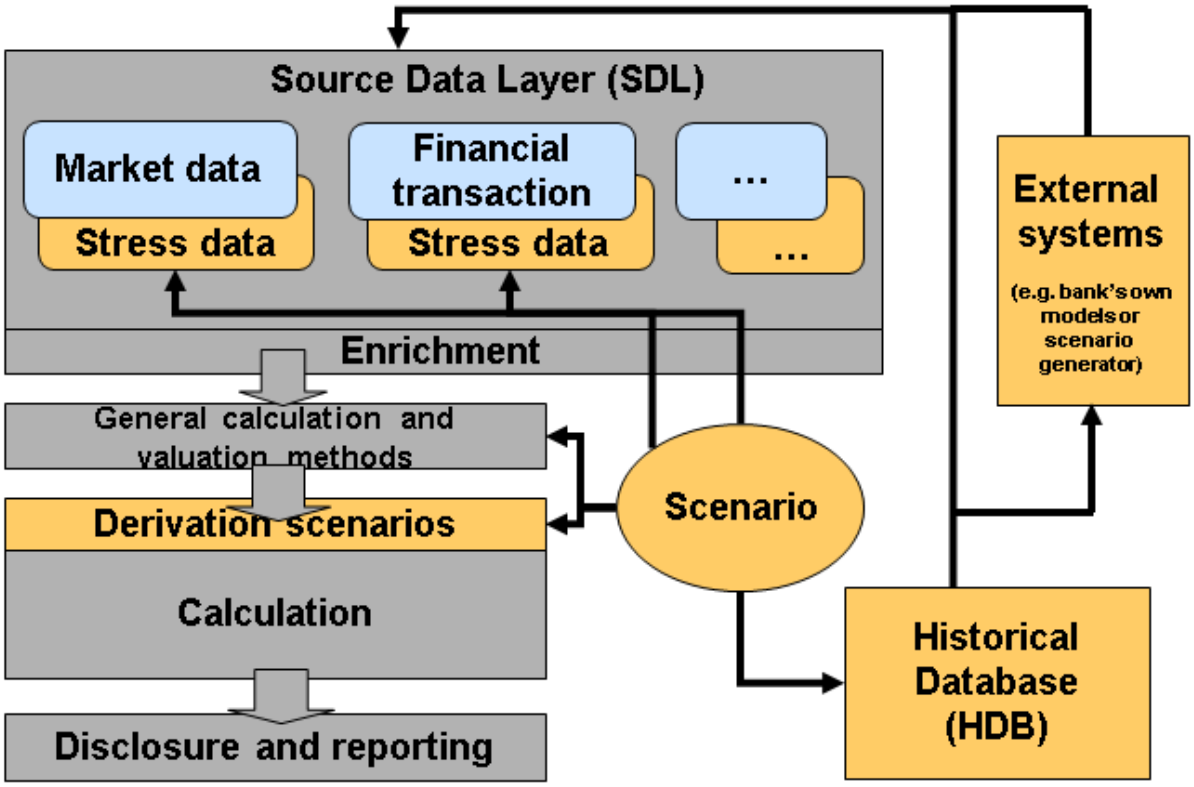
Stress tests use input data and control parameters that have been modified to reflect the (possibly hypothetical) result of a negative change in the economic conditions.

The following are examples of stress tests for Basel II:

- Modification of classic market data, such as exchange rates, interest rates, or security prices, and the effects of these on the market value of derivatives
- Modification of the PD and LGD as the result of a crisis in a particular sector or country
- Worsening of the ratings of counterparties in particular countries

Integration

The figure below shows which Bank Analyzer functions are used in stress tests.



Framework for Mapping Stress Tests

- Stress test scenarios in the Source Data Layer (SDL), for example:
  - Assumption of a worsening credit rating
  - Modified maturities
  - Modified positions
  - Modified market data

  For more information, see Scenario [page 1731] and Retrieving Stress Data [page 1734].
- Stress test scenarios in the calculation process (using derivation or using modules from the module editor), for example:
  - Country-specific rating changes

  You can carry out stress runs for the general functions and for the credit exposure calculation. For more information, see Support for Stress Tests in General Calculation and Valuation Methods [page 429] and Credit Exposure Stress Run [page 1261].
- Stress test scenarios are supported by the Historical Database (HDB), for example:
  - Modification of the input data used in banks' own models
  - Modification of position data in the Source Data Layer

  The HDB contains stress runs for banks' own models. For more information, see Support for Stress Tests in the Historical Database [page 1505].

You define which Bank Analyzer functions are involved in a stress test by creating a stress test scenario (see Stress Test Scenarios [page 428]). Here you can also combine multiple scenarios. By specifying whether factors are to be stressed implicitly or explicitly, for example, you define which scenario is depicted in a stress run.

4.3.7.4.1 Stress Test Scenarios

Definition

Different scenarios can be used for stress tests. These can be categorized as follows:

- Market data scenarios
  These are based on the use of modified classic or generic market data.
- Class determination scenarios
  These are based on the use of primary object data that has been modified.
- Modification of the input data used in banks' own models

When market data scenarios and class determination scenarios are used, the calculation process uses modified input data, which is stored along with the original data in the Source Data Layer (SDL). For more information, see Scenario [page 1731].

Use

To be able to map scenarios that reflect crises in particular regions or sectors, the scenarios do not always have to include all transactions and business partners during a calculation run. Instead, they can be limited to particular groups or individual entities. The following application areas are available for scenarios:

- **Local**
  Scenarios that are applied locally use data that is manipulated locally. This means that attributes are changed at *single transaction* or *business partner* level.
- **Global**
  Scenarios that are used globally apply modified values to the entire portfolio.
- **Subportfolio**
  Scenarios that are used for subportfolios apply modified values to a group of single transactions.

Integration

Bank Analyzer provides an infrastructure that enables you to combine stress test scenarios You define the combination of scenarios in scenario administration by assigning multiple single or composite scenarios to a composite scenario.

You define composite scenarios in Customizing for *Bank Analyzer* under ▶ *Infrastructure* ▶ *General Scenario Administration* ▶ *Edit Composite Scenarios* ◀. You can still change these settings after your system has gone live. You do this on the *SAP Easy Access* screen under ▶ *Bank Analyzer* ▶ *Infrastructure* ▶ *General Scenario Administration* ◀.

A composite scenario can contain only one scenario from each data category for scenarios, for example a market data scenario and a business partner scenario.

4.3.7.4.2 Support for Stress Tests in General Calculation and Valuation Methods

Use

The general calculation and valuation methods are mainly used to prepare data for processing in the calculation processes in Credit Exposure. The results data from these methods can therefore provide the input data for the credit exposure calculation. So that a consistent framework is used for Credit Exposure stress tests [page 1261], the general calculation and valuation methods provide the following stress tests:

- Stress run for account pooling
- Stress run for facility distribution
- Stress run for determination of the free line for facilities
- Stress run for collateral distribution
- Stress run for determination of default

> **Note**
>
> The stress tests do not modify data by applying scenarios. They read stressed data from previous processes, which means that the results of the stress tests are stressed indirectly.

Prerequisites

1. The Foundation component contains stressed market data, or the Source Data Layer (SDL) contains stressed data for the primary objects. For more information, see Retrieving Stress Data [page 1734].
2. You have created scenarios and assigned them to a composite scenario in Customizing for *Bank Analyzer* under *Infrastructure* > *General Scenario Administration*.
3. The same prerequisites apply for stress tests in the general calculation and valuation methods as for the corresponding update runs. For more information, see the following:
   - Account pooling [page 383]
   - Distribution of facilities [page 399]
   - Determination of the free line for facilities [page 395]
   - Collateral distribution [page 404]
   - Determination of default [page 411]

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods*:
   - *Account Pooling* > *Start Stress Run for Account Pooling*
   - *Account Pooling* > *Start Stress Run for Account Pooling for a Run Type*
   - *Free Line for Facilities* > *Start Stress Run for Determination of the Free Line*

- *Free Line for Facilities* > *Start Stress Run for Determination of the Free Line for a Run Type*
- *Free Line for Facilities* > *Start Stress Run for Facility Distribution*
- *Free Line for Facilities* > *Start Stress Run for Facility Distribution for a Run Type*
- *Collateral Distribution* > *Start Stress Run for Collateral Distribution*
- *Collateral Distribution* > *Start Stress Run for Collateral Distribution for a Run Type*
- *Determination of Default* > *Start Stress Run for Determination of Default*

2. To start a stress run, proceed in the same way as for an update run.
3. You also need to enter the ID of the composite scenario that contains the single or composite scenario [page 428] to be used.

The results of the stress tests in the general calculation and valuation methods are stored in the Results Data Layer (RDL) or the Result Database (RDB) and can be accessed from there and used in stress testing in Credit Exposure. However, the results from the stress tests can still be identified. You can display the results in run administration.

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Common Functions* > *Common Run Administration* > *Overview of General Method Runs*.

Recommendation

To display the results of stress runs in run administration, restrict the selection to the relevant run category (in this case `STR`).

4.3.7.5 Run Administration

Definition

Run administration includes the following functions:

- Execute or create run
- Display an overview of runs
- Display Application Log
- Edit run
- Manage run
- Replace run
- Select run for archiving
- Delete run
- Log of the deletion function

Note

The above functions are not all available for each application. For more information, see the application-specific documentation.

Use

The following table lists the runs available for each application:

Application	Run
General Calculation and Valuation Methods [page 376]	Account Pooling [page 383] or Account Pooling (CVPM) Facility Distribution [page 399] or Facility Distribution (CVPM) Determination of the Free Line for Facilities [page 395] or Determination of the Free Line (CVPM) Collateral Distribution [page 404] or Collateral Distribution (CVPM) Determination of Default [page 410] Stress Runs [page 429]: • Stress run for account pooling or stress run for account pooling (CVPM) • Stress run for facility distribution or stress run for facility distribution (CVPM) • Stress run for determination of the free line or stress run for determination of the free line (CVPM) • Stress run for collateral distribution or stress run for collateral distribution (CVPM) • Stress run for determination of default
Credit Risk [page 1191]	Credit Exposure Run [page 1238] or Credit Exposure Run (CVPM) Country Risk Run [page 1336] (prototype for country risk) **Stress run:** Credit Exposure Stress Run [page 1261] or Credit Exposure Stress Run (CVPM)

Application	Run
Historical Database [page 1478]	**Version management:** Run for generating historical data for data layers [page 1481] Run for generating historical data for the bank's own models [page 1484] Uploading of Files [page 1486] **Calculation functions:** Determining Default Rates [page 1488] Determining Average Default Rates [page 1491] Determining Default Figures [page 1493] Calculation of Migration Matrices [page 1496] **Data retrieval:** Run for supplying data to models [page 1498] Downloading of Files [page 1500] **Stress tests:** Stress Run for Supplying Models with Data [page 1505] Generating Scenario Data in the Source Data Layer [page 1506]
Generic BI Data Extraction [page 1638]	Testing the BI Extractor [page 1639] BI extraction run **Note** Extraction runs are created and executed in SAP NetWeaver Business Intelligence (BI). The system displays information about extraction runs in run administration of Bank Analyzer.
Regulatory Reporting Interface	Data Extraction Runs
Limit Manager [page 1529]	Limit Utilization Run [page 1551]
Fair Value Effectiveness Test for Hedging Relationships [page 1036]	Effectiveness Test Run
Cash Flow Hedge Analysis [page 1107]	Valuation Run [page 1128] Subvaluation Run Aggregation Runs [page 1129]

Application	Run
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169] Portfolio Item Runs [page 1172] • Period-end run for portfolio items (subledger scenario) • Period-start run for portfolio items (subledger scenario) • Portfolio item run (merge scenario)

> **Note**
>
> For some of the Bank Analyzer components, you can use the Schedule Manager to schedule and control jobs. If you use multiple applications, you can define the sequence in which the runs are to be carried out. For more information, see Schedule Manager [page 437].

More Information

Status Overview for Run Administration [page 1248]

4.3.7.5.1 Special Features of Run Administration

Use

Run administration manages information about the runs completed in General Calculation and Valuation Methods. This information can include, for example, the date of the run, the version of the run based on a key date, the user, the time and date the user started the run, the name of the user who made the latest changes to the run, and the time and date the user made those changes, the run category, and the current status of the run. You can branch from the overview screen to the logs for the results that were written to the RDB, and to the application log for the run.

Prerequisites

You have started at least one run in General Calculation and Valuation Methods.

Features

Run administration includes the following functions. You access these functions on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Common Functions* > *Run Administration*

- Overview of Runs [page 434]
- Deletion Function [page 436]
- *Log of the Deletion Function*: Provides an overview of the runs that were deleted.
- *Release Function*

4.3.7.5.2 Overview of Runs

Use

This report enables you to display in the the actual runs and stress runs you have carried out. In the detail log for a run, you can change the status of the run manually. For example, you can complete a run, flag a run for deletion, or set the status of a run to *Completed Successfully*.

Prerequisites

You have already started at least one run.

Features

To run the report, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Common Functions* > *Common Run Administration* > *Overview of General Methods Runs*.

The *Run Administration: Overview of General Methods Runs* screen appears.

Selection

Use the following input fields to select the run you require:

- Client of run administration (for example, `74` for determination of the free line for facilities)
- External ID of run
- Key Date
- Run Category
- Version Run
- System Status

Result

The system displays the results in the ALV. By double clicking on a row you can branch to the detailed view for the selected run. Depending on the run, the status of the run, and where you store results data, the system displays different pushbuttons that you can use to navigate to other functions:

> **Note**
>
> You define whether results data is stored in the Results Data Layer (RDL) or in the Result Database (RDB) in Customizing under *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Storage of Results* > *Define Results Storage*.

Navigation Using Pushbuttons	Use
Display	Switches from change mode to display mode.
Change	Switches from display mode to change mode.
Set Deletion Flag	Changes the status of the run to *Deletion Flag* (`Del.`). You have to flag a run for deletion before it can be deleted (see also Deletion of Credit Exposure Runs [page 436]).
Reset Deletion Flag	Resets the deletion flag previously set.
Close Run	Changes the status of the run to *Run Completed* (`Cmtd`).
Reset Status of Run	Resets the status of a run.
Detail Log	Displays additional information for a run, and allows you to navigate to the master data, for example, of the transaction.
Application Log	Displays the application log for a run. The filter variant ID you entered before you started the run defines which types of messages are displayed.
Filter Log	You can use a filter in the log to restrict the number of messages that are displayed. This is particularly useful if the log contains a large number of messages and you want to display a certain subset of them.
Flag Run as Obsolete	Changes the status of the run to *Run is Obsolete* (`ROBS`).
Clear Flag for Obsolete Run	Resets the *Run Obsolete* status change.
Permit Manual Changes	Enables you to make manual changes to a Determination of Default run [page 413].
Prevent Manual Changes	Prevents you from making manual changes to a Determination of Default run.
Restart	Restarts the run.
Stop Run	Cancels the run.

RDL Viewer	Can be used to navigate to the overview of the result objects stored in the RDL. In the overview of the result objects you can display more information about each result object, branch to online detail reporting [page 418] and change the status of a result object.
Time Stamp	Displays the timestamp collections that were used when the run was started.
Additional RDL Information	Displays additional information about the results data stored in the RDL.
Postprocessing	Displays information about postprocessing [page 416], such as the timestamp collection and the selection characteristics.

If you entered Determination of Default as the client of the run, the *Additional Information* section in the lower part of the screen contains tab pages that give you information about the following settings used for the run:

- Control Parameters
- Selection
- Aggregation
- Selection options in the worklist, and the BAdI implementations that were used
- Report parameters

4.3.7.5.3 Deletion of Runs

Use

You use the deletion report to delete individual runs, or to delete all the runs in General Calculation and Valuation Methods that you have flagged for deletion. You can also carry out a test run for the report. The system records a log of the deletion process for each run.

Note that once you have deleted a run, it is deleted permanently and cannot be recovered.

Prerequisites

The runs that you want to delete have to have the system status *Flagged for Deletion*. You can set and clear the deletion flag on the Overview of Runs [page 434] screen.

Activities

1. To run the report, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Common Functions* > *Common Run Administration* > *Deletion Function*.
   The *Run Administration: Delete Runs in General Methods* screen appears.
2. Make the following settings in the *Selection Criteria of Run* data group:
   - Client of run administration (for example, **74** for the determination of the free line for facilities)
   - External ID of run
   - Key Date
   - Version of Run
3. Choose one of the following *Control Parameters*:
   - Set the *Delete Immediately* checkbox to delete all the runs that are selected straight away.
   - Set the *Test Run* checkbox to display all the runs that can be deleted.
4. Choose *Execute*.
   - If you deleted the selected runs immediately, these are displayed in the .
   - If the run was a test run, the runs that can be deleted are shown in the ALV. You cannot delete any of these runs from the list.
   - If you did not set any checkbox, the runs that are to be deleted are shown in the ALV. From this list, you can delete the entire worklist or just individual runs. Once the runs are deleted, the system regenerates the list.

4.3.7.6 Schedule Manager

Definition

A tool in Bank Analyzer that you can use to execute and monitor complex business processes, such as period-end closing. When you schedule a run, you always have to specify the program with which the system calls the run. There is a separate program for each run category in an application.

The table below shows for which runs and run categories the Schedule Manager can be used, and their corresponding programs.

Application	Run Category	Program
Account Pooling	Account Pooling [page 383]	/BA1/DE_GM_MAIN_COMP
Account Pooling	Stress Run for Account Pooling [page 429]	/BA1/DE_GM_MAIN_COMP_SCEN
Determination of the Free Line for Facilities	Determination of the Free Line for Facilities [page 395]	/BA1/DE_GM_MAIN_FL

Application	Run Category	Program
Determination of the Free Line for Facilities	Stress Run for Determination of the Free Line for Facilities [page 429]	/BA1/DE_GM_MAIN_FL_SCEN
Distribution of Facilities	Distribution of Facilities [page 399]	/BA1/DE_GM_MAIN_RLV
Distribution of Facilities	Stress Run for Facility Distribution [page 429]	/BA1/DE_GM_MAIN_RLV_SCEN
Collateral Distribution	Collateral Distribution [page 404]	/BA1/DE_GM_MAIN_SV
Collateral Distribution	Stress Run for Collateral Distribution [page 429]	/BA1/DE_GM_MAIN_SV_SCEN
Determination of Default	Determination of Default [page 411]	/BA1/RA_DFLT4280_RUN
Determination of Default	Stress Run for Determination of Default [page 429]	/BA1/RA_DFLT4280_RUN_SCENARIO
Fair Value Effectiveness Test [page 1036]	Effectiveness Test Run [page 1039]	/BA1/R8_FVHE_MAIN_RUN
Fair Value Effectiveness Test	Generation of Time Series for Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MAIN_RUN
Fair Value Effectiveness Test	Modifying a Time Series for the Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MODIFY
Cash Flow Hedge Analysis [page 1107]	Subvaluation Run [page 1128]	/BA1/R5_CFHA_BTC_RUN
Cash Flow Hedge Analysis	Aggregation Run [page 1129]	/BA1/R5_START_BATCH_AGGR
Cash Flow Hedge Analysis	Joint call of Valuation Run [page 1111] and Aggregation Run [page 1123]	/BA1/R5_EXECUTE_CFHA
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169]	/BA1/R9_VPM_BTC_ID_RUN
Portfolio Fair Value Hedge	Portfolio Item Run (Merge Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_RUN
Portfolio Fair Value Hedge	Period-End Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PE_RUN
Portfolio Fair Value Hedge	Period-Start Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PS_RUN
Portfolio Fair Value Hedge	Define End of Valuation Period	/BA1HM/RAH_M_PFVH_SET_VP_END
Portfolio Fair Value Hedge	Adjust Initial Portfolio Fair Value Hedges for Next Valuation Period	/BA1HM/RAH_M_PFVH_ADJUST

Application	Run Category	Program
Portfolio Fair Value Hedge	(Re-) Designation of Initial Portfolio Fair Value Hedges	/BA1HM/RAH_M_PFVH_DESIG
Credit Exposure [page 1192]	Credit Exposure Run [page 1238]	/BA1/R2_CRA_MAIN_CRE
Credit Exposure	Credit Exposure Stress Run [page 1261]	/BA1/R2_CRA_MAIN_CRE_SCEN
Credit Exposure	Credit Exposure Run for Revolving Retail	/BA1/RR2_CRE_REVRET
Credit Exposure	Approximate Stress Run [page 1274]	/BA1/RR2_CRE_SCENARIO
Credit Exposure	Aggregation Run for Ad Hoc Calculation [page 1276]	/BA1/RR2_SCEN_AGGR
Country Risk [page 1326]	Country Risk Run [page 1336]	/BA1/R2_CRE_MAIN
Historical Database [page 1478]	Historization Run for Data Layers [page 1481]	/BA1/R6_HIST_RUN
Historical Database	Historization Run for Bank's Models [page 1484]	/BA1/R6_HIST_RUN_MODEL
Historical Database	Determine Default Rates [page 1488]	/BA1/R6_CF_DEFAULT_4210
Historical Database	Determine Average Default Rates [page 1491]	/BA1/R6_CF_DEFAULT_4220
Historical Database	Determine Default Figures [page 1493]	/BA1/R6_CF_DEFAULT_KEYFIGURES
Historical Database	Exporting Data to In-House Models [page 1498]	/BA1/R6_MOD_RUN
Historical Database	Calculation of Migration Matrices [page 1496]	/BA1/R6_CF_MIGRATIONS
Historical Database	Stress Run for Supplying Models with Data [page 1505]	/BA1/R6_MOD_RUN_SCENARIO
Historical Database	Stress Run for Supplying Data [page 1506]	/BA1/R6_FDB_RUN_SCENARIO
Regulatory Reporting Interface	Processing Framework Run Copying the Start ID	/BA1/S1_COPY_STARTID
Regulatory Reporting Interface	Processing Framework Run Creating Runs	/BA1/S1_CREATE_RUN

Application	Run Category	Program
Regulatory Reporting Interface	Processing Framework Run Release Run	/BA1/S1_RELEASE_RUN
Regulatory Reporting Interface	Processing Framework Run Creating a Worklist	/BA1/S1_CREATE_WRKL
Regulatory Reporting Interface	Processing Framework Run Processing a Run	/BA1/S1_PROCESS
Regulatory Reporting Interface	Data Extraction Runs	/BA1/S3_EXTRACT_RUN
Limit Manager [page 1529]	Limit Utilization Run [page 1551]	/BA1/R3_UTILIZATION_RUN_DPF

More Information

Schedule Manager

4.3.8 Cash Flow Generation Method

Use

In Bank Analyzer, you can generate cash flows with the *cash flow generation method*. Cash flows always refer either to the primary object financial transaction [page 122] or the primary object financial instrument [page 129]. A cash flow is comprised of a set of financial flows, which together depict an individual flow of money, (such as disbursement, interest, or repayment).

> **Note**
>
> Financial transactions and financial instruments can consist of a structure with several nodes. In this case, the system generates the cash flows on the level of the individual node. The system generates a cash flow for each node of a financial transaction or financial instrument from a particular business view.

Integration

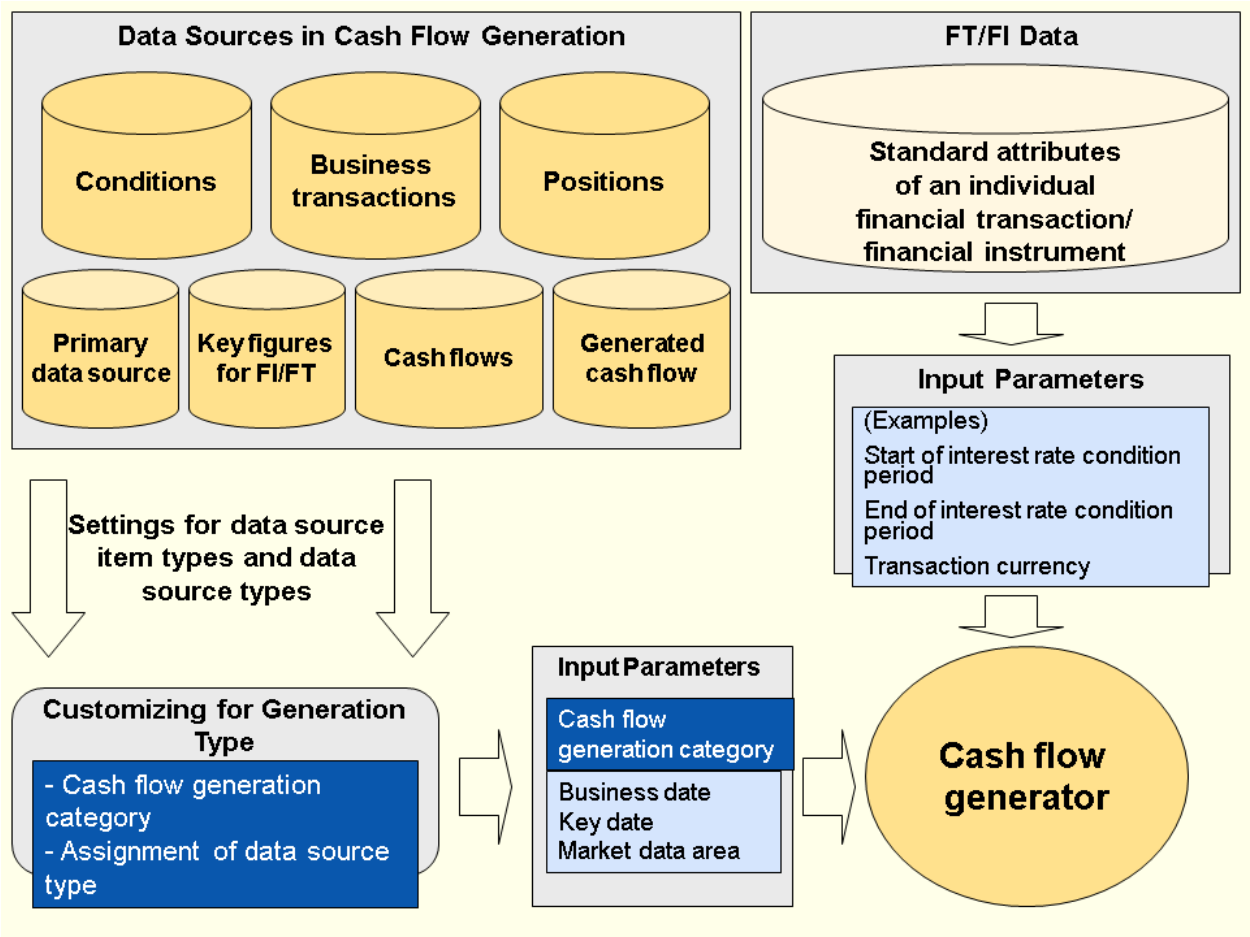
You can also import cash flows for a financial transaction or financial instrument to Bank Analyzer. The *Cash Flow Data* service module or a result type based on result category HKCFD is then used to manage the cash flows. This enables cash flows that represent different cash flow views to be classified using the relevant cash flow categories, and then transferred for further processing.

Prerequisites

You have made the required settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Generation*.

Features

Cash Flow Generation Process



Cash flows are usually generated on the basis of *condition data* that is defined for a financial transaction and, where required, on the basis of *business transactions* and *positions* that are maintained for a financial transaction (financial instrument). First, you have to determine which of these data sources are to be considered in order to generate a specific business view of a cash flow. You can make these settings in Customizing for cash flow generation.

Cash flow generation can also be carried out on stress financial transactions and stress financial instruments. Furthermore, cash flow generation is based on stress positions if these exist and are used.

Customizing Settings

You make all the required settings in Customizing for cash flow generation types [page 461]. The generation type is, therefore, the central parameter in the *Cash Flow Generation* method. When you customize the generation type,

you make the appropriate settings for the view to be generated, the data source types to be considered, and the data source item types of the data source types to be considered.

You use generation categories [page 467] to map the different business views involved in generating cash flows for a financial transaction or financial instrument. A generation category comprises a set of rules that is used to identify the data relevant for this view in the different cash flow data sources and to generate the relevant flows for a template.

A cash flow is always generated on the basis of data for a financial transaction or financial instrument as well as on the basis of data from cash flow data sources that are identified using appropriate data source types [page 464]. The data source types are grouped into data source categories [page 442] and are included in the standard SAP system.

Cash flow data source types consist of individual data source item types that determine how specific parts of a data source are incorporated into the cash flow generation process. This would, for example, be used to determine how individual key figures of a business transaction item from a data source in the *Business Transaction* category are to be incorporated into a cash flow. The data source item types [page 464] are grouped into data source item categories [page 449] provided by SAP.

4.3.8.1 Calculation Bases for Cash Flow Generation

4.3.8.1.1 Data Source Categories

Definition

Used to classify cash flow data sources, and specified when the data source types [page 464] and data source item types [page 464] are defined. A cash flow is always generated on the basis of data from one or more cash flow data sources.

The cash flow generation data sources are grouped into the following data source categories and are predefined by SAP:

- Conditions [page 443]
- Positions [page 444]
- Business Transactions [page 444]
- Key Figures of Transaction/Instrument [page 445]
- Primary Data Source [page 446]
- Cash Flow [page 447]
- Market Data [page 448]
- Generated Cash Flow [page 448]

Use

Uniquely identifying the data relevant for generating a cash flow merely on the basis of the generation type poses certain challenges to the granularity of data sources, which can be incorporated into the cash flow generation process via the data source types [page 464].

For more information, see Cash Flow Generation Type [page 461].

For this reason, when you define a data source type, you have to specify not just the data source category, but also additional attributes so that the data source and, consequently, the data source items can be identified. You have to enter values for certain attributes each time you create a data source type. The attributes you need to define depend on the data source category involved. In the case of the data source category *positions*, for example, the attributes are the *position class* and *account category* for which the position has been defined.

4.3.8.1.1.1 Data Source Category: *Condition*

Definition

An element of a financial transaction or financial instrument that is defined in an object. Conditions describe the structure details of objects (contracts) in terms of time and amount (for example, interest payments and repayments) and are defined in Bank Analyzer for the financial transaction or financial instrument using a special service module (see also Template [page 112]). They contain all the data required to generate condition-based flows when a cash flow is generated.

Use

You use the data source category *conditions* when you want to define a data source type [page 464] to incorporate individual elements of the *conditions* data source, for example, interest rate conditions and repayment conditions, into the cash flow generation process.

For more information, see Data Source Item Category: Condition [page 451].

To enable the relevant condition data to be incorporated into the cash flow generation process, you **always** have to enter values for the following attributes when defining a relevant data source type in the data source category *conditions*:

- Transaction type: Characterizes a set of conditions.
- Condition group type: The condition types that are relevant for a template [page 112] and, therefore, for the associated financial transaction or financial instrument are grouped in the condition group types. You can define your own condition group types using the condition group categories.
  The condition group type *loans*, for example, is shipped by SAP. The condition types *installment repayments* and *nominal interest rates*, for example, are assigned to this condition group type as fixed.

You define new condition group types in Customizing for *Bank Analyzer* under ▶ *Source Data Layer* ▶ *Primary Objects* ▶ *Master Data* ▶ *Settings for Attributes* ▶ *Condition Data* ▶ *Edit Condition Group Types* ■.

4.3.8.1.1.2 Data Source Category: *Position*

Definition

A position [page 142] is an entity for which key figures are recorded on a given date. An account [page 134] is assigned to each position. Key figures might include the daily balance, the end-of-month balance, or the monthly average value of the account. The key figures are taken from the key figure repository.

- You make Customizing settings for positions in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Transaction Data* > *Positions*.
- You create and edit positions from the *SAP Easy Access* screen by choosing *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Position*.

Use

You use the data source category *positions* when you want to define a data source type [page 464] to incorporate individual key figures of a position into the cash flow generation process as a flow or as the calculation base.

For more information, see Data Source Item Category: Flow [page 452] and Data Item Category: Calculation Base [page 454].

To enable the relevant elements of the data source *positions* to be incorporated into the cash flow generation process, you always have to enter values for the following attributes when defining a relevant data source type in the data source category *positions*:

- Position class
- Account category

4.3.8.1.1.3 Data Source Category: *Business Transaction*

Definition

An independent but logically connected business event that implies a change in value and/or quantity, or a result.

Business transactions therefore represent transaction data. A business transaction [page 153] generally entails several flows to several accounts. A business transaction is characterized by its business transaction class and the associated item class.

You make settings for business transactions in Customizing for Bank Analyzer under *Source Data Layer* > *Primary Objects* > *Transaction Data* > *Business Transactions*.

You create and edit business transactions from the *SAP Easy Access* screen by choosing *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Business Transaction*.

Use

You use the data source category *business transactions* when you want to define a data source type [page 464] to incorporate individual components of a business transaction into the cash flow generation process as one of the following:

- Flow
- Calculation base
- Interest rate fixing

For more information, see Data Source Item Category: Flow [page 452], Data Source Item Category: Calculation Base [page 454], and Data Source Item Category: Reference Interest Rate [page 456].

To enable the relevant elements of the data source *positions* to be incorporated into the cash flow generation process, you always have to enter values for the following attributes when defining a relevant data source type in the data source category *business transactions*:

- Business transaction class
- Item class
- Account category

4.3.8.1.1.4 Data Source Category: *Key Figures of Transaction/ Instrument*

Definition

A parameter defined by the user with a quantitative value. When using key figures in conjunction with the data source category *key figures of transaction/instrument*, you have the following options:

- Key figures for the *key figures* service module
  You can use the attributes of the "key figures" service module (see also Template [page 112]) to store time-based information about amounts, quantities, and percentages in a key figure category, financial transaction, or financial instrument. You can define the nominal volume of a loan for EUR 20,000 as amount information using the key figure category *nominal volume*, for example.
  You edit these key figures for a financial transaction (financial instrument) from the *SAP Easy Access* menu by choosing *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Transaction* or *Financial Instrument*.
  You make settings for key figure categories in Customizing for Bank Analyzer under *Source Data Layer* > *Primary Objects* > *Master Data* > *Settings for Attributes* > *Key Figures*.
- Key figures in the key figure repository
  In Bank Analyzer, the key figures are properties and values that the applications can use as a basis for carrying out calculations. These key figures are created in the key figure repository on the basis of the key figure classes shipped by SAP.
- Credit lines for the financial transaction/financial instrument
  You can use the attributes of the *general limit data* service module (see also Template [page 112]) to map credit lines of a current account overdraft facility (financial transaction), for example.

Use

You use the data source category *key figures of transaction/instrument* when you want to define a data source type to incorporate individual elements of the data source *key figures* into the cash flow generation process as one of the following:

- Flow
- Calculation base
- Simulated disbursement
- Distribution of discount
- Rollover portion

> **Note**
>
> Enter values for the attributes when defining a relevant data source type in the data source category *key figures of transaction/instrument* **only** in the following scenario:
>
> If, when you generate cash flows for a generation type node whose underlying template is the parent template of a structured template, you want to use a key figure that is defined for the child template rather than the parent template, you have to edit the parent and child template attributes when you define the data source type.

4.3.8.1.1.5 Data Source Category: *Primary Data Source*

Definition

A standard interface by means of which a worklist or a single record can be imported. A primary data source is a standardized means of accessing elementary data sources in Bank Analyzer and the relevant characteristics and key figures. The dataset determined by a primary data source is defined by the values of the selection characteristics and the values of the granularity characteristics. The primary data source defines which data from the Source Data Layer, Analyzers, or other sources that process data is imported by an application.

In the cash flow generation process, a distinction can be made between:

- General primary data source
  A server that provides data in the form of single records or worklists is always assigned to a general primary data source. You assign primary data sources with a server in Customizing.
  You make the relevant settings for primary data sources in Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources*.
- Object transfer structure (OTS)
  Object transaction structures deliver single records to primary objects from the Source Data Layer.
  You define object transfer structures in Customizing by choosing *Bank Analyzer* > *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Object Transfer Structures (OTS)*.

Use

You use the data source category *Primary Data Source* when you want to define a data source type [page 464] to incorporate individual key figures (from the key figure repository) from a primary data source into the cash flow generation process as one of the following:

- Flow
- Calculation base
- Rollover portion
- Forward interest rate
- Spread

For more information, see Data Source Item Category: Flow [page 452] and Data Source Item Category: Calculation Base [page 454].

To enable the relevant key figures of the primary data source to be incorporated into the cash flow generation process, you always have to enter values for the attributes when defining a relevant data source type in the data source category *Primary Data Source*:

- Primary data source ID
  Each primary data source has an ID that is unique for each primary data source category.
- Primary data source category
  You can use the following primary data source categories:
  - General primary data source
  - Object transfer structure

4.3.8.1.1.6 Data Source Category: Flows

Definition

Flows in the cash flow generation method [page 440] that are part of a cash flow that was generated in a source system and not by the conditions in the Source Data Layer.

Use

You use the data source category flows only when you want to define a data source type [page 464] to incorporate the flows of a cash flow that has already been generated in the **source system** (for example, interest payments and repayments) into the cash flow generation method.

You use the cash flow category to import the cash flow. For more information, see Data Source Item Category: Flow [page 452].

To enable the relevant elements of the cash flows data source to be forwarded, you always have to enter values for the following attribute when defining a relevant data source type in the data source category flows.

4.3.8.1.1.7 Data Source Category: *Market Data*

Definition

The data source category *Market Data* enables you to access the dividends defined in your market data for a financial transaction or instrument, and to generate flows from this.

Use

You use the data source category *Market Data* only if you want to define a data source type [page 464] in order to incorporate into the *cash flow generation* method the flows of a cash flow that has already been generated in the source system (for example, interest payments and repayments).

You use the cash flow category to import the cash flow. For more information, see Data Source Item Category: Flow [page 452].

4.3.8.1.1.8 Data Source Category *Generated Cash Flow*

Definition

A cash flow that is generated on the basis of another generation type within cash flow generation.

Use

You use the data source category *generated cash flow* when you want to define a data source type in order to incorporate into the *cash flow generation* method the components of a cash flow that has already been generated using another generation type (for example, interest payments and repayments).

The source generation type is used to generate the cash flow. You can define the components of the cash flow that are used. You have the following options:

- Conditions
- Items
- Cash flow

4.3.8.1.2 Data Source Item Category

Definition

Data source item categories are used to classify data source items for cash flow generation. The data source item category is stored with a data source category [page 442] when the data source item type [page 464] is defined.

The data source items determine how specific data source elements are incorporated into the cash flow generation process.

> **Example**
>
> This enables you to specify that a key figure from a data source of the category *positions* is incorporated as a credit balance and used as the calculation base for generating cash flows.

The data source items in the cash flow generation process are characterized by the data source item type, which describes one or more data source items.

The data source items are divided into the following data source item categories:

- Condition [page 451]
- Flow [page 452]
- Calculation Base [page 454]
- Reference Interest Rate [page 456]
- Simulated Disbursement [page 456]
- Residual Balance [page 458]
- Scaling [page 458]
- Move Key Figures [page 459]
- Distribution of Premium [page 459] and Distribution of Discount [page 460]
- Rollover Portion [page 460]
- Forward Interest Rate [page 461]
- Spread [page 461]

In the system, the data source category and data source item category assignments are fixed. This means that only certain data source items from a data source can be considered. The following assignments are available:

DSC	Description	DSIC	Description
01	Conditions	01	Condition
02	Cash flow	02	Flow
02	Cash flow	06	Scaling factor
02	Cash flow	07	Move key figures
03	Key figures of transactions and instruments	02	Flow

03	Key figures of transactions and instruments	03	Calculation base
03	Key figures of transactions and instruments	06	Scaling factor
03	Key figures of transactions and instruments	07	Move key figures
03	Key figures of transactions and instruments	08	Simulated disbursement
03	Key figures of transactions and instruments	09	Distribution of discount
03	Key figures of transactions and instruments	10	Rollover portion
03	Key figures of transactions and instruments	6	Scaling factor
04	Generated cash flow	05	Residual balance
04	Generated cash flow	06	Scaling factor
04	Generated cash flow	07	Move key figures
05	Business transactions	02	Flow
05	Business transactions	03	Calculation base
05	Business transactions	04	Reference interest rate
06	Positions	02	Flow
06	Positions	03	Calculation base
08	Market data	02	Flow
10	Primary data source	02	Flow
10	Primary data source	03	Calculation base
10	Primary data source	10	Rollover portion
10	Primary data source	11	Forward interest rate
10	Primary data source	12	Spread

Use

Uniquely identifying the data source items relevant for generating a cash flow merely on the basis of a generation type poses certain challenges to the data source items, which can be incorporated into the cash flow generation process by means of assigning the data source item types to the data source types [page 464]; for more information, see Cash Flow Generation Type [page 461] .

For this reason, when you define a data source item type, you have to define additional attributes so that the data source items can be identified. You have to enter values for certain attributes each time you create a data source item type. The attributes you need to define depend on the data source item category and data source category involved.

In the case of a data source item type with the data source category *positions* and the data source item category *flow*, for example, attributes are the *position class* and *account category* for which the position has been defined.

4.3.8.1.2.1 Data Source Item Category: Condition

Definition

Use

You use this object to define data source item types that characterize data source items from a data source of the *conditions* category.

Examples of data source items in the *condition* category are annuity repayments, credit interest, or fees/commission that are incorporated into the cash flow generation process as condition-based flows. You use the condition types defined in Customizing to define the condition data directly for the financial transaction (financial instrument).

To enable the condition data to be incorporated into the cash flow generation process as a flow, you always have to enter values for the following attributes when defining a relevant data source item type in the data source item category *condition*:

- Condition type
  Condition types are the particular characteristic values of the condition categories and describe the different options of levying interest and repayments. We provide the following condition types:
  - 101100 Nominal interest
  - 103110 Annuity repayment
  - 103120 Installment repayment
  - 103130 Final repayment
  - 103140 Special repayment

  You define additional condition types in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Master Data* > *Settings for Attributes* > *Condition Data* > *Edit Condition Types*.
- Flow type

The flow type is the smallest unit of classification for flows. Examples of flow types include credit interest, fees, and installment repayments.

You make settings for flow types in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Generation* > *Edit Flow Types for Flow*.
We do not provide any fixed flow types.
You can define financial mathematics information (using the *calculation category*) and business information (using the *classification flow type* and the *plus/minus sign of the flow type*) for the cash flow generation process in Customizing.

4.3.8.1.2.2 Data Source Item Category: *Flow*

Definition

Use

You use this object to define data source item types [page 464] that characterize data source items from the following data sources:

- Flows in the data source category *position*
  The flows in the data source category *position* are credit or debit balances, for example, that are to be set as individual flows in a cash flow, depending on the generation category [page 467]. You use a key figure from the key figure repository to define the amount-based value of the balances in the position. A credit or debit balance is set as a flow when you use the fixed-interest period view (see also Basic Views [page 468]) for financial transactions based on the template [page 112] for transactions valid until further notice, for example.
  To enable the key figures of a position to be incorporated into the cash flow generation process as a flow, you always have to enter values for the following attributes when defining a relevant data source item type in the data source item category *flow*.
  - Key figure (from the key figure repository)
    In Bank Analyzer, the key figures are properties and values that the applications can use as a basis for carrying out calculations. These key figures are created in the key figure repository on the basis of the key figure classes shipped by SAP. InfoObjects in the *key figures* category are generated in the associated SAP NetWeaver Business Intelligence (BI) for the key figures created in the core Bank Analyzer system.
    Item categories 02 (Flow) and 03 (Calculation Base) can now be used in conjunction with data source category 05 (Business Transactions) to access defined key figures in the key figure repository.

    > **Note**
    >
    > The InfoObjects generated in BI have the same names as the key figures in the core Bank Analyzer system.
    >
    > You make the relevant settings for key figures in Customizing for *Bank Analyzer*under *Basic Settings* > *Key Figures*.

  - Flow type

See the *flow type* section in Data Source Item Category: [page 451] Condition [page 451].

- Date type of the call date
  Depending on the generation category [page 467], you can incorporate the flows into the cash flow generation process at different points in time.

- Flow in the data source category *business transaction*
  Examples of flows in the data source category *business transactions* are business transaction items for loan disbursements that are to be set as flows in a cash flow, depending on the generation category [page 467]. It makes sense to set a loan disbursement as a flow when you use the plan view (see also Basic Views [page 468]) for a financial transaction based on the template [page 112] for loans, for example.
  To enable the business transaction items to be incorporated into the cash flow generation process as a flow, you always have to enter values for the flow type when defining a relevant data source item type in the data source item category *flow*.
- Flow in the data source category *flows*
  The flows in the data source category *flows* are flows of a cash flow that has already been generated in a source system. These flows are to be incorporated into a cash flow of the *cash flow generation* method, depending on the particular generation category [page 467]. If all generation categories are used, it might make sense to incorporate a flow that has already been generated.
  To enable cash flows that have already been generated in source systems to be incorporated into the cash flow generation process, you always have to enter values for the flow type when defining a relevant data source item type in the data source item category *flow*.
- Flow in the data source category *key figures of transaction/instrument*
  An example of a flow in the data source category *key figures* is the nominal volume of a bond that is to be incorporated once into a cash flow as a flow in the form of a disbursement amount, depending on the generation category [page 467]. The nominal volume can be mapped as a key figure of the *key figures* service module or as a key figure in the key figure repository. For more information, see Data Source Category: [page 445] Key Figures of Transaction/Instrument [page 445].
  To enable key figures to be incorporated into the cash flow generation process as a flow, you always have to enter values for the following attributes when defining a relevant data source item type in the data source item category *flow*.
  - Flow type (see the *Flow type* section in Data Source Item Category: [page 451] Condition [page 451]).
  - Date type of the call date (depending on the generation category [page 467], you can incorporate the flows into the cash flow generation process at different points in time).
- Flow in the data source category *primary data source*
  The flows in a primary data source are key figures that are to be incorporated into a cash flow as individual flows using a primary data source from one of the following categories. The key figures are key figures from the key figure repository:
  - InfoSet
  - Object transfer structure
  - General primary data source

  To enable the key figures for a primary data source to be incorporated into the cash flow generation process as a flow, you always have to enter values for the following attributes when defining a relevant data source item type in the data source item category *flow*.
  - Flow type
    See the *flow type* section in Data Source Item Category: [page 451] Condition [page 451].
  - Key figure (from the key figure repository)
    In Bank Analyzer, the key figures are properties and values that the applications can use as a basis for carrying out calculations. These key figures are created in the key figure repository on the basis of the key

figure classes shipped by SAP. InfoObjects in the *key figures* category are generated in the associated *SAP NetWeaver Business Intelligence (BI)* for the key figures created in the core Bank Analyzer system.

> **Note**
>
> The InfoObjects generated in BI have the same names as the key figures in the core Bank Analyzer system.
>
> You make the relevant Customizing settings for key figures in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Metadata* > *Key Figures*.

4.3.8.1.2.3 Data Source Item Category: *Calculation Base*

Definition

Use

You use this object to define data source item types [page 464] that characterize data source items from the following data sources:

- Calculation base in the data source category *positions*
  An example of a calculation base in the data source category *positions* is a credit balance or a debit balance that is to be used as the basis for calculating interest (debit interest or credit interest) for generating cash flows for a financial transaction. You use a key figure from the key figure repository to define the amount-based value of the balance in the position. A credit or debit balance is set as the calculation base when you use the fixed-interest period view (see also Basic Views [page 468]) for financial transactions based on the template [page 112] for transactions valid until further notice, for example.
  To enable the key figures of a position to be incorporated into the cash flow generation process as the calculation base, you always have to enter values for the following attributes when defining a relevant data source item type in the data source item category *calculation base*.
  - Key figure (from the key figure repository)
    In Bank Analyzer, the key figures are properties and values that the applications can use as a basis for carrying out calculations. These key figures are created in the key figure repository on the basis of the key figure classes shipped by SAP. InfoObjects in the *key figure* category are generated in the associated SAP NetWeaver Business Intelligence (BI) for the key figures created in the core Bank Analyzer system.

    > **Note**
    >
    > The InfoObjects generated in BI have the same names as the key figures in the core Bank Analyzer system. You make the relevant Customizing settings for key figures in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Metadata* > *Key Figures*.

  - Base reference
    Key value for the calculation base of an amount item.

- Start date/end date category
  You can make different settings for the usage period of the calculation base, depending on the generation category [page 467].

- Calculation base in the data source category *business transaction*
  To enable the key figures of a position to be incorporated into the cash flow generation process as the calculation base, you always have to enter values for the following attributes when defining a relevant data source item type in the data source item category *calculation base*.
  - Key figure (from the key figure repository)
    In Bank Analyzer, the key figures are properties and values that the applications can use as a basis for carrying out calculations. These key figures are created in the key figure repository on the basis of the key figure classes shipped by SAP. InfoObjects in the *key figure* category are generated in the associated SAP NetWeaver Business Intelligence (BI) for the key figures created in the core Bank Analyzer system.

    > **Note**
    >
    > The InfoObjects generated in BI have the same names as the key figures in the core Bank Analyzer system. You make the relevant Customizing settings for key figures in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Metadata* > *Key Figures*.
    >
    > Up to now, when a calculation base was generated from a business transaction, the *Nominal Amount* or *Quantity* field of the item was accessed. This procedure is still available, and if you opt for it, you do not have to specify a key figure at this point. However, as of Bank Analyzer Release 3.0, you also have the option of accessing key figures of the category *Amount* or *Quantity* that have been assigned to an item class.

  - Base reference
    Key value for the calculation base of an amount item.
    If you select debit interest, credit interest, or overdraft interest as the base reference (in other words, if a transaction valid until further notice is involved), both the fields for determining the transaction types will be activated.
    The payment date specified in the item is interpreted as the start date of the base reference. This base reference will be valid until it is replaced by another one.

- Calculation base in the data source category *key figures of transaction/instrument*
  An example of a calculation base in the data source category *key figures* is the nominal amount of a financial transaction that is to be used as the basis for calculating interest (nominal interest) for generating cash flows for a financial transaction. The nominal amount can be mapped as a key figure of the *key figures* service module or as a key figure in the key figure repository. For more information, see Data Source Category: Key Figures of Transaction/Instrument [page 445].
  To enable key figures to be incorporated into the cash flow generation process as the calculation base, you always have to enter values for the base reference when defining a relevant data source item type in the data source item category *calculation base*.

- Calculation base in the data source category *primary data source*
  The calculation base in a primary data source is a key figure that is to be incorporated into the cash flow generation process by means of one of the following primary data sources so that it can be used as the basis for calculating interest (the nominal interest, for example) for a cash flow.
  - InfoSet
  - Object transfer structure
  - General primary data source

  The key figures are always key figures from the key figure repository.

To enable key figures of a primary data source to be incorporated into the cash flow generation process as the calculation base, you always have to enter values for the following attributes when defining a relevant data source item type in the data source item category *calculation base*:

- Key figure (from the key figure repository)
  In Bank Analyzer, the key figures are properties and values that the applications can use as a basis for carrying out calculations. These key figures are created in the key figure repository on the basis of the key figure classes shipped by SAP. InfoObjects in the *key figure* category are generated in the associated SAP NetWeaver Business Intelligence (BI) for the key figures created in the core Bank Analyzer system.

  > **Note**
  >
  > The InfoObjects generated in BI have the same names as the key figures in the core Bank Analyzer system. You make the relevant Customizing settings for key figures in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Metadata* > *Key Figures*.

- Base reference
  This is the key value for the calculation base of an amount item.

4.3.8.1.2.4 Data Source Item Category: Reference Interest Rate

Definition

Use

You use this object to define data source item types [page 464] that are distinguished by data source items from a data source in the *business transactions* category.

The data source items in the *reference interest rate* category are reference interest rates that are mapped for the relevant interest rate fixing dates using business transactions. Reference interest rates are incorporated into the cash flow generation process when you use the interest rate condition period view for loan transactions with a variable interest rate.

For more information, see Basic Views [page 468].

4.3.8.1.2.5 Data Source Item Category: *Simulated Disbursement*

Definition

Use

You use this object to define data source item types [page 464] that characterize data source items from a data source in the *key figures of transaction/instrument* category.

The data source items in the *simulated disbursement* category are items that control the disbursement procedure for financial transactions that are not disbursed or that are only partly disbursed. The disbursement procedure controls the simulation of the non-disbursed capital during the cash flow generation process. The following simulated disbursement procedures exist:

- Disbursement procedure for individual financial transactions
  The Key Figures [page 445] service module allows you to define a value-based disbursement procedure for a specific financial transaction. The *valid from* and *valid to* dates of the relevant key figures have to be identical and correspond to the date of the planned incoming/outgoing payment. SAP provides a key figure category for mapping a disbursement procedure in this service module.

  ### Example

  For example, a loan for EUR 100,000 that starts on January 1, 2003 could have the following disbursement procedure in the *key figures* service module:

Valid From	Valid To	Key Figure Category	Amount	Currency
2002–01–15	2002–01–15	PLVAL	50,000	EUR
2002–02–01	2002–02–01	PLVAL	20,000	EUR
2003–01–01	2003–01–01	PLVAL	30,000	EUR

  - Universal disbursement procedure
    You can define disbursement procedures in Customizing. You use a simulated disbursement ID to define disbursement dates relative to the object start date. Due to its general character, information about the amount of the incoming/outgoing payment can only be entered in the form of percentages.
    You can define the simulated disbursement ID for the procedure in the financial transaction. If both a financial transaction-specific and a general disbursement procedure exist for a financial transaction, the financial transaction-specific procedure has priority.

    ### Example

    You use a general disbursement procedure to express the scenario from the example above as follows:

Shift	Shift Unit	Disbursement Amount as Percentage
15	Day	50
1	Month	20

1	Year	30

You make Customizing settings for universal disbursement procedures in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Master Data* > *Settings for Attributes* > *Condition Data* > *Simulated Disbursement Procedure*.
You use the fixed-interest period view for loan transactions that are only partly disbursed to incorporate a simulated disbursement procedure into the cash flow generation process.
For more information, see Basic Views [page 468].
To enable a simulated disbursement procedure to be incorporated into the cash flow generation process, you always have to enter values for the following attributes when defining a relevant data source item type in the data source item category *simulated disbursement*.

- Flow type
  For more information, see the *flow type* section in Data Source Item Category: Condition [page 451].
- Date type of the call date
  Depending on the generation category [page 467], you can incorporate the simulated disbursement procedure into the cash flow generation process at different points in time.

4.3.8.1.2.6 Data Source Item Category: *Residual Balance*

Definition

Use

You use this object only to define data source item types [page 464] that characterize data source items from a data source in the *business transactions* category.

In the case of data source items of the category *Residual Balance*, you can access the residual balance of a cash flow that was generated in the Source Data Layer on the basis of conditions. The residual balance of a cash flow can thus be used as the calculation base for generating another cash flow. You can use this item category in conjunction with data source category 04 (Generated Cash Flow).

4.3.8.1.2.7 Data Source Item Category: *Scaling*

Definition

Use

You use this object to define data source item types [page 464] that characterize data source items from a data source in the *business transactions* category.

In the case of data source items of the category *Scaling*, a standardized cash flow is scaled. The scaling factor can be determined either by specifying a relevant nominal amount or a number of shares (or other financial instrument) for which the cash flow should be generated. You can use this item category in conjunction with data source categories 02 (Flows), 03 (Key Figures of Transaction/Instrument), 04 (Generated Cash Flow).

4.3.8.1.2.8 Data Source Item Category: Move Key Figures

Definition

Use

You use this object to define data source item types that characterize data source items from a data source in the *business transaction* category.

In the case of data source items of the category *Move Key Figures*, key figures are moved within the hierarchy of a structured object so that they can be used in a data source for providing the calculation bases for cash flow generation. You can use this item category in conjunction with data source categories 02 (Flows), 03 (Key Figures of Transaction/Instrument), 04 (Generated Cash Flow).

4.3.8.1.2.9 Data Source Item Category: Distribution of Premium

Definition

Use

You use this object to define data source item types that characterize data source items from a data source in the *key figures of financial transaction/financial instrument* category.

Data source items of the category *distribution of premium* are used to distribute the percentage premiums that are stored in the service module *conditions* according to the distribution procedure selected. Premium flows are

defined for the relevant disbursements in the cash flow. To classify the flows, a flow type is assigned to the relevant data source item type. Simulated premium flows can be classified differently if a different flow type is assigned.

4.3.8.1.2.10 Data Source Item Category: Distribution of Discount

Definition

Use

You use this object to define data source item types that characterize data source items from a data source in the *key figures of financial transaction/financial instrument* category.

Data source items of the category *distribution of discount* are used to distribute the percentage discounts that are stored in the service module "Conditions" according to the distribution procedure selected. Discount flows are defined for the relevant disbursements in the cash flow. To classify the flows, a flow type is assigned to the relevant data source item type. Simulated discount flows can be classified differently if a different flow type is assigned.

4.3.8.1.2.11 Data Source Item Category *Rollover Portion*

Definition

Use

You use this object to define data source item types [page 464] that characterize data source items from a data source in the *primary data source* or *key figures of financial transaction/financial instrument* category.

Data source items of the category *rollover portion* are used to set the residual maturity amount to be extended in a maturity view. The percentage value of the rollover portion is taken from the key figure that was assigned to the relevant data source item category.

4.3.8.1.2.12 Data Source Item Category: Forward Interest Rate

Definition

Use

You use this object to define data source item types that characterize data source items from a data source in the *primary data source* category.

You define the interest rate for the maturity view "forward interest rate + spread" is defined using the data source items of category *forward interest rate*.

4.3.8.1.2.13 Data Source Item Category *Spread*

Definition

Use

You use this object only to define data source item types [page 464] that characterize data source items from a data source in the *primary data source* category.

You define the spread for the maturity view *forward interest rate + spread* using the data source items of category *spread*.

4.3.8.1.3 Cash Flow Generation Type

Definition

Determines how cash flows are generated on the basis of the properties of the relevant node of a financial transaction or financial instrument, in particular on the basis of the underlying template [page 112] and the template hierarchy.

Use

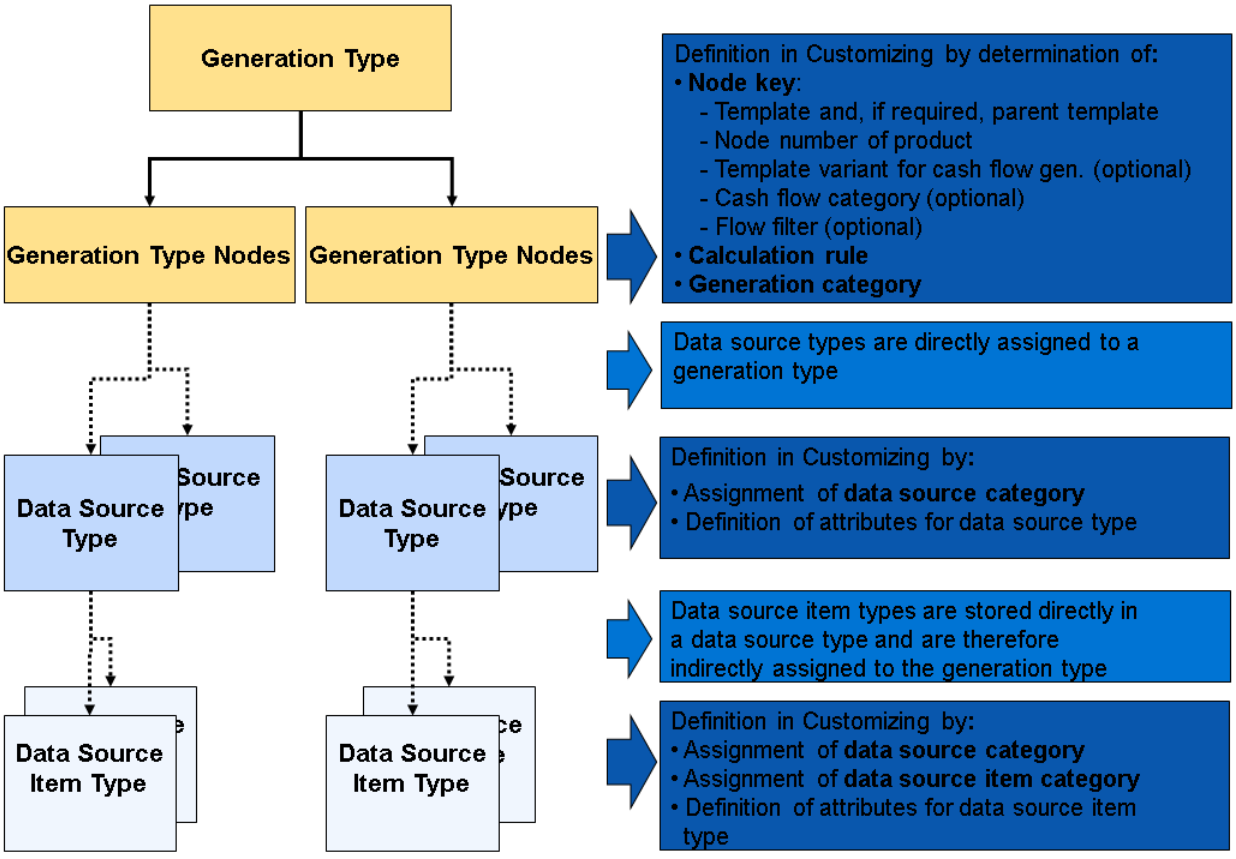
You define generation types in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Generation* > *Edit Generation Types*.

Structure

Generation types are made up of at least one generation type node [page 463].

Data source types [page 464] are also assigned to each generation type node. The data source types, therefore, have to be defined explicitly in Customizing before they are assigned.

A data source type is, in turn, assigned data source item types [page 464], which are assigned indirectly to the generation type as a result. The data source item types also have to be defined explicitly in Customizing before they can be assigned to a data source type.



Structure of the Generation Type

4.3.8.1.3.1 Generation Type Nodes

Definition

The generation type consists of one or more template-dependent generation type nodes. A generation type node determines how a cash flow is generated for a particular template. Since templates can also be included in a template hierarchy, the position of the template in the template hierarchy has to be taken into consideration.

Use

At financial transaction or financial instrument node level, the generation type node determines which cash flow view is to be generated for this node, and according to which calculation rule.

Structure

A generation type node is defined by the following elements:

- The underlying template [page 112] of the relevant financial transaction node
- The template of the higher-level node (parent template)
- Node number:
  Nodes with the same parent template/child template relationship are developed sequentially.
- Template variants:
  This is necessary if a generation type is to contain more than one generation type node for the same template/parent template/node number combination.

A generation type node is, therefore, assigned exactly one generation category [page 467], which describes the business semantics of the cash flow generation process and that controls how the incoming data is processed. Each cash flow view (plan view, interest commitment view, and so on) corresponds to exactly one generation category. The generation categories are predefined as fixed system settings.

A generation type node is assigned exactly one calculation rule, which defines the calculation algorithm for cash flow generation. The following calculation rules are available:

-
  By default, cash flows are generated using financial mathematics.
-
  You use Cash Flow Calculator for the financial transaction 'loan'. Two calculation algorithms can be used: Actual principle and planned principle. You use Cash Flow Calculator in particular for cash flows in the analytical application that are to consider fees, interest rates, and installments for business transactions related to incoming payments and imported from the operational applications.

You can use the following optional elements when you define the generation type node:

- Cash Flow Category
- Flow Filter
  You use this filter to restrict only certain flows of a flow type in the cash flow generation process.

You define generation type nodes in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Generation* > *Edit Generation Types*.

Integration

The settings for the data sources to be used are made in the settings for the data source types [page 464].

4.3.8.1.3.2 Data Source Type

The data source type enables the *cash flow generation* method to identify the data sources that are to be used to generate a cash flow at generation type node level.

The generation type node [page 463] determines the cash flow view and the relevant data sources for each financial transaction node or each financial instrument node. The data source type is used to assign the settings for the relevant data sources to a generation type node.

You define data source types in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Generation* > *Edit Data Source Types*.

To define the data source types, select a data source category [page 442] and enter values for the attributes of a data source type. The attributes you have to maintain for a data source type depend on the data source category assigned. If, for example, you have a data source type with the data source category "position", you need to define the *position class* and *account category*.

You can assign one or more data source types to each generation type node. You assign data source types to a generation type node in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Generation* > *Edit Generation Types*.

4.3.8.1.3.3 Data Source Item Type

You can assign one or more data source item types to a data source type [page 464].

A data source item type includes settings that enable data source data to be incorporated into the cash flow generation process as a flow or as the calculation base, for example. Assigning the data source type directly to a generation type node enables data source item types to be assigned indirectly to a generation type node.

You define data source item types in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Generation* > *Edit Item Types for Data Sources*.

To define the data source item type, select a data source category [page 442] and a data source item category [page 449], and enter values for the attributes of a data source item type. The attributes you need to maintain for a data source item type depend on the data source category and data source item category assigned. If, for example, you have a data source type with the data source category *condition* and the data source item category *condition*, you need to define the *condition type* and flow type.

You assign data source item types to a data source type [page 464] in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Generation* > *Edit Item Types for Data Sources*.

4.3.8.1.4 Standard Attributes for Generating Cash Flows

Definition

The financial transactions and financial instruments for which cash flows are generated are mapped as templates [page 112] in Source Data Layer.

You can create actual financial transactions or financial instruments by editing the individual standard attributes of a financial transaction or a financial instrument. The attributes are managed and grouped in individual service modules, which enables the attributes to be distinguished from a business perspective.

Use

Certain attributes are particularly important for the cash flow generation process and the associated generation category [page 467]. These attributes and their influence on the cash flow generation process are briefly described in the following tables.

Attributes in the 'Basic Data' Service Module

Attribute	Meaning
Expiry date of object	End of the calculation period for maturity views.
Start of interest rate condition period	Start of the calculation time for all views except for transactions valid until further notice.
End of interest rate condition period	End of the calculation time for all views except maturity views. In maturity views, the end of interest rate condition period marks the transition from the conditions stored in the transaction to simulated or derived conditions.
Object currency	Currency in which the cash flow is calculated. Object currency and condition currency (all relevant conditions) must be identical otherwise the program is terminated with an error message.
Interest calculation method BD	Used to calculate the effective interest rate.
Interest calendar BD	Used to calculate the effective interest rate.

Attribute	Meaning
Cash flow generation template variant	The template variant can be used to define different generation type settings in Customizing for a template in the same hierarchy relationship.
Call date, incl. time unit	Only relevant for call options (for example, for loans) for minimum and maximum view.
Calendar for working day shifts	Relevant for all working day shifts that do not result from conditions; for all views.
Alternative calendar	Relevant for all working day shifts that do not result from conditions; for all views. If an alternative calendar is set, both calendars have to be used to determine the next working day.

Attributes in the 'Cash Flow Data' Service Module

You can also import cash flows for a financial transaction or financial instrument to Bank Analyzer. The cash flow data service module is then used to manage the cash flows. This enables cash flows that represent different cash flow views to be classified using the relevant cash flow categories, and then transferred for further processing.

Attributes in the 'Financial Conditions' Service Module

Attribute	Meaning
Conditions: 'Valid from' date	Relevant when you roll over financial transactions or financial instruments for the *Multiple Fixed Interest Periods* view.

Attributes in the 'Instrument Data' Service Module

Attribute	Meaning
Nominal amount, incl. currency	The nominal amount for financial instruments is taken by default from this field in the instrument data. Alternatively, key figures can be used to define a nominal amount imported from an external source (for example, a position).

Attributes in the 'Option Data' Service Module

Attribute	Meaning
Option valid from date	Relevant for determining the relevant option on the key date for cash flow generation.

Attribute	Meaning
Option valid to date	Relevant for determining the relevant option on the key date for cash flow generation.
Current option class	The current option class *LSR* must be set for unscheduled repayments. *LFR* must be set for for notice rights.
Option style	The *European* or *American* option style determines when (within the exercise period) the exercise can be assumed in the optional views.
Multiple exercise possible indicator	Determines whether an exercise is assumed for all or just the first exercise period for the option.
Option exercise: Valid from and valid to	Interval within which an option can be exercised once.

Attributes in the 'Key Figures' Service Module

Key figures for financial transactions/instruments are stored for specific time periods and categories.

- As a rule, the key date of the cash flow generation is used as the criterion for selecting the key figures for the transaction or instrument.
  Exception:
  If the start of the interest rate condition period is before the key date of the cash flow generation, the start date is used as the criterion for selecting the key figures.
- Key figure category `NMVAL` is expected for nominal amounts.
- Key figure category `PLVAL` is expected for individual simulated disbursement.

Attributes in the 'Credit Limit Data' Service Module

Credit limit data in financial transactions is stored on the basis of the category and a time value.

- As a rule, the key date of the cash flow generation is used as the criterion for selecting the credit limit data for the transaction.
  Exception:
  If the start of the interest rate condition period is before the key date of the cash flow generation, the start date is used as the criterion for selecting the limits.
- In Customizing, you can specify which limit categories are to be used.

4.3.8.2 Results of Cash Flow Generation

4.3.8.2.1 Generation Category

The generation category describes the business semantics of the cash flow generation process, and controls how the data incorporated into the cash flow generation process is processed. In the *cash flow generation* method, a

generation category represents one specific view of the cash flow. The different views of a cash flow are required because financial transactions and financial instruments can be analyzed from various perspectives. Each view that is used provides a cash flow as a result. The cash flow views are divided into the following views:

- Basic Views [page 468]
- Optional Views [page 470]
- Effective Views [page 471]
- Technical Generation Categories [page 473]

> **Note**
>
> The generation categories are shipped by SAP and cannot be enhanced.

4.3.8.2.1.1 Basic Views

Definition

In the *cash flow generation* method, a basic view is a cash flow view that is not necessarily based on another cash flow view. A basic view generates a cash flow without using another cash flow with another cash flow view for this first. This distinguishes the basic view from the effective view [page 471] and the optional view [page 470].

Use

Basic views can be subdivided as follows:

- **Plan View**
  The plan view generates a cash flow from the start to the end of the interest rate condition period.
  No assumptions or simulations are made when the plan view is used. No existing simulated disbursement procedures are considered (for more information, see the “Simulated Disbursement” section in Data Source Item Category [page 449]).
  This means that the cash flow is generated solely on the basis of the financial transaction (financial instrument) master data that actually exists, together with the relevant business transaction data. Forward rates can be used for variable interest rates that have yet to be fixed.
- **Interest Commitment View**
  The interest commitment view is calculated for a key date. In this process, a cash flow is generated from the start to the end of the interest rate condition period. Any existing final repayments are made at the end of the fixed-interest period.
- **Contractual Cash Flow for Current Period**
  This view is essentially the same as the *Interest Commitment View* except that imported cash flows are used.
- **Fixed-Rate Period View**
  The fixed-rate period view generates a cash flow from the start to the end of the last available interest rate condition period.
- **Multiple Fixed Interest Periods**

If financial transactions or financial instruments are rolled over, and the end of the fixed-rate period is postponed, new conditions can be defined for one or more fixed-rate periods. You can use this view to generate a cash flow for every fixed interest period defined.
Cash flows are generated for a fixed-rate period starting from the start date of the fixed-rate period or from the date from which the condition is valid, and ending one day before the start date of the next condition. The valid-from date defines the start of the next fixed-rate period. Cash flows are generated in the last fixed-rate period up to the end of the overall fixed-rate period. The valid-from date of a condition must be before the end of the fixed-rate period.
Cash flows are generated for each fixed-rate period in turn for the fixed-rate periods that precede the fixed-rate period defined by the generation date, and for the generation date itself.
If the flows in a fixed-rate period mature in the next period, the cash flow generation function includes these flows in the calculation in the next period, but does not display them in that period.

- **End with Current Fixed Interest Rate**
  This maturity view generates a cash flow until the end of the current interest rate condition period (in relation to the generation key date). At the end of the interest rate condition period, a simulated total repayment can be made, for example. This view is relevant for accruals and deferrals or amortization.
  The difference between this and the fixed-rate period view becomes apparent in a loan that has been rolled over with a value date in the future: The original discount must only be amortized until the end of the previous interest rate condition period. For fair value calculations, however, the system has to use knowledge that is already available about the next interest rate condition period.
- **Contractual Cash Flows Up To End of Fixed-Rate Period**
  This view is essentially the same as the *End with Current Fixed Interest Rate* maturity view, except that imported cash flows are used.
- **End with simulated final payment**
  This view is the same as the *End with Current Fixed Interest Rate* maturity view. However, you can restrict the cash flows to the near future (for example, the current interest period) instead of generating them until the end of the interest rate condition period.
- **Maturity View** for "**Forward Rate + Margin**"
  The maturity view is calculated for a key date.
  Here, the interest rate conditions are simulated beyond the end of the interest rate condition period up to the end of the contract for the financial transaction (financial instrument). This means that the cash flow is generated from the start of the interest rate condition period until the contract end date.
  After the end of the interest rate condition period, the nominal interest rate is assumed as equal to the *Forward Rate + Margin* based on the last interest rate condition in the interest rate condition period. All other conditions, in particular the repayment conditions, still apply on the basis of the last condition in the interest rate condition period.
  The *forward rate*, *spread*, and *interest rate adjustment* parameters can be derived from the object data of the financial transaction (financial instrument). In other words, you can define a derivation to determine the three parameters from the object data.
- **Maturity View** for "**Previous Nominal Interest Rate**"
  The maturity view is calculated for a key date.
  Here, the interest rate conditions are simulated beyond the end of the interest rate condition period up to the end of the contract for the financial transaction (financial instrument). This means that the cash flow is generated from the start of the interest rate condition period up to the end of the contract for the financial transaction (financial instrument).
  From the end of the interest rate condition period, the nominal interest rate based on the last interest rate condition within the interest rate condition period is used. In other words, the last valid interest rate condition is applied to the remaining term. All other conditions, in particular the repayment conditions, still apply on the basis of the last condition in the interest rate condition period.

- **Maturity View "Constant Effective Interest Rate"**
  The maturity view is calculated for a key date.
  Here, the interest rate conditions are simulated beyond the end of the interest rate condition period up to the end of the contract. This means that the cash flow is generated from the start of the interest rate condition period until the end of the contract.
  From the end of the interest rate condition period, a fixed nominal interest rate is assumed based on the last interest rate condition in the interest rate condition period. The effective interest rate for the period from the start to the end of the interest rate condition period therefore also applies to the new cash flow (from the start of the interest rate condition period until the end of the contract). All other conditions, in particular the repayment conditions, still apply on the basis of the last condition in the interest rate condition period.
  As a result, a cash flow is first generated for the interest rate condition period. If the conditions are variable, this is done using forward rates. Then the effective interest rate is calculated using the AIBD/ISMA methods. The fixed nominal interest rate for the time after the interest rate condition period is then selected in such a way that the effective interest rate remains the same.

  > **Note**
  >
  > This view is not available if Cash Flow Calculator is used to generate cash flows.

- **Maturity View "Forward Rate + Margin with Rollover Rate"**
  This view is essentially the same as the maturity view for *forward rate + margin* without a rollover portion. The only difference is that the data from the end of the interest rate condition period to the end of the financial transaction (financial instrument) contract is calculated on the basis of rollover capital adjusted by the rollover portion. This rollover portion can be derived from the financial transaction (financial instrument) master data.

  > **Example**
  >
  > If the rollover portion is 60 %, for example, and the rollover capital at the end of the interest rate condition period is EUR 100,000, only EUR 60,000 are rolled over in the maturity view. Technically speaking, this is the same as a one-off repayment at the end of the interest rate condition period.

- **Maturity View "Current Nominal Interest Rate with Rollover Rate"**
  This view is the same as the maturity view *Previous Nominal Interest Rate*, but has a rollover rate as well.
- **Maturity View "Constant Effective Interest Rate with Rollover Rate"**
  This view is the same as the maturity view *Constant Effective Interest Rate*, but has a rollover rate as well.

  > **Note**
  >
  > This view is not available if Cash Flow Calculator is used to generate cash flows.

4.3.8.2.1.2 Optional Views

Definition

In the *cash flow generation* method, this is a cash flow view that is based on one or more basic views. This means that when you use an optional view, you can generate a cash flow only if a different cash flow has already been generated for this cash flow using a basic view.

Use

You use the optional views when you generate cash flows in the loans area, taking notice rights and options for making unscheduled repayments into consideration. The optional view is divided into the following cash flow views:

- Minimum view
  For loans with notice rights, the minimum view assumes that immediate notice is given for the earliest possible date.
  For loans with unscheduled repayment rights, the minimum view assumes that all unscheduled repayment options are exercised in full.
- Maximum view
  For loans with notice rights, the maximum view assumes that no notice is given. In other words, the notice rights are not exercised.
  The maximum view assumes that no unscheduled payment options are exercised.

For more information, see Recurring Call Rights [page 484].

4.3.8.2.1.3 Effective Views

Definition

A cash flow view in the *cash flow generation* method with no relation to the actual generation of cash flows from the master data of a financial transaction or financial instrument, conditions, or other data sources. As is the case with the optional views, the effective view does, however, require a cash flow as an input. The effective view uses cash flows that were generated on the basis of a basic or optional view.

Use

The effective view is based on calculating the effective interest rate in accordance with AIBD/ISMA methods and can be used to accrue/defer a discount/premium or additional transaction costs.

The effective view requires a cash flow as an input parameter and essentially provides the effective interest rate calculated using AIBD/ISMA methods and the corresponding interest rate/repayment history.

The flow types used in the calculation can be identified in Customizing in order to control the effective view. You can also connect the cash flow filter before determining the effective interest rate and interest rate/repayment history.

> **Note**
>
> This view is not available if Cash Flow Calculator is used to generate cash flows.

Example

Consider the following cash flow as a starting point:

- Interest rate 5 % p.a.
- Discount 3 % of the nominal value

Date	Amount	Currency	Plus/Minus Sign	Flow Type
01/01/2001	100,000	EUR	-	7000 Disbursement (asset side)
01/01/2001	3,000	EUR	+	70001 Discount (asset side)
12/31/2001	50,000	EUR	+	7004 Installment re-payment (asset side)
12/31/2001	5,000	EUR	+	7010 Interest (asset side)
12/31/2002	50,000	EUR	+	7004 Installment re-payment (asset side)
12/31/2002	2,500	EUR	+	7010 Interest (asset side)

If all the flow types are "relevant" for calculating the effective view, the effective interest rate is 7.19291 % and the interest/repayment history is as follows:

Date	Amount	Currency	Plus/Minus Sign	Flow Type
01/01/2001	97,000	EUR	-	Disbursement
12/31/2001	48,022.88	EUR	+	Repayment
12/31/2001	6,977.12	EUR	+	Interest rates
12/31/2002	48,977.12	EUR	+	Repayment
12/31/2002	3,522.88	EUR	+	Interest rates

4.3.8.2.1.4 Technical Generation Categories

Definition

The technical generation categories include the **initial** and the **optional** view of the cash flow.

Use

- Initial View (Generation Category 00)
  You use the initial view if you want to use data from assigned data source types without this data being recalculated. The data that is read was already generated within a data source type that accesses either imported cash flows or cash flows generated by a generation type.
- Optional View (Generation Category 12)
  You use this generation category to compare conditions and options.

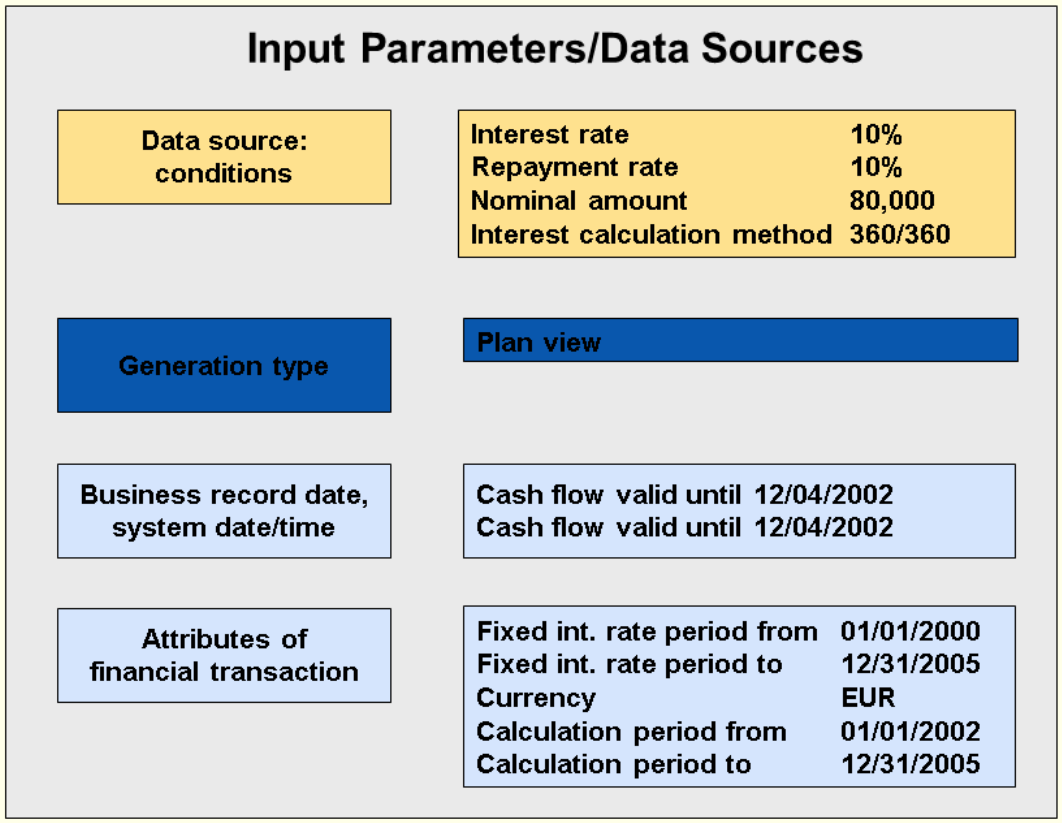
  Example

  You can set the start date of the unscheduled repayment condition to that on which the option is exercised.

4.3.8.2.2 Example 1: Generating Cash Flows for Loans

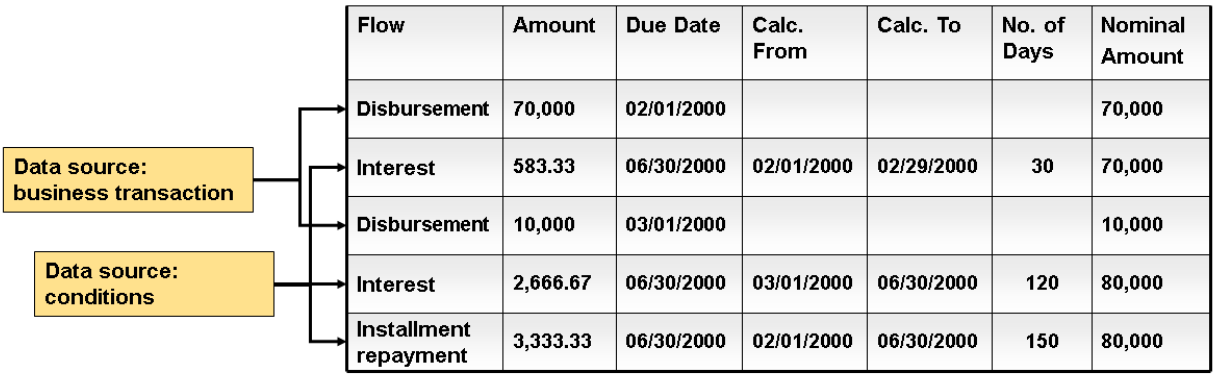
The *Cash Flow Generation* method generates cash flows for financial transactions and financial instruments on the basis of specific data sources and input parameters.

The following graphic is an extract from the result of a cash flow generated in the Source Data Layer and illustrates how the different input parameters and data sources are incorporated into the generation process. The cash flow was generated for the *installment loan* template [page 112] (see also Primary Object: Financial Transaction [page 122]) using a plan view (see also Basic Views [page 468]) and over an analysis time frame of five years. The interest payments and repayments are made every six months with an interest rate of 10% of the remaining nominal amount. The interest calculation method is based on 360/360 days.



Plan View of the Cash Flow for an Installment Loan

The following table gives an overview of how the input parameters and data sources affect a cash flow. For more information about the required Customizing settings, see the detailed description in Example 2 [page 475].



Flow	Amount	Due Date	Calc. From	Calc. To	No. of Days	Nominal Amount
Disbursement	70,000	02/01/2000				70,000
Interest	583.33	06/30/2000	02/01/2000	02/29/2000	30	70,000
Disbursement	10,000	03/01/2000				10,000
Interest	2,666.67	06/30/2000	03/01/2000	06/30/2000	120	80,000
Installment repayment	3,333.33	06/30/2000	02/01/2000	06/30/2000	150	80,000

4.3.8.2.3 Example 2: Generating Cash Flows for Loans

Example 1: Generating Cash Flows for Loans [page 473] outlined the result of generating a cash flow for an individual financial transaction. It provided an initial overview of the input parameters in the cash flow generation process, in which the central parameter is the generation type. This example describes in detail which input parameter settings you have to make in order to obtain the cash flow from the first example in the result.

You have to make input parameter settings for the:

- Cash flow generation type
- Individual standard attributes for the financial transaction

Example 1: Generating Cash Flows for Loans: Initial Situation

A plan view was used to generate the cash flow for the Installment Loan template [page 112].

The cash flow was generated over an analysis period of five years from January 1, 2000 to December 31, 2005 in the currency EUR. The analysis period is also the interest rate condition period. The interest payments and repayments are made every six months with an interest rate of 10% of the remaining nominal amount. The interest calculation method is based on 360/360 days.

The cash flow consists of the following flows:

- Loan disbursement

- Installment repayments
- Interest

The interest rate conditions and repayment conditions are stored in the condition data for the financial transaction.

Plan View of the Cash Flow for an Installment Loan

Flow Type	Amount	Due Date	Calculation From	Calculation To	Number of Days	Nominal Amount
Disbursement	70,000	02/01/00				70,000
Interest	583.33	06/30/00	02/01/00	02/29/00	30	70,000
Disbursement	10,000	03/01/00				10,000
Interest	2,666.67	06/30/00	03/01/00	06/30/00	120	80,000
Installment repayment	3,333.33	06/30/00	02/01/00	06/30/00	150	80,000

4.3.8.2.3.1 Editing the Cash Flow Generation Type

To enable the cash flow to be generated for the installment loan, you first have to define the cash flow generation type, which is the central cash flow generation parameter.

For more information, see Cash Flow Generation Type [page 461].

The generation type consists of generation type nodes [page 463], data source types [page 464], and data source item types [page 464].

You define the generation type and the associated elements in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Generation*.

To generate a cash flow for the installment loan using a plan view, you have to carry out the following Customizing activities for defining the generation type.

- Activity for Defining Flow Types [page 477]
- Activity for Defining Data Source Item Types [page 477]
- Activity for Defining Data Source Types [page 478]
- Activity for Defining Generation Type Nodes [page 478]

In this example, only one generation type is defined with one generation type node. This is to make the process of generating cash flows easier to comprehend.

4.3.8.2.3.1.1 Activity for Defining Flow Types

A cash flow is a collection of individual flows. For instance, the cash flow of the installment loan comprises several different types of flow, such as disbursements, repayments, and interest.

These flows have to be classified using flow types.

To generate the cash flow for the installment loan, three flow types, therefore, have to be defined:

- Flow type: RATT (installment repayment)
- Flow type: ZINZ (interest payments)
- Flow type: AUS (disbursements)

You have to specify flow types when you define a data source item type of the item category *Flow* or *Condition*.

4.3.8.2.3.1.2 Activity for Defining Data Source Item Types

The following data source item categories are required to generate cash flows for the installment loan:

· Data Source Item Category: Condition [page 451]

You can use the data source items in the *Condition* category to incorporate repayments and interest payments into the cash flow generation process as condition-based flows. The flows are taken from a data source in the *Condition* category.

· Data Source Item Category: Flow [page 452]

You can use the data source items in the *Flow* category to incorporate loan disbursements into the cash flow generation process as flows on the basis of business transactions. The flows are taken from a data source in the *Business Transaction* category.

This means that to use the data source items described to generate cash flows for the installment loan, you have to define three data source item types [page 464] with the following descriptions and properties.

- Data source item type: *Condition: Installment Repayment*
  - Data source category: *Condition*
  - Data source item category: *Condition*
  - Condition type: *103120 Installment Repayment*
  - Flow type: RATT
- Data source item type: *Conditions: Interest*
  - Data source category: *Condition*

- Data source item category: *Condition*
- Condition type: *101100 Nominal Interest*
- Flow type: ZINZ

- Data source item type: *Loan Disbursement: Business Transaction*
  - Data source category: *Business Transaction*
  - Data source item category: *Flow*
  - Flow type: AUS

  Once the data source item types have been defined, they can be assigned to a data source type.

4.3.8.2.3.1.3 Activity for Defining Data Source Types

On the basis of the data source items required to define the data source item types for generating cash flows for the installment loan, two data source categories were identified: *Condition* and *Business Transaction*.

Two data source types [page 464] must, therefore, be defined for the example with the following properties and descriptions. The descriptions chosen for the business transaction class and item class are examples:

- Data source type: *Conditions for Loans*
  - Data source category: *Condition*
  - Condition group type: *Loans*
  - Transaction type: *Standard*
- Data source type: *Loan Disbursement via Business Transaction*
  - Data source category: *Business Transaction*
  - Business transaction class: *Business Transactions for Loans*
  - Item class: *Loan Disbursement*
  - Account category: *Settlement Account*

Once the data source types have been defined, they can be assigned the data source item types [page 464]. Note that data source types can only be assigned data source item types if the relevant data source categories and data source item categories have also been linked in the system.

For more information, see Data Source Item Category [page 449].

The following assignments of data source item types to data source types are made for the example:

- Data source type: *Conditions for Loans*
  - Data source item type: *Conditions: Installment Repayment*
  - Data source item type: *Conditions: Interest*
  - Data source type: *Loan Disbursement via Business Transaction*
  - Data source item type: *Loan Disbursement via Business Transaction*

Once you have made the assignment, you can define the generation type node for a generation type and specify the relevant data source types for the generation type node.

4.3.8.2.3.1.4 Activity for Defining Generation Type Nodes

Before you can create a generation type node for the installment loan, you have to define a description for the generation type. This description should indicate which cash flow view (see Generation Category [page 467]) the

generation type uses to generate cash flows. This information makes it easier to select the correct generation type when you call up the "cash flow generation" method. In this example, the description of the generation type is:

SX 3: Plan View Financial Transactions and Financial Instruments

You could define several generation type nodes with the same generation category for this generation type.

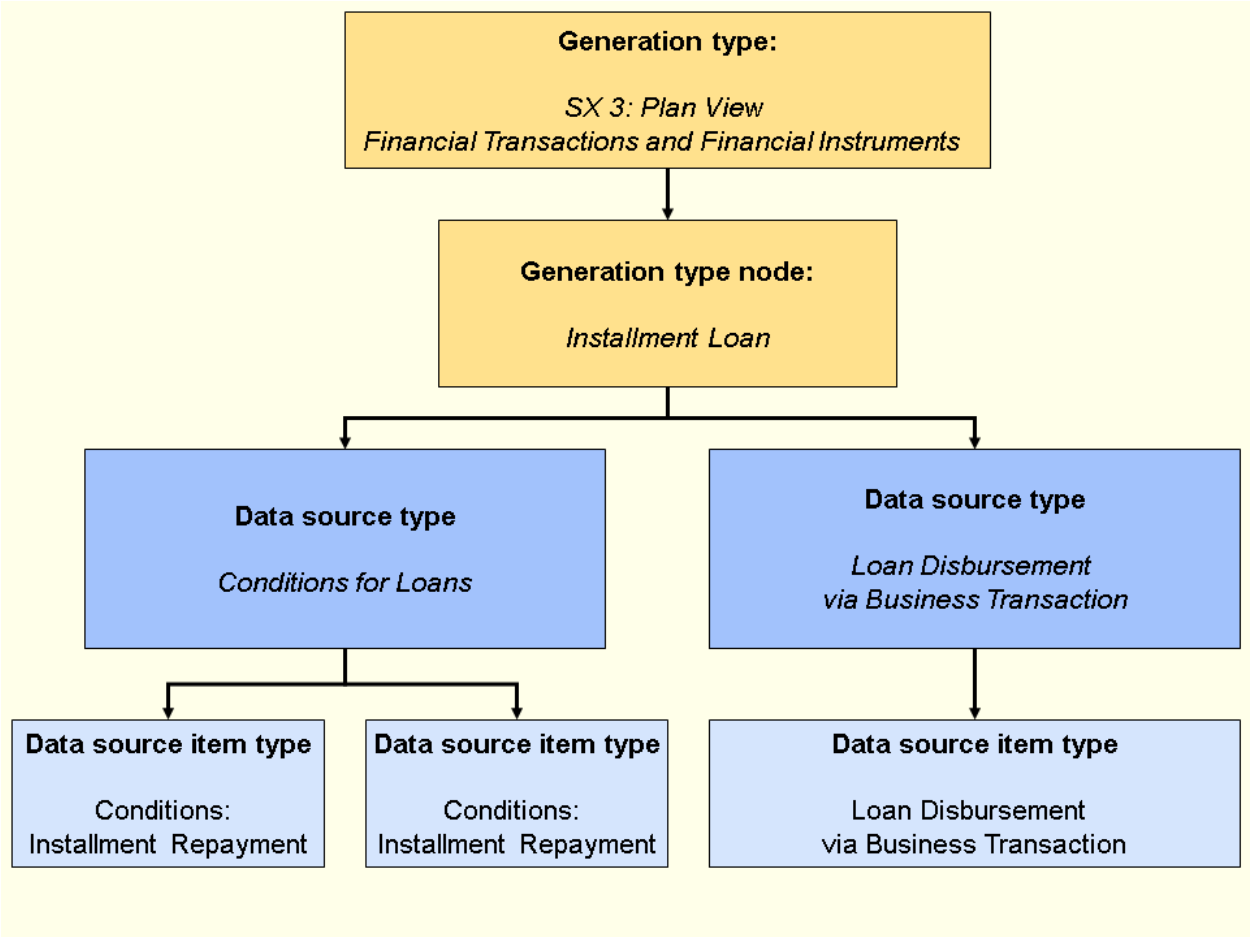
To generate cash flows for the installment loan, you have to define the following properties for the generation type node for generation type *SX 3: Plan View Financial Transactions and Financial Instruments* (see also Generation Type Node [page 463]). The template name is one of the identifying features of the generation type node:

Generation type node: *Installment Loan*

- Template: 900_100 installment loan
- Parent template
  The parent template is not required in this case because the template is unstructured.
- Node number: 1
- Template variant
  No variants exist for this template.
- Generation category: *Plan View*
- Calculation rule: financial mathematics

Once the generation type node has been defined, the data source types can be assigned. In this example, the generation type *SX 3: Plan View Financial Transactions and Financial Instruments* has the following structure.

Structure of the Cash Flow Generation Type



4.3.8.2.3.2 Editing Individual Standard Attributes

Certain standard attributes of a template [page 112] are particularly important for generating cash flows and, therefore, have to be taken into account when you create financial transactions and financial instruments.

For more information, see Standard Attributes for Generating Cash Flows [page 465].

When you create a financial transaction for template *900_100 Installment Loans*, there are specific attributes that you have to edit because their values have a particular effect on the generation of cash flows using the generation category *Plan View*. Compare the data provided in Example 2: Generating Cash Flows for Loans [page 475].

- Start of interest rate condition period: January 1, 2000
- End of interest rate condition period: December 31, 2005
- Currency: EUR

4.3.8.2.4 Generating Cash Flows Manually in the Source Data Layer

Prerequisites

- You have made the appropriate settings in Customizing for the cash flow generation type.
- You have entered or created a financial transaction (financial instrument).

Context

You can also call up the cash flow generation process from the *SAP Easy Access* screen using the manual processing transactions for financial transactions and financial instruments.

Financial transaction primary object:

Bank Analyzer > *Source Data Layer* > *Primary Objects* > *Financial Transaction* > *Display Financial Transaction*

Bank Analyzer > *Source Data Layer* > *Primary Objects* > *Financial Transaction* > *Change Financial Transaction*

Financial instrument primary object:

Bank Analyzer > *Source Data Layer* > *Primary Objects* > *Financial Instrument* > *Display Financial Instrument*

Bank Analyzer > *Source Data Layer* > *Primary Objects* > *Financial Instrument* > *Change Financial Instrument*

Procedure

1. In the display or change view for the financial transaction or financial instrument, choose *Cash Flow Generation*.
2. In the *Enter Generation Parameters* dialog box, enter the following data:
   - Generation type
   - Generation key date
   - Posting date
   - System date
   - System time
   - Account
     When you generate cash flows for financial instruments, you can specify an account number that enables the system to access the relevant flow data (positions, business transactions). You do not need to enter an account category because the system automatically uses the account category that has been defined in the relevant data source type of category 05 or 06.
   - Required from

Date from which the cash flow is required (value date)

- Market data area
  Enter the market data area only if market data is required for the cash flow view.

3. Choose *Continue*.

   The system displays the cash flow.

4.3.8.2.5 Generating Cash Flows Automatically in the Source Data Layer

When you generate cash flows for financial instruments from a secondary data source, you can use the attribute EXT_CONT_NUMBER in the basic data of an object to specify an account number and thus access the transaction/flow data. You can call function module /BA1/FP_API_PO_41_BAS_SET in the Module Editor for this purpose.

4.3.8.2.6 Display Cash Flows stored in Results Data Layer

Use

You can call up the cash flow generation process for financial transactions and financial instruments for the manual processing transactions in the Results Data Layer (RDL).

Financial Transaction Primary Object:

- *SAP Area Menu* > *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Transaction* > *Display Financial Transaction*.
- *SAP Area Menu* > *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Transaction* > *Change Financial Transaction*.

Financial Instrument Primary Object:

- *SAP Area Menu* > *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Instrument* > *Display Financial Instrument*.
- *SAP Area Menu* > *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Instrument* > *Change Financial Instrument*.

Prerequisites

- You have customized the cash flow category to enable it for RDL storage in Customizing *Financial Services* > *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Generation* > *Edit Cash Flow Categories*.

- You have customized the cash flow type in Customizing *Financial Services* > *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Generation* > *Edit Cash Flow Type*. Cash Flow Type defines the results data area and result type in RDL from where cash flow data is retrieved for further refinement and generation. You can also assign a generation type to the cash flow type.
- You have entered or created a financial transaction (financial instrument).

Procedure

1. To edit financial transactions, go to the *SAP Easy Access* screen, and choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Transaction* > *Create Financial Transaction*.
2. To edit financial instruments, go to the *SAP Easy Access* screen, and choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Instruments* > *Create Financial Instruments*.
3. Choose *Cash Flow Generation*.
4. In the `Enter Generation Parameters` dialog box, enter the following data:
   - Generation type
   - Generation key date
   - System date
   - System time
   - Market data area
     Enter the market data area only if forward rates are required for the cash flow view.
     When you go to generate cash flows for financial instruments, you can specify an account number as a generation parameter. This will enable the system to access the relevant transaction data (positions, business transactions). You do not need to enter an account category - this is because the system automatically uses the account category specified in the relevant data source type of category 05 or 06.
5. Choose *Continue*.

Result

The system displays the cash flow which is stored in RDL under result category HKCFD.

4.3.8.3 Additional Functions

4.3.8.3.1 Recurring Call Rights

Definition

Repeated call rights that are defined by means of a start date and a call cycle. The call cycle specifies the interval between the individual call dates.

Structure

You map recurring call rights in a three-level template hierarchy that is made up of a root node, option nodes, and condition nodes. You define recurring call rights (as with call rights) with fixed dates in the option nodes and condition nodes.

The following example uses the template hierarchy for cash flow transactions to illustrate the mapping of recurring call rights. You use this template hierarchy for loans and collateral agreements, for example.

Mapping in the Source Data Layer (SDL)

The nodes of the template hierarchy have the following properties:

- Transaction nodes
  You store non-optional in the node for the cash flow transaction. When the system is calculating the call dates, it uses the calendar and the call date from the basic data of the transaction node.
- Option nodes
  You do not store any fixed exercise dates in the option data. Instead the call cycle that defines the exercise dates is stored here. To do this, you can extend the template version by activating the fields for the exercise frequency. *Note:* The dates are used in the optional views of the cash flow generation in order to take into account unscheduled repayment options or call options. If you have a monthly call cycle and a period end that falls at the end of the month, you can determine the extent to which the system updates the exercise dates by means of the *Key Date for Exercise* attribute.
  You can control the updating of the exercise dates by means of the option class. The option classes LFR (call date) and LFA (exercise date) are used here.
  The price calculator differentiates calls by borrowers and lenders by means of the purchase/sale indicator in the basic data of the option node.
- Condition nodes
  The condition node is assigned (lower-level) to the option node. You have to assign a call condition of the *final repayment* category to the condition node. The system reads information about the inclusive/exclusive indicator and about working day shifts (among other things) from the defined call conditions.

  > ⚠ Caution
  >
  > The valid from date of the call conditions and the valid from date of the exercise dates that are defined in the corresponding option node must be the same.

You can define a number of call rights for a cash flow transaction if, for example, you want to map separate call rights for borrowers and lenders. To differentiate between borrowers and lenders, you must use different templates or template variants on the level of both the option nodes and the condition nodes. In this way, you can map calls by borrowers and lenders with different transaction types. This enables you to control cash flow generation more precisely.

Customizing Settings

To ensure that the system considers recurring call rights during cash flow generation, you have to make the following settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Generation* > *Edit Generation Types*:

- Generation type for recurring call rights
  You have assigned the following attributes to the generation type node for the uppermost node in the template hierarchy:
  - Generation category 25 (minimum view) or generation category 26 (maximum view)
    This is necessary so that the system can read information about the exercise dates in the lower-level option node.
  - Data source types of category 04 that read conditions and items from the lower-level option nodes (in both cases for the call rights of borrowers and lenders).
- Generation type of data source types for reading conditions and items
  You have assigned the following attributes to the generation node for call rights of the borrower and lender:
  - Generation category 12 (optional view)
  - Data source types (category 04) that read conditions and items from the lower-level condition nodes (in both cases for the call rights of borrowers and lenders).
- Generation type for reading from the condition nodes
  You have created generation type nodes for the call conditions and have defined the following attributes there:
  - Generation category 00 (initial view)
  - Data source types (category 01) for reading the call conditions
    Note that you should use different data source types for the call conditions of borrowers and lenders. In particular, different item types should be assigned to the data source types in the transaction type.

Integration

The system accesses the call dates by means of the views in cash flow generation. It considers the call rights in the minimum and in the maximum view only. These two optional views [page 470] differ in the way they expect call rights to be exercised.

- Minimum view
  For the minimum view, the system presumes that the call rights will be exercised at the earliest possible date. It calls the determination of the next call date [page 486] function and generates a final repayment for the date returned.
- Maximum view
  For the maximum view, the system does not consider any call rights itself. Instead it returns all of the call dates that are possible for the future. This list contains both the call dates for the borrower and for the lender.

When it calculates call dates, the cash flow generation process considers the conditions of the financial transaction, in particular the inclusive/exclusive indicator and a possible working day shift.

4.3.8.3.1.1 Determination of the Next Call Date

Use

The system uses this function to determine the next possible call date for a financial transaction or financial instrument that has call rights. The starting point is a key date that the function receives as an input parameter.

Integration

The function is implemented as a function module (/BA1/FP_API_CFG_OPT_COND_GET) that is used in the following areas:

- Cash Flow Generation
  Consideration of call rights in the minimum and in the maximum view.
- Generic mapper for risk basis
  Processing of call rights in the mapping of the SDL templates to valuation structures [page 496]
- SAP Basel II
  Calculation of the residual maturity or the original maturity for financial transactions and financial instruments that have call rights.

Prerequisites

You have made the appropriate settings in Customizing for cash flow generation so that optional rights such as call rights are considered. For more information, see Recurring Call Rights [page 484].

Features

After you call the function, the system calculates the next possible call date for both the borrower and the lender, and returns the earlier of the two dates that are calculated. The system does the following:

1. It calculates a series of exercise dates for the period from the valid from date (of the exercise dates) up to and including the valid to date. The exercise date corresponds to the end of the exercise period (European style).
2. The system reads the call date from the basic data of the option node. If the call date is not available there, the system reads it from the basic data of the root node of the template hierarchy.
3. To calculate the next call date, the system uses the option class from the option header:
   - Option Class LFA (Exercise Date)
     The system adds the call date to the key date. If a calendar is stored in the basic data of the cash flow transaction, the system shifts the calculated date to the next working day.
     The system selects the date that comes exactly after the date just calculated from the sequence of exercise dates that were calculated previously.
     The system adjusts the calculated exercise date according to the inclusive/exclusive indicator, and shifts the date depending on the call conditions to the next working day.

- Option Class LFR (Call Date)
  The system selects the date that comes exactly after the key date from the series of exercise dates that was calculated.
  The system adds the call date to this exercise date. If a calendar is stored in the basic data of the cash flow transaction, the system shifts the calculated date to the next working day.
  The system adjusts the calculated exercise date according to the inclusive/exclusive indicator, and shifts the date depending on the call conditions to the next working day.

Example

The system is to calculate the next call date for the following input parameters:

- Key date: 03/20/2004
- Call option of the borrower
  - Call date: 15 days
  - Valid from: 01/01/2004
  - Valid to: 01/01/2005
  - Payment frequency: Three months (month end)
  - Option class: LFA (exercise date)
- Call option of the lender
  - Call date: 15 days
  - Valid from: 01/01/2004
  - Valid to: 01/01/2005
  - Payment frequency: Three months (month end)
  - Option class: LFR (call date)

The system calculates the exercise dates and the call dates for the borrower and the lender.

- Borrower
  - Exercise dates: 03/31/2004, 06/30/2004, 09/30/2004, 12/31/2004l
  - Call dates: 6/30/2004, 9/30/2004, 12/31/2004
    Because of option class LFA, the system first added the call date to the key date (result: 04/04/2004). The system takes into account only exercise dates that are after this date.
- Lender
  - Exercise dates: 03/31/2004, 06/30/2004, 09/30/2004, 12/31/2004
  - Call dates: 04/15/2004, 07/15/2004, 10/15/2005, 01/15/2005
    Because of option class LFR, the system added the call date to the exercise dates that are after the key date.

The system returns the first call date of the lender (04/15/2004) as the next possible call date.

> **Note**
>
> Note that this example does not take any working day shifts into account and that the calculation of the exercise intervals always includes the 'to' date of these intervals.

4.3.8.3.2 Restart Points For Cash Flow Generation

Use

If you use the Cash Flow Calculator for generating cash flows, you need to take into account the information for all the business transactions since the transaction start date.

From the time of generation onwards, you can use restart points to archive the business transactions with an earlier date because the restart points contain the information for these business transactions. In addition to the restart points, the system only needs to select the new business transactions.

If you later want to generate the cash flow for an earlier point in time, for example when you make retroactive changes or corrections, you must have first generated the restart points using the indicator for the partial cash flow.

> **Note**
>
> Note that you cannot make any changes to the saved partial cash flow.
>
> Therefore, do not archive your business transactions until you no longer expect any retroactive changes or corrections that affect this part of the cash flow.
>
> For example, at the end of a fiscal year you could generate restart points for December 31. Afterwards, you could archive all business transactions up until then.

Features

Generate Worklist for Restart Points

You can use this function to create a worklist of transactions for which restart points for the cash flow generation need to be generated. The system uses the general primary data source `SBT_CLASSES` of the server *Business Transaction Classes* `BT_CLASS` to choose the worklist. From Customizing for cash flow generation, it reads all business transaction classes and item classes for data sources of the category Business Transactions `05` that are assigned to a generation node with the calculation rule `01`, `02` or `03` (in other words with Cash Flow Calculator). The system then reads all the business transactions of the corresponding business transaction classes and item classes, and determines the relevant transactions. The system saves the list of transactions in the versioned data store.

> **Note**
>
> To avoid inconsistencies, you cannot restrict the generation of restart points to specific cash flow generation types.
>
> Use SAP Note 1663316.

Generate Restart Points

This function generates the restart points for the cash flow generation for the transactions determined by the first process. The system uses the general primary data source `SPO_FOR_RI` of the server *Worklist Primary Objects for*

Restart Points `PO_FOR_RI` to select the worklist that was generated by the first process. For each transaction from the worklist, the system then generates the cash flow with restart points (on the key date of the first process). The system then saves the restart points in the versioned data store.

4.3.9 Storage of Yield Curves

In the SAP Easy Access menu under *Financial Services* > *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Yield Curves* , you can use the following transactions to save and delete yield curve data.

Save Yield Curves

The system creates yield curves based on the following input fields:

- Market Data Area
- Currency
- Yield Curve Type
- Validity Date
- Due Date
- Date of System Date/Time
- Time of System Date/Time
- Grid Point Density

The report calculates Discount Factors, Continuous Compounding Zero Rates and Parbond Yields, which are stored in a database table in the system.

> **Note**
>
> If the report is executed with the same set of input fields (i.e. Market Data Area, Currency, Yield Curve Type, Validity Date), but a different Grid Point Density, the previously calculated values will be implicitly deleted from the system.

Delete Yield Curves

This report deletes the persisted values. Values calculated for all due dates for which the Market Data Area, Yield Curve Type, Currency and Validity Date were given will be deleted from the system.

4.3.10 Cash Flow Refinement (FS-BA-PM-GM-CFR)

This component provides processes and methods that you can use to prepare data for the analyses:

- Valuation Rule [page 490]
- Valuation Structure [page 495]
- Mapping of SDL Templates to Valuation Structures [page 496]
- Methods for Cash Flow Refinement [page 499] and Method Schemas [page 534]

4.3.10.1 Valuation Rule

Definition

Customizing entity that you use to define the calculation and valuation methods at financial transaction or financial instrument level.

Use

Valuation rules contain information that is used by various applications in Bank Analyzer. These include the Fair Value Server, price calculator, and cash flow refinement methods, such as cash flow determination.

The valuation rule is part of the valuation Customizing, and is assigned to financial transactions and financial instruments at runtime. The system assigns a valuation rule to each financial transaction and financial instrument. This makes it possible to define how the methods are applied at financial transaction and financial instrument level.

You use valuation rules only in exceptional circumstances to control how methods are used at transaction level. It is usually better to use valuation rules to define how an entire product group is valued, or how transactions are to be valued across an organizational unit.

Example

In Cash Flow Hedge Analysis, the system derives valuation rules at two points, the second of which is at subcontract level. For more information about how the system derives valuation rules, see Derivation of Global and Local Valuation Rules [page 1116] in the documentation for Cash Flow Hedge Analysis.

Structure

The system stores the valuation rule by default in characteristic `/BA1/C69_VALRULE`. However, for the methods that you manage using the valuation rule, you can define that the system stores the valuation rule in a different characteristic. For more information, see Application Categories for Valuation Rules [page 492].

Integration

Customizing Settings

- You define valuation rules in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivations* > *Determination of the Valuation Rule* > *Create Valuation Rules*.
- You can enter additional settings in Customizing for the individual applications that allow their valuations to be governed by valuation rules. You control these applications by entering in each valuation rule specific settings for valuations.
- The valuation rules are assigned to the individual financial transactions and financial instruments by means of the derivation tool in Bank Analyzer. The derivation tool is also used to determine other Customizing entities, such as the market data set, that are relevant for the valuation process.
  The system contains a set of source characteristics that can be use by the derivation tool. These include the characteristics that are stored for the respective financial transaction in the Source Data Layer (SDL), for example. Using the source characteristics, you can define which rules the system is to use to define the values for the characteristic used.
  You enter the settings that define how valuation rules are derived in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivations* > *Determination of the Valuation Rule* > *Create Derivation Environments*, and the Customizing activities shown below this.

Operational Use

The system uses one of the following methods to derive valuation rules:

- **Derivation in a Secondary Data Source**
  You specify how the valuation rule is to be derived in the step sequence of the secondary data source (for example, in the selection module in Cash Flow Hedge Analysis).
- **Derivation Outside of a Secondary Data Source**
  You specify a derivation environment and a derivation module in the form of a derivation structure in Customizing for the respective application (for example in the valuation structure type in the Fair Value Server).
- **Automatic Definition without Customizing Settings**
  In the cash flow filter [page 513] in Cash Flow Hedge Analysis, although the system access the valuation rule, it does not use the Bank Analyzer derivation tool.

Applications of the Valuation Rule

You can use the valuation rule to manage the following applications:

Application	Component
Cash Flow Hedge Analysis [page 1107]	Hedge processes
Assignment to the Short Term Category.	Cash Flow Hedge Analysis
Gap Analysis	Strategy Analyzer
Fair Value Server [page 628]	General Calculation and Valuation Methods

Application	Component
Price calculator [page 547]	General Calculation and Valuation Methods
Cash Flow Determination [page 504]	Cash Flow Refinement
Use of Due Date Scenarios [page 508]	Cash Flow Refinement
Breaking Down of Implied Options	Cash Flow Refinement
Cash Flow Splitting [page 518]	Cash Flow Refinement
Cash Flow Filter [page 513]	Cash Flow Refinement
Rollover [page 510]	Cash Flow Refinement
Derivation of the Market Data Set	Cash Flow Refinement

Example

- **Import of NPVs for the Price Calculator**
  You can use valuation rules to tell the price calculator for which financial transactions and financial instruments NPVs were imported from feeder systems. The source field for the derivation of the valuation rule is usually a *Feeder System* characteristic, which you store in the financial transaction or financial instrument in the SDL.
- **Assignment of Financial Transactions to the Short Term Category**
  In Cash Flow Hedge Analysis, you can use valuation rules to define which financial transactions the system is to display as short-term transactions in the reporting functions. You do this in Customizing for Cash Flow Hedge Analysis by entering conditions in each valuation rule that have to apply in order for the financial transaction to be included in the short-term category.
  In this case, the valuation rule is not used to control valuations, since it governs only how the data is displayed.

4.3.10.1.1 Application Categories for Valuation Rules

Definition

Method that you define by means of the valuation rule, and for which you can store your own custom characteristic. The system then uses this custom characteristic to store the valuation rule.

Use

Application categories for valuation rules are an extension to the valuation rule [page 490] concept. You use the application categories of valuation rules in order to separate the methods governed by the valuation rules. You may need to do this if you use multiple methods at once, and want to apply them to data at different levels of detail.

The system stores by default the valuation rule for all methods in the same characteristic, /BA1/C69_VALRULE, which is predefined in the system. However, in certain circumstances this feature can lead to a very large number of valuation rules.

Example

If, for example, you want to use the standard valuation rule to define how five methods are used, and for two groups of financial transactions and financial instruments within these methods, then you would need 32 valuation rules, in other words, one valuation rule for each combination of method with two groups of financial transactions or financial instruments.

To avoid the need for a large number of valuation rules, you can enter your own custom characteristics for each method. The system then uses these characteristics to store the valuation rule for the method in question. In this way, you separate the methods from each other. The methods that can be controlled in this way come under the concept of *Application Categories for Valuation Rules*.

Example

If you register separate characteristics for the application categories of valuation rules, then you need separate valuation rules just for the combination of the method and its grouping of financial transactions or financial instruments. This means that if you have five methods, and each of them has two groupings of financial transaction or financial instrument, then you need only 10 valuation rules.

You register your custom characteristics for the application categories of valuation rules in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivations* > *Determination of the Valuation Rule* > *Register Characteristics for Valuation Rules*.

Structure

We provide the following application categories for valuation rules. They are stored in a system table, and cannot be customized.

Application Categories for Valuation Rules	Function Name of the Application Category for the Valuation Rule
Initial value	Standard valuation rule
101	Cash Flow Splitting [page 518]
102	Cash Flow Determination [page 504]

103	Due date scenarios [page 508]
104	Break down implied options [page 503]
105	Rollover [page 510]
107	Cash Flow Filter [page 513]
201	Price calculator [page 547]
202	Fair Value Server [page 628]
301	Gap analysis in Strategy Analyzer
302	Cash Flow Hedge Analysis [page 1107]
401	Determination of market data sets

> **Note**
>
> Although cash flow filters [page 513] are stored in the table for the application categories of valuation rules, the derivation of valuation rules in the cash flow filter is defined in the coding, so that when you register a characteristic for the cash flow filter, you cannot influence how the valuation rule is derived and stored.

Integration

- Standard Characteristic for the Valuation Rule
  You do not need to store separate characteristics for each application category of the valuation rule. If you do not register any characteristics, the system uses the default standard characteristic /BA1/C69_VALRULE, which is predefined in the system.
- Registered Characteristics
  You can register your own custom characteristics so that they can be used for the individual application categories of valuation rules. Only those characteristics are permitted that you have defined in reference to the data element /BA1/R0_DTE_VALRULE, which is predefined in the system. For more information about creating characteristics, see the document General Information about Characteristics [page 21].

  > **Note**
  >
  > If you have registered your own characteristic for an application category of a valuation rule, then the relevant method looks for the valuation rule in this characteristic. If it cannot find a valuation rule, it uses the valuation rule stored in the standard characteristic.

- Derivation Module for Deriving the Valuation Rule
  The system normally uses the derivation tool in Bank Analyzer to derive valuation rules. You enter settings for the derivation tool in Customizing for the *Cash Flow Refinement* component. To be able to use the application categories for the derivation rule in the derivation process, in the associated derivation environments you must enter as target fields the characteristics that you registered for the storage of derivation rules. Only then can the system fill the characteristics with data during the derivation process, which in turn enables it to use

them to define how it applies the methods. You define derivation environments in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivations* > *Determination of the Valuation Rule* > *Create Derivation Environments*.

- Features Specific to Cash Flow Hedge Analysis
  In Cash Flow Hedge Analysis, the system derives a global and a local valuation rule. You use the application categories of valuation rules for the derivation of local valuation rules only. For more information see the document Derivation of Global and Local Valuation Rules [page 1116] in the documentation for Cash Flow Hedge Analysis.

4.3.10.2 Valuation Structure

Definition

Structures used at runtime to store the data for financial transactions and financial instruments that is required for the analyses.

Use

Valuation structures are purely internal structures that are used as input for the cash flow refinement methods. Valuation structures are not stored in the database; they exist only during processing.

Structure

Valuation structures are mapped as a hierarchy, just as is the case for the templates in the Source Data Layer (SDL). Each node contains information about the financial transaction or financial instrument, and about any substructures it may contain. As is the case for SDL templates, the individual nodes of a valuation structure contain the following information:

- General Information
- Header and item information
- Cash flow header and items
- Option Data
- Characteristics and key figures

The system uses the transaction form to classify each node in the valuation structure. Transaction forms define how a transaction is measured.

> **Note**
>
> Valuation structures are implemented in the class interface /BA1/CL_R0_RISK_OBJECT. The data for an individual node is represented by type /BA1/R0_STR_RO_ECONTRACT and the information about the subnodes is represented by table type /BA1/R0_TAB_RISK_OBJECT.

Integration

The system uses the mapping of SDL templates to valuation structures [page 496] to generate valuation structures. In this process, depending on which SDL template category is used and its Risk Basis classification, the system selects a mapper and a transaction form. SAP provides some mappers for this process. If you use templates other than the standard templates in the SDL, then you can create your own mappers in Customizing for Cash Flow Refinement.

> **Note**
>
> Valuation structures do not contain any information about financial conditions. Therefore, cash flow generation has to be called before the mapping process in order to trigger the conditions in cash flows. These conditions can then be measured by the price calculator.

When the mapping process is complete, all the information that is relevant for the measurement process is stored in the valuation structures. This means that during the measurement process, the system only has to access the valuation structures. The Calculation and Valuation Methods are created in such a way that they use only market data, and the cash flows in the valuation structure.

Example

The price calculator [page 547] is one of the applications that uses valuation structures to measure transactions. The mapping in the valuation structure is used to provide the price calculator with the fields from the SDL templates. For information about which fields in the SDL templates are needed by the price calculator, see the section Valuation-Relevant Fields in the SDL Templates [page 581].

4.3.10.2.1 Mapping of SDL Templates to Valuation Structures

Use

To be able to measure financial transactions and financial instruments, the system maps the templates of the Source Data Layer (SDL) to a valuation structure that the system uses in the analyses. This structure is the valuation structure [page 495].

Mapping is required because the data in the SDL is stored in a generic structure, which customers can define to meet their own requirements, whereas the Calculation and Valuation Methods in Bank Analyzer need a structure that is not based on Customizing settings, and that has clear semantics that the system can interpret. For example, the price calculator can price financial transactions at theoretical prices only if the system can provide it with the required data in a stable structure.

In addition to the functions for generating valuation structures, there is a function that enables other components to recognize the structure of a valuation structure. This function is referred to as the *structure mapper*, and is implemented by means of an Application Programming Interface (API). For example, the Accounting Processes use the structure mapper to determine which parts of a financial transaction or financial instrument can be measured in order to show these separately if required.

The mapping is called internally only.

Integration

The conditions stored in the SDL template **cannot** be transferred to the valuation structure during mapping. Therefore, the system calls the cash flow generation method [page 440] before mapping. It also calls cash flow generation for cash flows that were imported from non-SAP systems. This is because the system needs the cash flow view for the evaluations, which it obtains by means of the cash flow generation type. It then writes this to the valuation structure as part of the cash flow generation method.

Price Calculator

The valuation structure contains a tree structure, the nodes of which contain the information that is needed to measure the financial transaction or financial instrument. During the mapping process, the system also stores the information needed to value subtransactions. This information is the transaction form, which the price calculator uses in its valuations.

The fields in the SDL templates that are needed for the measurement are retained in the mapping process, and the price calculator [page 547] can then access them in the valuation structure. The mapping for the sample products delivered by SAP is set up in such a way that the SDL and price calculator can interact without problems. If you create your own SDL templates, then you may need to create a new mapper, or assign the SDL templates to the mappers, and extend the transaction forms in Customizing so that the price calculator can measure correctly the financial transactions and financial instruments mapped using the new templates.

> **Note**
>
> For information about which fields in the SDL templates for financial transactions and financial instruments are relevant for the price calculator, see the price calculator documentation under Valuation-Relevant Fields in the SDL Templates [page 581].

Accounting Processes

The system uses the mapping in the valuation structure in the Accounting Processes [page 693] to generate financial positions and to measure financial transactions and financial instruments. For this reason, the mapping defines the granularity of the financial positions created by the Accounting Processes. Financial positions can always have a lower level of detail (granularity) than that defined by their mapping to the valuation structure, but they can never be at a greater level of detail than this mapping.

Prerequisites

Mapping requires the following settings in Customizing for the *Cash Flow Refinement* component:

- Assignment of Template/Risk Basis Classification and Mapper/Transaction Form
  In order to transfer the SDL view to the view of financial transactions and financial instruments in the Calculation and Valuation Methods, you assign the mapper and the transaction form to the SDL templates. You can use the Risk Basis classification to make a distinction between the templates, which you use to map multiple different financial transactions and financial instruments.

- Assignment of SDL Relationships and Risk Basis Relationships
  The SDL uses object relationships to create complex templates. Object relationships are used to map the relationships between options and their underlyings, for example, but also to document where individual transactions are covered by one general contract. Not all of these relationships are relevant for the measurement of financial transactions and financial instruments. In the mapping function, the system separates the other object relationships, in particular those that are used for documentation purposes only.
- Assignment of the Properties of Cash Flows
  The SDL and the valuation methods each have their own views of cash flows. For this reason, the cash flow generation type [page 461] and the flow type of the SDL have to be mapped to the *cash flow view*, the *cash flow set*, and the *cash flow display indicator*.

> **Caution**
>
> Note that you should not classify the flow types in the SDL that are relevant for the IAS solution as effective. The system always creates the cash flow category *Effective View* during cash flow generation. The Calculation and Valuation Methods do not normally use this effective view because the original cash flows should be used as the basis for calculating the effective interest rate. The *Subclassification* field in the flow type defines which cash flow category the valuation methods access. If the system contains a cash flow that has at least one flow type that has the subclassification *Effective*, then the system uses the effective view. In this case, it uses only those cash flows from the SDL that belong to flow types that are flagged as *Effective*. This can lead to errors in later calculations.
>
> You flag flow types as*effective* or *not effective* in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Generation* > *Edit Flow Types for Flow*.
>
> The cash flows are transferred in the generic mappers GENINT and GENSTR.

> **Note**
>
> For more information about the Customizing settings for mapping, see the documentation in the Implementation Guide (IMG) under *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *SDL Connection*.

Features

The system contains predefined generic mappers to map SDL templates to valuation structures. These mappers can be used for multiple templates:

- GENINT (Mapper for General Templates)
  This mapper maps a single node in the SDL hierarchy to a node in the hierarchy of the valuation structure. The system does this by filling the node of the valuation structure with the data from the SDL template.
- GENSTR (Mapper for General Structured Templates)
  This mapper operates on the level of the individual nodes in an SDL template in the same way as mapper GENINT. It also initiates the processing of all child nodes of the node in question. On the child nodes, the system calls the associated mapper, and transfers to it the subhierarchies of the SDL template that relate to this child node.
  In this way, it is possible to process the entire hierarchy in an SDL template recursively.

In addition to the generic mappers, SAP also provides the following mappers for specific financial transactions and financial instruments, and the parts of them that can be priced:

- LOAN_GEN (Generic Mapper for Loans)
  This can be used for loans without special rights or call rights, as well as for loans with these rights. Mapper LOAN_GEN replaces mappers LOAN, LOAN_OPT (loan with option rights) and LOAN_OPTR (option rights for loan).
- SWAP (Mapper for Swaps)
- FWD_FX (Mapper for Forward Exchange Transactions)
- CAPFLOOR (Mapper for Caps and Floors)
- MCB (Mapper for Multi-Currency Bonds)

> **Note**
>
> All mappers are implemented as Business Add-Ins (BAdIs) and can be found in the Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *SDL Connection* > *BAdI: Mapper* .This gives you the option of creating your own custom mappers. This is usually only necessary if you create your own transaction forms. Even if you define new templates in the SDL, the mappers provided by SAP should still be adequate.

4.3.10.3 Methods for Cash Flow Refinement

Definition

Methods that, at runtime, change financial transactions and financial instruments after they have been selected from Source Data Layer (SDL), and in this way prepare them for the analysis process.

Use

Bank Analyzer uses cash flow refinement methods in the following applications:

- Fair Value Server [page 628]
- Cash Flow Hedge Analysis [page 1107]

The 'due date scenario' cash flow refinement method is used only in Strategy Analyzer and Cash Flow Hedge Analysis. You use the 'due date scenario' cash flow refinement method to simulate the 'maturity' of account balances, and to apply this simulation before the financial transactions in question are actually analyzed. Similarly, the other cash flow refinement methods prepare the financial transactions and financial instruments for the Bank Analyzer analyses. The settings made in Customizing define which application can use which cash flow refinement methods.

The system calls the cash flow refinement methods by means of the method schema [page 534], in which multiple cash flow refinement methods are grouped together. Therefore, you assign only the method schema, and not the individual cash flow refinement methods, to the Bank Analyzer applications. Since the selection and the order of the cash flow refinement methods is largely defined for the individual applications, SAP delivers suitable method schemas.

Integration

The cash flow refinement methods are applied to the valuation structure [page 495], which contains all the information that is needed to measure financial transactions, financial instruments, or subcontracts. There are two main types of cash flow refinement method:

- Modification of the valuation structure
  Some of the methods for cash flow refinement only modify the content of the valuation structure. This is the case in the 'cash flow filter' refinement method, which you use to remove certain cash flows from the valuation structure.
- Creation of multiple valuation structures
  Other cash flow refinement methods generate multiple valuation structures from one valuation structure. This is because later evaluations require a particular business view of the financial transaction or financial instrument in question, which can be mapped only be means of two or more valuation structures. The 'cash flow splitting' refinement method creates a separate valuation structure for each splitting level and splitting value, for instance.

Control Parameters

You use the control parameters to define how cash flow refinement methods are to be used. You do this by entering in the control parameters the respective cash flow refinement method for the valuation rule [page 490]. This enables you to treat certain groups of financial transactions or financial instruments in a particular way. You configure the methods for cash flow refinement in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas*. You make the settings that are specific to valuation rules from within the method environment under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Edit Method Environments*.

The system calls some of the cash flow refinement methods more than once. An example is market data determination. Strategy Analyzer calls this cash flow refinement method once before cash flow splitting, and again after cash flow splitting. This enables you to assign different market data to each valuation structure generated in cash flow splitting. Configuration values are used for these cash flow refinement methods to tell the system when to call the methods.

Enhancement

SAP delivers the cash flow refinement methods that are required for the applications in Bank Analyzer. However, you can create your own custom methods for cash flow refinement. You do this using a BAdI, which is available in Customizing for *Bank Analyzer*, and for which you can store implementations. You can find this BAdI in Customizing under *Processes and Methods* > *Cash Flow Refinement* > *Method Schemas* > *BAdI: Refine Valuation Structures*.

Example

SAP provides the following methods for cash flow refinement:

Method ID	Method Name	Function

1005	Breaking Down of Implied Options [page 503]	The system breaks down financial transactions whose interest rate conditions contain variable components, and hence implied options, into their component parts. This enables the price calculator to price the component parts as options.
1010	Due Date Scenarios [page 508]	The system applies the specified due date scenarios to the balance in question in order to simulate the maturity of account balances.
1011	Market Data Scenarios	The system creates additional valuation structures from the valuation structure of the transaction or position in question, and based on the scenario set. It creates new valuation structures for the current market data set, and for all that the transferred scenario set contains. The only difference between the new valuation structures and the original valuation structure is the market data that they contain.
1012	Cash Flow Fixing [page 506]	The system determines future cash flows with variable interest rates on the basis of the market data and market data scenarios.
1013	Interest Capitalization	The system recalculates the cash flows and the nominal amount for loans for which an interest capitalization is agreed instead of an outgoing interest payment. From a technical point of view, on the payment date the system generates two cash flows of the same amount, each of which has a different sign (+/-). The cash flows are a nominal cash flow and an interest cash flow, and both describe the interest that is to be capitalized.
1014	Zero Conditions	For loans, the system displays the interest as zero for specific periods. One period is between the conclusion of the contract and the first interest rate condition period. The other is between the last interest condition period and the final repayment.

1015	Interest Calculation Methods	The system applies one interest calculation method to all transactions. Here the system adjusts only the interest rates; the actual amounts of interest remain the same.
1016	Cash Flow Splitting [page 518]	The system creates new valuation structures for the current transaction or position for all the splitting levels that are specified in the splitting environment. The splitting level and splitting value are listed in the valuation structures, and the transaction or position cash flows are adapted as required.
1017	Cash Flow Filter [page 513]	The system deletes cash flows from the valuation structure as required.
1020	Cash Flow Determination [page 504]	The system determines cash flows for loans with unscheduled repayment options or call options.
1030	Distribution of the Premium/Discount [page 507]	The system distributes premiums and discounts to the interest payment dates of the relevant transactions.
1040	Rollover [page 510]	Depending on the setting, the system rolls cash flows over the actual end of the term of the financial transaction.
1060	Generation of Hypothetical Derivatives [page 515]	The system generates a hypothetical derivative in order to test the fair value effectiveness for cash flow hedging relationships.
1070	Generation of Par Bonds [page 517]	The system generates a par bond in order to prepare for the calculation of the hedge adjustment for the hedge items and hedging instruments in portfolio fair value hedging relationships.
1100	Market Data Assignment	The system derives the market data set that is required for valuing the transactions and positions without any information about the cash flow view and the cash flow splitting.

Example

SAP delivers method schema CFHA1 for Cash Flow Hedge Analysis. Method schema CFHA1 contains a selection of the predefined cash flow refinement methods listed above. For more information, see Method Schemas [page 534].

4.3.10.3.1 Break-Down of Implied Options

Use

The system uses this function to break down financial transactions whose interest rate conditions contain implied options so that the Fair Value Server can price the parts as options.

For example, floaters contain both a reference interest rate and an interest rate cap. Floaters with interest rate caps can be mapped as a combination of a floater without a cap, and a cap. The cap is a short-term option on the reference interest rate, which is exercised at the maximum value that was agreed for the reference interest rate of the original floater. Floaters with floors can be mapped as simple floaters plus the floor as the option.

Integration

The function that breaks down implied options is one of the cash flow refinement methods [page 499] provided by SAP.

The Fair Value Server [page 628] calls this method as part of the pricing of financial transactions and financial instruments.

Prerequisites

In Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Edit Method Environments*, you have already defined for which financial transactions and financial instruments the system is to use the *Break Down Implied Options* method. You use valuation rules [page 490] to define this.

Features

One manipulation method is used to break down the implied options of simple products such as floaters, and another to break down the implied options of structured products such as swaps and forward rate agreements (FRAs). In both methods, the system analyzes the upper node to determine whether the implied options are to be broken down. You use valuation rules to define this process.

The system does the following:

- Simple products (for example, floaters)
  The system analyzes the first level that contains the interest cash flows. It then breaks down the implied options for this level by deleting the cap or floor and creating a new subnode for the corresponding option.
- Structured products (for example, swaps or forward rate agreements that have caps or floors)
  For each subnode, the system calls the break-down of implied options for simple products, and replaces the existing nodes with the newly generated options.

> **Note**
>
> So that variable loans with call options or repayment options are not processed, the system checks whether there are any subnodes in addition to the interest rate condition, or whether these nodes contain any references to other transactions. If not, then the system breaks down the implied options. Variable loans with call options or repayment options are not broken down into separate options because the loan acts as the underlying of the options, so the call options and repayment options are options on underlyings with implied options.

4.3.10.3.2 Cash Flow Determination

Use

In this function, the system calculates cash flows for loans with unscheduled repayment options or call options in order to determine prepayments.

The system contains the following methods for cash flow determination:

- **Percentage Rate**
  The system uses a percentage rate stored in Customizing to calculate the remaining debt. It does this by multiplying the current remaining debt by the stored percentage rate and the delta of the unscheduled repayment or call option.
- **Using Multiple Percentage Rates in a Due Date Scenario**
  The calculation is the same as that used in the method above. Note that you have to use an absolute due date scenario, and that the time points in the due date scenario have to match the dates on which the unscheduled repayment option or call option are exercised. For American options and Bermuda options the system takes the earliest possible exercise date.
- **Using the Price Calculator to Price the Option**
  The system calls the price calculator in order to calculate the delta of the unscheduled repayment option or call option. It then multiplies the existing remaining debt with the delta and the percentage rate of the unscheduled repayment right or call option in order to calculate the remaining debt.

> **Note**
>
> You can also store your own methods for cash flow determination. You do this by using a BAdI as a separate cash flow refinement method [page 499] to create the methods, and then adding them into the appropriate method schema [page 534].

Integration

The cash flow determination method is one of the cash flow refinement methods predefined in the system. As is the case for all the cash flow refinement methods [page 499], the system uses a method schema [page 534] to call the cash flow determination method. The following method schemas, which are predefined in the system, contain the cash flow determination method:

- Fair Value Server (calculation of the full fair value)
- Cash Flow Hedge Analysis
- Strategy Analyzer

Prerequisites

In Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schema* > *Basic Settings for Methods* > *Edit Cash Flow Determination Methods*, you have created an ID for the cash flow determination method and defined the method that the system is to use to determine the cash flows.

You have then assigned the ID of the cash flow determination method to a method environment and a valuation rule. You make these settings in Customizing under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Edit Method Environments*.

> **Note**
>
> If you use transaction forms other than those defined by SAP, then in Customizing under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Basic Settings for Methods* > *Assign Cash Flow Determination to Transaction Form*, you need to specify which transaction forms the system is to include when it determines the cash flows. This step is required so that the system can recognize which transaction forms are those for loans with unscheduled repayment options or call options.

Features

Depending on the type of loan, the system uses one of the following approaches in the cash flow determination method:

- **Loan with Annual Repayment**
  Once the system has determined that there is an unscheduled repayment option or a call option, the annual repayment installment does not change. The system merely shortens the term.
- **Loan with Final Repayment or Repayment in Installments**
  The system adjusts only the interest cash flows; the principal for which the interest is calculated is reduced. The system also reduces the repayment installment at the end of the term by the value of the unscheduled repayment option or call option.

4.3.10.3.3 Cash Flow Fixing

Use

This function uses market data and market data scenarios to determine variable-rate cash flows. The cash flows are fixed in a way that is consistent with the forward rates of the associated reference interest rates, and in accordance with the conditions stored in the transaction (such as the date on which the interest rate is fixed).

Besides fixing the cash flows, the system also translates the cash flows into the payment currency if the condition currency and the payment currency are different.

Integration

The cash flow fixing method is one of the cash flow refinement methods [page 499] predefined in the system. The system calls the cash flow fixing function either from a method schema [page 534], or directly from the price calculator [page 547].

The following method schemas, which are predefined in the system, contain the cash flow fixing method:

- Cash Flow Hedge Analysis
- Fair Value Server (calculation of the full fair value)
- Price Calculator
- Strategy Analyzer
  Note that when cash flow fixing is called from the method schema in Strategy Analyzer, the program does not use forward rates; the cash flows are fixed for the horizon date.

The cash flow fixing function calls functions in the Market Data application in order to calculate forward interest rates. For more information, see the section and the documentation about the and .

Prerequisites

- So that the price calculator calls the cash flow fixing function, you enter settings for cash flow fixing in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *Edit Price Calculator Environments*. Here you can enter the following settings:
  - Switch off the that are before the evaluation date
  - Switch on the convexity adjustment [page 580]
  - Specify whether bid or ask rates are to be used for currency translation
  - Specify that the system is to ignore the exchange rate stored in the transaction

  For more information, see the documentation for the Customizing activity *Edit Price Calculator Environments*.
- There are no Customizing settings for the cash flow fixing function when it is called from the method schema. In this case, the system uses the standard setting. In the standard setting, interpolation is used for historical interest rate fixing as well, and the convexity adjustment is switched off. The system takes middle rates for currency translation, and uses the exchange rates stored in the transactions.

> ⚠ Caution
>
> Note that the Customizing settings you enter in the price calculator have no effect when the system calls cash flow fixing from the method schema. In the same way, it ignores the Customizing settings in the price calculator if cash flow fixing is called from the method schema before the price calculator is called.

4.3.10.3.4 Distribution of Premiums and Discounts

Use

This function is used to prepare data for analyses that are based on cash flows such as the gap analysis in Strategy Analyzer.

The purpose of this preparation function is to ensure that premiums and discounts are shown in later analyses in chronological order. The system does this by distributing the premiums and discounts to the interest payment dates of the transaction in question. During the analysis, the system then displays for each interest payment date all the interest payments that have already been made, and the premium/discount distribution.

Integration

The function that distributes premiums and discounts is one of the cash flow refinement methods [page 499] provided by SAP. As in the other cash flow refinement methods, the system uses a method schema [page 534] to call the function for distributing premiums and discounts.

No settings are made in Customizing for the premium/discount distribution method of cash flow refinement.

You define how the premium/discount distribution is displayed in the reporting functions for the gap analysis in Strategy Analyzer by setting the *Net/Gross Display* checkbox in Customizing for *Bank Analyzer* under *Analytics* > *Strategy Analyzer* > *Valuation* > *Create Valuation Environment and Assign Market Data Set to Der. Strategy*. Under *Settings for Gap Analysis*, you define how the system is to handle premiums and discounts in the position evaluation. If you set the *Net/Gross Display* checkbox to *Gross Display*, the development of the nominal principal and of the premium and discount are shown on different sides of the balance sheet. If you set the checkbox to *Net Display*, the netted payments are shown on the same side of the balance sheet.

Features

Based on the premiums and discounts, the system generates cash flows for each interest payment date. These premium/discount cash flows have the following properties:

- The total of the premium or discount cash flows is equal to the original premium or discount.
- The premium or discount cash flows are weighted according to the nominal amount that is the basis of the interest payments made on the respective interest payment date.

- The system flags the cash flows that were generated in the distribution of premiums and discounts as fictitious cash flows.

4.3.10.3.5 Use of Due Date Scenarios

Use

You use this function to apply due date scenarios to account balances to simulate cash flow maturities. The system uses the due date scenarios that you defined in Customizing to simulate the development of balances. The due date scenarios are usually based on empirical values about how balances change.

Due Date Scenarios

You can define the following types of due date scenarios:

- Absolute Due Date Scenarios
  In absolute due date scenarios, you define fixed time points, and for each of these time points you specify the percentage by which the balance is reduced or increased. If you enter a positive value for the percentage change, this reduces the balances. If you enter a negative value, this increases the balance.
- Relative Due Date Scenarios
  In relative due date scenarios, you define the time points at which amounts are to flow in or flow out relative to a start date. This start date is then the start date of due date scenario. If you do not enter a date for the start of the due date scenario, then the system takes the evaluation date of the relevant application as the start date, for instance the key date in cash flow analysis.
  To define the inflows and outflows for relative due date scenarios, you use due date periods. A due date period can be used more than once to reflect seasonal effects.

  > **Example**
  >
  > Let us assume that you want to simulate the development of the balances of a number of conventional checking accounts. The balances develop in similar way over the course of a year, but within one year they can fluctuate considerably. Therefore, you could define one due date period that reflects the way in which the balances typically develop over the course of a year, and then use this due date period in the due date scenario as many times as you want to simulate the development of the balances.

If you assume that a certain portion of an account's balance always remains on the account, then you can enter this portion as a base amount in the due date scenario. If you do not enter a base amount, the system uses the value 0%.

Simulated Interest Payments

You can simulate interest payments for accounts and products that are similar to accounts. These reflect both the interest rate stored for the account, and the development of the account's balance, which is determined when the due date scenario is applied. When you define how the interest payments are to be simulated, you can store an interest calculation method, which the system uses to calculate the interest.

> **Note**
>
> You can enter an end date to define the period for which the system is to simulate the interest payments. You enter the end date in the associated due date scenario. We recommend that you enter an end date for

performance reasons, in particular if you are using interest payment patterns that contain small periods of time, such as daily or monthly interest payments.

If you do not enter an end date, the system stops calculating the simulated interest after a defined time period. For the *Daily* days method, the system calculates the simulated interest for a period of 10 years; in the *Monthly* and *Quarterly* methods it does so for 100 years; for the *Half-Yearly* method for 300 years; and for the *Annual* method for 500 years.

Integration

The use of due date scenarios is one of the cash flow refinement methods [page 499] predefined in system. The system calls this method in the same way as the other cash flow refinement methods, which is by means of a method schema [page 534]. Due date scenarios are part of the predefined method schemas for Cash Flow Hedge Analysis and Strategy Analyzer

Prerequisites

- You have defined a due date scenario in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Basic Settings for Methods* > *Edit Due Date Scenarios*.
- You have defined to what extent the system is to include interest when it simulates the development of account balances. You configure this in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Basic Settings for Methods* > *Edit Simulated Interest Payments*.
- You have also assigned the ID of the due date scenario to a method environment and a valuation rule. You enter these settings in Customizing under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Edit Method Environments*.

  > **Note**
  >
  > If you use transaction forms other than those defined by SAP, then in Customizing activity *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Basic Settings for Methods* > *Assign Transaction Forms to Due Date Scenarios*, you need to specify which transaction forms the system is to include when it processes due date scenarios.

4.3.10.3.6 Rollover

Use

You use this function to roll over cash flows from loans beyond the maturity dates of the loans in order to simulate the reinvestment of the nominal amount.

Integration

The rollover function is one of the cash flow refinement methods [page 499] predefined in the system. The system calls this method in the same way as the other cash flow refinement methods, which is by means of a method schema [page 534]. The rollover method is contained in one of the predefined method schemas for Cash Flow Hedge Analysis

Prerequisites

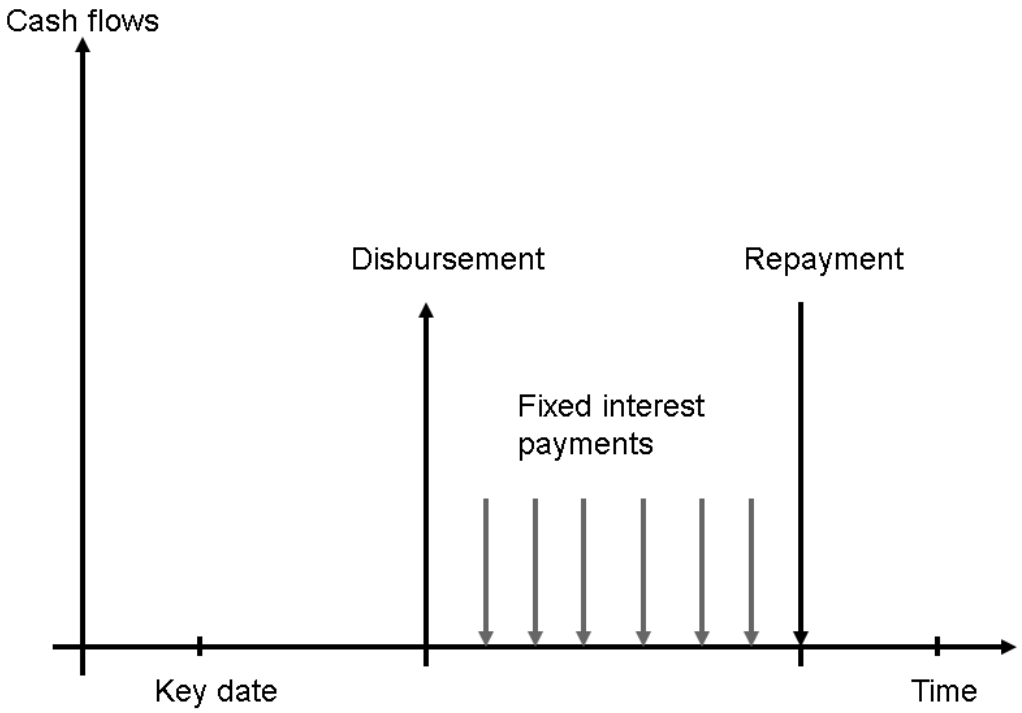
You have already defined the rollover method in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Edit Method Environments*, and assigned a valuation rule.

You have defined maturity bands in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Processes* > *Edit Valuation Run Type and Edit Aggregation Run Type*.

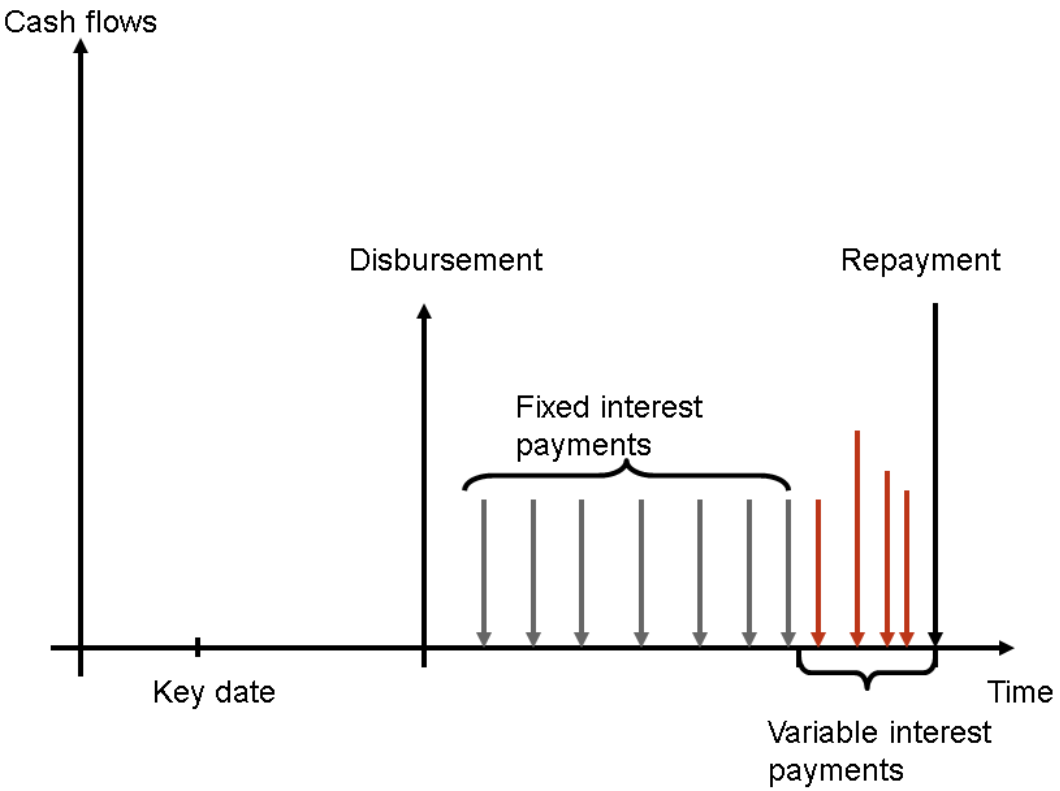
Features

The system shifts the repayment date of the loan, and fills the period between the original repayment date and the new repayment date with variable interest payments. You use the residual maturity and overall maturity to define the financial transactions that are to be rolled over. To define this for each type of transaction, you can use valuation rules [page 490].

The following figures show how the rollover method works. The system shifts the repayment date, and fills the period between the original repayment date and the new repayment date with variable interest payments:



Loan Before the Application of the Rollover Method



Loan After the Application of the Rollover Method

You can use the following parameters to define how the transaction is rolled over:

- Length of the rollover
  You have the following options for defining the new repayment date, and hence the length of the rollover:
  - Entering a fixed date
    The system shifts the repayment of the loan to the date you entered, and rolls the transaction over for the period up to this date.
  - Entering a time period
    The system rolls the loan over for a fixed period beyond its maturity date. The repayment of the loan is then shifted by this period.
- Time points for simulated interest payments
  You have the following options for generating simulated interest payments:
  - Maturity band
    The system places simulated payments in all the time buckets of the maturity band that you specified in Cash Flow Hedge Analysis.
  - Underlying transaction
    The system uses the interest payment frequency of the financial transaction to generate simulated interest payments.

  You use the *Underlying* settings to prevent the system from displaying more simulated interest payments than are possible for the interest payment cycle of the financial transaction.

> **Example**
>
> For example, if you use the *Maturity Band* setting for a loan for which interest is paid every three months and a maturity band in which interest is paid each month, the system displays the simulated interest payments for each month. If you use the *Interest Cycle* setting, the system displays the simulated interest payments only in every third section of the maturity band.

- Amount of the simulated interest payments
  The system uses the reference interest rate from the financial transaction to generate the simulated interest payments. You can also specify a margin that the system then adds to the individual cash flows.

  > **Note**
  >
  > Note that you can roll over interest payments only, and not annuities.

4.3.10.3.7 Cash Flow Filter

Use

You use this function to remove cash flows that have a particular business role from financial transactions so that the system does not include them in analyses.

Cash flow filters can be used in Cash Flow Hedge Analysis [page 1107] to exclude from the analyses the interest cash flows of financial transactions that are already used as hedged items in a fair value hedging relationship.

Integration

The cash flow filter is one of the cash flow refinement methods [page 499]. You use valuation rules to define how the cash flow filter is used.

Prerequisites

- You have already assigned the cash flow filter refinement method to the method schema that you want to use. You configure method schemas in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Edit Method Schemas*.

  > **Note**
  >
  > The cash flow filter refinement method is part of method schema CFHA1 that is predefined in Cash Flow Hedge Analysis.

- You have already created one or more cash flow filters for the method environment that you want to use. You configure cash flow filters in Customizing for *Bank Analyzer* under *Processes and Methods* > *General*

Calculation and Valuation Methods > *Cash Flow Refinement* > *Method Schemas* > *Edit Method Environments*.

- You have already assigned the cash flow filter to the application required. For cash flow hedge analysis, for example, you assign the cash flow filter in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Valuation* > *Edit Cash Flow Hedge Environment*.

Example

You use this function in Cash Flow Hedge Analysis to exclude, for example, the following financial transactions and cash flows from analyses:

- Hedging instruments that are already designated in fair value hedging relationships
  Financial transactions that are already designated must not also be used as hedging instruments in cash flow hedging relationships.
- Interest cash flows of financial transaction and financial instruments that are already used, either partially or wholly, as hedged items in fair value hedging relationships
  In fair value hedging relationships, the interest cash flows of financial transactions or financial instruments are swapped for variable interest cash flows, which has the effect of hedging them. Therefore, these interest cash flows must not be included in Cash Flow Hedge Analysis as well.

In order to filter out the relevant hedging instruments and interest cash flows, do the following:

1. Define a valuation rule for hedged items, and another for hedging instruments. You then use these rules to define how the cash flow filter is used. You define valuation rules in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivations* > *Determination of the Valuation Rule* > *Create Valuation Rules*.

Valuation Rule	Meaning
GG_FILTER	Valuation rule for hedged items in Cash Flow Hedge Analysis
SG_FILTER	Valuation rule for hedging instruments in Cash Flow Hedge Analysis

> **Caution**
>
> Note that we do not deliver any valuation rules, and that the entries GG_FILTER and SG_FILTER are give here as examples only.

1. In the required method schema, define the cash flow filters as follows:

Valuation Rule	Cash Flow Indicator	Activity
GG_FILTER	Interest	Delete
SG_FILTER	All cash flows	Delete

> **Note**
>
> In the entry for cash flow filter SG_FILTER for hedging instruments, you can see that the cash flow filter is defined just on the basis of cash flows. By making this entry, you define that the system is to delete all interest cash flows from the hedging instruments that are already designated. This means that in cash flow hedge analysis, the system ignores these hedging instruments, which are then "empty".

1. Assign the cash flow filter to the cash flow hedge environment.

Cash Flow Hedge Environment	Filter is active	Valuation Rule for Hedged Items	Valuation Rule for Hedging Instruments
CFHA	Checkbox set	GG_FILTER	SG_FILTER

> **Caution**
>
> The entry CFHA for the cash flow hedge environment is used here as an example only. You have to use the cash flow hedge environment that you want to use in the cash flow hedge analysis. Note that you have to assign the method schema that you defined for cash flow filters GG_FILTER and SG_FILTER to this cash flow hedge environment.

> **Note**
>
> Also note that the system automatically assigns the valuation rules GG_FILTER and SG_FILTER to the hedged items and hedging instruments that are already used in fair value hedging relationships. Since the valuation rules are not derived by the Bank Analyzer derivation tool, no further Customizing settings are required.

4.3.10.3.8 Generation of Hypothetical Derivatives

Use

You use this function to generate hypothetical swaps. This prepares the fair value effectiveness test [page 1036] for cash flow hedging relationships under the hypothetical derivative approach. The system generates a hypothetical swap for each hedging instrument in the cash flow hedging relationship.

Integration

The function for generating hypothetical derivatives is one of the cash flow refinement methods [page 499] provided by SAP and is called by the following components:

- Cash Flow Hedge Analysis [page 1107]
  The function for generating hypothetical derivatives is called at the start of the valuation run in order to analyze hypothetical derivatives, and then to display them in the reporting functions in Cash Flow Hedge Analysis.

- Fair Value Server [page 628]
  The function for generating hypothetical derivatives is called when the fair value effectiveness of a cash flow hedging relationship is being tested.

SAP provides method schemas that contain the cash flow refinement method *Generation of Hypothetical Derivatives*. So that hypothetical derivatives are generated as required, the method schemas contain the following manipulation methods in the order given below:

1. Assignment of market data using Method Parameter 1 (general derivation)
2. Generation of Hypothetical Derivatives
3. Assignment of market data using Method Parameter 4 (derivation of market data for hypothetical derivatives)

The method schema calls the cash flow refinement methods for each transaction that is to be processed. For this reason, the component calling the cash flow refinement method tells the method whether it is to generate a hypothetical derivative for the current transaction.

If so, the system generates a separate valuation structure for the hypothetical derivative. This valuation structure exists only at runtime, and is not stored on the database.

Prerequisites

- Fair Value Server
  In *Bank Analyzer* Customizing under *Processes and Methods* > *General Valuation and Calculation Functions* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Structure Types*, you need to have assigned a method schema that contains the cash flow refinement method *Generation of Hypothetical Derivative* to the valuation structure type.
- Cash Flow Hedge Analysis
  In *Bank Analyzer* Customizing under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Valuation* > *Edit Cash Flow Hedge Environment*, you need to have assigned the cash flow hedge environment a method schema that contains the cash flow refinement method *Generation of Hypothetical Derivative*.
  In *Bank Analyzer* Customizing under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Valuation* > *Edit Environment for Hypothetical Derivative Approach*, you need to have created an environment for the hypothetical derivative approach (hypo environment).
  In *Bank Analyzer* Customizing under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Processes* > *Edit Valuation Run Type*, or when you call the online runs, you need to have set the *Use Approach for Hypothetical Derivatives* indicator.
- Fair Value Effectiveness Test
  When you created effectiveness test runs, or when you called the online runs, you set the *Approach for Hypothetical Derivatives* indicator

Features

The system calls the cash flow refinement method *Generation of Hypothetical Derivative* for each swap.

The process is as follows for hedging instruments in a hedging relationship: The current swap, the market data, the reference interest rate stored in the hedging relationship, and the date on which the swap in question was placed in the hedging relationship are used as the input parameters.

The hypothetical derivative has the following properties:

- The nominal volume is equal to the nominal volume of the current swap.
- The variable side is based on the reference interest rate used for the hedging relationship.
  The system uses the last payment date of the current swap as the last payment date of the hypothetical swap. Based on this payment date, the system then generates the other payment dates until it reaches the designation date. It does this by using the interest payment cycle and the interest calculation method of the reference interest rate of the hedging relationship.
  The system calls the price calculator to calculate the cash flows and the present value of the variable side of the hypothetical swap. It takes the reference interest rate of the hedging relationship for the coupons of the hypothetical swap.
- The present value is zero on the designation date.
  The system calculates the coupons of the fixed side in such a way that the present value of the hypothetical swap is zero on the designation date. The current reference interest rate from the market data set is used to discount the cash flows.

In Cash Flow Hedge Analysis, the cash flow refinement method *Generation of Hypothetical Derivative* is also called for non-designated hedging instruments. In this case, instead of using the reference interest rate of the hedging relationship, the system takes the reference interest rate stored in the environment for the hypothetical derivative approach. Also, it does not use the designation date, but instead the key date of the associated valuation run in Cash Flow Hedge Analysis.

4.3.10.3.9 Generation of Par Bonds

Use

You use this function to generate par bonds. These par bonds are used by the portfolio fair value calculator [page 1175] to calculate, for a particular time bucket, the hedge adjustment for the portfolio of hedged items in a portfolio fair value hedging relationship.

> **Note**
>
> The system uses par bonds to calculate the hedge adjustment only if you have mapped nominal cash flows (and no other type of cash flow) to the time buckets.

Integration

The generation of par bonds function is one of the cash flow refinement methods [page 499] predefined in system. The portfolio fair value calculator calls the method during the pricing of portfolio items. For more information, see the documentation for the Portfolio Fair Value Hedge [page 1156] component.

Prerequisites

- You have defined a yield curve in Customizing under *Financial Services* > *Foundation* > *Market Data* > *Interest* > *Edit Reference Interest Rates and Yield Curves*. The interest payment cycle of the yield curve should be adjusted to fit the time buckets. To ensure that accrued interest is calculated in a standard way, you need to have entered the following settings for the par bond method: *linear interest calculation*, *broken periods at end of term*.
- You have created a market data set in Customizing for *Financial Services*, under *Foundation* > *Market Data* > *Create Market Data Set*. This market data set must contain the yield curve you defined.
- In Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivations* > *Determination of the Market Data Set and the Spread Curve Type*, you have defined a derivation strategy for your market data set. For example, you can use the characteristics of the hedging relationship for the derivation process.
- In Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Valuation* > *Edit Portfolio Item Environment*, you have created a valuation environment. You must also have assigned a method schema to the valuation environment. This method schema must contain the cash flow refinement method *Generation of a Par Bond*. The derivation strategy for determining the market data set must also have been assigned to the calculation environment

Features

The system creates a par bond that has the following properties:

- The start of the term is the designation date of the portfolio fair value hedging relationship.
- The maturity date is the end of the current time bucket.
- Upon maturity, the par bond is quoted at 100 percent (in other words, the nominal is 1).
- The coupons are constructed in such a way that the present value at the start of the term is also 1. The system uses the interest payment cycle of the yield curve that you stored in the derived market data set.
  All the coupons have more or less the same value, but this depends on the interest calculation method stored in the yield curve. If the end of the time bucket does not coincide with the interest payment date, the system takes the interest that accrued by the end of the time bucket as the last coupon (due to the par bond method *Linear Interest Rate Calculation*, *Broken Periods at End of Term*, which is stored in the yield curve).

4.3.10.3.10 Cash Flow Splitting

Use

You use this function to split cash flows into their various business components, such as a covered and uncovered portion. The cash flows you break down in this way can be displayed in the reporting functions in Strategy Analyzer.

The following splitting levels can be used:

- Syndication split [page 520]
- Disbursement split [page 522]
- Coverage split [page 523]
- Margin split [page 527]

Integration

The cash flow splitting method is one of the cash flow refinement methods [page 499] predefined in the system. As for the other cash flow refinement methods, the system uses a method schema [page 534] to call cash flow splitting.

Prerequisites

You have entered the following settings in Customizing for *Bank Analyzer* under ▶ *Processes and Methods* ▶ *General Calculation and Valuation Methods* ▶ *Cash Flow Refinement* ▶ *Method Schemas* ▶ *Edit Splitting Environments* ▶:

- You have defined the splitting levels that you want the system to use.
- You have defined which method the system is to use for each splitting level (applies to the syndication split, coverage split, and margin split only).
- For the margin split, you have also defined whether, for the covered portion or uncovered portion, the system is to apply the market interest rate from the input data, or whether it is to add a predefined margin to this rate based on a current market rate.
- If required, you have also flagged individual splitting levels as not relevant. In this case, you decide for which splitting values the system flags the entire transaction.

The splitting environment settings are made for each valuation rule [page 490] to enable you to control the cash flow splitting method for each transaction. You have to enter the input parameters that the system requires for the individual splitting levels in each position. Key figures from the key figure repository in Bank Analyzer are provided for this purpose.

If you want to take into account only certain splitting levels, the cash flows supplied by the Source Data Layer (SDL) have to have already been split for the splitting levels omitted.

You store the splitting environment in the Customizing for the respective application:

- Strategy Analyzer
  You enter the splitting environment in Customizing for *Bank Analyzer* under ▶ *Analytics* ▶ *Strategy Analyzer* ▶ *Valuation* ▶ *Create Valuation Environment and Assign Market Data Set to Derivation Strategy* ▶.
- Fair Value Server
  You enter the splitting environment in Customizing for *Bank Analyzer* under ▶ *Processes and Methods* ▶ *General Calculation and Valuation Methods* ▶ *Present Value Calculation* ▶ *Fair Value Server* ▶ *Edit Valuation Structure Types* ▶.

- Extended price calculator
  You enter the splitting environment in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *Edit Extended Price Calculator Environment*.

The cash flow splitting method can be used by *Strategy Analyzer* and *Accounting Processes*, but is not a prerequisite.

Features

The cash flow splitting refinement method is called from within the *Cash Flow Engine* method schema during the valuation run in Strategy Analyzer. Usually either no splitting levels or all splitting levels are taken into account. If the cash flows selected from the Source Data Layer (SDL) have already been split into one or more splitting levels, the system splits them into the remaining splitting levels only. The system processes the splitting levels in the following order: syndication split, disbursement split, coverage split, and margin split.

1. Syndication split
2. Disbursement split
3. Coverage split
4. Margin split

4.3.10.3.10.1 Syndication Split

Use

In a syndication split, the system splits financial transaction cash flows into two parts: the part provided by the bank and the part provided by other parties. The total transaction (the view that has not been pooled) is also transferred.

To describe the bank's risk, you can split cash flows into the part provided by the bank and the part provided by the other parties for transactions such as syndicated loans, asset backed securities, and overdraft facilities for current accounts where the balances and interest are netted. The syndication split comprises two methods: the linear method, in which all repayments are split proportionally, and the near/far method, in which the near repayments are displayed as the part provided by other parties and the far repayments as the part provided by the bank; linear splitting is the default method.

Integration

The syndication split is the first splitting level used by the cash flow splitting method. The disbursement split, coverage split, and margin split splitting levels follow on from the syndication split.

The system creates a separate valuation structure for the splitting values "part provided by the bank" and "part provided by other parties", as well as for the transaction as a whole. Any financial transactions selected from the Source Data Layer (SDL) that have already been split are transferred unchanged.

Prerequisites

In Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Edit Splitting Environments*, you specify whether the system is to take into account the syndication split splitting level, and which method it should apply.

If the system does not apply the syndication split method, then it needs a valuation structure that already contains the values for the syndication split splitting level. If the system applies the syndication split method, it needs the total pooling amount, the pooled amount, and the non-pooled amount (see the table below). The pooled amount and the non-pooled amount add up to the total pooled amount.

Input Key Figures for the Syndication Split

Key Figure	Meaning
63GKOBT	Total pooled amount
63KOBT	Pooled amount
63UKOBT	Amount that is not pooled

Features

The system splits the financial transaction cash flows in such a way that the ratio of the part provided by the bank to the part provided by other parties corresponds to that of the disbursement. You can use the following methods:

- Linear (default setting)
  The system splits each individual cash flow into the part provided by the bank and the part provided by other parties in such a way that the ratio of the two parts corresponds to that of the disbursement.
- Near/far
  The system assigns the latest cash flows to the part provided by other parties and the earlier cash flows to the part provided by the bank, and the ratio of the totals of the two parts corresponds to that of the two parts of the disbursement. The system does this as follows:
  1. First, the part of the disbursement provided by other parties is distributed. The system keeps adding the last financial transaction cash flow and the cash flows before it to the part provided by other parties until this is no longer possible.
  2. If an amount remains once the part provided by other parties has been split, the system assigns the next earliest cash flow from the financial transaction to the part provided by other parties to the same amount as the remaining amount. The remaining amount is assigned to the part provided by the bank.
  3. All the cash flows before this cash flow are assigned to the part provided by the bank.

4.3.10.3.10.2 Disbursement Split

Use

The system splits cash flows into a settled (disbursed) and an unsettled portion. Amounts that have already been settled belong to the disbursed portion. Amounts that can still be disbursed belong to the unsettled portion. This includes external open loan commitments, for example. The total of the unsettled portions calculated is the nominal principal minus the disbursements that have been made.

Integration

The disbursement split is the second splitting level after the syndication split level that uses the cash flow splitting method. In the disbursement split, the system splits only the part provided by the bank, as determined in the syndication split. Following the disbursement spit, the method processes the coverage split and margin split splitting levels.

The system creates a separate valuation structure for the splitting values *Disbursed* and *Unsettled*, and for the transaction as a whole. Any financial transactions selected from the Source Data Layer (SDL) that have already been split are transferred unchanged.

Prerequisites

In Customizing for *Bank Analyzer* under ▶ *Processes and Methods* ▶ *General Calculation and Valuation Methods* ▶ *Cash Flow Refinement* ▶ *Method Schemas* ▶ *Edit Splitting Environments* ▶, you specify whether the system is to take into account the disbursement split splitting level.

The input data are the input key figures for the syndication split or a complete valuation structure. By means of the valuation type or cash flow indicator, the system calculates the disbursed and unsettled amounts of the cash flows.

Features

The disbursement split is always linear. The system splits each cash flow into disbursed and unsettled portions so that they have the same relationship as the balance for the unsettled portion of the entire financial transaction.

Example

A loan is recorded in the system with a nominal principal of EUR 60,000, annuity installments of EUR 6,000 per year, a nominal interest rate of 6%, and a term and fixed-rate period of 10 years. The balance of the loan is EUR

48,000, the unsettled amount is EUR 12,000. Assuming that unsettled amounts are immediately fully disbursed, the following disbursement split results:

Disbursement Split of the Loan

Year	Total Repayment	Total Interest	Settled Repayment	Unsettled Repayment	Interest Paid	Unsettled Interest
0	-60000	0	-48000	-12000	0	0
1	2400	3600	1920	480	2880	720
2	2544	3456	2035	509	2765	691
3	2697	3303	2157	539	2643	661
4	2858	3142	2287	572	2513	628
5	3030	2970	2424	606	2376	594
6	3212	2788	2569	642	2231	558
7	3404	2596	2724	681	2076	519
8	3609	2391	2887	722	1913	478
9	3825	2175	3060	765	1740	435
10	32421	1945	25937	6484	1556	389
Total	0	28366	0	0	22693	5673

4.3.10.3.10.3 Coverage Split

Use

The coverage split is applied to loans in order to plan the coverage of lending products, such as mortgage loans or public sector loans. The system splits the cash flows into the following portions:

- Portion eligible for cover: portion that can be refinanced by mortgage bonds or public mortgage bonds
- Portion that cannot be registered: portion that cannot be refinanced by mortgage bonds until it has been registered
- Portion that is not eligible for cover: portion that cannot be refinanced by mortgage bonds

Integration

The coverage split is the third splitting level after the syndication split and the disbursement split that the cash flow splitting method uses. The system divides the settled and the unsettled portion again. Following the coverage spit, the method processes the margin split splitting level.

The system creates a separate valuation structure for each splitting value. Any financial transactions selected from the Source Data Layer (SDL) that have already been split are transferred unchanged.

> **Note**
>
> In the Fair Value Server, you can use the coverage split in the calculation of the full fair value key figure. For more information, see Coverage Split in the Full Fair Value Calculation [page 647].

Prerequisites

In Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Edit Splitting Environments*, you specify whether the system is to take into account the coverage split splitting level.

If the system does not apply the coverage split method, then it needs a valuation structure that already contains the values for the coverage split splitting level. If it uses the coverage split splitting method, the system needs valuation structures for the previous splitting levels, and the portions that are and are not eligible for cover, the portion that cannot be registered, and the portions of the unsettled amount that are and are not eligible for cover. All these values have to be available for the pooled amount and the amount that has not been pooled (see the table below).

Input Key Figures for the Coverage Split

Key Figure	Meaning
63BKSD	Pooled, unsettled amount that can be covered
63BKSND	Pooled, unsettled amount that cannot be covered
63BKVD	Pooled, settled amount that can be covered
63BKVND	Pooled, settled amount that cannot be covered
63BKVNR	Pooled, settled amount that cannot be registered
63BUSD	Unsettled amount that can be covered and is not pooled
63BUSND	Unsettled amount that cannot be covered and is not pooled
63BUVD	Settled amount that can be covered and is not pooled
63BUVND	Settled amount that cannot be covered and is not pooled

Key Figure	Meaning
63BUVNR	Settled amount that cannot be registered and is not pooled

Features

The coverage split is applied to lending transactions only. The system flags borrowing transactions as *Eligible for Cover*, *Not Eligible for Cover* or *Cannot Be Registered* depending on the transaction data transferred.

- Relative distribution
  The system divides the individual cash flows into covered and uncovered portions in proportion to the total covered and uncovered portion.
- Absolute distribution
  For lending transactions, the cash flows are split in such a way that the loan develops into the covered portion over time. Based on your Customizing settings, the system does the following:
  - The system first assigns the long-term repayments and interest to the portion that is eligible for cover. It then assigns the cash flows of the portion that cannot be registered. Lastly, it assigns the cash flows of the portion that is not eligible for cover.
  - The system first assigns the nearest (in terms of time) repayments and interest to the portion that is not eligible for cover, and then the cash flows of the portion that cannot be registered, and finally the cash flows of the portion that is eligible for cover.

The following constraints apply:

- Unsettled cash flows can be split only into the portions that are and are not eligible for cover.
- Settled cash flows can be split into the portions that are and are not eligible for cover, and the portion that cannot be registered.
- Cash flows are split only if the covered portion and the non-covered portion of the unsettled cash flows are not equal to zero. For settled cash flows, two of the three possible values must not be equal to zero.
- If only one amount of the settled or unsettled portions from the input parameters is not equal to zero, no coverage split is carried out. The system flags the financial transaction as *Eligible for Cover*, *Not Eligible for Cover*, or *Cannot Be Registered* as per the transaction data.

Example

A loan is entered in the system with a nominal principal of EUR 60,000, an annuity installment of EUR 6,000 per year, a nominal interest rate of 6%, and a maturity and fixed-rate period of 10 years. The balance of the loan is EUR 48,000; the unsettled amount is EUR 12,000. Unsettled amounts are settled in full straight away. Of the settled amount of EUR 48,000, EUR 40,000 is eligible for cover (EFC) and EUR 8,000 is not eligible for cover (not EFC); the portion that cannot yet be registered is zero. The total unsettled amount is not eligible for cover. Absolute distribution is used.

Disbursement Split of the Loan

Year	Repayment Before Split	Interest Before Split	Settled Repayment	Interest Paid	Unsettled Repayment	Unsettled Interest
0	-60000	0	-48000	0	-12000	0
1	2400	3600	1920	2880	480	720
2	2544	3456	2035	2765	509	691
3	2697	3303	2157	2643	539	661
4	2858	3142	2287	2513	572	628
5	3030	2970	2424	2376	606	594
6	3212	2788	2569	2231	642	558
7	3404	2596	2724	2076	681	519
8	3609	2391	2887	1913	722	478
9	3825	2175	3060	1740	765	435
10	32421	1945	25937	1556	6484	389
Total	0	28366	0	22693	0	5673

Coverage Split of the Loan

Year	Repayment Before Split	Interest Before Split	Settled Repayment Not EFC	Interest Paid Not EFC	Settled Repayment EFC	Interest Paid EFC	Unsettled Repayment Not EFC	Unsettled Interest Not EFC
0	-60000	0	-8000	0	-40000	0	-12000	0
1	2400	3600	1920	480	0	2400	480	720
2	2544	3456	2035	365	0	2400	509	691
3	2697	3303	2157	243	0	2400	539	661
4	2858	3142	1887	113	399	2400	572	628
5	3030	2970	0	0	2424	2376	606	594
6	3212	2788	0	0	2569	2231	642	558
7	3404	2596	0	0	2724	2076	681	519
8	3609	2391	0	0	2887	1913	722	478
9	3825	2175	0	0	3060	1740	765	435

Year	Repayment Before Split	Interest Before Split	Settled Repayment Not EFC	Interest Paid Not EFC	Settled Repayment EFC	Interest Paid EFC	Unsettled Repayment Not EFC	Unsettled Interest Not EFC
10	32421	1945	0	0	25937	1556	6484	389
Total	0	28366	0	1201	0	21492	0	5673

The system therefore splits the fully disbursed loan into two subloans.

4.3.10.3.10.4 Margin Split

Use

The system splits cash flows to show the values with and without the margin.

The system contains the following methods:

- **Nominal Method**
  Nominal approach, in which the margin is based on the remaining capital.
- **Effective Method**
  Nominal approach, whereby the margin is deducted on the basis of the remaining capital, which results in a cleared effective interest rate account at the end of the fixed-rate period.
- **Method for Effective Cash Flows**
  Effective approach oriented towards the development of the effective remaining capital.

Integration

The margin split is the last splitting level that uses the cash flow splitting method. The system divides the settled and the unsettled portion again. Following the coverage spit, the method processes the *margin split* splitting level.

The system creates a separate valuation structure for each splitting value. Any financial transactions selected from the Source Data Layer (SDL) that have already been split are transferred unchanged.

Prerequisites

In Customizing for Bank Analyzer under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Edit Splitting Environments* you specify whether the system is to take into account the margin split splitting level. Here you also define which methods the system applies for the margin split, and whether the opportunity interest rate is recalculated for the coverage and non-coverage parts.

If the system does not carry out the margin split, then it needs a valuation structure that already contains the values for the *margin split* splitting level. If the system carries out a margin split, then it needs valuation structures

that already contain the values for the previous splitting levels, plus the following input data: margin spread 1, margin spread 2, effective interest rate, market interest rate or opportunity interest rate of the covered portion and the non-covered portion, plus a net volume factor.

The input key figures for the margin split are stored in the primary objects in the SDL. To store this data in a financial transaction, on the SAP Easy Access screen choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Transaction* > *Change Financial Transaction*. Choose the Key Figures tab page, which contains the Percentages tab page. On the *Percentages* tab page, choose *Insert Row* and enter the key figure categories listed in the table below.

Input Key Figures for the Margin Split

Key figure	Description
EFFZ	Effective interest rate
MZD	Market interest rate (covered)
MZND	Market interest rate (not covered)
NVF	Net volume factor
FSPD	Margin spread 1 (covered)
FSPND	Margin spread 1 (uncovered)
OSPD	Margin spread 2 (covered)
OSPND	Margin spread 2 (uncovered)

Features

The system calculates the margin using the method specified in the splitting environment.

> **Note**
>
> Any margin spreads specified are removed before the margin is calculated.

> **Caution**
>
> Note that not all cash flow views are needed for the margin split methods in gap analysis reporting.
>
> - In the method for effective cash flows, only the effective view is relevant.
> - In the nominal and effective methods, all the cash flow views, with the exception of the effective view, are relevant.

Example

A margin split is to be carried out for a mortgage loan using the nominal approach.

The nominal principal of the loan is EUR 100,000, EUR 80,000 of which is disbursed and EUR 20,000 of which is unsettled. EUR 60,000 is eligible for cover and EUR 40,000 is not. An annuity of EUR 10,000 per year is agreed upon as the form of repayment. The nominal interest rate is 6 percent and the discount 5 percent. The fixed-rate period is set at 5 years. The term of the loan is 16 years.

To keep the example simple, only the fixed-rate period view is shown below. The unsettled premium is settled immediately. The discount is broken down yearly instead of monthly. The margin split is applied only to the nominal cash flows. The cash flows in the table are shown as nominal values.

Interest and Repayment Schedule for the Loan

Time of Payment	**0**	**1**	**2**	**3**	**4**	**5**	**Total**
Full disbursement	-100000.00						-100000.00
Disagio	5000.00						5000.00
Annuity		10000.00	10000.00	10000.00	10000.00	10000.00	50000.00
Interest amount		6000.00	5760.00	5505.60	5235.94	4950.09	27451.63
Repayment		4000.00	4240.00	4494.40	4764.06	5049.91	22548.37
Account balance	100000.00	96000.00	91760.00	87265.60	82501.54	77451.63	
Discount		1092.83	1049.12	1002.78	953.67	901.60	5000.00
Cash flow	-95000.00	10000.00	10000.00	10000.00	10000.00	10000.00	

The linear method is used for the disbursement split.

Disbursement Split of the Loan

Time of Payment	**0**	**1**	**2**	**3**	**4**	**5**	**Total**
Total repayment	-10000.00	4000.00	4240.00	4494.40	4764.06	82501.54	0.00
Settled repayment	-80000.00	3200.00	3392.00	3595.52	3811.25	66001.23	0.00

Unsettled repayment	-20000.00	800.00	848.00	898.88	952.81	16500.31	0.00
Total interest		6000.00	5760.00	5505.60	5235.94	4950.09	27451.63
Interest paid		4800.00	4608.00	4404.48	4188.75	3960.07	21961.30
Unsettled interest		1200.00	1152.00	1101.12	1047.19	990.02	5490.33
Total premium/ discount		1092.83	1049.12	1002.78	953.67	901.60	5000.00
Settled premium/ discount		874.27	839.29	802.23	762.93	721.28	4000.00
Unsettled premium/ discount		218.57	209.82	200.56	190.73	180.32	1000.00

Coverage Split of the Loan

Time of Payment	**1**	**2**	**3**	**4**	**5**	**Total**
Settled repayment	3200.00	3392.00	3595.52	3811.25	66001.23	80000.00
Settled repayment eligible for cover	0.00	0.00	0.00	0.00	60000.00	60000.00
Settled repayment not eligible for cover	3200.00	3392.00	3595.52	3811.25	6001.23	20000.00
Unsettled repayment	800.00	848.00	898.88	952.81	16500.31	20000.00
Unsettled repayment not eligible for cover	800.00	848.00	898.88	952.81	16500.31	20000.00
Interest paid	4800.00	4608.00	4404.48	4188.75	3960.07	21961.30
Settled interest eligible for cover	3600.00	3600.00	3600.00	3600.00	3600.00	18000.00

Settled interest not eligible for cover	1200.00	1008.00	804.48	588.75	360.07	3961.30
Unsettled interest	1200.00	1152.00	1101.12	1047.19	990.02	5490.33
Unsettled interest not eligible for cover	1200.00	1152.00	1101.12	1047.19	990.02	5490.33
Settled premium/discount	874.27	839.29	802.23	762.93	721.28	4000.00
Settled premium/discount eligible for cover	655.70	655.70	655.70	655.70	655.70	3278.50
Settled premium/discount not eligible for cover	218.57	183.59	146.53	107.23	65.58	721.50
Unsettled premium/discount	218.57	209.82	200.56	190.73	180.32	1000.00
Unsettled premium/discount not eligible for cover	218.57	209.82	200.56	190.73	180.32	1000.00

The nominal approach is applied for the margin split. The transferred effective interest rate is 7.3356, the opportunity interest rate of the covered portion is 5.6330, and the opportunity interest rate of the non-covered portion is 5.7056. The net volume factor is 0.9000. The transferred values are applied to the effective interest rate that was calculated by the system without any correction or adjustment being made to them.

Margin Split of the Loan

Time of Payment	**1**	**2**	**3**	**4**	**5**	**Total**
Settled repayment eligible for cover	0.00	0.00	0.00	0.00	60000.00	60000.00
Settled repayment not eligible for cover	3200.00	3392.00	3595.52	3811.25	6001.23	20000.00

Unsettled repayment not eligible for cover	800.00	848.00	898.88	952.81	16500.31	20000.00
Interest paid that is eligible for cover, with margin	3600.00	3600.00	3600.00	3600.00	3600.00	18000.00
Interest paid that is not eligible for cover with margin	1200.00	1008.00	804.48	588.75	360.07	3961.30
Unpaid interest not eligible for cover with margin	1200.00	1152.00	1101.12	1047.19	990.02	5490.33
Settled remaining capital eligible for cover	60000.00	60000.00	60000.00	60000.00	60000.00	
Settled remaining capital, not eligible for cover	20000.00	16800.00	13408.00	9812.48	6001.23	
Unsettled remaining capital, not eligible for cover	20000.00	19200.00	18352.00	17453.12	16500.31	
Settled margin eligible for cover	919.40	919.40	919.40	919.40	919.40	
Settled margin, not eligible for cover	293.40	246.46	196.70	143.95	88.04	
Unsettled margin, not eligible for cover	293.40	281.66	269.22	256.04	242.06	
Interest paid that is eligible for cover, with margin	2680.60	2680.60	2680.60	2680.60	2680.60	

Interest paid that is not eligible for cover with margin	906.60	761.54	607.78	444.80	272.03
Unpaid interest not eligible for cover with margin	906.60	870.34	831.90	791.15	747.96

The spread is the product of the remaining capital, the net volume factor, and the difference between the transferred effective interest rate and the market interest rate.

In the example, the nominal interest rate without a spread for the covered portion is 4.46766 and the nominal interest rate without a spread for the non-covered portion is 4.53300.

4.3.10.3.11 Liquidation Scenario

Use

You use this function to simulate potential sales of lending transactions (such as securities and loans). For example, if, for liquidity reasons, you want to restructure your lending business, you can use liquidity scenarios to test the effect of the restructuring on your liquidity risk.

Integration

In the cash flow engine, Strategy Analyzer applies the liquidation scenarios you created to the transactions. Strategy Analyzer uses the net present value calculated based on the settings in the liquidation scenario, and not the original value of a transaction (for example, the nominal value).

Prerequisites

You have created liquidation scenarios in Customizing for *Cash Flow Refinement* under *Method Schemas* > *Basic Settings for Methods* > *Edit Liquidation Scenarios*. You also have to have made the settings for derivation in IMG activity *Edit Liquidation Scenarios* as the system applies liquidation scenarios only to those transactions to which the system can assign a liquidation scenario by means of the combination of valuation rule and liquidation ID. You must also have set the *Apply Liquidity Scenario for Valuation Rule* indicator for the valuation rule in question in IMG Activity *Edit Method Environments*.

After your system has gone live, you edit the liquidation scenarios on the *SAP Easy Access* screen under *Financial Services* > *Bank Analyzer* > *Analytics* > *Strategy Analyzer* > *Current Settings* > *Edit Liquidation Scenarios*.

Features

- You can use the liquidation scenario, as with other methods for cash flow refinement, to influence the cash flows of all realizable transactions from Source Data Layer.
- To avoid short sales, the system considers only those transactions that currently have a nominal volume, a position, and so on. The total liquidation amount cannot exceed 100 % (short sale), but by no means does it have to be 100 %. If the liquidation amount is under 100 %, the system does not use the residual volume in the simulation.
- A liquidation scenario can contain one or more 'times of liquidation' that have different degrees of liquidation (in percentage).
- The early sale of transactions can cause a loss, which you can consider using the repayment installment (in percentage).

Example

In the Implementation Guide (IMG) for *Cash Flow Refinement* under *Method Schemas* > *Basic Settings for Methods* > *Edit Liquidation Scenarios*, you can find an example of how the system assigns liquidation scenarios to transactions.

4.3.10.4 Method Schemas

Definition

Grouping of cash flow refinement methods in the order in which they are called.

Use

You use method schemas to group cash flow refinement methods [page 499] together that are to be used to prepare financial transactions and financial instruments for the actual evaluation process. The method schema defines which cash flow refinement methods are to be used for a particular application, and in which order the system is to call them. SAP provides method schemas for the following applications:

- Cash Flow Refinement [page 489]
- Fair Value Server [page 628]
- Cash Flow Hedge Analysis [page 1107]

The method schemas provided by SAP are system settings that you cannot change. However, you can copy the predefined method schemas, using the customer namespace, and then change them. In this way you can add your own cash flow refinement methods. Since the selection and order of cash flow refinement methods is largely already defined for the individual applications, if you create your own custom method schemas, SAP advises you not to change the order of the cash flow refinement methods, and not to delete any of the cash flow refinement methods.

Integration

To be able to use a method schema, you must assign it in Customizing to the relevant application. For example, you assign the method schema for Cash Flow Hedge Analysis to the Cash Flow Hedge Analysis environment. You also need to add the settings for the cash flow refinement methods that the method schema contains to the respective application. You store the settings for the cash flow refinement methods in Customizing for the method environment. You then assign the method environment (separately from the method schema) to the application. In this way you can use different method environments for one method schema.

> **Note**
>
> For an overview of the manipulation methods provided by SAP, see the Methods for Cash Flow Refinement [page 499] document.

Example

SAP provides a method schema for Strategy Analyzer, for example. This has the technical name SRCFE, and is usually referred to as the Cash Flow Engine. The system calls the Cash Flow Engine in Strategy Analyzer at the beginning of the valuation run in order to prepare the cash flows of the selected transactions and positions, and to transfer them to the gap analysis or the NPV analysis. Within the Cash Flow Engine, the system calls multiple cash flow refinement methods, which modify the valuation structures of the selected transactions and positions.

After the system has called the Cash Flow Engine, each transaction or position has a valuation structure for each cash flow view, the market data scenarios, and the splitting values. In the valuation structures, the cash flows are modified in such a way that Strategy Analyzer can analyze the transactions and positions appropriately.

The following table shows which cash flow refinement methods the method schema in Strategy Analyzer contains.

Method Schema SRCFE (Cash Flow Engine)

Step Number	Method for Cash Flow Refinement
10	1100 (Market Data Assignment)
15	1011 (Market Data Scenarios)
20	1012 (Cash Flow Fixing)
30	1014 (Zero Conditions)
35	1020 (Cash Flow Determination)
40	1016 (Cash Flow Splitting)
45	1030 (Distribution of Premium/Discounts)
50	1013 (Interest Capitalization)

60	1100 (Market Data Assignment)
70	1010 (Due Date Scenarios)
80	1015 (Interest Calculation Methods)

> **Note**
>
> The step numbers specify the order in which the system calls the cash flow refinement methods. Here it is the relationship between the numbers, and not the numbers themselves, that defines the actual order in which the methods are called.

4.3.11 Fair Value Production Process (FS-BA-PM-FVP)

Use

This component calculates the fair value (with and without accrued interest) for financial products, and saves the results in the Result Data Layer (RDL). The various components, such as Accounting for Financial Products, can access the fair values stored in the RDL here.

The component can carry out currency translations for the *Accounting for Financial Instruments* (AFI) scenario.

Integration

The Fair Value Production Process (FVPP) uses the Fair Value Server [page 628] to calculate the fair value.

Calculation and Valuation Process Manager (CVPM)

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Since CVPM governs the Fair Value Production Process (FVPP), its processes can be run as parallel jobs.

Accounting for Financial Products

The Fair Value Production Process calculates the fair value for the financial positions in accounting. It does this by reading the financial instrument spreads that were calculated when the internal business transactions were updated.

Other Components

The Fair Value Production Process calculates the fair value for financial positions and financial instruments.

Features

The following process categories are available in the Fair Value Production Process:

- Fair Value Production: Batch Process
- Fair Value Production: Delta Process
- Fair Value Production: Online Process

For more information, see Fair Value Production Process [page 537].

As of *banking services from SAP 8.0*(FSAPPL 400) Support Package 9, an additional fair value production process is available for the AFI scenario only. For more information, see Fair Value Production Process as of FSAPPL 400, SP09 [page 540].

4.3.11.1 Fair Value Production Process

Use

Several components in Bank Analyzer use the same fair value. These include the processes of the Accounting for Financial Products [page 678] component, for example. The Fair Value Production Process calculates the fair value (with and without accrued interest), and can save it in the Results Data Layer (RDL).

Prerequisites

To calculate fair values using the Fair Value Production Process, you need to have made the following settings in Customizing for *Bank Analyzer*:

- Results Type
  You have created a results type that is assigned to the results category `HKMVL` (market value) under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*.
- Determination of the market data area
  - **Accounting for Financial Products**
    You have assigned a market data area to the legal entity and the accounting system. You make the assignment under *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Assign Market Data Areas to Legal Entities and Accounting Systems*..
  - **Other users**
    You have made the settings for determining the market data area under *General Calculation and Valuation Methods* > *Fair Value Production Process* > *Derivation* > *Derivation of the Market Data Area*.
- Fair Value Run Types
  You have defined fair value run types under *Processes and Methods* > *General Calculation and Valuation Methods* > *Fair Value Production Process* > *Edit Fair Value Run Types*.
- Worklist Creation

Accounting for Financial Products
If you want to calculate the fair value of financial transactions, you need to have created an enhanced worklist for the Fair Value Production Process under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Index Function* > *Edit Enhanced Worklists*.
If you calculate fair values for positions, the primary data source uses Results Data Layer. You need to have assigned results types and an aggregation level to the accounting system under *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Define Accounting Systems*.
Other users
You have created an enhanced worklist under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Index Function* > *Edit Enhanced Worklists*.

- Step Sequences
  You have determined step sequences for the analytical processes `/BA1/AL_Q1_FVPP_BATCH`, `/BA1/AL_Q1_FVPP_DELTA` and `/BA1/AL_Q1_FVPP_ONLINE` under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Fixed Step Sequences for Analytical Processes*. In this storage step you enter the results type defined in the Results Data Layer.
- Fair Value Server
  You can find the Customizing activities relevant for the Fair Value Server under *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server*.
- Price Calculator
  You can find the Customizing activities for the Price Calculator under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator*.

Process

To run the fair value production process (FVPP) as a batch, delta or online process on the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Fair Value Production Process*.

The process categories available differ in the following ways:

- **Fair Value Production: Batch Process**
  In the batch process, the Fair Value Production Process calculates the fair value for the financial positions, financial transactions, and financial instruments, and saves the results in the RDL.

> **Note**
>
> In the batch process, the server *Enhanced Worklist Service* (`EWL`) can be used for the primary data source for the financial transactions. Depending on the application in which the batch process is running, the following servers can be used for the primary data sources for financial instruments:
>
> - Accounting for Financial Products:
>   Primary data source *Aggregation to RDL Financial Object Balances*
> - Other Users
>   Primary data source *Enhanced Worklist Service* (`EWL`) for the Source Data Layer position

- **Fair Value Production: Delta Process**
  In the delta process, incorrect objects from the batch process are processed again (postprocessing). Before you start the delta process, you can correct the information stored in Customizing, the market data, and the transaction data, for instance.
  You define the delta process `/BA1/AL_Q1_FVPP_DELTA` in CVPM Customizing in the same way as the batch process `/BA1/AL_Q1_FVPP_BATCH` apart from the fact that you define only one step sequence, and not two as is the case for the batch process for accounting. The delta list contains incorrect financial transactions and incorrect positions for the positions for financial instruments.

  > **Note**
  >
  > Server *FVPP: Delta List* can be used for the primary data source for the delta process.

- **Fair Value Production: Online Process**
  You can use the online process to calculate fair values for individual objects on an ad hoc basis. The system displays the results on the screen. Since the online process is intended for analyzing errors, the primary data source should be defined in the same way as in the batch process.

  > **Note**
  >
  > You can display the error messages for each transaction here. Unlike in the batch process, you can display a detail log for the Fair Value Server. You can also branch to the market data that the system used.

The Fair Value Production Process does the following:

1. Enriches the parameters
   The system reads the Customizing for the process and uses the dynamic selection to fix the primary data source.
   It also initializes the delta list in the batch process.
2. Creates the worklist
   The system reads the primary data source to get the packages for parallel processing. It then creates the object lists and packages for parallel processing.
3. Enriches the data
   The system calculates a fair value for each entry in the object list, and creates a list of the objects that contain errors. This is called the delta list. It processes the delta list later on in the delta process. It enriches the data in parallel processing.
4. The system stores the data in the RDL (batch and delta process) or displays the results on the screen (online process).

> **Note**
>
> You use delta process `/BA1/AL_Q1_FVPP_DELTA` to correct objects that contain errors without having to restart the entire batch job. The system processes the incorrect objects in the delta list in a separate batch job.

> **Note**
>
> You use the online process for test purposes only. You run this process for a small number of transactions. You specify the key date, time stamp, and a worklist of the type “primary data source”. The system displays the results on the screen, and does not write them to the RDL.

4.3.11.2 Fair Value Production Process as of FSAPPL 400, SP09

Use

Multiple components in Bank Analyzer use the same fair value. For example, the Accounting for Financial Products [page 678] component uses the fair value. The Fair Value Production Process (FVPP) calculates the fair value (with and without accrued interest), and can save it in the Results Data Layer (RDL).

The Fair Value Production Process available as of *banking services from SAP 8.0* (FSAPPL 400), Support Package 9 provides the following additional functions that can be used for the *Accounting for Financial Instruments* (AFI) scenario:

- The process can translate the results into all currencies that are defined for the legal entity.
- If the Fair Value Server has determined the FI spread, the Fair Value Production Process can store this in the RDL for accounting.

Prerequisites

You have configured the process `/BA1/AL_Q1_FVPP` (Fair Value Production Process) in the Calculation and Valuation Process Manager. For more information, see Calculation and Valuation Process Manager (CVPM) [page 1622].

You have made the following settings in Customizing for *Bank Analyzer*:

- Results Type
  You have created a results type that is assigned to the results category `HKMVL` (market value) under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*.
- Determination of the Market Data Area
  You have assigned a market data area to the legal entity and the accounting system. You make the assignment under *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Assign Market Data Areas to Legal Entities and Accounting Systems*.
- Fair Value Run Types
  You have defined fair value run types under *Processes and Methods* > *General Calculation and Valuation Methods* > *Fair Value Production Process* > *Edit Fair Value Run Types*.
- Accounting System
  If you use the standard system step sequence, you have assigned a results data area and a results type for the market value to your accounting system under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Basic Settings* > *Assign Global Valuation Environments to Accounting Systems*.
- Worklist Generation
  You have defined a primary data source for the selection of financial positions under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.
  This process creates the worklist of financial positions on the server `BA_FP_MIDX` (Financial Position ID Express) with the data reading group `02` (Results Data and Analytical Data). For more information, see Servers for General Primary Data Sources [page 1597].
- Step Sequences

You have edited a standard step sequence for the analytical process `/BA1/Q1_FVPP` under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Fixed Step Sequences for Analytical Processes*. Alternatively, you have created your own step sequence under *Edit Custom Step Sequences for Analytical Processes*.

- Fair Value Server
  You can find the Customizing activities relevant for the Fair Value Server under *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server*.
- Price calculator
  You can find the Customizing activities for the Price Calculator under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator*.

Procedure

To start the Fair Value Production Process (FVPP), on the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Fair Value Production Process* > *Fair Value Production Process*:

- Restrict the run according to legal entity.
- Enter a key date and a fair value run type.
- Choose a step sequence.
  If you use the standard system step sequence, you must enter an accounting system.

> **Note**
>
> If you select *Detail Log* in the selection screen for the process, the Fair Value Server generates and saves a detail log during the calculation. You can display this within the application log after you have executed the process.

You can schedule the process as a periodic background job.

Depending on how you have configured the step sequence for the CVPM process, corresponding activities run in the system. The following is an example of the process flow in the standard step sequence for an integrated scenario with accounting:

1. **Parameter enrichment**
   Step execution using class `/BA1/CL_AL_Q1_FVPP`
   The system reads the relevant Customizing settings and compiles the selection criteria for the creation of the worklist.
2. **Worklist creation**
   Data source category: *General primary data source*
   In this step you enter the data source that you created in Customizing. The system reads the primary data source to get the packages for parallel processing. It then creates the object lists and packages for parallel processing.
3. **Data enrichment**
   Step execution using class `/BA1/CL_AL_Q1_FVPP`
   The system calculates a fair value for each entry in the object list provided no fair value has been determined yet. In other words, if no object currency has been defined. It sets incorrectly edited objects aside and processes them again later. It enriches the data in parallel processing.

Master data and imported results are read with the source data time stamp; market data with the market data time stamp.
The system calculates the results in the position currency `/BA1/C55AOCURR`. It then translates the results into functional currency. In Customizing, you can also make settings for translation into up to three local currencies (local currency, group currency, hard currency). To do so, you have to define these currencies for the legal entity and add the corresponding key figure from the following table to the result type:

Key Figure	Description	Assignment to Result Type
`64MVCLF`	Clean Price in Functional Currency	Provided by SAP
`64MVCLT`	Clean Price in Transaction Currency	Provided by SAP
`64MVCLG`	Clean Price in Group Currency	Optional
`64MVCLH`	Clean Price in Hard Currency	Optional
`64MVCLL`	Clean Price in Local Currency	Optional
`64MVDIF`	Dirty Price in Functional Currency	Provided by SAP
`64MVDIT`	Dirty Price in Transaction Currency	Provided by SAP
`64MVDIG`	Dirty Price in Group Currency	Optional
`64MVDIH`	Dirty Price in Hard Currency	Optional
`64MVDIL`	Dirty Price in Hard Currency	Optional

> **Note**
>
> The system does not calculate fair values for aggregation objects.

4. **Saving the results**
   - **Save**
     Step category: *Data Enrichment* or *Save*
     Step execution using class `/BA1/CL_AL_Q1_STORAGE_ACC`
   - **Display**
     Step category: *Results Display*
     Step execution using class `/BA1/CL_AL_FW_DISPLAY`

   In this step you enter the results type defined in the Results Data Layer. The system stores the data in the RDL or displays the results on the screen

You can restart the process by using the corresponding CVPM standard function. For more information, see Capability to Restart Parallel Processes [page 1632].

4.3.12 Maturity Grouping (FS-BA-PM-MTG)

Use

Maturity grouping is not an accounting process, but a support process that has to be executed at the end of a period after the accounting processes have run completely. The system determines how the cash flows or the book value for contracts are to be distributed over the remaining term of the contract so that these can be explained in the notes to the financial statement:

- For cash-flow-based products, the process distributes the cash flows across the maturity groups.
- For products that are not cash-flow-based, the process distributes the book value across the maturity groups.

The maturity is calculated from the days between the key date for balance sheet preparation and the due date of the payment, which has to be assigned to a maturity group. Maturity grouping is part of the *Financial Instruments Calculation and Valuation Process Management* process component, and is configured in the Calculation and Valuation Process Manager (CVPM).

Prerequisites

You have configured the `/BA1/AL_Q1_MATGRP` (Maturity Grouping) process in the Calculation and Valuation Process Manager. For more information, see Calculation and Valuation Process Manager (CVPM) [page 1622].

You have made the following settings in Customizing for *Bank Analyzer*:

- *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*
  You define a result type here that you assign to the result category `HKMGR` (Maturity Grouping).
- *Infrastructure* > *Communication and Worklist Service* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*
  You define a primary data source here for the maturity grouping worklist. Select all the characteristics that you will later need in the derivation.
  If you create the worklist from financial positions, you exclude financial position categories that are not relevant, (for example, funding positions) in the *Attributes* area. Make settings in the attributes so that the posting date is taken into account.
- *Infrastructure* > *Communication and Worklist Service* > *Data Sources* > *General Derivations*
  You make settings for at least one derivation module for deriving the method for maturity grouping here.
- *Infrastructure* > *Communication and Worklist Service* > *Data Sources* > *Module Editor for Processes and Methods* > *Edit Modules for Processes and Methods*
  You create a module here that is used to call the derivation, and you include it in the step sequence in CVPM Customizing. You can also create a module for custom manipulations. If you implement custom fields, you must take these into account throughout all CVPM steps. For example, in the derivation module they need to be defined as input and output fields and assigned to each other.
- You have edited a standard step sequence for the analytical process `/BA1/AL_Q1_MATGRP` under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Fixed Step Sequences for Analytical Processes*. Alternatively, you have created your own step sequence under *Edit Custom Step Sequences for Analytical Processes*.

- *Processes and Methods* > *General Calculation and Valuation Methods* > *Maturity Grouping*:
  - *Create Maturity Band*
    You create a maturity band here, in other words, a time grid in the future that defines the date values for calculating and displaying the results. You can enter the date values for the maturity band individually, or generate them relative to the last generated date value or on the start date of the maturity band. Note that the first date of the maturity band is the start date of the first interval; all other date entries are end dates of the intervals.
  - *Define Methods for Maturity Grouping*
    You define methods that the process calls during maturity grouping and their properties, for example, the determination category.
  - *Define Results Data Area and Result Type for RDL*
    If you use the standard system step sequence, you define here where in the Results Data Layer (RDL) the system stores the results.

Procedure

To run this process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Maturity Grouping* > *Execute Maturity Grouping*:

- Restrict the run according to legal entity.
- Enter a key date.
- Choose a step sequence.
  If you use the standard system step sequence, you must enter an accounting system.

You can schedule the process as a periodic background job.

Depending on how you have configured the step sequence for the CVPM process, corresponding activities run in the system. The following is an example of the process flow in the standard step sequences, `MATGRP_ACC` and ,`MATGRP_AGG` for an integrated scenario with accounting:

1. **Parameter enrichment**
   Step execution using class `/BA1/CL_AL_MATGRP_PE`
   In this step, the system determines the results data area based on the accounting system. The system also reads the relevant Customizing settings, for example, the global valuation environment.
2. **Worklist creation**
   Data source category: *General primary data source*
   The process creates the worklist of financial positions on the server `BA_FP_MIDX` (financial position ID express), in other words, the same server that the accounting processes use. This ensures consistency between the balance sheet and notes to the financial statement. For more information, see Servers for General Primary Data Sources [page 1597].
   Only contracts that are legally binding at the time of balance sheet creation are relevant for the worklist. You exclude irrelevant financial position categories in the primary data source (see *Prerequisites*). You can limit positions that are relevant for analysis using dynamic selections for financial position characteristics.
3. **Derivation of methods for maturity grouping**
   A method bundles the following settings:
   - Determination category
     - Generation of cash flows

- Fair Value Server
- Determination based on nominal value
- Determination based on fair value
- Determination based on amortized cost
- No cash flows

- Cash flow indicator to be used
- Book value as target value (optional)
  If you have selected the corresponding checkbox, the process reads the book value as the target value (book value category: *Book Value* - BV) from accounting.
- Distribution mode
- Maturity band

You can use step category *Data Enrichment Using Module Editor* to call a derivation module. Using the derivation module, you define which method the process uses in which case (see *Prerequisites*). You can use financial position characteristics, such as the financial position type or the holding category as criteria for this, for example.

Example

Holding Category of the Contract	Method
Market value (fair value)	Method with determination based on fair value and with proportional distribution
Other liabilities (OLI)	Method with determination category *No Cash Flows*, the book value as the target value and weighting-based distribution to the first maturity group.
Loans and receivables (LAR)	Method with determination based on amortized cost and with proportional distribution

Note

Note that for aggregation objects in source data aggregation you need to derive a method that uses the determination category *No Cash Flows*.

4. **Determination of cash flows**
   Step category: *Data Enrichment*
   Step execution using class `/BA1/CL_AL_MATGRP_CF`
   In this step, the process determines the cash flows for each financial position based on the determination category of the derived methods. The system includes only cash flows with the cash flow indicator that are defined in the method. For financial instruments, cash flows are scaled according to the position.
   For cash-flow-based products, the process can determine the cash flows by calling cash flow generation directly (determination category *Cash Flow Generation*) or by calling the Fair Value Server (determination categories *Based on Nominal Value*, *Based on Fair Value* and *Based on Amortized Cost*).
   The process reads the target value from accounting using book value determination.

> **Note**
>
> After this step you can manipulate the cash flows or the target value in a custom step sequence with the module editor, for example, by enriching custom fields.
>
> Step category: *Data Enrichment Using Module Editor*
>
> The contract data is transferred by the complex key figure Contract (`/BA1/X_KCKF40` or `/BA1/X_KCKF41`); the target value is transferred by key figure `/BA1/K64TGVAL`.

5. **Distribution of cash flows**
   Step category: *Data Enrichment*
   Step execution using class `/BA1/CL_AL_MATGRP_DE`
   The system assigns the relevant cash flows to the maturity groups. It distributes any difference between the sum of the cash flows and the target value to the maturity groups according to the distribution mode.
   The system then translates the results into functional currency. In Customizing, you can also make settings for translation into up to three local currencies (local currency, group currency, hard currency). To do so, you have to define these currencies for the legal entity and add the corresponding key figure from the following table to the result type:

Key Figure	Description	Assignment to Result Type
`64MTGRT`	Amount in maturity band in transaction currency	Provided by SAP
`64MTGRF`	Amount in maturity band in functional currency	Provided by SAP
`64MTGRG`	Amount in maturity band in group currency	Optional
`64MTGRH`	Amount in maturity band in hard currency	Optional
`64MTGRL`	Amount in maturity band in local currency	Optional

6. **Preaggregation (optional)**
   Step category: *Aggregation*
   This step is only contained in the standard step sequence `MATGRP_AGG`.
   To reduce the data volume before saving the results, you can aggregate data (for example, all cash flows of a maturity group). You can either aggregate by package (the granularity of the single transactions remains) or across several packages. If the data to be aggregated is in one package, aggregating the data by package is preferable because it is considerably faster.
7. **Saving the results**
   Step category: *Storage*
   Step execution using class `/BA1/CL_AL_Q1_STORAGE_ACC`

You can restart the process by using the corresponding CVPM standard function. For more information, see Capability to Restart Parallel Processes [page 1632].

Result

The process stores the key date results in the Results Data Layer (RDL). The results are always structured in the following results node types according to result category `HKMGR`:

- Overall contract
- Subcontract
- Maturity group

When maturity grouping has been completed, you can extract results data to a financial data mart (for example, SAP NetWeaver Business Warehouse).

4.3.13 Present Value Calculation (FS-BA-PM-GM-PVC)

This component provides methods with which you can calculate (present value) key figures for financial products.

4.3.13.1 Price Calculator

Use

The Price Calculator calculates present values and other key figures for financial transactions and positions. These key figures include the accrued interest, clean price, and durations and options Greeks.

Integration

The applications in Bank Analyzer call the Price Calculator either by means of the Fair Value Server, or using a secondary data source. When the Price Calculator is called, the system reads the Customizing settings in the price calculator environment. If it is called up from a secondary data source, the extended price calculator environment is used. You configure the price calculator in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator*.

- **Importing Net Present Values**
  The Price Calculator can also process net present values that have been imported from feeder systems or are stored in RDL. You store the imported data as generic market data or positions in the Source Data Layer (SDL). For more information, see Data Transfer [page 576].
- **External Pricing Models**
  SAP provides pricing models that you can use to measure the sample products that are shipped with the system. You can also create new pricing models. For more information see Pricing Models [page 549] and the documentation under Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *External Pricing Models*.
- **Market Data and Market Data Scenarios**

The Price Calculator is provided with all the from the Foundation component. If an application supplies the price calculator with market data scenarios, the price calculator calculates the NPV key figures on the basis of these scenarios.

Features

The Price Calculator uses multiple pricing models to price transactions and positions. Each of these pricing models is suitable for particular transaction forms only. Depending on the pricing model, the Price Calculator determines a suitable selection of the following key figures:

- Present value
- Accrued interest
- Clean price
- Modified duration
- Macaulay duration
- Fisher-Weil duration
- Key rate duration
- Convexity
- Delta, gamma, vega, theta and rho of an option
- Price value of one basis point
- Implied volatility
- Amortized cost
- Effective capital over time

The application informs the Price Calculator which key figures it is to calculate.

You can use market data scenarios with the Price Calculator. The Price Calculator then values a transaction using the parameters defined in the market data scenario. If you do not specify a scenario, the Price Calculator uses the market data valid on the evaluation date. Certain pricing models use a scenario, which you specify, and market data for the evaluation date. One such model is that used to price future contracts.

The Price Calculator can process various pricing models. Each of these models is suitable for particular transactions only. A transaction form can usually be measured by multiple pricing models. For example, loans can be measured using the *mark-to-market* pricing model (if price information is available) or by discounting the cash flow (theoretical price). The subtransactions of a transaction can also be measured by different pricing models. In Customizing for the Price Calculator, you define which pricing model the Price Calculator uses for which transaction form.

> **Note**
>
> For an overview of the pricing models predefined in the system, the key figures that they calculate, plus the transaction forms that can be used, see Pricing Models [page 549].

More Information

Valuation of Transactions with Subtransactions

In the Price Calculator, transactions are stored in a valuation structure, which is a hierarchy. A transaction or subtransaction with its own transaction form can be stored on each node. Depending on the assignment of the

pricing models and transaction forms, the Price Calculator prices the subtransactions using various pricing models.

The financial transaction *swap* is stored as a parent node that has the transaction form `SWAP`. There are two subnodes that have the transaction form `SWAPLEG`. The parent node is measured by the *structured products* pricing model, while the subnodes are measured by the *cash flow discounting* pricing model. You enter these settings in the valuation category in Customizing for the Price Calculator.

Consideration of Bid and Ask Rates

In the Customizing for the Price Calculator, you specify whether the Price Calculator is to consider bid or ask rates. At runtime, the Price Calculator decides for each individual subnode of a transaction and for each cash flow whether the bid or ask rate is to be used for the valuation. This decision is based on whether the purchase/sale indicator is set in the root node or subnode of the transaction. The Price Calculator receives this information from the SDL. The bid rate is used for a sale, and the ask rate used for a purchase. Note the following:

- Interest Rate Options
  For interest rate options, you use the reference interest rate to specify whether the bid or ask rate is to be used for the spot rate.
- Structured Products
  For structured products, the root node governs whether the bid or ask rate is used. For subtransactions, it is the relevant higher-level node that governs this. All subnodes of the transaction have a purchase/sale indicator. If the higher-level node is set to *purchase*, the subnode setting remains the same. However, if the higher-level node is set to *sale*, the subnode setting changes: from *purchase* to *sale,* and vice versa.

  Example

  In a swap, for example, the fixed and variable leg each have a purchase and a sale indicator. Usually, one leg is defined as a sale, and the other leg as a purchase. If the whole swap is defined as a purchase, the Price Calculator interprets the subtransactions as specified by the purchase/sale indicator. If the whole swap is defined as a sale, then the Price Calculator interprets the subtransactions in the opposite way to the indicator's setting. The side that was originally defined as a purchase is now interpreted by the Price Calculator as a sale; the side that was originally defined as a sale is now interpreted as a purchase.

4.3.13.1.1 Pricing Models

Definition

Methods for calculating the present value key figures.

Use

SAP delivers the following pricing models that are used to calculate the present value key figures given below for the various transaction forms.

Pricing Model	Key Figures	Transaction Forms
Mark-to-Market [page 551]	Present value, clean price, accrued interest, spot price adjusted for dividends	Listed transactions, such as stocks, bonds, and index, stock, or bond futures
Cash flow model: • Cash Flow Discounting [page 553] • Cash Flow Compounding [page 556]	• Cash flow discounting: Present value, clean price, accrued interest, amortized cost • Cash flow compounding: Amortized cost	Cash-flow-based transactions, such as loans, bonds, fixed-term deposits, forward exchange transactions
Forward Transactions [page 557]	Present value	Stock, bond, and currency forwards, forward rate agreements (FRAs)
Futures Transactions [page 559]	Present value	Stock, bond, index, and interest rate futures
Structured Products [page 561]	Present value	Caps, floors, collars, forward rate agreements, repurchase agreements, swaps
Standard Model for Options [page 564]	Present value, delta	European and American standard options, digital options, barrier options
Data Transfer [page 576]	Key figures that were calculated externally	Any
External Price Calculator for Interest Rate Options [page 563]	Present value key figures for interest rate options	American and Bermudan options on interest rate swaps, American and Bermudan options on bonds

> **Note**
>
> Some of the pricing models call other pricing models during the pricing process. For example, this is the case for the subtransactions of structured products. The pricing model for options also calls up the *cash flow discounting* pricing model to obtain the modified duration that it uses to translate the interest rate and price volatilities.

Integration

- External pricing models
  Besides the pricing models predefined in the system, you can use pricing models that you create yourself. These models enable you to use different methods to measure templates (products) for the transaction forms provided in the system and to measure templates (products) for new transaction forms that you created and that you cannot price using the pricing models predefined in the system. You use BAdIs to connect external pricing models to the price calculator. Within your pricing models, you can an RFC destination to call external price calculators. You create new pricing models in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *External Pricing Models*.

- Configuring pricing models
  You can define how different transactions are priced by a pricing model. You do this by using the configuration values provided in the pricing models. For example, the pricing model for standard options is used for stock options and currency options, but is configured differently in each case.
- Priority lists for transaction forms
  In Customizing for the price calculator, you define which pricing models are to be used for which transaction forms. You can specify multiple pricing models for a transaction form, which you have to prioritize. The price calculator applies the pricing model with the highest priority to the transaction form. If it cannot measure the transaction using this model, the price calculator then uses the pricing model that has the next highest priority.
  The price calculator may be unable to measure a transaction if, for instance, it tries to measure a listed bond at the market price using the *mark-to-market* pricing model but there is no suitable market data available in the system. The pricing model with the next highest priority might be *cash flow discounting*, which the price calculator uses to derive the theoretical price of the bond from its transaction data.

4.3.13.1.1.1 Mark-to-Market

Use

The price calculator uses the *mark-to-market* pricing model to calculate the market value of listed financial transactions and positions on the horizon date. For long positions, the price calculator determines the realizable price. For short positions, it determines the redemption value. In addition to the present value (dirty price), the price calculator determines the following key figures: clean price, accrued interest, and spot price adjusted for dividends.

The pricing model contains the following configuration values:

- **Standard Style**
  Pricing of transactions such as spot stock transactions and stock or bond positions
- **Future Style**
  Measurement of index, stock, or bond futures
- **Similar Instrument**
  Measurement of financial instruments for which no market data is available. To measure these transactions, the system uses financial instruments that are flagged as similar transactions by means of the setting for the contract relationship.

Integration

Any future cash flows contained in listed transactions are transferred by the pricing model to the Source Data Layer (SDL), where they are then discounted.

Prerequisites

The pricing model needs market data, which is provided by the Source Data Layer (SDL). It is unable to price transactions if:

- The SDL does not contain either the clean price or the dirty price
- The quotation type in the position does not match the quotation type in the market data

Features

The calculation formula varies depending on the configuration:

- **Standard Style**
  Depending on its quotation type, the price calculator uses the following formulas to price transactions and positions. The purchase/sale indicator for short positions (sale) is 1 and for long positions (purchase) +1.
  - **Direct or Indirect Quotation (stocks, for example)**

$$PV = \pm 1 \cdot n \cdot p'_{stock}$$

    where *PV* = the present value, *p' stock* = the price of a stock (market value), and *n* = the number of these stocks in the position.
  - **Quotation in Points (index futures, for example)**

$$PV = \pm 1 \cdot index\ value \cdot n \cdot p'_{pt}$$

    where *PV* = the present value, *p' pt* = the value of a point, *index value* = the price of the index (market value), and *n* = the number of futures; for pricing indexes *n=1*.
  - **Percentage Quotation (bonds, for example)**

$$PV = \pm 1 \cdot n \cdot \frac{NV}{100} \cdot p'_{percent}$$

    where *PV* = the present value, *p' percent* = the value of a percentage (market value), *NV* = the nominal value, and *n* = the number of bonds.
  - **Inverse Percentage Quotation (interest rate futures, for example)**

$$PV = \pm 1 \cdot n \cdot w \cdot \frac{NV}{100} \cdot p'_{percent}$$

where *PV* = the present value, *p' percent* = the percentage value (market value), *NV* = the nominal value, *n* = the number of bonds, and *w* = a weight, which is calculated as days/365 or months/12, for example, depending on the unit of measurement.

> **Note**
>
> The reference interest rate is required for the calculation of weight *w*. The system uses the reference interest rate stored for the underlying.

- **Future Style**
  The pricing of transactions depends on whether the maturity date of the future is before or after the horizon and whether the price calculator is provided with a scenario.
  If the horizon is after the maturity date, the present value of the future is zero. If the horizon is before the maturity date of the future, the price calculator calculates the present value (depending on whether a scenario has been provided) as follows:
  - **Without a Scenario**
    The present value of a future up to its maturity date is zero, since the margin account is used for daily marking to market. The margin account is held at the exchange and used to net the profits and losses that result from the everyday price fluctuations of the future. The collateral required for a futures contract is also posted to the margin account. Margin accounts are not mapped in the system.
  - **With a Scenario**
    If a scenario is specified, the price calculator calculates the present value of the future as the difference between the present value based on the specified scenario and the present value without a scenario, in other words, with the market data read from the SDL that is valid for the horizon.
- **Similar Instrument**
  You use this configuration value in the following instances:
  - Pricing for a similarly structured contract
    The system prices financial instruments for which there is no market data by measuring a financial instrument that is flagged as being similar by a setting in the contract relationship.
  - Measurement of subcontracts that do not have their own IDs
    It can be the case for structured contracts that the system has to measure subcontracts for which no IDs are stored. The IDs of warrant bonds, for example, are stored on the parent nodes only (template *Warrant Bond*), and not on the child nodes for the associated bonds. If you use the *Residual Method* pricing model in the Fair Value Server, it can be the case that the system has to price the bond on the child node separately. So that the system can measure the bond, the SDL must contain a contract relationship for a similarly structured bond for the relevant subcontract. The system prices the instrument by using the market data for the financial instrument that is flagged as a similar instrument.

4.3.13.1.1.2 Cash Flow Discounting

Use

The price calculator uses the *cash flow discounting* pricing model to calculate the present values of cash-flow-based transactions, such as loans, time deposits, and currency forwards. Cash flow discounting involves discounting future cash flows to determine their value on the horizon. The present value of a transaction is then the sum of the discounted cash flows.

Prerequisites

To be able to use the *cash flow discounting* pricing model, the system requires yield curves. Discount factors are calculated using yield curves.

Features

The pricing model receives the transaction and its cash flows, the market data for the evaluation date, and a scenario (if specified). The pricing model calculates the following key figures: *Present Value*, *Clean Price*, *Accrued Interest*, *Macaulay Duration*, *Modified Duration*, and *Amortized Cost*. It also returns individual cash flows in the cash flow or evaluation currency. The effective interest rate is also calculated for continuous compounding, though it is only for internal use.

> **Note**
>
> In Customizing for the price calculator, you specify whether the price calculator also considers cash flows with a value date that matches the horizon when it calculates NPV key figures. You use the *Cash Flow on Horizon* indicator to make these settings for each valuation rule. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *Edit Price Calculator Environments*.

Present Value

The pricing model calculates the present value of the transaction as follows:

1. By discounting the individual cash flows to the horizon
   The system multiplies the individual cash flows by a discount factor. The discount factor d_t of a cash flow CF_t at point in time t reduces the cash flow to the value that reaches the value CF_t when it is reinvested on the horizon at point in time t. The discount factors are based on continuous compounding.
2. By converting the cash flows to the evaluation currency
   The system uses the spot rate on the horizon to convert cash flows in a foreign currency to the evaluation currency.
3. By summing all the discounted cash flows

The system calculates the present value of a cash-flow-based transaction as follows:

$$NPV = \sum_{i=1}^{n} e^{-y \cdot (t_i - t_{horizon})} \cdot CF_{t_i}$$

Where NPV is the present value of the transaction on the horizon, CF_{ti} is the cash flows after the horizon, y is the yield using continuous compounding t_i is the point in time of cash flow CF_{ti}, and $t_{horizon}$ is the horizon.

Accrued Interest

The pricing model calculates accrued interest for cash flows for which the interest accrued up to the horizon has not yet been considered. This applies only to cash flows for which the horizon falls in the interest period of the cash flow but before the value date.

Clean Price

The pricing model calculates the clean price of a transaction as the present value of the transaction less the accrued interest.

Amortized Cost

The pricing model calculates the amortized cost as the sum of the cash flows discounted using the effective interest rate less the accrued interest. The effective interest rate specifies the yield achieved or costs involved in a transaction. The pricing model calculates the effective interest rate as the interest rate with which compounding and summing all the cash flows before the horizon produces the same value as summing the discounted cash flows that fall after the horizon. This calculation is usually based on the day count convention `act/365` and continuous compounding.

If the *Day Count Convention for Financial Transaction* checkbox is selected in Customizing, the system uses the day count convention that is defined for the financial transaction to be valuated. You make these settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings*. This setting only has an effect if you have set the *Value Date for Deferrals* indicator to "True" in the Customizing activity *Make Settings for Value Date for Deferrals*.

Macaulay Duration

The Macaulay duration reflects the elasticity of a present value to changes in the market interest rate and can, therefore, be used as a key figure for interest rate risk. For cash-flow-based transactions, the Macaulay duration is calculated as a quotient of the cash flows weighted by the payment dates and discounted to the horizon and of the present value of the transaction on the horizon:

$$Macaulay's\ Duration = \frac{\sum (t_i - t_{horizon}) \cdot e^{-y \cdot (t_i - t_{horizon})} \cdot CF_{t_i}}{\sum e^{-y \cdot (t_i - t_{horizon})} \cdot CF_{t_i}}$$

Where CF_{ti} is the cash flows after the horizon, y is the yield with continuous compounding and day count convention act/365, t_i is the point in time of cash flow CF_{ti}, and $t_{horizon}$ is the horizon. Time is measured in years.

Modified Duration

The modified duration can be calculated from the Macaulay duration. The modified duration reflects the change in the present value of a cash-flow-based transaction in response to a change in the market interest rate level.

$$Modified\ Duration = e^{-y} \cdot Macaulay's\ Duration$$

Where y is the yield with continuous compounding and day count convention act/365.

> **Note**
>
> The price calculator uses the modified duration to convert interest rate volatilities to price volatilities. It uses the following formula:
>
> $$price\,volatility = Modified\,Duration \cdot yield \cdot yield\,volatility$$
>
> The price volatility is required to price options. If the pricing model for options is provided with an interest rate volatility but no price volatility, it calls up the *cash flow discounting* pricing model to obtain the duration.

4.3.13.1.1.3 Cash Flow Compounding

Use

The price calculator uses the *cash flow compounding* pricing model in special situations to calculate the *Amortized Cost* key figure for cash-flow-based transactions, such as loans, time deposits, and currency forwards.

This model works in the same way as the *cash flow discounting* pricing model except that payments are compounded instead of discounted. This means that this model is only used if there is a valid effective interest rate. It aims to determine the amortized cost without using flows after the horizon. This is needed in connection with backvalued valuations.

However, you can force the price calculator to use the *cash flow discounting* pricing model by making the corresponding settings in Customizing.

Prerequisites

To be able to use the *cash flow compounding* pricing model, the system requires yield curves. Compounding factors are calculated using yield curves.

There must also be a valid effective interest rate.

Features

The pricing model receives the transaction and all its cash flows, the market data for the evaluation date, and a scenario (if specified). The pricing model calculates the *Amortized Cost* key figure. It also returns individual cash flows in the cash flow or evaluation currency.

> **Note**
>
> In Customizing for the price calculator, specify whether you want the *cash flow discounting* pricing model to be used as well when the system is compounding the cash flows. For example, this makes sense for impairments if you want to access future flows.
>
> You use the *Force CF Discounting* indicator to make these settings for each valuation rule. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *Edit Price Calculator Environments*.

Amortized Cost

The pricing model calculates the amortized cost as the sum of the cash flows compounded using the effective interest rate less the accrued interest. In contrast to the *cash flow discounting* model, however, the flows after the horizon are not discounted. Instead, the target value (the result of the last amortized cost calculation) and the flows between the target value and the horizon are compounded to the horizon. This calculation is usually based on the day count convention `act/365` and continuous compounding.

If the *Day Count Convention for Financial Transaction* checkbox is selected in Customizing, the system uses the day count convention that is defined for the financial transaction to be valuated. You make these settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings*. This setting only has an effect if you have set the *Value Date for Deferrals* indicator to "True" in the Customizing activity *Make Settings for Value Date for Deferrals*.

The results are identical to those of a calculation using cash flow discounting. Only the approach is different.

4.3.13.1.1.4 Forward Transactions

Use

The price calculator uses the *forward transactions* pricing model to the calculate present values of forward transactions involving stocks or bonds and the present values of forward rate agreements, for example.

In this case, the term "forward transaction" does not refer to futures transactions; it denotes forward contracts in the strict sense. The date on which the purchase or sale takes place and the price for the underlying transaction for the purchase or sale are defined at the outset. In contrast to futures, forwards usually involve physical delivery of the asset.

Prerequisites

The pricing model requires market data, in particular a yield curve or discount factors that are stored in the Foundation component.

Features

The present value of a forward transaction depends on whether the horizon is before or after the maturity date of the transaction. If the horizon is after the maturity date, the transaction has already been settled. The system assigns the forward transaction a present value of zero.

If the underlying of a forward transaction is a security such as a stock, the purchased securities are included in the position, or are deducted from the position in the event of a sale. Since the position is priced independently of the forward transaction, the price calculator assigns the forward transaction a present value of zero if the horizon is after the maturity date.

If the horizon is before the maturity date, the system calculates the present value of the forward transaction as the difference between the underlying price defined in the contract and the relevant forward price of the underlying.

Because the agreed interest period has to be considered in forward rate agreements, the present value is calculated in different ways:

- **Forward rate agreements (FRAs)**
  The pricing model calculates the present value as follows:
  - It calculates the forward rate of the agreed reference interest rate.
  - It calculates the difference between this rate and the agreed interest rate.
  - It uses this value to calculate the interest on the underlying notional value until the end of the agreed period.
  - It discounts the result to the start of the interest period using spot rate of the reference interest rate.
  - It discounts the result to the horizon using the interest rate supplied with the market data set.
  - It sets the purchase/sale indicator.
    The pricing model therefore uses the following formula:

$$PV = \pm 1 \cdot d_{(t,T)} \cdot d_{(T,T')} \cdot NV \cdot (R(T,T'|t) - R_{contract}) \cdot (T'-T)$$

where *PV* = the present value, *NV* = the notional amount of the underlying, *t* = the horizon, *T* = the start and *T'* = the end of the interest period, *R(T,T'|t)* = the forward rate of the agreed reference rate for the interest period from T to T' at point in time t, $R_{contrac\ t}$ = the interest rate agreed upon in the contract, $d_{(T,T')}$ = the discount factor for discounting from the end of the interest period to its start, and $d_{(t,T)}$ = the discount factor for discounting to the horizon from the end of the interest period to its start.

- **Other types of forward contract**
  The pricing model calculates the present value as follows:
  - It calculates the spot price of the underlying on the maturity date.
    For this valuation, the *forward transactions* pricing model calls up the pricing model for the transaction form of the underlying. For example, it calls up the *mark-to-market* pricing model for stocks, and the *cash flow discounting* pricing model for bonds.
  - It calculates the difference between this price and the price defined in the contract.
  - It discounts the result to the horizon.
    The system uses the interest rate supplied with the market data set for this purpose.
  - It sets the purchase/sale indicator.
    The pricing model therefore uses the following formula:

$$PV = \pm 1 \cdot d_t \cdot (p_{forward} - p_{contract})$$

where *PV* = the present value, $p_{forward}$ = the spot price of the underlying on the maturity date, $p_{contrac\ t}$ = the underlying price agreed upon in the contract, d_t = the discount factor for discounting from the maturity date to the horizon *t* . The system already contains these values in the evaluation currency.

Example

The formula used to calculate the present value varies depending on the underlying of the forward contract:

- **Forward contracts on stocks**

  $$PV = \pm 1 \cdot d_{(t_{contract};t_{horizon})} \cdot n \cdot (p_{forward} - p_{contract})$$

  where *PV* = the present value, $p_{forward}$ = the spot price of the stock on the maturity date, $p_{contrac\ t}$ = the stock price defined in the contract, n = the number of stocks, and $d_{(t,t')}$ = the discount factor for discounting from point in time *t* to point in time *t'* ; $t_{contract}$ = the maturity date, $t_{horizon}$ = the horizon. The plus/minus sign is used as the purchase/sale indicator.
- **Bond forwards**

  $$PV = \pm 1 \cdot d_{(t_{contract};t_{horizon})} \cdot (p_{forward} - p_{contract})$$

  where:

  $$p_{forward} = \left(\sum_{t_i \geq t_{contract}} d_{(t_i,\ t_{contract})} \cdot CF_{t_i} \right) - AI$$

  PV = the present value, CF_{ti} = the cash flows of the bond that have already been converted to the evaluation currency (including payment and repayment), *AI* = the relevant accrued interest, $p_{contrac\ t}$ = the bond price agreed upon in the contract, $d_{(t,t')}$ = the discount factor for discounting from point in time *t* to point in time *t'* ; $t_{contract}$,= the maturity date, $t_{horizon}$ = the horizon.

4.3.13.1.1.5 Futures

Use

The price calculator uses the *futures transactions* pricing model to calculate the change in the present value of futures in the market data scenarios specified by the application when the price calculator is called up. The system can also calculate the present value for futures in portfolio fair value hedging relationships without using scenarios.

Typical futures transactions are futures contracts involving stocks, bonds, indexes, or interest rates. Unlike forward contracts, in futures payments are made during the life of the futures contract. The forward transaction that is part of the futures contract is not usually exercised. This is for the following reasons:

A margin account is used.

Futures contracts are traded on the stock exchange and involve payments to a margin account. This type of account is maintained on the exchange to clear the profits and losses that result from daily price fluctuations of the future. The collateral required for a futures contract is also posted to the margin account. Margin accounts are not mapped in the system.

The contract is not exercised.

Futures are usually closed out before their maturity date.

Because a margin account is used, the present value of a future is always zero. In view of this, the main purpose of the *futures transactions* pricing model is to estimate how the margin account will develop in a given market data scenario. The pricing model therefore supplies a "real" present value (not equal to zero) only if the price calculator is provided with a market data scenario.

The pricing model also calculates the theoretical present value of a future. This value is displayed in the detail log.

Prerequisites

The pricing model requires market data and a market data scenario. Otherwise, the present value is always zero.

Futures are listed and should therefore be priced using the *mark-to-market* pricing model (configuration value: FUTURE). The *futures transactions* pricing model provides an alternative calculation function that can be used if a valuation using the *mark-to-market* pricing model fails.

In Customizing for the price calculator, you can specify a list of pricing models for a transaction form and assign priorities to these models. The price calculator then starts by using the pricing model with the highest priority. If a model cannot be applied, the price calculator uses the pricing model with the next highest priority. You make these settings in Customizing for *Bank Analyzer under Processes and Methods General Calculation and Valuation Methods Present Value Calculation → Price Calculator → Edit Valuation Categories*

You can specify two pricing models for futures contracts: The *mark-to-market* pricing model with priority 1 and the *futures transactions* pricing model with priority 2.

Features

The NPV key figure returned by the pricing model depends on whether the horizon lies before or after the maturity date and whether the price calculator is supplied with a market data scenario by the calling application. If the horizon is after the maturity date, the present value of the transaction is zero.

If the horizon is before the maturity date and no market data scenario was specified, the present value of a future is zero, since the present value is cleared each day on the margin account. If a market data scenario is used, the price calculator calculates the difference between the future value of the underlying transaction (based on the specified market data scenario) and the value of the underlying transaction on the horizon (based on the market data valid on the evaluation date). The cash flows are not discounted to the horizon.

$$\Delta PV = futurevalue_{szenario} - futurevalue_{horizon}$$

To calculate the future value of an underlying transaction, the pricing model calls up the pricing model assigned in Customizing for the relevant transaction form. For example, for loans, it calls up the *cash flow discounting* pricing model.

When the price calculator calculates hedge key figures, it calculates the present value of futures, even though you have not specified a scenario. In such cases, the system returns the full fair value that it calculates as the theoretical price of the underlying of the future. For more information, see Calculation of the Fair Value [page 633] and Calculation of Amortized Cost [page 636] .

4.3.13.1.1.6 Structured Products

Use

The price calculator uses the *structured products* pricing model to calculate the present value of transactions such as caps, floors, collars, swaps, or forwards involving loans, fixed-term deposits, and swaps.

The price calculator treats these transactions as structured products, since they are stored in the processing structure as transactions that have the following properties: The sum of the present values of the subtransactions equals the present value of the overall transaction.

> **Note**
>
> Transactions that are usually classed as structured products are not necessarily regarded as structured products by the price calculator. Examples of these transactions include loans with early repayment and unscheduled repayment options. Likewise, the structured products in the price calculator are not necessarily regarded as structured products normally. For example, the price calculator treats swaps as structured products although they are not normally regarded as such.

Prerequisites

The pricing model requires market data, in particular yield curves and exchange rates. Market data is stored in the Foundation component.

Features

The pricing model calls up the price calculator for each subtransaction of a structured product. It then adds up the present values calculated for the subtransactions. Any commission or charges incurred are measured using the cash flow discounting pricing model and added to the subtotal. This gives the present value of the overall transaction.

If subtransactions cannot be measured, the pricing model either (depending on the entire transaction) returns an error or a present value based on the existing present value of the subtransactions and ignores the subtransactions that were not priced. You can define how the pricing model handles a transaction form by entering the following settings:

- Value entire transaction only
  Both legs of a swap have to be included in the present value calculation, for example (see the example below).

- Transaction is the sum of the individual parts
  When the present value of caps is calculated, some of the caplets might be irrelevant (see the example below). The pricing model for options treats caplets with exercise dates before the horizon as expired options, so it does not measure them. The present value of the cap equals the sum of the present values of all the remaining caplets.

> **Note**
>
> So that the transaction can be measured, a pricing model must be assigned to each subtransaction of a structured product. This assignment is stored in a system table for the transaction forms shipped by SAP. For customer-defined transaction forms or newly defined pricing models, you have to make this assignment in Customizing for the price calculator You do this in Customizing activity *Edit Valuation Category*. You assign transaction forms and configuration values in Customizing activity *Assign Configuration Values*.
>
> For more information about the treatment of bid and ask rates in structured products, see Price Calculator [page 547].

Example

Typical products that are treated as structured products in the price calculator are caps and swaps. In the valuation structure, these transactions have a root node, which does not contain transactions, and subnodes that contain subtransactions.

- Caps
  A cap is an agreed maximum interest rate, and can be regarded as a series of "caplets". While caps span multiple interest periods, a caplet describes a cap that relates to precisely one interest period. Since the sum of the present values of the caplets equals the present value of the relevant cap, caps are stored as structured products in the valuation structure. The caplets are stored on subnodes below the root node. All the caplets are located on the same hierarchy level.
  Caplets can be priced as European call options on a reference interest rate. When pricing caps, the *structured products* pricing model calls up the pricing model for options for each caplet. If the end of the fixed-rate period and the exercise date of a caplet are before the horizon, the caplet is not priced and the option has expired. The present values calculated for the caplets are then added to produce the present value of the cap.
- Swaps
  In the valuation structure, swaps comprise a root node, which contains both legs of the swap as subtransactions.
  The price calculator first measures the two legs of the swap using the *cash flow discounting* pricing model. The *structured products* pricing model then adds up the present values of the two legs of the swap. Finally, the pricing model considers cash flows that are incurred in the form of charges, for example. These cash flows are measured using the *cash flow discounting* pricing model and added to the total of the present values of the legs of the swap. Examples of this type of cash flow include settlement payments that contain the change in the value of the swap after the conclusion of and before the start of the transaction (up-front payments).

4.3.13.1.1.7 External Price Calculator for Interest Rate Options

Use

You can use the pricing model *External Pricing of Interest Rate Options* to connect (by means of Web Services) to a third-party provider of prices for interest rate options.

This enables you to use a third-party provider to calculate the NPV key figures for interest rate options that the *Standard Options* pricing model cannot price.

The *External Pricing of Interest Rate Options* pricing model contains the following configuration values:

- Swaption
  Pricing of American and Bermudan options on interest rate swaps
- Bond option
  Pricing of American and Bermudan options on bonds. Cross-currency interest rate swaps and multicurrency bonds cannot be priced.

Integration

The pricing model receives market data (yield curves, volatilities) from the Source Data Layer (SDL), and transfers this data and the transaction data to the external provider by means of Web Services.

Prerequisites

Before you can use the *External Pricing of Interest Rate Options* pricing model (or before you can use an external provider to price interest rate options) the following requirements must be met:

- In Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *Settings for External Price Calculator* > *Create Calibration Instruments* you have created at least one calibration instrument.
- The SDL must contain the volatilities for the calibration instruments you created.
- In Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *Settings for External Price Calculator* > *Create External Pricing Models* you have created at least one pricing model.
- In Customizing for*Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *Edit Price Calculator Environments* you have defined that the system is to use the special pricing model *External Pricing of Interest Rate Options*. You use valuation rules [page 490] to define this. You have also specified the pricing model.

Features

You can use the following models to use an external provider to price interest rate options:

- Hull-White (1, 2 or 3 factors)
  The Hull-White 1-factor model is based on the assumption that the short-term rate is normally distributed. The multifactor models assume that the short-term interest rate is the sum of multiple correlated normal distributions.
- Black-Karasinski
  The Black-Karasinski model is based on a lognormal distribution of the short-term interest rate.
- Spot Skew
  The spot-skew model uses a skew parameter to model the short rate. The short rate can be normally distributed (Hull-White), lognormal distributed (Black-Karasinski), or anything in between these.

> **Note**
>
> You define which of these models the external provider is to use in the pricing model type.

You can also use the external price calculator for interest rate options to calculate the Fisher-Weil duration and key rate duration for options. The prices area calculated as approximate values using the difference quotients of two present values from a shifted yield curve.

4.3.13.1.1.8 Standard Options

Use

The Price Calculator uses the pricing model for standard options to calculate the present value, and the delta, gamma, vega, theta, and rho for the following options:

- **European options**
  (options that can be exercised on the expiration date only)
  Possible underlyings:
  - Stocks with or without dividend payments
  - Bonds
  - Indices
  - Foreign exchange
  - Interest rates
  - Swaps
  - Forward contracts
  - Futures
- **American options**
  (options that can be exercised at any point in time until the expiration date)
  Possible underlyings:
  - Stocks without dividend payments
  - Indices
  - Foreign exchange

- Forward contracts
- Futures

The following American-style options currently cannot be priced:

- Options on stocks with dividend payments
- Stock or bond options
- Options on swaps or interest rates

Integration

The pricing model uses the market data stored in the Foundation component. To calculate spot prices, the pricing model for standard options calls up the *mark-to-market* pricing model. To calculate the modified duration required to determine price volatilities, it calls up the *cash flow discounting* pricing model.

Prerequisites

To calculate the NPV of options, the pricing model requires the following information:

- Spot price of the underlying
- Exercise price of the option
- Risk-free interest rate
- Volatility of the underlying
- Term in days from the horizon until the expiration date of the option

Furthermore, the pricing process depends upon the following date values:

- Evaluation date of the option
- Horizon of the option
- Expiration date of the option

For options on forwards and futures or related transactions, the expiration date of these transactions is also considered. For caplets and floorlets, the end of the fixed-rate period is also considered.

To price American options, the price calculator must also be informed of the number of steps to be included in the binomial tree used for the calculation.

Features

The pricing model calculates the *present value* and the option Greeks.

> **Note**
>
> The transaction data for options states whether they are to be priced as futures. If you set the relevant indicator and the pricing model is not provided with a scenario, the NPV of the option is always zero. If the calling application provides the price calculator with a scenario, the pricing model for options calculates the difference between the value of the option with a scenario and the value of the option without a scenario.

Present value

The price calculator uses the Black-Scholes formula to price European options [page 566]. The formula is applied differently depending on the underlying. Configuration values define how exactly the options are priced.

The price calculator uses the binomial model to price American options [page 574]. The calculation is the same for all underlyings.

> **Note**
>
> The pricing model recognizes from the transaction data whether an option is a European or American style option, and whether it has to use the Black-Scholes formula or the binomial model for the valuation.

Delta

The delta of an option is defined as a derivation of the price of an option on the basis of the spot price of the underlying. For European options, the pricing model calculates the delta directly from the derivation of the Black-Scholes formula. For American options, the pricing model uses a difference quotient for this purpose and varies the spot price of the underlying. You set the shift of the underlying in the *price calculator environment* in Customizing for the price calculator; the standard setting is 0.01%.

4.3.13.1.1.8.1 European Options

Use

Features

The pricing model for standard options calculates the net present value (NPV) of European-style options using the Black-Scholes formula.

Since the exact method used to calculate the individual underlyings varies, the pricing model is controlled by means of configuration values. The following configuration values and pricing functions are provided for the option classes given below:

Option Group	Option	Option Class (Transaction Data)	Configuration Value (for Option Class: Standard Option)	Underlying
Stock, Bond, Index, and Currency Options [page 567]	Stock, bond, index, and currency options	Standard option	Standard option	Stocks, bonds, indexes, and exchange rates

Stock, Bond, Index, and Currency Options [page 567]	Swaption	Standard option	Swaption	Bond or swap rate
Unscheduled repayment or call option for loan [page 567]	Unscheduled repayment or call option for loan	Call option, unscheduled repayment option		Stocks, bonds, indexes
Interest Rate Options [page 569]	Caplet/floorlet	Standard option	Caplet/floorlet	Interest rate
Interest Rate Options [page 569]	Interest rate	Standard option	Option on a forward rate agreement (FRA) - interest rate guarantee	Interest rate
Options on Futures and Forwards [page 571]	Options on futures	Standard option	Options on futures	Future
Options on Futures and Forwards [page 571]	Option on forward	Standard option	Standard option	Forward
Barrier Options [page 573]	Barrier option	Knock-in barrier option, knock-out barrier option, double knock-in option, double knock-out option		Stocks, bonds, indexes
Digital Options [page 574]	Digital option	Digital hit-at-end option, digital one-touch option		Stocks, bonds, indexes

> **Note**
>
> You define whether an option is a digital option, a barrier option, a call option on a loan, or an unscheduled repayment option in the transaction data for the option class.

4.3.13.1.1.8.1.1 General Standard Options

Features

The pricing model uses a slightly modified version of the Black-Scholes formula to price options on stocks, bonds, indexes, and repayment and call options on loans. The pricing of options on stocks and bonds is used as the basis for the pricing of other options. The calculation formula is considered only when the horizon lies between the purchase or sale of the option and its expiration date. In all other cases, the value of the option is zero.

- **Stock or bond options**

The $c_s(t,T,X)$ value of a call option and the $p_s(t,T,X)$ value of a put option are calculated as follows:

$$c_s(t,T,X) = \tilde{S}(t,T) \cdot N(x) - e^{-r(T-t)} \cdot X \cdot N\left(x - \sigma\sqrt{T-t}\right)$$

$$p_S(t,T,X) = -\tilde{S}(t,T) \cdot N(-x) + e^{-r \cdot (T-t)} \cdot X \cdot N\left(-x + \sigma\sqrt{T-t}\right)$$

with

$$x = \frac{\ln\left(\frac{\tilde{S}(t,T)}{e^{-r \cdot (T-t)} \cdot X}\right)}{\sigma\sqrt{T-t}} + \frac{1}{2}\sigma\sqrt{T-t}$$

where *t* = the horizon, *T* = the expiration date of the option, *X* = the strike of the option,*s* = the volatility of the underlying , *Š(t,T)* = the spot price of the underlying at point in time *t* adjusted by the dividends accrued in the period from *t* to *T*, and *r* = the risk-free interest rate; *N(x)* = the standard normal distribution.

> **Note**
>
> This pricing model discounts strike X using continuous compounding of the risk-free interest rate and the day count convention act/365.

- **Index options**
  Index options are treated in the same way as stock options. The index value is used as the spot price.
- **Currency options**
  Currency options are treated in the same way as stock options. The exchange rate is used as the spot price. The risk-free interest rate is calculated as the difference between the interest rate of the local currency and the interest rate of the foreign currency.
- **Call options and unscheduled repayment options for loans**
  Call options and unscheduled repayment options for loans are treated as call options (bond options) on loans. They are always short positions. For technical reasons, the pricing model prices only loans with precisely one call option and unscheduled repayment option correctly.

  > **Note**
  >
  > For loans with call options and unscheduled repayment options, the option cash flows can also be calculated by using due date scenarios, or on a delta-weighted basis. In Strategy Analyzer, cash flows are calculated in this way in the Cash Flow Engine. You make the settings for this in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Basic Settings for Methods* > *Edit Cash Flow Determination Method*.

- **Swaptions**
  A swaption is an option on an interest rate swap that involves a fixed and variable leg. The buyer of the swaption gains the right to enter into a swap on a specified future date. The pricing model can price swaptions as interest rate options or as bond options.
  If the pricing model prices swaptions as bond options, it calculates the present value of the swaption in the same way as the present value of an option on a fixed-rate bond. The option is exercised if the value of the

fixed-rate bond exceeds the value of the corresponding floating-rate bond. The bond here is described by the fixed leg of the swap. The system uses the interest rate volatility and Macaulay duration to calculate the price volatility of the swaption (meaning the price volatility of the fixed leg). The price calculator uses the *cash flow discounting* pricing model to calculate the Macaulay duration.
If the pricing model prices swaptions as interest rate options, it calculates the present value of the swaption in the same way as the present value of an option on a reference interest rate. The swap rate is used as the interest rate. The swap rate is calculated as the fair interest rate for a bond that corresponds to the fixed leg of the swap. The interest rate of the fixed leg of the swap is used as the strike. If the swaption, as an interest rate option, has a spread on the floating leg of the swap, the system includes this in the measurement of the swaption and adjusts the strike accordingly.

> **Note**
>
> In Customizing for the price calculator, you define whether the pricing model prices swaptions as interest rate options or bond options. You make this setting in a price calculator environment in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *Edit Price Calculator Environments*.

4.3.13.1.1.8.1.2 Interest Rate Options

Features

The pricing model uses a slightly modified version of the Black-Scholes formula to price interest rate options. The calculation formula is considered only in cases where the horizon lies between the purchase or sale of the option and its expiration date. In all other cases, the value of the option is zero.

- **Caplets and Floorlets**
  Caplets and floorlets are priced as options on the reference interest rate of the relevant cap or floor. Caplets and floorlets always involve precisely one interest period. The date on which the interest rate is fixed and the expiration date of the option usually coincide.

  > **Note**
  >
  > The system uses two date fields for the date on which the interest rate is fixed and the expiration date of the option, which means that you can also create caplets and floorlets with different dates for the date on which the interest rate is fixed and expiration of the option.

  The calculation formula for caplets is as follows:

$$c_R^{caplet}(t,T,C|t) = NV \cdot (T'-T) \cdot d(t,T') \cdot \left(R(T,T'|t) \cdot N(x) - C \cdot N\left(x - \sigma\sqrt{T-t}\right)\right)$$

$$p_R^{floorlet}(t,T,C|t) = NV \cdot (T'-T) \cdot d(t,T') \cdot \left(-R(T,T'|t) \cdot N(-x) + C \cdot N\left(-x + \sigma\sqrt{T-t}\right)\right)$$

Where the following applies:

$$x = \frac{\ln\left(\frac{R(T,T'|t)}{C}\right)}{\sigma\sqrt{T-t}} + \frac{1}{2}\sigma\sqrt{T-t}$$

Key
t = horizon date
T = expiration date of the option
T' = end of the fixed-rate period
C = maximum interest rate agreed upon in the cap
s = volatility of the underlying reference interest rate
R(T,T'|t) = forward rate of the interest rate in the period from *T* to *T'* at point in time t
$d_{(t,T')}$ = discount factor with the risk-free interest rate for the period from *t* to *T'*
NV(T'-T) = proportion of the nominal value for the period *T* to *T'*
N(x) = standard normal distribution

> **Note**
>
> The system considers only cases in which a payment is always made at the end of a fixed-rate period. For discounting with factor $d_{(t,T')}$, only the risk-free interest rate provided with the market data set is used. The reference interest rate of a caplet or floorlet is not used.

> **Note**
>
> Caplets and floorlets are abstract transactions into which caps or floors are broken down for calculation purposes. The caps and floors are broken down on the basis of the length of the period of the reference interest rate that is to be limited by the cap or floor. The individual caplets of a cap or floorlets of a floor each cover a period of this type and do not overlap. A cap based on the three-month EURIBOR is broken down into caplets that each have a period length of three months, for example. In the internal processing structure in Risk Basis, caps and floors are mapped as structured products with the subtransactions caplets and floorlets. To calculate the net present value of caps and floors, the structured products pricing model is used, which calls up the pricing model for standard options for each caplet or floorlet.

- **Options on Forward Rate Agreements (FRA) Interest Rate Guarantees**
  Interest rate guarantees are priced in the same way as caplets and floorlets:

$$c_R(t,T,C|t) = NV \cdot (T'-T) \cdot d(t,T') \cdot \left(R(T,T'|t) \cdot N(x) - C \cdot N\left(x - \sigma\sqrt{T-t}\right)\right)$$

$$p_R(t,T,C|t) = NV \cdot (T'-T) \cdot d(t,T') \cdot \left(-R(T,T'|t) \cdot N(-x) - C \cdot N\left(-x - \sigma\sqrt{T-t}\right)\right)$$

Where the following applies:

$$x = \frac{\ln\left(\frac{R(T,T'|t)}{C}\right)}{\sigma\sqrt{T-t}} + \frac{1}{2}\sigma\sqrt{T-t}$$

Key
t = horizon date
T = expiration date of the option
T' = end of the fixed-rate period
C = maximum interest rate agreed upon in the cap
s = volatility of the underlying reference interest rate
R(T,T'|t) = forward rate of the interest rate in the period from *T* to *T'* at point in time t
$d_{(t,T')}$ = discount factor with the risk-free interest rate for the period from *t* to *T'*
NV(T'-T) = proportion of the nominal value for the period *T* to *T'*
N(x) = standard normal distribution

4.3.13.1.1.8.1.3 Options on Forwards and Futures

The valuation model uses a slightly modified version of the Black-Scholes formula for options on forwards and futures, and for currency options. The method used to valuate stock or bond options is used as a basis for valuating the other options. The calculation formula is considered only when the horizon lies between the purchase or sale of the option and its expiration date. In all other cases, the value of the option is zero.

- **Options on Forwards**
  If a purchase or sale is made on the expiration date for options on forwards, the value of the forward transaction is paid. The estimated actual forward price/rate is, therefore, incorporated as the spot price/rate and the agreed forward price/rate as the strike in the Black-Scholes formula. Three dates have to be considered in the calculation: The horizon *t*, the exercise date of the forward *T'*, and the expiration date of the option T over the time to maturity of the option.

$$c_f(t,T,X) = d_{(t,T')} \cdot \left(S(t,T') \cdot N(x') - X \cdot N\left(x' - \sigma\sqrt{T-t}\right)\right)$$

$$p_f(t,T,X) = d_{(t,T')} \cdot \left(-S(t,T') \cdot N(-x') - X \cdot N\left(-x' + \sigma\sqrt{T-t}\right)\right)$$

with

$$x' = \frac{\ln\left(\frac{\tilde{S}(t,T)}{e^{-r\cdot(T-t)}\cdot X}\right)}{\sigma\sqrt{T-t}} + \frac{1}{2}\sigma\sqrt{T-t}$$

where *t* = the horizon, *T* = the expiration date of the option, T'= the exercise date of the forward, *X* = the strike of the option, *S(t,T')* = the agreed forward price/rate of the forward for the exercise date *T'* from horizon *t*, *σ* = the volatility of the underlying, and $d_{(t,T')}$ = the discount factor (discounting with the risk-free interest rate); *N(x)* = the standard normal distribution.

- **Options on Futures**
  Options on futures are exercised in the same way as options on forwards. If a purchase or sale is made on the expiration date, the value of the futures transaction is paid. The estimated actual forward price/rate is, therefore, incorporated as the spot price/rate and the agreed forward price/rate as the strike in the Black-Scholes formula. Since a margin account is maintained for futures, discounting is required only from the expiration date of the option to the horizon (unlike in options on forwards).

$$c_F(t,T,X) = d_{(t,T)}\cdot\left(S(t,T')\cdot N(x') - X\cdot N\left(x' - \sigma\sqrt{T-t}\right)\right)$$

$$p_F(t,T,X) = d_{(t,T)}\cdot\left(-S(t,T')\cdot N(-x') - X\cdot N\left(-x' + \sigma\sqrt{T-t}\right)\right)$$

with

$$x' = \frac{\ln\left(\frac{S(t,T')}{X}\right)}{\sigma\sqrt{T-t}} + \frac{1}{2}\sigma\sqrt{T-t}$$

where *t* = the horizon, *T* = the expiration date of the option, T'= the exercise date of the future, *X* = the strike of the option, *S(t,T')* = the agreed forward price/rate of the future for the exercise date *T'* from horizon *t*, *σ* = the volatility of the underlying, and $d_{(t,T')}$ = the discount factor (discounting with the risk-free interest rate); *N(x)* = the standard normal distribution.

4.3.13.1.1.8.1.4 Barrier Options

Use

Features

The price calculator prices only European-style barrier options. The option class in the transaction data enables a distinction to be made between the following types of barrier options:

- **Simple Barrier Options**
  The price calculator prices simple barrier options using the Rubinstein and Reiner model (1991) [Breaking down the barriers, Risk 4, p. 28-35 (September 1991)].
  The system distinguishes between the following simple barrier options:
  - **Knock-in**
    The option comes into being (in other words, "knocks in") if the spot price of the underlying reaches the barrier.
  - **Knock-out**
    The option expires worthless (in other words, "knocks out") if the spot price of the underlying reaches the barrier.

  Depending on whether the barrier option is a call or put option and (in the case of valid options) whether the spot price is above or below the barrier (up-and-in, up-and-out, down-and-in, down-and-out), the system distinguishes between eight simple barrier options.
  A rebate can also be incorporated. A rebate is a fixed amount that is paid if the option cannot be exercised as a result of the barrier being reached or not reached. With an up-and-out option, the rebate is paid if the spot price of the underlying reaches or exceeds the barrier. With a down-and-in option, the rebate is paid if the spot price of the underlying does not reach the barrier during the life of the option.
- **Double Barrier Options**
  The price calculator prices double barrier options using the Ikeda and Kunitomo model (1992) [Pricing options with curved boundaries, Mathematical Finance 2 (1992), 275-298].
  The system distinguishes between the following double barrier options:
  - **Knock-in**
    The option comes into being (in other words, "knocks in") if the spot price of the underlying reaches the lower or upper barrier.
  - **Knock-out**
    The option expires worthless (in other words, "knocks out") if the spot price of the underlying reaches the lower or upper barrier.

  Double barrier options do not incorporate rebates.

> **Note**
>
> Barrier options change their status in line with the behavior of their underlying. The status of an option is stored in the transaction data in the Source Data Layer (SDL). If the status of the option changes, the system adjusts the transaction data by means of a business transaction.
>
> If the spot price of the underlying for a knock-in option reaches the predetermined barrier, for example, the option becomes a standard option. The system changes the option class in the transaction data from *Knock-In Barrier Option* to *Standard Option*. Knock-out options expire worthless if the spot price of the underlying reaches the barrier.

The price calculator works on the assumption that all the options it receives for the valuation are valid.

> **Note**
>
> Note that the rates you enter for foreign currency barrier options have to be the quoted prices. Forward exchange transactions, on the other hand, are defined by absolute amounts. When you enter barrier prices, you also need to take into account the exchange rate categories, and the exchange rate factors stored for the related currencies.

4.3.13.1.1.8.1.5 Digital Options

Use

Features

The price calculator valuates all digital options (binary options) using the Rubinstein and Reiner model (1991) [Unscrambling the Binary Code, Risk 4, p.75-83 (October 1991)].

The option class in the transaction data enables a distinction to be made between the following types of barrier options:

- Hit-at-End Binary Options
  On the exercise date, a fixed amount (rebate) is paid if the spot price of the underlying is higher than the strike (call option or digital up) or lower than the strike (put option or digital down).

  > **Example**
  >
  > A digital up (call option) at a strike of EUR 1.04 to the US dollar is a "bet" that the US dollar will be worth more than EUR 1.04 when the option is exercised. The exchange rate before or after the exercise date is irrelevant.

- One-Touch Binary Options
  A fixed amount (rebate) is paid if the spot price of the underlying has reached the agreed barrier. This can take place at any time during the life of the option.

4.3.13.1.1.8.2 American Options

Features

The pricing model for standard options prices American options using the Cox-Ross-Rubinstein binomial model (1979) [*Option Pricing: A Simplified Approach*, Financial Economics, Vol. 7, p. 229-263].

To enable the model to be used, the pricing model breaks the time to maturity of the option down into time intervals of equal length. The pricing model uses 30 time intervals, in other words, it determines 30 possible exercise times for the option. The calculation is also based on the assumption that the value of the underlying only ever increases and decreases by the same factor between these time intervals.

The pricing model starts by constructing a (recombining) tree, which contains the possible values of the underlying at the exercise times. The root node contains the value of the underlying on the horizon. The end nodes contain the possible values of the underlying on the expiration date of the option.

The pricing model then calculates the value of the option at each exercise time by starting at the end nodes and working through the nodes step by step until it reaches the root node. The values are discounted using the risk-free interest rate. This produces the value of the option on the horizon for the root node.

The formula for calculating the present values of American options is similar to the Black-Scholes formula, although binomial distribution is used for American options rather than normal distribution. Because no standard formula is available here, as in the Black-Scholes formula, the price of American options is calculated using an algorithm that normally takes longer to run than the algorithm used to price European options.

Example

S(t) represents the value of the underlying (of a stock, for example) on the horizon. The probability that the value of the underlying will increase at a subsequent time is p, and the probability that it will decrease is 1-p. If three time intervals are considered in the calculation, the recombining binomial tree has the following form:

Point in Time t	t+t	t+2t	t+3t
			p 3S(t)
		p 2S(t)	
	p S(t)		p 2(1-p) S(t)
S(t)		p (1-p) S(t)	
	(1-p) S(t)		p (1-p) 2S(t)
		(1-p) 2S(t)	
			(1-p) 3S(t)

4.3.13.1.1.9 Data Transfer

Use

You can use this function to import present value key figures from source systems into the Fair Value Server. You have the following options:

- **Import of Key Figures Using Primary Data Sources**
  The system uses primary data sources to read present value key figures from the Source Data Layer (SDL). Present value key figures can be stored as generic market data or as positions.
- **Import of Key Figures Using Secondary Data Sources**
  The system reads present value key figures from a key figure that was already calculated before the price calculator was called. The key figure is assigned its value in a secondary data source within Bank Analyzer.
- **Import of Data into the Results Data Layer (RDL)**
  The system uses a results data area, result type, and the name of the key figure to read the present value key figures from the RDL.

Integration

This function is designed so that the price calculator reads the imported key figures, When the Fair Value Server calls the price calculator, the Fair Value Server tells the Price Calculator where it can find the key figures.

In order to be able to import present value key figures into the Fair Value Server, you need to have already entered the Customizing settings for the Fair Value Server and for the price calculator. If you want to do so by using secondary data sources, then you also need to enter Customizing settings for the application that the Fair Value Server calls:

- **Price Calculator**
  In Customizing for the *price calculator* you use the *Transfer Type* field to define how present value key figures are to be imported.
  In the *Settings for the Price Calculator Environment*, you can also use the valuation rule to define for which financial transactions and financial instruments the key figures are to be transferred.
  You must set the valuation rule for fair value for the free line and for the free line at the start of day to “Transfer Only”.
  You configure the price calculator in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *Edit Price Calculator Environments*.

  > **Note**
  >
  > For more information about using valuation rules to define how transactions are priced, see Valuation Rules [page 490]. The system derives a valuation rule for each financial transaction according to the settings made in Customizing. The system can also derive a valuation rule for each key figure of the Fair Value Server (for example, fair value of the free line for HFS loans).

> **Note**
>
> The price calculator does not scale the imported key figures. Therefore, you need to make sure that the key figures for financial instruments are already normed before they are transferred to the position. For a stock position, it is not the stock price, for example, but the value of the entire position that is to be transferred.
>
> In Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings* you can enter a scaling key figure that the system uses to norm imported key figures to the position in Bank Analyzer.
>
> You can use the scaling key figure only if you use a primary data source or the Results Data Layer to import key figures.

- **Fair Value Server**
  In Customizing for the *Fair Value Server* specify which present value figures you want to import. You also enter the information about where the present value key figures are stored that you want to import. When the Fair Value Server calls the price calculator, it tells the system where it can find the imported present value key figures.
  For the following key figures you can define whether the system interprets the imported values as clean (without accrued interest) or dirty (net present value; with accrued interest):
  - Full Fair Value
  - Full Fair Value, Start of Day
  - Hedge Fair Value
  - Hedge Fair Value, Start of Day

  All other key figures are always interpreted as clean values.
  You enter the settings for the Fair Value Server in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings*.
- **Bank Analyzer Application that Calls the Fair Value Server**
  If you use a secondary data source to import present value key figures, then you need to set up the secondary data source so that it is compatible with your settings in the Fair Value Server. In the secondary data source, the system has to read the present value key figures in question from the SDL. You have imported the present value key figures to the key figures in the valuation settings, and entered them in the calculation bases for the Fair Value Server. You use the Business Add-In *BAdI: Determine Calculation Basis* to do this.

Prerequisites

Import of Key Figures Using Primary Data Sources

You need to have already imported the present value key figures that were calculated externally into the SDL, and stored them in generic market data classes or SDL positions.

- Generic Market Data Classes
  You have created a generic market data class in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Generic Market Data* > *Edit Generic Market Data Classes*.

For each generic market data class, you enter defining characteristics, which are later used as a key when the system reads the data. Therefore, you need to make sure that the defining characteristics exist in the transaction or position that is to be valued.
Once you have created a generic market data class, the system automatically generates a primary data source that enables you to select the data from the generic market data class. The primary data source is always a *general primary data source*.
You can store up to five values in each generic master data class.

- To store values manually in generic market data classes, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Generic Market Data* > *Edit Generic Market Data*.
- If you want to import values automatically, use the *Create* method in the *GenericMarketData* BAPI.

For more information, see Generic Market Data [page 254].

- SDL Positions
  You need to have already created an SDL position, and stored in it the values that are to be transferred. On the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Position* > *Create Position*. When you create positions, you always set up a reference to a position class.
  Note that unlike generic market data classes, primary data sources for reading data from SDL positions are not generated automatically. To be able to select SDL positions, you have to create a primary data source of the category *object transfer structure* (OTS). You do this in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Object Transfer Structure (OTS)* > *Edit OTS for Primary Objects* > *Edit OTS for Positions*.
  You can store any number of values in SDL positions.
  - To store values manually in the SDL positions, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Position* and then either *Create Position* or *Change Position*, as required.
  - If you want to transfer values automatically, use the *AddNewVersion* method in the *PositionBA* BAPI.

  For more information, see Primary Object: Position [page 142].

> **Note**
>
> When you configure the import of key figures, it does not matter whether you store the key figures in generic market data classes or in SDL positions. You need only to specify the relevant primary data source and the name under which the key figure is to be stored. You need to provide the system with this information, as you can store multiple key figures for one SDL position or one generic market data class.

Import of Key Figures Using Secondary Data Sources

You have defined the secondary data sources for import of present value key figures so that the system can determine the present value key figures. You have imported the present value key figures to the key figures in the valuation settings, and entered them in the calculation bases for the Fair Value Server. You use the Business Add-In *BAdI: Determine Calculation Basis* to do this.

> **Note**
>
> Make sure that this process is complete before the price calculator is called.

Transfer of Present Value Key Figures By Reading Data from the RDL

The RDL must contain the relevant present value key figures. For more information about the Results Data Layer, see Results Data Layer (FS-BA-RD) [page 1364].

Features

You can transfer the following present value key figures:

- Full fair value (FFV)
- Amortized cost (AC)
- Fair value for free line (FVFL)
- Hedge fair value (HFV)
- Hedge amortized cost (HAC)
- Hedge adjustment (HAD)
- Effective capital over time (ECOT)
- Remaining debt
- Accrued interest

> **Note**
>
> The fair value level is imported together with the present value key figures.

Import of Present Value Key Figures for the Valuation of Business Transactions

The system usually measures financial transactions not only in the key date valuation in Accounting Processes, but it also measures particular business transactions for a day. To support this process for the imported present value key figures (*Full Fair Value*, *Amortized Cost*, *Fair Value for Free Line*, *Hedge Fair Value*, *Hedge Amortized Cost*, and *Hedge Adjustment*), you can define two values for each of these key figures:

- One value for the calculation at the end of the day (key figures `FFV`, `AC`, `FVFL`, `HFV`, `HAC` and `HAD`).
- One value for the start of the day, before the business transactions were posted (key figures: `FFV_EARLY`, `AC_EARLY`, `FVFL_EARLY`, `HFV_EARLY`, `HAC_EARLY`, `HAD_EARLY`)

To price the financial instruments for the individual business transactions, you have to set up the Fair Value Server for the import of values via a secondary data source. The system provides the secondary data source with information about the position and the scaling for the individual business transactions. You store the normed key figures in the generic market data or positions.

> **Note**
>
> For more information, see the documentation in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings*.

Consistency Check for Imported Key Figures

You can use the detail log [page 652] to check the key figures that were imported. The detail log gives you an overview of each step in the calculation process in the Fair Value Server. If there are errors in the key figures that were imported, this may be because the valuation results were incorrect. For example, when you import key figures, you need to make sure that the system treats the full fair value as a clean price.

You can use the detail log in analyses to call the Fair Value Server.

4.3.13.1.2 Auxiliary Functions

The price calculator uses the auxiliary function convexity adjustment [page 580] to adjust forward interest rates.

4.3.13.1.2.1 Convexity Adjustment

Use

You use this function to adjust forward interest rates that are used to price interest rate instruments. You usually carry out convexity adjustment if the actual value date of a cash flow stored in the transaction is different from the "natural" value date stored in the reference interest rate.

This is the case for financial transactions in which the term of the reference interest rate is different from the length of the period in which the interest rate is applied. Constant maturity swaps are one example of this type of financial transaction. They pay interest for the 10-year bond rate, for example, every six months against a fixed interest rate.

Prerequisites

In Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *Edit Price Calculator Environment*, you have specified a minimum value for the deviation from the value date of a transaction, and from the value date of the reference interest rate. If the deviation is at least as large as the minimum specified, the system carries out convexity adjustment. If you enter 1 as the minimum value, the system always carries out convexity adjustment; if you enter 0, it does not carry out convexity adjustment. You enter the minimum value on the basis of the valuation rule [page 490].

Features

The system uses the following formula to calculate the convexity adjustment:

where *term* is the period of time between the fixing date through to the valuation date; *forward rate* is the unadjusted forward rate; and *volatility* is the reference interest rate for the period of time from the valuation through to the fixing date. The system reads the volatility from the SDL; it does not calculate forward volatility.

The calculation of the convexity adjustment is based on the idea that first the key figures *Modified Duration* and *Convexity* are to be calculated for a bond that has the same term as the reference interest rate. These are the *modified duration* and *convexity* key figures that are used in the formula. In order to calculate these, the system first calculates the term of the reference interest rate in years. It then generates a fictitious bond that has the same term and interest payments for the whole year that are the same as the forward interest rate. The system uses difference quotients to calculate the *Modified Duration* and *Convexity* key figures

4.3.13.1.3 Valuation-Relevant Fields in the SDL Templates

In order to value financial transactions and financial instruments, the price calculator accesses certain fields in the templates of the Source Data Layer (SDL). Different fields are relevant, depending on the template used, and the financial transaction or financial instrument it maps. The following documents list the fields that are relevant for the valuation processes of the price calculator:

Template or Part of Template	Examples of Financial Transactions and Financial Instruments
Individual Cash Flows [page 582]	Relevant for general cash flows contained in financial transactions and financial instruments
Listed option [page 584]	• Options on Securities [page 584] • Options on Futures [page 587] • Options on Indexes [page 588]
OTC Options [page 591]	• Options on Securities (OTC) [page 591] • Swaptions (OTC) [page 593]
OTC Currency Options [page 595]	-
Interest Rate Caps and Floors [page 598]	Caps, Floors, and Collars
Stock [page 600]	-
Listed Bonds [page 601]	-
OTC Bonds [page 602]	-
Warrant Bonds [page 603]	-
Cash flow transaction [page 605]	Loans with unscheduled repayment options or call options, plus fixed-term deposits and deposits at notice.
Swap [page 610]	-
Current accounts [page 613]	-
Future [page 613]	• Index Futures [page 614] • Bond future [page 615] • Interest Rate Futures [page 617]
Forward Rate Agreement [page 618]	-
Forward Transactions [page 621]	-
Forward Security Transaction [page 621]	-
Foreign Exchange Transactions [page 623]	-

Template or Part of Template	Examples of Financial Transactions and Financial Instruments
Securities Lending [page 624]	-
Real Repurchase Agreement [page 626]	-

4.3.13.1.3.1 Individual Cash Flows

Use

You can store plan values for cash flows in the templates for financial transactions. You do so by using the Create Financial Transaction [page 123] function, and choosing the *Cash Flow* tab page, and entering the header and flow data for the cash flow.

Structure

The tables below contain the fields that the price calculator uses to value cash flows. The fields listed in the third table are required for variable cash flows that have been fixed before the valuation process.

> **Note**
>
> Note that the *Purchase/Sale* indicator in the basic data of the associated template is relevant here too. This applies regardless of which pricing model the system uses. The system uses the *Purchase/Sale* indicator when it translates currencies, for example.

Header data

Field Name	Comment
Valid From	
Valid To	
Cash Flow Category	You use this key to differentiate between different types of cash flows

Flow Data for Fixed Cash Flows

Field Name	Comment

Flow Type	In Risk Basis, the flow type is mapped to the cash flow indicator and the cash flow display indicator.
Amount	
Currency	
Maturity	The price calculator uses this field if no date was entered in the *Value Date* field.
Value Date	The date entered here is used to calculate the NPV.
Calculation From	Relevant for the calculation of accrued interest on bonds
Inclusive (Calculation From)	
Month-End (Calculation From)	
Calculation To	Relevant for the calculation of accrued interest on bonds
Inclusive (Calculation To)	
Month-End (Calculation To)	
Percentage Rate	
Condition Currency	If the condition currency is different from the amount currency, the price calculator takes the *Customer Rate* (see the *Basic Data* tab page when you are editing financial transactions or financial instruments).
Exchange Rate	Is used for currency translation
Interest category	The value *Fixed* is used for fixed cash flows.

Additional Flow Data for Variable Cash Flows (Interest Category *Variable*)

Field	Comment
Interest Calculation Method	
Days	
Base Days	
Nominal Amount	
Fixing	Date on which the interest rate is fixed
Offset on Reference Interest Rate	
Formula	

> **Note**
>
> If you define your own templates for financial transactions and then want to store cash flows for them, in Customizing for the Source Data Layer you must add the *Cash Flow Data* service module [page 118] to your template, and release the fields shown in the tables. The price calculator can value the cash flows from your templates only if you do this.

4.3.13.1.3.2 Listed Options

Definition

Template: S41OPTLIST (Option (Listed))

Use

You use template S41OPTLIST to map the class data for listed options and warrants. These include the following types of financial instruments:

- Options on securities [page 584]
- Options on futures [page 587]
- Options on indexes [page 588]
- Warrants on stocks
- Warrants on bond futures
- Warrants on indexes

4.3.13.1.3.2.1 Options on Securities

Use

You use two elementary templates to map the option (or warrant) and the security in the Source Data Layer (SDL). These templates are linked to one another by means of the contract relationship *B is an Underlying for A*. You store the data for the contract relationship in the option, since this is the starting point for the contract relationship. Using attributes in the option data, you can map different option classes (for example, barrier options and digital options), and different settlement methods (future or standard style).

Templates for Options on Stocks

Template	Description	Template Category	Comment
S41OPTLIST	Option (Listed)	Financial instrument	You store the contract relationship *B is an Underlying for A* in the option.
S41SHARE	Stock	Financial instrument	-

> **Note**
>
> For options on bonds, you use template S41BONDEX for the underlying.

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S41OPTLIST	STOPT	GENSTR	OPT_STOCK
S41OPTLIST	BOOPT	GENSTR	OPT_BOND
S41SHARE	-	GENINT	STOCK
S41BONDEX	-	GENINT	BOND

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
OPT_STOCK	OPTION MODEL	1	Standard option
OPT_BOND	OPTION MODEL	1	Standard option

Structure

The following table contains the fields that the price calculator uses to price options on securities.

Risk Basis Fields in the SDL Template

Tab Page	Field Name	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Currency	-

Tab Page	Field Name	Comment
Basic Data	Classification in Risk Basis	• For options on stocks: `STOPT` • For options on bonds: `BOOPT`
Object Relationships	Relationship Category	-
Object Relationships	Multiplicity	For stock options, the reference unit results from the multiplicity and the subscription ratio.
Option Data	Option Style	The price calculator currently prices only bond options that have the exercise type *European*.
Option Data	Category	-
Option Data	Settlement Method	-
Option Data (Details)	Subscription Ratio	-
Option Data (Details)	Strike Amount	Stock Options
Option Data (Details)	Currency for Strike Amount	Stock Options
Option Data (Details)	Exercise Price	Bond options
Option Data (Details)	Exercise Date: Valid From/Valid To	Time period in which the option can be exercised. For European options, the fields for the valid-from date and the valid-to date have to contain the same date value.
Key Figures	Key figure category: `NMVAL`	For bond options, the reference unit results from the key figure NMVAL (in the underlying) and the subscription ratio.

Note

For stock options, the system takes into account any dividends that fall due during the term of the option. You make the settings required in Customizing for *Financial Services* under *Foundation* > *Market Data* > *Prices for Financial Instruments* > *Corporate Actions*.

4.3.13.1.3.2.2 Options on Futures

Use

You use three elementary templates to map options on futures in the Source Data Layer (SDL). These templates are linked to one another by means of the contract relationships *B is Underlying for A*. The uppermost template maps the option itself. The contract relationship forms the reference between the option and the future contract. The underlying of the future is a synthetic bond.

Templates for Mapping Options on Bond Futures

Template	Description	Template Category	Comment
S41OPTLIST	Option (Listed)	Financial instrument	The option is linked to the future by means of the contract relationship *B is underlying for A*.
S41FUTURE	Future	Financial instrument	The future is linked to the bond by means of the contract relationship *B is underlying for A*.
S41BONDEX	Bond	Financial instrument	-

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S41OPTLIST	FTOPT	GENSTR	OPT_FU
S41FUTURE	FTBND	GENSTR	FUT_BND
S41BONDEX	-	GENINT	BOND

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
OPT_FU	OPTION MODEL	6	Options on Futures

Structure

The following table contains the fields that the price calculator uses to value options on futures.

Risk Basis Fields in the SDL Template

Tab Page	Field Name	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Currency	-
Basic Data	Classification in Risk Basis	`FTOPT`
Object Relationships	Relationship Category	-
Object Relationships	Multiplicity	The multiplicity specifies the number of contracts to which the option relates. Usually, the option is based on one contract.
Instrument Data	Nominal Amount / Currency	-
Key Figures	Key figure category: `NMVAL`	The key figure category can be used as an alternative to the nominal amount.
Option Data	Option Style	Here you enter the option category *A (American)*.
Option Data	Category	-
Option Data	Settlement Method	Here you enter the settlement method *2 (Future Style)*.
Option Data (Details)	Subscription Ratio	-
Option Data (Details)	Exercise Price	-
Option Data (Details)	Exercise Date: Valid From/Valid To	Time period in which the option can be exercised. For European options, the fields for the valid-from date and the valid-to date have to contain the same date value.

4.3.13.1.3.2.3 Options on Indexes

Use

You use multiple elementary templates to map options on indexes in the Source Data Layer (SDL). These templates are linked to one another by means of contract relationships. The uppermost template maps the option itself. It references the index by means of the relationship category *B is Underlying for A*.

Using attributes in the option data, you can map different option classes (for example, barrier options and digital options), and different settlement methods (future or standard style).

Templates for Options in Indexes

Template	Description	Template Category	Comment
S41OPTLIST	Option (Listed)	Financial instrument	The option is linked to the index by means of the contract relationship *B is underlying for A*.
INDEX	Index	Financial instrument	You store the contract relationship *B is Underlying for A* in the index.
S41SHARE	Stock	Financial instrument	-

> **Note**
>
> Note that you have to create the template `INDEX` yourself, since it is not part of the sample Customizing. You usually map indexes in the SDL so that they reference a multivalue contract relationship with the individual stocks in the portfolio. However, this structure is not required for the evaluations in Risk Basis.

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S41OPTLIST	INOPT	GENSTR	OPT_INDEX
INDEX	-	GENINT	INDEX

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
OPT_INDEX	OPTION MODEL	1	Standard Options
INDEX	MARK2MARKET	1	Standard security transaction

Features

The following table contains the fields that the price calculator uses to value options on indexes.

Risk Basis Fields in the SDL Template

Tab Page	Field Name	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Currency	-
Basic Data	Classification in Risk Basis	INOPT
Index Data	Index Point Value	The index data is relevant only if no data is entered in the *Point Value* and *Currency of Point Value* fields in the option data. If values are entered in these fields, the system overwrites the index data.
Index Data	Index Point Currency	The index data is relevant only if no data is entered in the *Point Value* and *Currency of Point Value* fields in the option data. If values are entered in these fields, the system overwrites the index data.
Object Relationships	Relationship Category	-
Option Data	Option Style	-
Option Data	Category	-
Option Data	Settlement Method	Here you enter the settlement method *2 (Future Style for Listed Options)*.
Option Data	Point Value	If you enter a point value, then this overwrites the index point stored on the *Index Data* tab page.
Option Data	Point Value Currency	If you enter a currency for the point value, then this overwrites the currency of the index point stored on the *Index Data* tab page.
Option Data (Details)	Subscription Ratio	-
Option Data (Details)	Exercise Price	-
Option Data (Details)	Exercise Date: Valid From/Valid To	Time period in which the option can be exercised. For European options, the fields for the valid-from date and the valid-to date have to contain the same date value.

4.3.13.1.3.3 OTC Options

Definition

Template: S40OPTOTC (Option/Swaption (OTC))

Use

You use template S40OPTOTC to map the following OTC options:

- Options on securities (OTC) [page 591]
- Options on indexed (OTC)
- Options on futures (OTC)
- Swaptions (OTC) [page 593]

4.3.13.1.3.3.1 Options on Securities (OTC)

Use

In the Source Data Layer (SDL), the option and the underlying are mapped as separate elementary templates. You use the template category *Financial Transaction* for the OTC option, and the template category *Financial Instrument* for the underlying. The two templates are linked by an object relationship, the starting point of which is the OTC option. You store the data for the contract relationship in the option.

Templates for OTC Options on Stocks

Template	Description	Template Category	Comment
S40OPTOTC	Option/Swaption (OTC)	Financial transaction	You store the contract relationship *B is an Underlying for A* in the option.
S41SHARE	Stock	Financial instrument	-

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S40OPTOTC	STOPT	GENSTR	OPT_STOCK
S40OPTOTC	BOOPT	GENSTR	OPT_BOND

Template	Classification in Risk Basis	Mapper	Transaction Form
S41SHARE	-	GENINT	STOCK
S41BONDEX	-	GENINT	BOND

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
OPT_STOCK	OPTION MODEL	1	Standard Options
OPT_BOND	OPTION MODEL	1	Standard Options

Structure

The following table contains the fields that the price calculator uses to value OTC options on securities.

Risk Basis Fields in the SDL Template

Tab Page	Field Name	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Currency	-
Basic Data	Purchase/Sale	Enter whether the OTC option is a purchase or a sale.
Basic Data	Classification in Risk Basis	• For options on stocks: STOPT • For options on bonds: BOOPT
Object relationships	Relationship category	-
Object relationships	Multiplicity	For stock options, the reference unit results from the multiplicity and the subscription ratio.
Option Data	Option Style	-
Option Data	Category	-

Tab Page	Field Name	Comment
Option Data	Purchase/Sale	If you set the *Purchase/Sale* indicator on the *Option Data* tab page, then this setting overwrites the setting in the *Purchase/Sale* indicator on the *Header Data* tab page.
Option Data (Details)	Subscription Ratio	-
Option Data (Details)	Strike Amount	Stock Options
Option Data (Details)	Currency for Strike Amount	Stock Options
Option Data (Details)	Exercise Price	Bond Options
Option Data (Details)	Exercise Date: Valid From/Valid To	Time period in which the option can be exercised. For European options, the fields for the valid-from date and the valid-to date have to contain the same date value.
Key Figures	Key figure category: `NMVAL`	For bond options, the reference unit results from the key figure NMVAL (in the underlying) and the subscription ratio.

> **Note**
>
> For stock options, the system takes into account any dividends that fall due during the term of the option. You make the settings required in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Market Data* > *Prices for Financial Instruments* > *Corporate Actions*.

4.3.13.1.3.3.2 Swaptions (OTC)

Use

You map swaptions in the Source Data Layer (SDL) as structured templates. The parent node (option) points to the child node (swap) by means of the template hierarchy. The swap contains a structure as well.

> **Note**
>
> Note that the price calculator can value only plain vanilla swaps as underlyings.

Templates for Swaptions

Template	Description	Template Category	Comment
S40OPTOTC	Option/Swaption (OTC)	Financial transaction	A swaption is linked to the swap by means of the template hierarchy.
S40SWAP	Swap	Financial transaction	The swap is linked to the swap legs by means of the template hierarchy.
S40SWAPLEG	Swap Leg	Financial transaction	-

Cardinalities for Structured Templates

Template	Subtemplate	Minimum Cardinality	Maximum Cardinality
S40OPTOTC	S40SWAP	0	1
S40SWAP	S40SWAPLEG	2	2

> **Note**
>
> Since template S40OPTOTC is also used for other options, a minimal cardinality of 0 has to be entered in the template hierarchy of subtemplate S40SWAP.

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S40OPTOTC	SWPTN	GENSTR	OPT_SWAP
S40SWAP	STDR	SWAP	SWAP_STD
S40SWAPLEG	-	GENINT	SWAP_LEG

Settings in the Pricing Model

Transaction Form	Pricing Model	Classification in Risk Basis	Mapper
OPT_SWAP	OPTION MODEL	2	Swaption
SWAP_STD	STRCTRD PRODUCTS	1	Priced as one entity
SWAP_LEG	S40SWAPLEG	-	-

Structure

The following table contains the fields that the price calculator uses to value the option (parent node).

Risk-Basis-Relevant Fields in the SDL Templates for the Option

Tab Page	Field Name	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Purchase/Sale	By setting this indicator, you define whether the option is a purchase or a sale.
Basic Data	Classification in Risk Basis	SWPTN
Option Data	Option Style	The price calculator values only those swaptions that have the exercise type *European*.
Option Data	Category	-
Option Data	Purchase/Sale	If you set the *Purchase/Sale* indicator on the *Option Data* tab page, then this setting overwrites the setting in the *Purchase/Sale* indicator on the *Header Data* tab page.
Option Data (Details)	Subscription Ratio	The subscription ratio must have the value **1**.
Option Data (Details)	Exercise Date: Valid From/Valid To	Time period in which the option can be exercised. For European options, the fields for the valid-from date and the valid-to date have to contain the same date value.

> **Note**
>
> For a description of the fields that are relevant for the swap, see Swaps [page 610].

4.3.13.1.3.4 OTC Currency Options

Definition

Template: S40FXOPT (FX Option (Compound))

Use

You use template S40FXOPT to map OTC currency options. In the Source Data Layer (SDL), these are stored as structured templates. The currency option itself is the parent node, and has the template category *Financial Transaction*. The currency transaction is the child node, which is assigned to the parent node. The child node contains the information about the underlying in the form of cash flows.

Templates for OTC Currency Options

Template	Description	Template Category	Comment
S40FXOPT	(FX Option (Compound))	Financial transaction	A currency option is linked to the currency transaction by means of the template hierarchy.
S40FXFORWD	Foreign Exchange Transaction (Forward)	Financial transaction	-

Cardinalities for Structured Templates

Template	Subtemplate	Minimum Cardinality	Maximum Cardinality
S40FXOPT	S40FXFORWD	1	1

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S40FXOPT	-	GENSTR	OPT_FX
S40FXFORWD	-	GENINT	FWD_FX

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
OPT_FX	OPTION MODEL	3	Currency options
FWD_FX	-	-	-

Features

The following tables contain the fields that the price calculator uses to value OTC currency options.

Risk-Basis-Relevant Fields in the SDL Templates for the Option

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Purchase/Sale	By setting this indicator, you define whether the transaction is a purchase or a sale.
Option Data	Option Style	-
Option Data	Category	-
Option Data	Purchase/Sale	If you set the *Purchase/Sale* indicator in the option data, this setting overrides the setting you made for the *Purchase/Sale* indicator in the header data.
Option Data (Details)	Subscription Ratio	-
Option Data (Details)	Exercise Date: Valid From/Valid To	Time period in which the option can be exercised. For European options, the fields for the valid-from date and the valid-to date have to contain the same date value.

Risk-Basis-Relevant Fields in the SDL Templates for the Underlying

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Purchase/Sale	By setting this indicator, you define whether the transaction is a purchase or a sale.
Cash Flow (Header)	Cash Flow Category	-
Cash Flow (Header)	Cash Flow Valid From	-
Cash Flow (Header)	Cash Flow Valid To	-

Tab Page	Field	Comment
Cash Flow (Item)	Flow Type	Each of the flow types of the cash flows has to be mapped with different signs to cash flow indicator 5. The amount with the plus sign (+) is interpreted as the foreign currency amount for the calculation of the strike. Then, using the option category, you define whether the foreign currency amount is to be bought (call) or sold (put).
Cash Flow (Item)	Amount/Currency	-
Cash Flow (Item)	Value Date	-

> **Note**
>
> The underlying has to contain two cash flows, which the price calculator uses to calculate the base amount and the strike.

4.3.13.1.3.5 Caps, Floors, and Collars

Definition

Template: S40CAFLCOL (Interest Rate Cap/Floor)

Use

You use template S40CAFLCOL to map caps, floors, and collars as unstructured transactions in the Source Data Layer (SDL). Unlike for other options, the system does not use the *Option Data* tab page to map caps, floor, and collars. The option information is contained within the condition data and in the cash flow generated on the basis of the condition data.

This means that before you use the mapper you must first call the cash flow generation function. The system creates the caplets and floorlets from the cash flow that is then generated. The caplets and floorlets are then priced by the price calculator. In this process, the system generates a caplet from the variable cash flow that has the maximum interest rate, and a floorlet from the variable cash flow that has the minimum interest rate. A collar comprises a maximum and a minimum interest rate, which means that the system generates both a caplet and a floorlet from the variable cash flow. In the valuation structure, the system creates a node that maps an underlying for each caplet and floorlet.

Templates for OTC Caps and Floors

Template	Description	Template Category	Comment
S40CAFLCOL	OTC Cap/Floor	Financial transaction	-

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S40CAFLCOL	CAP	CAPFLOOR	CAP
S40CAFLCOL	FLOOR	CAPFLOOR	FLOOR
S40CAFLCOL	COLLR	CAPFLOOR	COLLAR

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
CAP	STRCTRD PRODUCTS	2	Total of the parts that can be priced
FLOOR	STRCTRD PRODUCTS	2	Total of the parts that can be priced
COLLAR	STRCTRD PRODUCTS	2	Total of the parts that can be priced
CAPLET	OPTION MODEL	5	Caplet/floorlet
FLOORLET	OPTION MODEL	5	Caplet/floorlet

Structure

The table below contains the fields that the price calculator uses to price caps, floors, and collars. The table contains only the main fields required for pricing, and does not consider the conditions.

Risk Basis Fields in the SDL Template

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Start Fixed Int.	-
Basic Data	End of Fixed Int.	-

Tab Page	Field	Comment
Basic Data	Currency	-
Basic Data	Purchase/Sale	By setting this indicator, you define whether the transaction is a purchase or a sale.
Basic Data	Classification in Risk Basis	Cap, floor, or collar
Conditions	Condition Group Type	To map the individual caplets and floorlets, you use condition group type *2004* (derivatives).
Conditions (Details)	Condition Type	You use the condition type *Nominal Interest (CAP)* or *Nominal Interest (FLOOR)*.
Conditions (Details)	Currency of Condition	-
Conditions (Details)	Reference Interest Rate	-
Conditions (Details)	Maximum Interest Rate	The cap rate is relevant for caps and collars only.
Conditions (Details)	Minimum Interest Rate	The floor rate is relevant for floors and collars only.
Key Figures	Key Figure Category	To map the nominal amount you can use the key figure category `NMVAL`.

> **Note**
>
> The nominal amount is stored by means of a key figure in the *Key Figures* tab page.

4.3.13.1.3.6 Stocks

Definition

Template: S41SHARE (Stock)

Use

You use template S41SHARE to map stocks and financial instruments, such as shareholdings that are similar to stocks, in the Source Data Layer (SDL). You use `STOCK` as the transaction form.

Structure

Market rates (pricing model `MARK2MARKET`) are used to value the transaction form `STOCK`. Therefore, no special fields are relevant for the price calculator.

4.3.13.1.3.7 Listed Bonds

Definition

Template: S41BONDEX (Listed Bond)

Use

You use template S41BONDEX to map class data for interest-bearing securities in the Source Data Layer (SDL) (the template category is *Financial Instrument*). These include the following types of security:

- Coupon bonds
- Floating rate notes
- Reverse floating rate notes
- Step-up bonds
- Multicurrency bond
- Zero bond

You can use conditions, which you store in the financial instrument, to generate the various cash flows related to the bonds.

> **Note**
>
> When you map bonds with call rights, you do so in the same way as for cash flow transactions [page 605]. You have to create the same *Financial Instrument* category templates for the call option and the call condition.

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S41BONDEX	-	GENINT	BOND

> **Note**
>
> To be able to analyze each currency part of multicurrency bonds separately so that the currencies to not have to be translated, you use mapper `MCB` to map each currency component on a different node in the valuation structure. You can use the transaction forms `MCB` and `MCB_CURR`, plus the classification `MCB` in Risk Basis to map multicurrency bonds and conventional bonds separately.

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
BOND	MARK2MARKET	1	Standard security transaction
BOND	CASH-FLOW DISC	-	-

Structure

The following table contains the fields that the price calculator uses to price listed bonds. Note the following constraints:

- If transaction form `BOND` is measured using market prices only (using the `MARK2MARKET` pricing model), no special fields are relevant for the price calculator.
- If pricing model `CASH-FLOW DISC` - *Cash Flow Discounting* is used, the fields used to generate the cash flow (conditions and so on) or the fields for the transferred cash flows (cash flow data service module) are relevant.

Risk Basis Fields in the SDL Template

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Start Fixed Int.	-
Basic Data	End of Fixed Int.	-
Basic Data	Currency	-
Conditions	Condition Group Type	To map listed bonds, you use the condition group type 2002 (Securities).

4.3.13.1.3.8 OTC Bonds

Definition

Template: S40CASHFLW (Cash Flow Transaction)

Use

You use template S40CASHFLW of template category *Financial Transaction* to map bonds in the Source Data Layer (SDL) as cash flows. You can generate the cash flow by means of the conditions, or you can use the *Cash Flow Data* tab page to let the system do this. For OTC bonds, you use the transaction form `LOAN` and the pricing model `CASH_FLOW DISC` – *Discounting of Cash Flows*.

Structure

The fields used to generate the cash flow (conditions and so on) or the fields for the cash flow data are relevant for the valuation process.

More Information

Individual Cash Flows [page 582]

4.3.13.1.3.9 Warrant Bonds

Definition

Template: S41WARRLBD (Warrant Bond Cum)

Use

You use template S41WARRLBD to map the warrant bonds in the Source Data Layer (SDL). Warrant bonds are structured financial instruments to which are assigned two child templates: one for the bond, and one for the warrant. The warrant bond itself is only a means of linking its two parts, the warrant and the bond.

Templates for Warrant Bonds

Template	Description	Template Category	Comment
S41WARRLBD	Listed Warrant Bond Listed (Cum)	Financial instrument	The warrant bond is linked to both the bond and the option by means of template hierarchies.
S41OPTLIST	Option (Listed)	Financial instrument	-

Template	Description	Template Category	Comment
S41BONDEX	Bond (Listed)	Financial instrument	-

Cardinalities for Structured Templates

Template	Subtemplate	Minimum Cardinality	Maximum Cardinality
S41WARRLBD	S41OPTLIST	0	N
S41WARRLBD	S41BONDEX	0	1

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S41WARRLBD	-	GENSTR	WARR_BOND
S41SHARE	-	GENINT	BOND
S41OPTLIST	STOPT	GENSTR	OPT_STOCK

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
WARR_BOND	STRCTRD PRODUCTS	1	Priced as one entity

Features

The following table contains the fields that the price calculator uses to value warrant bonds. For a description of the fields that are relevant for the child templates *Bond* and *Warrant* , see Listed Bonds [page 601] and Listed Options [page 584].

Risk-Basis-Relevant Fields in the SDL Template for the Warrant Bonds

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Purchase/Sale	By setting this indicator, you define whether the option is a purchase or a sale.

You must also store the subscription ratio in the option. You use the key figures `WRNOM` and `WRSTK` to do this. The subscription ratio specifies how many warrants are allocated to which nominal value.

The position for the warrant bond is usually at parent template level. It is stored in the complex key figure in the parent template before the price calculator is called. However, the system needs the position to be in the child node so that it can generate cash flows on the bond node, and calculate the number of warrants on the option node. It does this by using a data source item type of the category *Move Key Figure* to read the position from the parent template and transfer it to a complex key figure.

4.3.13.1.3.10 Cash Flow Transactions, Such As Loans with Option Rights

Definition

Template: S40CASHFLW (Cash Flow Transaction)

Use

You use template S40CASHFLW to map the following transactions in the Source Data Layer (SDL):

- **Loans with option rights for the following types of loans:**
  - Simple loans without option rights
  - Loans with options for making special (unscheduled) repayments
  - Loans with call (cancellation) options
  - Loans with special repayment options and call options

  Loans with option rights are mapped as structured financial transactions in the SDL. They consist of the main template (parent node) *Cash Flow Transaction* and multiple subtemplates (child nodes). The subtemplates on the second level in the hierarchy map the special repayment right or call right. These subtemplates are themselves structured and have their own subtemplates, or child nodes, on the third hierarchical level. These child nodes map the underlyings of the corresponding options, and can contain the repayment conditions in the form of special repayment options or call options.
  The system calculates the cash flows that occur during the term of the transaction using the conditions that you store in the loan itself.
- **Fixed-term deposits and deposits at notice**
  You can create fixed-term deposits and deposits at notice without having to create options rights by storing cardinalities for the subtemplates *Special Repayment Option* and *Call Option*.

Templates for Cash-Flow-Based Transactions

Template	Description	Template Category	Comment
S40CASHFLW	Cash flow transaction	Financial transaction	The cash-flow-based transaction is linked to the special repayment option and call option by means of the template hierarchy.
S40UNSCHED	Special repayment option	Financial transaction	The special repayment option is linked to the call condition by means of the template hierarchy.
S40NOTICE	Call option	Financial transaction	The call option is linked to the call condition by means of the template hierarchy.
S40UNSCCON	Special repayment condition	Financial transaction	-
S40NOTCOND	Call condition	Financial transaction	-

Cardinalities for Structured Templates

Template	Subtemplate	Minimum Cardinality	Maximum Cardinality
S40CASHFLW	S40NOTICE	0	1
S40UNSCHED	S40UNSCHED	0	1
S40NOTICE	S40NOTCOND	1	1
S40UNSCHED	S40UNSCCON	1	1

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S40CASHFLW	LOAN	GENINT	LOAN_STD
S40CASHFLW	LOANM	LOAN_OPT	LOAN_MOPT
S40CASHFLW	LOANS	LOAN_OPT	LOAN_SOPT
S40NOTICE	-	LOAN_OPTR	OPT_CANC
S40UNSCHED	-	LOAN_OPTR	OPT_REP

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
LOAN_STD	CASH-FLOW DISC	-	-
LOAN_SOPT	STRCTRD PRODUCTS	1	Priced as one entity
OPT_NODE	STRCTRD PRODUCTS	1	Priced as one entity
OPT_CANC	OPTION MODEL	1	Standard model
OPT_REP	OPTION MODEL	1	Standard model

> **Note**
>
> Note that there is no standard pricing model for the pricing of cash flow transactions that have multiple notice rights. In order to price these types of financial transactions, you need to store a method schema [page 534] that contains the cash flow refinement method cash flow determination [page 504]. For this reason, the table above does not contain a pricing model for transaction form LOAN_MOPT.

Structure

The following tables contain the fields that the price calculator uses to price cash-flow-based transactions.

Risk Basis Fields in the SDL Template for Cash Flow Transactions (Parent Node)

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Start Fixed Int.	-
Basic Data	End of Fixed Int.	-
Basic Data	Currency	-
Basic Data	Purchase/Sale	By setting this indicator, you define whether the transaction is a purchase or a sale.
Basic Data	Classification in Risk Basis	LOAN, LOANM or LOANS
Basic Data	Period of Notice	Depending on the call options, the cancellation dates can change when you enter the notice period.
Basic Data	Time Unit	-

Risk Basis Fields in the SDL Template for Call Options (Subtemplate on Second Level)

Tab Page	Field	Comment
Option Data	Valid From/Valid To	-
Option Data	Style	The price calculator supports only European-style call options.
Option Data	Category	-
Option Data	Expiry of Option Indicator	This indicator defines whether all subsequent options (headers) expire once an option (for a header) is exercised.
Option Data	Multiple Exercise Possible Indicator	This indicator defines whether all subsequent options within a header expire once an option is exercised.
Option Data	Option Class	You use the option class LFR (*Call Right (Notice Period)*).
Option Data (Details)	Exercise Price	The exercise price refers to the base reference in the call condition. This entry is relevant only if the strike amount is not specified.
Option Data (Details)	Strike Amount/Currency	The strike amount corresponds to the amount that can be cancelled. The system checks whether this amount is larger than, or equal to, the remaining debt.
Option Data (Details)	Exercise Date: Valid From/Valid To	Time period in which the option can be exercised. For European options, the fields for the valid-from date and the valid-to date have to contain the same date value.

Risk Basis Fields in the SDL Template for Call Conditions (Subtemplate for Call Option)

Tab Page	Field	Comment
Basic Data	Currency	-
Basic Data	Purchase/Sale	The *Purchase/Sale* indicator defines whether the option is a short or long option.

Tab Page	Field	Comment
Conditions	Condition Group Type	To map the call condition, you use the condition group type 2001 (loans).
Conditions (Details)	Condition Type	You use the condition type 103130 (Final Repayment).
Conditions (Details)	Currency of Condition	-

Risk Basis Fields in the SDL Template for Special Repayment Options (Subtemplate on the Second Level)

Tab Page	Field	Comment
Option Data	Valid From/Valid To	-
Option Data	Style	The price calculator currently supports only European-style special repayment options.
Option Data	Category	-
Option Data	Expiry of Option Indicator	This indicator defines whether all subsequent options (headers) expire once an option (for a header) is exercised.
Option Data	Multiple Exercise Possible Indicator	This indicator defines whether all subsequent options within a header expire once an option is exercised.
Option Data	Option Class	You use the option class LSR (Unscheduled Repayment Right).
Option Data (Details)	Exercise Price	The exercise price refers to the base reference in the repayment conditions. This entry is relevant only if the strike amount is not specified.
Option Data (Details)	Strike Amount/Currency	The strike amount corresponds to the amount that can be cancelled. The system checks whether this amount is larger than, or equal to, the remaining debt.
Option Data (Details)	Exercise Date: Valid From/Valid To	Time period in which the option can be exercised. For European options, the fields for the valid-from date and the valid-to date have to contain the same date value.

Risk Basis Fields in the SDL Template for Special Repayment Conditions (Subtemplate for Special Repayment Option)

Tab Page	Field	Comment
Basic Data	Currency	-
Basic Data	Purchase/Sale	The *Purchase/Sale* indicator defines whether the option is a short or long option.
Conditions	Condition Group Type	To map the special repayment conditions, you use the condition group type 2001 (loans).
Conditions (Details)	Condition Type	You use the condition type 103140 (Unscheduled Repayment).
Conditions (Details)	Currency of Condition	-

Caution

Note the following considerations when you use option data and condition data:

During the cash flow generation process, the system reads the option header valid for the key date from the option node. As a result only the exercise dates that belong to this header are considered. The exercise dates for the option header valid on this date are then used to determine the exercise dates. The currently valid condition is read for each exercise date. In this condition, the system then sets the maturity date or value date (DFAELL or DVALUT) to the exercise date.

The generation of the cash flows for loans with call rights or special repayment rights depends on the generation category that is used in the top node.

4.3.13.1.3.11 Swap

Definition

Template: S40SWAP (Swap)

Use

You use template S40SWAP to map the following swap types in the Source Data Layer (SDL):

- Plain vanilla swaps
- Basic swaps

- Cross-currency swaps
- Step-up swaps and step-down swaps (amortizing swaps)
- Roller-coaster swaps
- Reverse floating swaps
- Eonia swaps

The template hierarchy is the same for all types of swaps: A swap is made up of two subtransactions (the legs) that contain the conditions of the cash flows that are to be swapped. The two legs are mapped by the same template.

> **Note**
>
> The conditions of each cash flow determine the dates and how the system calculates the cash flows. Therefore, any differences between the swaps do not arise from the hierarchy, but from the conditions.

Templates for Swaps

Template	Description	Template Category	Comment
S40SWAP	Swap	Financial transaction	The swap is linked to the swap legs by means of the template hierarchy.
S40SWAPLEG	Swap Leg	Financial transaction	-

Cardinalities for Structured Templates

Template	Subtemplate	Minimum Cardinality	Maximum Cardinality
S40SWAP	S40SWAPLEG	2	2

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
500_010	STRD	SWAP	SWAP_STD
500_020	-	GENINT	SWAP_LEG

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
SWAP_STD	STRCTRD PRODUCTS	1	Can be priced as one entity
SWAP_LEG	CASH-FLOW DISC	-	-

Structure

The following tables contain the fields that the price calculator uses to value swaps.

Risk-Basis-Relevant Fields in the SDL Templates for the Swap

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Purchase/Sale	By setting this indicator, you define whether the swap is a purchase or a sale.
Basic Data	Classification in Risk Basis	STDR

Risk-Basis-Relevant Fields in the SDL Template for Swap Legs

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Start Fixed Int.	-
Basic Data	End of Fixed Int.	-
Basic Data	Currency	Currency of the leg
Basic Data	Purchase/Sale	By setting this indicator, you define whether the swap leg is a purchase or a sale.
Conditions	Condition Group Type	Use condition group type 2004 (Derivatives)
Conditions (Details)	Condition Type	Enter condition type 101100 (Nominal Interest –Asset Side) or 101105 (Nominal Interest –Liability Side).
Conditions (Details)	Currency of Condition	The currency of the condition is the same as that of the swap leg.

> **Note**
>
> The system generates the cash flow for the swap by generating the cash flows for each leg of the swap, and then merging them on the highest level of the hierarchy, which is the swap itself.

4.3.13.1.3.12 Current Accounts

Definition

Template: S40CURRACC (Current Account)

Use

You use template S40CURRACC to map current accounts in the Source Data Layer (SDL).

For accounts you use the transaction form `ACCOUNT`. The system uses the *Discounting of Cash Flows* – `CASH-FLOW DISC` pricing model to value the accounts.

Structure

Since the `CASH-FLOW DISC` valuation model is used, all the fields in the SDL template that are relevant for cash flow generation [page 440] are relevant here.

> **Note**
>
> Note that the Purchase/Sale indicator in the basic data of the associated template is relevant here too. This applies regardless of which pricing model the system uses. The system uses the Purchase/Sale indicator when it translates currencies, for example.

We recommend that you use a position class, which contains the key figures for the debit and credit balances, to map the balance.

The system contains the following settings for cash flow generation:

- You can use condition group type 9901 (Current Account) for cash flow generation. This type contains, for example, conditions for debit and credit balances.
- In the generation type used, you can create data source types of the category 01: *Conditions* to read the conditions, and of the category 06: *Positions* to read positions.
- You can use item types of category 03: *Calculation Basis* for the debit/credit balance that then populate the respective base references *0001: Debit Balance* or *0002: Credit Balance* from the key figures.

4.3.13.1.3.13 Futures

Definition

Template: S41FUTURE (Future)

Use

You use template S41FUTURE to map the following futures in the Source Data Layer (SDL):

- Index Futures [page 614]
- Bond Futures [page 615]
- Interest Rate Futures [page 617]

4.3.13.1.3.13.1 Index Futures

Definition

Template: S41FUTURE (Future)

Use

You use template S41FUTURE to map class data for index futures in the Source Data Layer (SDL). Index futures are mapped by means of multiple elementary templates, which are linked to one another by contract relationships.

> **Note**
>
> The index can refer to the individual stocks in its portfolio by means of a multiple-value object relationship. However, this system does not need to use this relationship to value the index future.

Templates for Interest Rate Futures

Template	Description	Template Category	Comment
S41FUTURE	Future	Financial instrument	The future is linked to the index by means of the contract relationship B is underlying for A.
INDEX	Index	Financial instrument	The index is linked to the stocks by means of the contract relationship B is underlying for A.
S41SHARE	Stock	Financial instrument	-

> **Note**
>
> Note that you have to create the template INDEX yourself, since it is not part of the sample Customizing.

Mapping

Template	Classification in Risk Basis	Mapping	Transaction Form
S41FUTURE	FTIND	GENSTR	FUT_INDEX
INDEX	-	GENINT	INDEX

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
FUT_INDEX	FUTURE	-	-
INDEX	MARK2MARKET	1	Standard security transaction

Structure

The following table contains the fields that the price calculator uses to value index futures.

Fields in the SDL Template for the Index Future that are Relevant for Risk Basis

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Currency	-
Basic Data	Classification in Risk Basis	FTIND
Instrument Data	Quote Type	Quotation *02 Percentage Quotation* is used for index futures.
Index Data	Index Point Value	-
Index Data	Index Point Currency	-

4.3.13.1.3.13.2 Bond Futures

Definition

Template: S41FUTURE (Future)

Use

You use template S41FUTURE to map class data for bond futures in the Source Data Layer (SDL). The future contract uses the contract relationship to reference the bond.

Templates for Bond Futures

Template	Description	Template Category	Comment
S41FUTURE	Future	Financial instrument	The future is linked to the bond by means of the contract relationship B is underlying for A.
S41BONDEX	Bond (Listed)	Financial instrument	-

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S41FUTURE	FTBND	GENSTR	FUT_BND

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
FUT_BND	FUTURE	-	-

Structure

The following table contains the fields that the price calculator uses to value bond futures.

Fields in the SDL Template for the Bond Future that are Relevant for Risk Basis

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Currency	-
Basic Data	Classification in Risk Basis	FTBND
Instrument Data	Quote Type	Quotation *01 Percentage Quotation* is used for bond futures.

Tab Page	Field	Comment
Instrument Data	Nominal Amount	You can also use the key figure `NMVAL` here.
Instrument Data	Nominal Currency	You can also use the key figure `NMVAL` here.

4.3.13.1.3.13.3 Interest Rate Futures

Definition

Template: S41FUTURE (Future)

Use

You use template S41FUTURE to map class data for interest futures in the Source Data Layer (SDL). The future contract uses the contract relationship to create a reference to the underlying reference interest rate.

Templates for Interest Rate Futures

Template	Description	Template Category	Command
S41FUTURE	Future	Financial instrument	The future is linked to the reference interest rate by means of the contract relationship *B is an underlying for A*.
INTEREST RATE	Reference interest rate	Financial instrument	-

> **Note**
>
> The reference interest rate is mapped as a financial instrument that contains only the reference interest rate as the required information. Note that you have to create the template ZINS yourself, since it is not part of the sample Customizing.

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S41FUTURE	FTINT	GENSTR	FUT_INT

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Command
FUT_INT	FUTURE	-	-

Structure

The following table contains the fields that the price calculator uses to value interest rate futures.

Fields in the SDL Template for the Interest Rate Future that are Relevant for Risk Basis

Tab Page	Field	Command
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Currency	-
Basic Data	Classification in Risk Basis	FTINT
Instrument Data	Quote Type	Quotation *04: Inverse* is used for interest rate futures.
Instrument Data	Nominal Amount	You can also use the key figure `NMVAL` here.
Instrument Data	Nominal Currency	You can also use the key figure `NMVAL` here.

> **Note**
>
> In the underlying, you must enter one cash flow in which a value is entered in the *Reference Interest Rate* field for the flow data.

4.3.13.1.3.14 Forward Rate Agreement

Definition

Template: S40FRA (Forward Rate Agreement)

Use

You use template S40FRA to map forward rate agreements (FRAs) in the Source Data Layer (SDL). In the SDL, an FRA is mapped as a structured financial transaction that has two subtransactions (FRA legs). One of the FRA legs contains the interest rate conditions of the base interest rate, and the other contains the interest rate conditions of the market rate. The system uses both sets of interest rate conditions to calculate the future cash flow. Since any differences in the legs are due to the conditions, and not the hierarchy, the same template is used to map the FRA legs.

Templates for Forward Rate Agreements

Template	Description	Financial Transaction	Comment
S40FRA	Forward Rate Agreement	Financial transaction	The forward rate agreement is linked to the FRA leg by means of the template hierarchy.
S40FRALEG	FRA_LEG	Financial transaction	-

Cardinalities for Structured Templates

Template	Subtemplate	Minimum Cardinality	Maximum Cardinality
S40FRA	S40FRALEG	2	2

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S40FRA	-	GENSTR	S40FRA
S40FRALEG	-	GENINT	S40FRALEG

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
FRAs	STRCTRD PRODUCTS	1	Can be priced as one entity
FRA	FORWARD	2	Forward Rate Agreement
FRA_LEG	CASH-FLOW DISC	-	-

> **Note**
>
> When an FRA is priced, the difference between the fixed leg and the variable leg is not usually paid at the end of the interest rate hedging period. Instead, it is paid in a discounted version at the beginning of the interest rate

hedging period. To ensure that the price calculator values the FRA correctly, you therefore need to use pricing model `FORWARD` with configuration value 2.

Structure

The following tables contain the fields that the price calculator uses to value forward rate agreements. Make sure that the FRA legs contain the information needed for the valuation.

Risk-Basis-Relevant Fields in the SDL Template for Forward Rate Agreements

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Currency	-
Basic Data	Purchase/Sale	By setting this indicator, you define whether the transaction is a purchase or a sale.

Risk-Basis-Relevant Fields in the SDL Template for the Legs of the FRA

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Start Fixed Int.	-
Basic Data	End of Fixed Int.	-
Basic Data	Currency	This field is based on the currency of the FRA leg.
Basic Data	Purchase/Sale	By setting this indicator, you define whether the transaction is a purchase or a sale.
Conditions	Condition Group Type	Use condition group type 2004 (*Derivatives*)
Conditions (Details)	Condition Type	You use the condition type *Nominal Interest*.
Conditions (Details)	Valid From/Valid To	Interest rate condition period

Tab Page	Field	Comment
Conditions (Details)	Currency of Condition	The currency of the condition is the same as that of the FRA legs.

> **Note**
>
> The system generates the cash flow for the forward rate agreement by generating cash flows for each FRA leg and then merging them on the parent node (FRA). It does this by using generation category 04 (*generated cash flow*) data sources, which applies the generation type (leg) to both FRA legs.
>
> For each FRA leg, you can use a generation type that reads the conditions by means of appropriate data source types, and reads the nominal value from a key figure, for example. In addition, a fixed interest rate from an imported business transaction can be taken into account for the variable side by means of a data source type of category 05 (*business transactions*).

4.3.13.1.3.15 Forward Transactions

The following forward transactions are supported:

- Currency Forward [page 623]
- Forward Security Transaction [page 621]

4.3.13.1.3.16 Forward Security Transaction

Definition

Template: S40SECURTR (Forward Security Transaction)

Use

You use template S40SECURTR to map security transactions in the Source Data Layer (SDL). Forward transactions are mapped as elementary templates of template category *Financial Transaction*. This object then references the associated security by means of the setting for the contract relationship.

The following tables contain the data required for forward stock transactions.

Templates for Forward Security Transactions

Template	Description	Template Category	Comment
S40SECURTR	Forward Security Transaction	Financial transaction	The forward security transaction is linked to the stock by means of the contract relationship *B is underlying for A*.
S41SHARE	Stock	Financial instrument	-

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S40SECURTR	-	GENSTR	FWD_ASSET
S41SHARE	-	GENINT	STOCK
S41BONDEX	-	GENINT	BOND

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
FWD_ASSET	FORWARD	1	Standard forward transaction

Structure

The following table contains the fields that the price calculator uses to value forward security transactions.

Risk Basis Fields in the SDL Template

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Purchase/Sale	By setting this indicator, you define whether the transaction is a purchase or a sale.
Object Relationships	Relationship Category	-
Object Relationships	Multiplicity	Number of units for stocks
Object Relationships	Ext. Target Object Number	Enter *Stock/Bond* as the ID of the target object.

Tab Page	Field	Comment
Key Figures	Category	To map the forward rate, you can store the following key figure categories: • 00001 (*Amount with Currency*) • 00002 (*Percentage*) • NMVAL (*Nominal values* are required for bonds, for example)

4.3.13.1.3.17 Foreign Exchange Transactions, Forward Exchange Transactions

Definition

Template: S40FXFORWD (Forex Transaction)

Use

Use

You use template S40FXFORWD to map foreign exchange and forward exchange transactions. In the Source Data Layer (SDL), these forwards are mapped as unstructured financial transactions that contain both the cash flows of the foreign exchange transaction. When this data is mapped to the valuation structure in Risk Basis, the forward exchange transaction is given a complex structure in which the forward transaction and the cash flows have their own separate valuation structures. The forward transaction and the cash flows also have their own transaction forms.

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S40FXFORWD	-	FWD_FX	FWD_FX

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
FWD_FX	STRCTRD PRODUCTS	1	Transaction can be priced as one entity
FX_LEG	CASH-FLOW DISC	-	-

Features

The table below contains the fields that the price calculator uses to price foreign exchange transactions.

Risk Basis Fields in the SDL Template

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Purchase/Sale	By setting this indicator, you define whether the transaction is a purchase or a sale.
Cash Flow (Header)	Cash Flow Category	-
Cash Flow (Header)	Cash Flow Valid From	-
Cash Flow (Header)	Cash Flow Valid To	-
Cash Flow (Item)	Flow Type	One of the flow types for the cash flows has to be a positive value and one has to be negative value, and they have to be mapped to cash flow indicator *5*.
Cash Flow (Item)	Amount/Currency	-
Cash Flow (Item)	Maturity	The price calculator uses this field only if no date was entered in the value date field. If you entered a date in the value date field, you do not need to enter any data in this field.
Cash Flow (Item)	Value Date	-

4.3.13.1.3.18 Securities Lending

Definition

Template: S40SECULEN (Securities Lending)

Use

You use template S40SECULEN to map securities lending transactions in the Source Data Layer (SDL). Securities lending transactions are mapped in the SDL as unstructured financial transactions, which are linked by means of the contract relationship to the security transferred to the counterparty, and to any collateral that is to be stored.

Templates for Securities Lending Transactions

Template	Description	Template Category	Comment
S40SECULEN	Securities Lending	Financial transaction	The securities lending transaction is linked to the stock or collateral by means of the contract relationship *B is an Underlying for A*.
COLLATERAL	Collateral	Financial transaction	-
S41SHARE	Stock	Financial instrument	-

> **Note**
>
> Note that you have to create the template SICHERHEIT yourself, since it is not part of the sample Customizing.

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S40SECULEN	-	GENINT	ASSET_LOAN

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
ASSET_LOAN	CASH-FLOW DISC	-	-

Structure

The following table contains the fields that the price calculator uses to value securities lending transactions. Only the cash flows are relevant for the valuation of securities lending transactions.

Risk Basis Fields in the SDL Template

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Purchase/Sale	By setting this indicator, you define whether the securities lending transaction is a purchase or a sale.
Cash Flow (Header)	Cash Flow Category	-
Cash Flow (Header)	Cash Flow Valid From	-
Cash Flow (Header)	Cash Flow Valid To	-
Cash Flow (Item)	Flow Type	-
Cash Flow (Item)	Amount/Currency	-
Cash Flow (Item)	Maturity	The price calculator uses this field only if no date was entered in the value date field. If you entered a date in the value date field, you do not need to enter any data in this field.
Cash Flow (Item)	Value Date	-

4.3.13.1.3.19 Real Repurchase Agreement

Definition

Template: S40REPO (Real Repurchase Agreement)

Use

You use template S40REPO to map real repurchase agreements and reverse repurchase agreements in the Source Data Layer (SDL). Real repurchase agreements are mapped as structured templates. Template S40REPO is the parent node, and two identical subtemplates for the security transactions are assigned to it as child nodes. Each of these security transactions contains a reference in the form of a contract relationship to the security that is transferred to the other party. The security could be a share, for example.

Template for Repurchase Agreements

Template	Description	Template Category	Comment
S40REPO	Real Repurchase Agreement	Financial transaction	The real repurchase agreement is linked to the security transactions by means of the template hierarchy.
S40SECURTR	Forward Security Transaction	Financial transaction	The forward security transaction is linked to the stock by means of the contract relationship *B is underlying for A*.
S41SHARE	Stock	Financial instrument	-

Cardinalities for Structured Templates

Template	Subtemplate	Minimum Cardinality	Maximum Cardinality
S40REPO	S40SECURTR	2	2

Mapping

Template	Classification in Risk Basis	Mapper	Transaction Form
S40REPO	-	GENSTR	REPO
S40SECURTR	-	GENSTR	FWD_ASSET
S41SHARE	-	GENINT	STOCK
S41BONDEX	-	GENINT	BOND

Settings in the Pricing Model

Transaction Form	Pricing Model	Configuration Value	Comment
REPO	STRCTRD PRODUCTS	2	Total of the parts that can be priced
FWD_ASSET	FORWARD	1	Standard forward transaction

Features

The table below contains the fields that the price calculator uses to price repurchase agreements. The fields required for the security transactions are not discussed in detail here. For a description of these fields, see Forward Security Transactions [page 621].

Fields in the SDL Template that are Relevant for Risk Basis for Repurchase Agreements

Tab Page	Field	Comment
Basic Data	Start of Term	-
Basic Data	End of Term	-
Basic Data	Purchase/Sale	By setting this indicator, you define whether the repurchase agreement is a purchase or a sale.

4.3.13.2 Fair Value Server

Use

This component supports you during the measurement of financial instruments, financial assets, and financial receivables under IAS 32 and IAS 39.

Integration

The following applications use the Fair Value Server:

- Accounting for Financial Products [page 678]
- Fair Value Effectiveness Test for Hedging Relationships [page 1036]
- Credit Exposure [page 1192]
- Fair Value Production Process [page 536]

The Fair Value Server calls the Price Calculator [page 547] in order to calculate present values.

> **Note**
>
> The system uses the pricing model *discounting of cash flows* to calculate the fair value key figures of cash flow transactions. Hence all cash flows that fall after the evaluation date are to be measured, and have to be available as input data for the Fair Value Server. The cash flows are determined by the cash flow generation method [page 440].
>
> If key figures are calculated using an effective interest rate or a spread, there may be even more requirements as regards the existing cash flows: If a spread or an effective interest rate needs to be updated then not only the cash flows up the evaluation date are required but also those dating back to the date of the last valuation. In the case of variable-rate transactions it may be necessary to use even later flows to fix interest rates.

Customizing settings

Besides the Customizing settings stored specifically for the Fair Value Server, the Fair Value Server also requires certain Customizing settings in the Price Calculator and in the calling applications.

- You enter the settings for the Fair Value Server in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server*.

- You enter the settings for the price calculator in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator*.
- You make the assignments for the subledger scenario in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Basic Settings* > *Assign Global Environments to Accounting Systems*.
- You make the assignments for the fair value effectiveness test in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness Test* > *Edit Environment for Effectiveness Test*.
- You make the assignments for credit exposure in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *General Settings* > *Enter Market Data Settings*.

Features

The Fair Value Server contains the following functions:

- **Calculation of the following key figures:**
  - Full fair value (FFV)
  - Fair value with initial credit spread (CFV)
  - Hedge fair value (HFV)
  - Amortized cost (AC)
  - Hedge amortized cost (HAC)
  - Hedge adjustment (HAD)
  - Effective capital over time (ECOT)

  For more information, see Calculation of the Fair Value [page 633] and Calculation of Amortized Cost [page 636].
- **Data Transfer**
  You can import key figures that were calculated externally to the Fair Value Server. For more information, see Data Transfer [page 576].
  The key figures *Fair Value for Free Line* (`FVFL`) and *Fair Value for Free Line, Start of Day* (`FVFL_EARLY`) must be imported because the Fair Value Server cannot calculate them.
- Calculation of auxiliary values for the key figure calculation
  The Fair Value Server usually receives all the parameters that are relevant for the valuation process from the calling application. Sometimes it has to recalculate the interest rates or spreads that it needs for the discounting of cash flows. This is the case for effective interest rates, hedge effective interest rates, financial instrument spreads (retail spreads), and for hedge spreads. For more information about how the parameters are calculated see Calculation of Effective Interest Rate and Spreads [page 641].
- **Use of exogenous spreads**
  The Fair Value Server can use exogenous spreads for each transaction. In the Customizing for the Fair Value Server, you can set up the derivation process for market data in such a way that the system calculates one or more exogenous spread curves for each transaction. It adds these spread curves to the base curve before it calls the price calculator for the evaluation.
- **Support for the cash flow splitting method**

In the calculation of the key figures, the Fair Value Server can use the following splitting levels for the cash flow splitting [page 518] method of cash flow refinement:
    - Syndication split
    - Disbursement split
    - Coverage split
    - Margin split
- Use of various market data sets
  As already described for the Price Calculator, you can derive a set of market data for each transaction.
- Overriding of the standard valuation
  As already described for the Price Calculator, you can override the standard valuation process for each transaction. You use the valuation rule [page 490] to do this.
- The *residual method* pricing model
  You can use the residual method pricing model [page 644] to price embedded derivatives. This is required if, for example, you are posting structured products that have optional components.
- Filtering cash flows
  You can use the cash flow filter to remove certain flows from the cash flow before financial transactions are measured. You define which flows the filter removes for each valuation rule. The system deletes these flows from the valuation structure, and does not include them in the evaluations.
- Processing the fair value level
  You can define how the system reacts if the expected fair value level from the feeder system is not the same as the fair value level calculated in the valuation. You can either specify that the system does not check this or specify whether a warning or error is displayed.
- Detail log [page 652]
  For more information, see Processes in the Fair Value Server [page 630].

> **Note**
>
> You can use programs `/BA1/R8_FVAC_SUBCON_EVAL` and `/BA1/R8_FVAC_HEDGE_EVAL` to test the Fair Value Server's valuations.

4.3.13.2.1 Processes in the Fair Value Server

Use

The Fair Value Server is called by the Bank Analyzer applications to calculate the following key figures:

- Fair value
- Fair value with initial credit spread
- Hedge fair value
- Amortized cost
- Hedge amortized cost
- Hedge adjustment
- Effective capital over time

The Fair Value Server is called in the background only. You cannot call the Fair Value Server directly. You can only call the Fair Value Server using the test programs `/BA1/R8_FVAC_SUBCON_EVAL` and `/BA1/R8_FVAC_HEDGE_EVAL`.

Process

The Fair Value Server processes data in the following steps:

1. Determination of the basic data:
   - Position data
   - Financial transaction or financial instrument data
   - Target values or spreads and effective interest rate

   The application that calls the Fair Value Server provides it with the required position data and parameters. This is done using Business Add-Ins (BAdIs). The position data also contains characteristics that are used to identify the relevant accounting objects or to define how key figures calculated externally are imported.

   > **Note**
   >
   > You can also use these characteristics to derive the valuation rule [page 490] in order to override the standard valuation process for particular transactions.

   Depending on which key figure is required, the Fair Value Server needs the following parameters:
   - Retail spread (to calculate the full fair value if you want to use spreads to calculate the fair value)
   - Initial credit spread (to calculate the fair value with the initial credit spread)
   - Hedge spread (to calculate the hedge fair value and the hedge adjustment)
   - Effective interest rate (to calculate the amortized cost)
   - Hedge effective interest rate (to calculate the hedge amortized cost and the hedge adjustment)
2. Formatting the data

   The Fair Value Server formats the data that is transferred:
   1. Selection of the financial transaction or financial instrument

      The Fair Value Server reads the data for the financial transaction or financial instrument from the Source Data Layer (SDL).
   2. Generation of cash flows

      The Fair Value Server calls the cash flow generation [page 440] function. It first checks whether it needs the entire cash flow to calculate the key figure in question, or whether a partial cash flow is sufficient. Partial cash flows are based on restart points. You use restart points to optimize system performance. In the standard system, the Fair Value Server calls partial cash flows because they contain enough information to calculate most key figures. To calculate certain key figures, such as the effective capital over time, the system needs the entire cash flow. If, in the course of calculating key figures, the system establishes that the information provided by the partial cash flow it has called up is not sufficient, cash flow generation is automatically repeated, this time for the entire cash flow.
   3. Mapping to the valuation structure

      The Fair Value Server transfers the data for the financial transaction or financial instrument to the valuation structure.
   4. Derivation of the valuation rule

      The Fair Value Server derives the valuation rule.

> **Note**
>
> You define how the valuation rule is derived in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivation* > *Determination of the Valuation Rule* > *Create Derivation Strategies*. You assign the rule to the global valuation environment by means of the valuation structure type. You do so in Customizing under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Structure Types*.

5. Determination of the subtransaction to be valued
   The Fair Value Server determines which subtransaction is to be calculated, and adjusts the valuation structure accordingly.
6. Calling the method schema
   The Fair Value Server calls a method schema to derive the market data set, for example. We provide a method schema for the Fair Value Server that contains the following methods for cash flow refinement:
   - Derivation of the Market Data Set
   - Break-down and calculation of options that are implied by interest rate formulas
   - Generation of hypothetical derivatives [page 515]
   - Cash flow fixing
   - Cash Flow Determination
   - Cash Flow Splitting

> **Note**
>
> You can create your own method schemas and methods for cash flow refinement in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas*.
>
> You define how the market data set is derived in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivation* > *Determination of Market Data Sets and Spread Curve Types* > *Create Derivation Strategies*. You assign the market data set to the global valuation environment by means of the valuation structure type. You do so in Customizing under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Structure Types*.

3. Calculation of spreads and the effective interest rate (if required)
   The Fair Value Server calculates the spreads and the (hedge) effective interest rate depending on the required key figures. This is required in the following cases:
   - A spread or effective interest rate has not yet been calculated.
   - The most recently calculated spread or effective interest rate is no longer valid because the parameters that influence this have changed. This means that the spread or effective interest rate has to be recalculated. The Fair Value Server receives the information about whether this is necessary from the caller.
   - Interest rates have been fixed since the spread or effective interest rate was last calculated. The spread or effective interest rate is recalculated depending on the Customizing settings.

   For more information, see Calculation of Effective Interest Rate and Spreads [page 641].
4. Calling the price calculator

To calculate the key figures, the Fair Value Server calls the price calculator [page 547]. Before it does so, the Fair Value Server adjusts the yield curve for the market rate for the required key figure by adding a spread to the yield curve for the market rate if required.

> **Note**
>
> You configure the price calculator in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *Edit Price Calculator Environments*.

For more information, see Calculation of the Fair Value [page 633] and Calculation of Amortized Cost [page 636].
It is possible to import NPV key figures that have already been calculated to the Fair Value Server. The system interprets the imported values as clean (without accrued interest). For specific key figures, you can define whether the system interprets them as dirty (net present value; with accrued interest). For more information, see Data Transfer [page 576].

5. Transfer of key figures
   The Fair Value Server returns the required key figures to the calling application. You can access a detail log [page 652] once the system has completed the measurement using the Fair Value Server.

   > **Note**
   >
   > The Fair Value Server only calculates key figures; it does not store them, or process them any further.

4.3.13.2.2 Valuation

The Fair Value Server contains the following functions, which can be used to value financial transactions, financial instruments, and subtemplates:

- Calculation of the Fair Value [page 633]
- Calculation of Amortized Cost [page 636]
- Calculation of the Effective Interest Rate and Spreads [page 641]
- The Residual Method Pricing Model [page 644]

4.3.13.2.2.1 Calculation of the Fair Value

Use

This function calculates the following key figures:

- Full fair value (or fair value)
  Required for financial reporting for IAS positions. The Fair Value Server calculates the fair value as a market price or as a theoretical price using customary market pricing models in line with the International Accounting Standards (IAS).

- Fair value with initial credit spread
  Required on the one hand for the financial reporting of securitized liabilities (own bonds) that have been designated to the *Fair Value* category (fair value option).
  On the other hand, for all assets and liabilities that are held in one of the fair value holding categories, FINREP requires that any fair value changes due to a change in credit rating be reported separately.
- Hedge fair value
  Required for financial reporting for IAS positions and to measure the effectiveness of hedging relationships. The hedge fair value describes the change in value that results solely from changes to hedged market parameters; this is currently only the interest rate risk.

Features

The Fair Value Server calls the price calculator [page 547] in Bank Analyzer. It can therefore use all of the pricing models that the price calculator contains:

- Data transfer
  The system is provided with the key figures *Full Fair Value* and *Hedge Fair Value* from external systems. For more information, see Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings*.
  You cannot import the *Fair Value with Initial Credit Spread* key figure from a feeder system.
- Mark-to-market [page 551]
  The system uses the market price of the financial instrument or financial transaction (for example, the market price of a stock).
- Mark-to-model
  The system calculates a theoretical price for the financial instrument or financial transaction, using market data, such as yield curves, volatilities, or spreads. The following valuation methods are available:
  - Cash flow discounting [page 553]
    The system discounts the future cash flows of the financial instrument or financial transaction for the evaluation date. This pricing model is useful only for transactions with cash flows.
    The Fair Value Server always returns the clean price.
    - Full fair value
      The system uses the market interest rate for discounting purposes. If an acquisition value is available for the position, the system adds the financial instrument spread to the market yield curve before it prices the position.
      You can measure the covered and uncovered parts of a financial transaction separately when calculating the full fair value by including a cash flow splitting method that has the splitting level coverage split [page 647].
    - Fair value with initial credit spread
      The fair value with initial credit spread can be calculated only using the *Mark-to-Model* pricing model. The Fair Value Server measures the fair value using the credit-risk-free market yield curve that is shifted by a constant credit spread. This initial credit spread is determined from the market value when the contract is created or initialized.
    - Hedge fair value
      The system uses the market interest rate plus the hedge spread to discount the cash flows.
      If the hedge fair value is required for transactions without cash flows, such as options, or for cash flow transactions used as hedging instruments, the system calculates the full fair value. For futures used as

hedging instruments in fair value hedges, the system uses the future price as the hedge amortized cost.

> **Note**
>
> If the calling application has not specified the required spreads, the Fair Value Server first calls the *cash flow discounting* pricing model to calculate the spreads. For more information, see Calculation of Effective Interest Rate and Spreads [page 641].
>
> The financial instrument spread, the hedge spread or the credit spread is always added to the basic curve after the yield curve has been created and the interest rates have been interpolated, and before the discount factor is calculated.

- Option pricing models [page 564]
  The system calculates the theoretical price of individual exotic options and of standard options with European and American exercise type.
- External pricing models
  You can define your own pricing models that you can use instead of or in addition to those provided by SAP. You do this by using Business Add-Ins (BAdIs). In Customizing, you can define your own implementations for these BAdIs. The external pricing models can use the market and transaction data provided by the SAP system. You can also call up external price calculators that use market and transaction data that is stored in a non-SAP system. An RFC interface is available for this purpose.
- Fair value level
  It is a requirement of IFRS 7 that a level (fair value level 1 to 3) is assigned to each financial position valued for the fair value. A source system provides the expected level for a product to Bank Analyzer. You can enter your own primary data source for reading fair value levels. If you do not, the system uses generic market data.

  You assign a fair value level to a pricing model in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *Assign Fair Value Level to Pricing Model*.
  You can define how the system reacts if the expected fair value level from the feeder system is not the same as the fair value level calculated in the valuation. You make these settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings*. In this Customizing activity, you also enter the primary data source for reading fair value levels.

Comments

- The system does not check whether the key figures and valuation results transferred from the external pricing model are IAS compliant.
- For detailed information about the pricing models provided by SAP, see Price Calculator [page 547].

4.3.13.2.2.2 Calculation of Amortized Cost

Use

In this function the system calculates the following key figures:

- **Amortized cost**
  Required for financial reporting for IAS positions. Under IAS 39, the acquisition costs of financial liabilities or assets are amortized using the effective interest rate. The amortized cost of the liabilities and assets does not reflect fluctuations in market rates.
- **Hedge amortized cost**
  Required for financial reporting for IAS positions and to measure the effectiveness of hedging relationships. In contrast to the *amortized cost* key figure, in the *hedge amortized cost* the values are amortized to the designation date of the hedging relationship. Market rate fluctuations that occur after the designation date are not considered.

Integration

- The Fair Value Server calls the price calculator [page 547] in Bank Analyzer.
- If the effective interest rate or the effective interest rate of the hedge was not imported into the Fair Value Server, the system first calls the function calculation of effective interest rate and spreads [page 641] before starting the valuation.
- You can import the key figures *amortized cost* and *hedge amortized cost* from non-SAP systems. For more information, see Data Transfer [page 576].

> **Note**
>
> The system does not check whether the measurements that are imported are IAS compliant.

Features

The key figures are either imported from non-SAP systems, or Bank Analyzer calculates them as follows using the price calculator and the pricing model *Cash Flow Discounting*:

- **Amortized cost**

The amortized cost is the effective capital minus accrued interest, discounted at the effective interest rate:

$$AC = \sum_{i=1}^{n} CF_i \cdot e^{-r \cdot (t_i - t_0)} - AI$$

Where r is the effective interest rate, AI is the accrued interest, and CF_i is the cash flows at time t_i. t_0 is the key date or evaluation date.
The Fair Value Server discounts only future cash flows:
During end-of-day valuation, the cash flows at time $t_i > t_0$. If the target value was provided using start-of-day logic (based on the enhanced value date concept for deferrals), during end-of-day valuation the system sets t_0 to t_0+1, in other words, to the following day.
The following applies to the start-of-day valuation: $t_i >= t_0$.

> **Note**
>
> Note that accrued interest is included in this calculation. By contrast, the business definition of amortized cost does not include accrued interest.

Amortized cost resulting from interest accumulation
The method of amortized cost calculation shown above is used if the effective interest rate has to be recalculated or if the system forces you to use discounting. In other cases, the amortized cost is determined by interest accumulation: The system uses the applicable effective interest rate and the most recently calculated amortized cost to add the unaccrued interest for all payments between the last calculation and the key date. The current effective capital can then be determined. The accrued interest is then adjusted.

- **Amortized cost per payment**
  The function calculates the amortized cost of the overall cash flow transaction and the amortized cost of the individual payments. These are cash flows discounted at the effective interest rate. The system discounts the accrued interest for the most recent cash flow only.
- **Hedge amortized cost**

The system calculates the hedge amortized cost for cash flow transactions by discounting all the cash flows that flow after the designation date using the effective interest rate of the hedge:

$$HAC = \sum_{i=1}^{n} CF_i \cdot e^{-r \cdot (t_i - t_0)} - AI$$

Where r is the effective interest rate of the hedge, AI is the accrued interest, and CF_i is the cash flows at time t_i. t_0 is the key date or evaluation date.

> **Note**
>
> For **transactions without cash flows,** such as options, the Fair Value Server calculates the full fair value at the designation date as the key figure *hedge amortized cost*.
>
> For **futures** used as hedging instruments in fair value hedging relationships, the system uses the future price as the hedge amortized cost.
>
> For **variable-rate cash-flow transactions**, a different method is used to calculate the amortized cost than those given above. You can influence this method by means of the Customizing settings. For more information, see Amortized Cost for Variable-Rate Financial Transactions [page 638].

4.3.13.2.2.2.1 Amortized Cost for Variable-Rate Financial Transactions

Use

In this function, the system calculates the *amortized cost* key figure for variable-rate cash-flow transactions, such as floaters and variable-rate loans.

The system contains two methods for amortizing these types of transaction; both methods are IAS-compliant. Which method is used depends upon whether the premiums and discounts are interest-rate induced, or arise due to the credit rating.

Prerequisites

In Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings*, you have defined

whether the system is to calculate the exact effective interest rate for the key date by taking into account all interest rate fixing dates.

In Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings*, you have defined which method is to be used to amortize which types of transactions. You define this in the *Amortization for 1st Fixing Date* area.

> **Note**
>
> By specifying the valuation rule [page 490] you define which method the system is to use to amortize the transactions. Note, however, that you can define how transactions are amortized at financial transaction level only. This means that you cannot use one method to amortize the premium of a financial transaction, and a different method to amortize the discount.

Features

You can use the following methods for the premiums and discounts of variable-rate transactions:

- **Straight-line amortization up to the first interest rate fixing date**
  Premiums and discounts are treated as being only interest-rate induced.
  The system amortizes the variable-rate cash-flow transactions to the first interest rate fixing date using the straight-line amortization method. In this method, the purchase price at the time of purchase is linked to the remaining debt as at the first interest rate fixing date. After this fixing date, the amortized cost up to the maturity of the financial transaction is equal to the remaining debt.

$$AC(t) = \begin{cases} 0 & t < T_0 \\ \frac{t - T_0}{T_1 - T_0} \cdot (Nom(t) - AV) + AV & T_0 \le t \le T_1 \\ Nom(t) & t > T_1 \end{cases}$$

  Where t is the key date, T_0 is the time point at which the transaction was acquired, T_i is the first interest rate fixing date, Nom(t) is the remaining debt on the key date and AV is the acquisition value.

  > **Note**
  >
  > The system uses the straight-line amortization method only to calculate the *amortized cost* key figure.
  >
  > Note the following additional requirements regarding the use of straight-line amortization up to the first interest rate fixing date. These requirements have to be met by the financial transactions and the data provided by Balance Analyzer:

  - The system processes only simple variable-rate financial transactions such as floaters and variable-rate loans. The system cannot process variable-rate annuity loans.
  - The remaining debt and the acquisition price must be in the same currency.

- The remaining debt must not change in the first interest-rate fixing period, which is the period up to the first interest rate fixing date.

- **Effective interest method (constant amortization until the maturity of the financial transaction) (default setting)**
  Premiums and discounts are assessed as being based on credit standing; any interest-driven premiums and discounts that occur between two interest rate fixing dates are not considered.
  The system amortizes variable-rate cash-flow transactions as fixed-rate transactions but it does not use forward rates to determine the future cash flows. Instead, it uses the reference interest rate that was last fixed.

$$AC = \sum_{i=1}^{n} CF_i \cdot e^{-r \cdot (t_i - t_0)} - AI$$

Where AI is the accrued interest, r is the effective interest rate, and CF_i is the cash flows at time t_i. t_0 is the key date or evaluation date.

The latest fixed reference interest rate is used because these transactions are amortized to their maturity, and not to the next interest rate fixing date. The effective interest rate does not change between two interest-rate-fixing periods as a result of continuing the fixed interest. If the transaction data does not contain the fixed reference interest rate, the system takes the reference interest rate from the market data for the evaluation date.

Calculation of the Exact Effective Interest Rate

In the effective interest rate method (constant amortization), you can define how the system updates the effective interest rate. If you use the default setting, the system updates the effective interest rate for the current key date by taking the latest effective interest rate.

For variable-rate financial transactions such as floaters, this can lead to inaccurate values in the calculation of the amortized cost and hedge amortized cost if the interest rate was fixed between the date on which the effective interest rate was last updated and the current key date. To prevent this, you can activate the calculation of the exact effective interest rate.

When the system calculates the exact effective interest rate, it updates the effective interest rate for all interest rate fixing dates that fall between the latest update of the effective interest rate and the key date. In this case, the effective interest rate calculated on the key date is usually more accurate than that calculated if you do not set the indicator.

> **Note**
>
> Note that if you set the indicator, the calculation process is complex, and can reduce system performance.

Example

To amortize a group of floaters to the first interest rate fixing date, define a valuation rule that is used for all of them, and in Customizing for the Fair Value Server for this valuation rule, activate amortization to the first interest rate fixing date.

4.3.13.2.2.3 Calculation of the Effective Interest Rate and Spreads

Use

This function calculates the following parameters for the Fair Value Server:

- **Effective Interest Rate**
  The effective interest rate is required to calculate the amortized cost and expresses the yield of a cash flow transaction. It considers nominal interest and any premiums or discounts for the entire transaction term.
- **Hedge Effective Interest Rate**
  The hedge effective interest rate is used to calculate the hedge amortized cost and is comparable to the effective interest rate. However, it is calculated only for financial instruments in a hedging relationship and is valid as of the designation date of the hedging relationship.
- **Financial Instrument Spread (Retail Spread)**
  The financial instrument spread is calculated for cash flow transactions that have to be priced using the full fair value and for which an acquisition value exists. It is not tied to a particular term, and is used to calibrate the yield curve for each financial instrument. The financial instrument spread can also be negative.
- **Credit Spread**
  The Fair Value Server calculates the credit spread for cash-flow-based products for which the *Fair Value with Initial Credit Spread* key figure is requested and for which a full fair value exists. The credit spread remains constant over the entire term, and reflects the initial credit risk of the cash flow instrument inherent in the market price using the credit-risk-free market yield curve.
- **Hedge Spread**
  The hedge spread is calculated when the hedging relationship is designated and if any conditions change. It is used to calculate the hedge fair value. It is used so that the valuation takes account of only those fluctuations in value that result from the hedged risk.

Integration

The system uses the cash flow discounting [page 553] pricing model in the price calculator to calculate the parameters. The pricing model needs the identifier of the financial instrument, financial transaction, or partial contract, the valuation key date, the cash flows that are to be considered, a yield curve, and target values in order to calculate the interest rates and spreads.

Subledger Scenario

Condition changes are not automatically taken into account in the subledger scenario. The required key figure is calculated for the current horizon only. A new SDL version is delivered and condition changes are included in a

business transaction. This business transaction must trigger a valuation that results in the adjustment of the target values.

Prerequisites

If the **default setting** is used, the Fair Value Server calculates the effective interest rate and the spreads for the current time stamp.

For the calculation of the amortized cost, **IAS 39 AG8** permits the use of the effective interest rate that was determined at the start of the term of the financial transaction on the basis of the expected cash flows. For this reason you can switch off the automatic update of the effective interest rate and the spreads. In Customizing for *Bank Analyzer* you have selected the area *Measurement as per IAS 38 AG8* and set the *Valuation as per AG8* checkbox under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Edit Valuation Settings*.

You switch off the automatic update of effective interest rates and spreads separately for each valuation rule [page 490]. You can trigger the update of effective interest rates manually using your own business transactions. For more information, see Amortization (Subledger Scenario) [page 890].

> **Caution**
>
> If you switch off the automatic update of the effective interest rate and spreads, then the system uses the effective interest rate that was originally calculated. This is also the case if you create new versions of a transaction.

You can define processing according to **IAS 39 AG7** in the same way as for AG8 in the Customizing activity *Edit Valuation Settings*. Under AG7, interest rate fixing for variable-rate transactions leads to an adjustment of the effective interest rate. The effective interest rate, however, remains unchanged for other changes to expected payments.

For more information, see Amortization, Valuation, and Hedge Adjustment (Subledger Scenario) [page 883].

Features

The system calculates the parameters as follows:

- **Effective Interest Rate**
  The effective interest rate is the interest rate that, during discounting, would result in a fair price for the financial instrument. The system solves the following equation iteratively for r:

$$WAP = \sum_{i=1}^{n} CF_i \cdot e^{-r \cdot (t_i - t_0)} - AI$$

Where WAP (=VAP) is the target value on the key date or evaluation date t_0, r is the effective interest rate and CF_i is the future cash flows of the financial instrument at time $t_i > t_0$. When the effective interest rate is first calculated on the disbursement date, the target value is the same as the acquisition or disbursement amount.

> **Note**
>
> To calculate the effective interest rate, the system uses continuous compounding and the day count convention act/365 (to determine the number of days t_i t_0).
>
> If the *Day Count Convention for Financial Transaction* checkbox is selected in Customizing, the system uses the day count convention that is defined for the financial transaction to be valuated. You make these settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings*. This setting only has an effect if you have set the *Value Date for Deferrals* indicator to "True" in the Customizing activity *Make Settings for Value Date for Deferrals*.

- **Hedge Effective Interest Rate**
  The hedge effective interest rate is used to price financial instruments that are in hedging relationships. It is the interest rate that, during discounting, would result in a fair price for the financial instrument on the designation date.
  The system calculates the hedge effective interest rate in the same way as the effective interest rate. However, the designation date of the hedging relationship is used as the valuation key date. The target value is the full fair value plus accrued interest for the designation date (that is, the dirty price).
- **Financial Instrument Spread (Retail Spread)**
  The financial instrument spread is the interest rate (which is not tied to a particular term) by which the yield curve has to be shifted so that the acquisition or disbursement amount results when the expected future cash flows results are discounted. The yield curve can comprise the basic curve and an exogenous spread.
  The system solves the following equation iteratively for RS:

$$A = \sum_{i=1}^{n} CF_i \cdot e^{-(r_i + s_i + RS)(t_i - t_0)}$$

Where the target value A is the acquisition or disbursement amount, CF_i is the future cash flows of the financial instrument, t_i is the times of the cash flows, r_i is the interest rate at time t_i and s_i is the exogenous spreads at time t_i. RS is the financial instrument spread (retail spread).
For transactions with cash flows that are split by covered and uncovered parts, the financial instrument spread is calculated as follows:

$$A = \sum_{i=1}^{n} CF_i^u \cdot e^{-(r_i + u_i + RS)\cdot(t_i - t_0)} + \sum_{j=1}^{m} CF_j^c \cdot e^{-(r_j + c_j + RS)(t_j - t_0)}$$

Where A is the acquisition or disbursement amount, CF^u_i or CF^c_j is the cash flows of the unhedged or hedged part of the financial instrument, t_i is the times of the cash flows, r_i is the interest rate at time t_i. u_i and c_j are the exogenous spreads of the unhedged or hedged part at time t_i. RS is the financial instrument spread (retail spread).

The system always uses the day count convention for the transaction to calculate the financial instrument spread.

- **Credit Spread**
  The credit spread is the interest rate that is the same for all maturities and by which the basic curve is shifted so that the target value is achieved when the future expected cash flows are discounted.
  The system solves the following equation iteratively for CS:

$$A = \sum_{i=1}^{n} CF_i \cdot e^{-(r_i+CS)\cdot(t_i-t_0)}$$

  Where CF_i is the future cash flows of the contract, t_i is the times of the cash flows, r_i is the interest rate at time t_i and CS is the credit spread. On the date on which the business transaction is concluded, the full fair value is used as target value A.
- **Hedge Spread**
  The hedge spread is the interest rate that is the same for all maturities and by which the basic curve is shifted so that the target value is achieved when the future expected cash flows are discounted.
  The system solves the following equation iteratively for HS:

$$A = \sum_{i=1}^{n} CF_i \cdot e^{-(r_i + HS)(t_i - t_0)}$$

  Where CF_i is the future cash flows of the financial instrument, t_i is the times of the cash flows, r_i is the interest rate at time t_i and HS is the hedge spread. The acquisition amount or disbursement amount is used as target value A for hedge designation when the transaction is concluded. If the hedge is designated after the transaction has been concluded, the full fair value at the time of the hedge designation is used as target value A; any interest that has accrued is not deducted here (that is, the dirty price is used).

4.3.13.2.2.4 The Residual Method Pricing Model

Use

You can use the residual method to price individual parts of structured products.

To determine the value of a particular part of the financial instrument or transaction, the residual method first calculates the present values of the remaining parts and of the overall financial instrument or contract. The residual method calculates the required value as the difference between the two values.

> **i** Note
>
> You can use the residual method pricing model to price embedded derivatives, for example. This pricing model can be used in the Accounting Processes for the key date valuation.

Integration

The residual method is a method in the Fair Value Server. This method calls the Fair Value Server recursively. When the system measures the transactions, it calls the method schema [page 534] and the price calculator [page 547].

Prerequisites

To use the residual method, you need to make settings in Customizing for the *Fair Value Server*. You define the valuation rules for which the system is to use the residual method in Customizing for *Bank Analyzer* by choosing *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings*.

Example

- **Warrant Bonds**
  You can use the residual method pricing model to price the warrant part of a warrant bond, for example. You might want to do so if the subinstruments or partial contracts can be separated and market prices are available for the warrant bond, the warrant, and the bond. To ensure that the warrant bond and its component parts are priced consistently, you can use the residual method pricing model to calculate the value of the warrant as the difference between the values for the warrant bond and the bond (ex-dividend).
- Loans with Rights of Notice [page 645]

4.3.13.2.2.4.1 Loans with Notice or Repayment Options

Definition

Loans can include notice and repayment rights, which are legally required or agreed upon in the contract. For example, in Germany the debtor is allowed to give 6 months notice on a fixed-interest loan after 10 years (German Civil Code § 609 (1), Paragraph 3).

Use

Loans with notice and repayment options are financial instruments, which you enter as templates in the Source Data Layer [page 100] (SDL). You can also import them into the SDL from external systems. These loans are measured and analyzed in Accounting Processes [page 693].

Structure

Loans with notice and repayment options are stored in the Fair Value Server as structured products. The root node represents the whole product. Directly below the root node, there are subnodes that contain the loan without the options, and other subnodes that contain the individual options.

Integration

The price calculator [page 547] is not a separate pricing model for pricing loans with notice and repayment rights. Instead, the system uses the cash flow discounting pricing model [page 553] to measure these products. This pricing model can be used only if the options contained in the template have been broken down into their component parts. The cash flow determination method is used for this purpose. This method is one of the cash flow refinement methods in Bank Analyzer. The cash flows are determined before the product is measured, rather than in the price calculator.

When determining the cash flows, the system removes the options from the valuation structure, and replaces these options with cash flows that map the development of the financial instrument as realistically as possible. After the cash flows have been determined, the valuation structure of the price calculator contains only those cash flows that can be priced using the *Cash Flow Discounting* pricing model.

You define how the cash flow determination function handles options in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Basic Settings for Methods* > *Edit Cash Flow Determination Method*. The cash flow determination method uses valuation rules and is transferred to the calling application using the Customizing concept *method environment*.

> **Note**
>
> In IAS accounting, some of the notice and repayment options are priced as embedded options. For this reason, the loan without options and the embedded options have to be disclosed separately in certain circumstances. The Fair Value Server uses the residual method pricing model [page 644] to price the options.

Example

The Fair Value Server is to be used to price a loan that can be called after 5 years.

To prepare for the measurement, in Customizing for the cash flow refinement methods you have defined that the price calculator is to be used to measure the call option. You have saved this setting under a cash flow determination method ID, which you have assigned to the method schema used by the Fair Value Server. As an alternative to using the price calculator to price the option, you could have specified a due date scenario or a percentage in the cash flow determination ID that states the probability of the option being exercised.

The Fair Value Server prices the loan as follows:

1. The Fair Value Server receives the IAS position ID from the Accounting Processes, and selects the loan from the SDL.

2. The Fair Value Server calls a function module that maps the loan to the valuation structure.
3. The Fair Value Server calls up the cash flow determination method.
   The cash flow determination method calls the price calculator to price the option. The price calculator uses the usual option pricing model for this purpose. The system then deducts the present value of the option (as calculated by the price calculator) from the nominal volume of the loan, and removes the option parts from the valuation structure.
4. The Fair Value Server calls the price calculator [page 547] to price the loan without the option.
5. The Fair Value Server returns the value of the loan with the notice right to the Accounting Processes.

4.3.13.2.2.5 Splitting of Financial Transactions in the Calculation of Key Figures

Use

You use this function to include the split parts of financial transactions in the calculation of the key figures *Full Fair Value*, *Amortized Cost*, and *Effective Capital Over Time*.

- **Full fair value**
  Here the system takes the same target value for the split parts of the transaction, and the same financial instrument spread, but it uses different exogenous spreads.
- **Amortized cost/effective capital over time**
  The system can derive different market data for the split parts of financial transactions, and different exogenous spreads can be used. To calculate the effective interest rate, the system aggregates the split parts and nets them off with the target value, which is the value for the product as a whole (not split).

> **Note**
>
> For more information, see Fair Value Calculation [page 633] and Calculating Effectiveness Interest Rates and Spreads [page 641].

Integration

To prepare for the measurement of the split parts of transactions, the system calls the cash flow splitting [page 518] method for cash flow refinement. The *cash flow splitting* method for cash flow refinement generates for the Fair Value Server one valuation structure for the each part of the financial transaction.

Prerequisites

- You need to have already entered the characteristic for splitting relevance, /BA1/C63SPLITR, in the attributes of the financial transaction, and set the characteristic to *relevant*. To enter these settings, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Transaction* > *Change Financial Transaction*.

- You need to have already created a position of the position class *Cash Flow Splitting*, and stored the splitting key figures in this position. The position account must be the same as the financial transaction account (category 02 *Settlement Account*). Splitting key figures are characteristics that contain the series of characters &63B. To create a position, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Position* > *Create Position*.
  Alternatively, you can store the splitting indicators of the secondary data source in Accounting Processes [page 693] (subledger scenario) in the complex key figure that you use.
- You need to have created a splitting environment in Customizing for *Bank Analyzer*, and entered the splitting values as required. You create splitting environments in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Edit Splitting Environments*.
- In Customizing for *Bank Analyzer* under *Processes and Methods* > *General Valuation and Calculation Functions* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Structure Types*, you have assigned the position class and the splitting environment to the Fair Value Server.

4.3.13.2.3 Data Transfer

Use

You can use this function to import present value key figures from source systems into the Fair Value Server. You have the following options:

- **Import of Key Figures Using Primary Data Sources**
  The system uses primary data sources to read present value key figures from the Source Data Layer (SDL). Present value key figures can be stored as generic market data or as positions.
- **Import of Key Figures Using Secondary Data Sources**
  The system reads present value key figures from a key figure that was already calculated before the price calculator was called. The key figure is assigned its value in a secondary data source within Bank Analyzer.
- **Import of Data into the Results Data Layer (RDL)**
  The system uses a results data area, result type, and the name of the key figure to read the present value key figures from the RDL.

Integration

This function is designed so that the price calculator reads the imported key figures, When the Fair Value Server calls the price calculator, the Fair Value Server tells the Price Calculator where it can find the key figures.

In order to be able to import present value key figures into the Fair Value Server, you need to have already entered the Customizing settings for the Fair Value Server and for the price calculator. If you want to do so by using secondary data sources, then you also need to enter Customizing settings for the application that the Fair Value Server calls:

- **Price Calculator**
  In Customizing for the *price calculator* you use the *Transfer Type* field to define how present value key figures are to be imported.

In the *Settings for the Price Calculator Environment*, you can also use the valuation rule to define for which financial transactions and financial instruments the key figures are to be transferred.
You must set the valuation rule for fair value for the free line and for the free line at the start of day to “Transfer Only”.

You configure the price calculator in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *Edit Price Calculator Environments*.

> **Note**
>
> For more information about using valuation rules to define how transactions are priced, see Valuation Rules [page 490]. The system derives a valuation rule for each financial transaction according to the settings made in Customizing. The system can also derive a valuation rule for each key figure of the Fair Value Server (for example, fair value of the free line for HFS loans).

> **Note**
>
> The price calculator does not scale the imported key figures. Therefore, you need to make sure that the key figures for financial instruments are already normed before they are transferred to the position. For a stock position, it is not the stock price, for example, but the value of the entire position that is to be transferred.
>
> In Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings* you can enter a scaling key figure that the system uses to norm imported key figures to the position in Bank Analyzer.
>
> You can use the scaling key figure only if you use a primary data source or the Results Data Layer to import key figures.

- **Fair Value Server**
  In Customizing for the *Fair Value Server* specify which present value figures you want to import. You also enter the information about where the present value key figures are stored that you want to import. When the Fair Value Server calls the price calculator, it tells the system where it can find the imported present value key figures.
  For the following key figures you can define whether the system interprets the imported values as clean (without accrued interest) or dirty (net present value; with accrued interest):
  - Full Fair Value
  - Full Fair Value, Start of Day
  - Hedge Fair Value
  - Hedge Fair Value, Start of Day

  All other key figures are always interpreted as clean values.
  You enter the settings for the Fair Value Server in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings*.
- **Bank Analyzer Application that Calls the Fair Value Server**
  If you use a secondary data source to import present value key figures, then you need to set up the secondary data source so that it is compatible with your settings in the Fair Value Server. In the secondary data source, the system has to read the present value key figures in question from the SDL. You have imported the present

value key figures to the key figures in the valuation settings, and entered them in the calculation bases for the Fair Value Server. You use the Business Add-In *BAdI: Determine Calculation Basis* to do this.

Prerequisites

Import of Key Figures Using Primary Data Sources

You need to have already imported the present value key figures that were calculated externally into the SDL, and stored them in generic market data classes or SDL positions.

- Generic Market Data Classes
  You have created a generic market data class in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Generic Market Data* > *Edit Generic Market Data Classes*.
  For each generic market data class, you enter defining characteristics, which are later used as a key when the system reads the data. Therefore, you need to make sure that the defining characteristics exist in the transaction or position that is to be valued.
  Once you have created a generic market data class, the system automatically generates a primary data source that enables you to select the data from the generic market data class. The primary data source is always a *general primary data source*.
  You can store up to five values in each generic master data class.
  - To store values manually in generic market data classes, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Generic Market Data* > *Edit Generic Market Data*.
  - If you want to import values automatically, use the *Create* method in the *GenericMarketData* BAPI.

  For more information, see Generic Market Data [page 254].
- SDL Positions
  You need to have already created an SDL position, and stored in it the values that are to be transferred. On the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Position* > *Create Position*. When you create positions, you always set up a reference to a position class.
  Note that unlike generic market data classes, primary data sources for reading data from SDL positions are not generated automatically. To be able to select SDL positions, you have to create a primary data source of the category *object transfer structure* (OTS). You do this in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Object Transfer Structure (OTS)* > *Edit OTS for Primary Objects* > *Edit OTS for Positions*.
  You can store any number of values in SDL positions.
  - To store values manually in the SDL positions, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Position* and then either *Create Position* or *Change Position*, as required.
  - If you want to transfer values automatically, use the *AddNewVersion* method in the *PositionBA* BAPI.

  For more information, see Primary Object: Position [page 142].

> **Note**
>
> When you configure the import of key figures, it does not matter whether you store the key figures in generic market data classes or in SDL positions. You need only to specify the relevant primary data source and the name under which the key figure is to be stored. You need to provide the system with this information, as you can store multiple key figures for one SDL position or one generic market data class.

Import of Key Figures Using Secondary Data Sources

You have defined the secondary data sources for import of present value key figures so that the system can determine the present value key figures. You have imported the present value key figures to the key figures in the valuation settings, and entered them in the calculation bases for the Fair Value Server. You use the Business Add-In *BAdI: Determine Calculation Basis* to do this.

> **Note**
>
> Make sure that this process is complete before the price calculator is called.

Transfer of Present Value Key Figures By Reading Data from the RDL

The RDL must contain the relevant present value key figures. For more information about the Results Data Layer, see Results Data Layer (FS-BA-RD) [page 1364].

Features

You can transfer the following present value key figures:

- Full fair value (FFV)
- Amortized cost (AC)
- Fair value for free line (FVFL)
- Hedge fair value (HFV)
- Hedge amortized cost (HAC)
- Hedge adjustment (HAD)
- Effective capital over time (ECOT)
- Remaining debt
- Accrued interest

> **Note**
>
> The fair value level is imported together with the present value key figures.

Import of Present Value Key Figures for the Valuation of Business Transactions

The system usually measures financial transactions not only in the key date valuation in Accounting Processes, but it also measures particular business transactions for a day. To support this process for the imported present value key figures (*Full Fair Value*, *Amortized Cost*, *Fair Value for Free Line*, *Hedge Fair Value*, *Hedge Amortized Cost*, and *Hedge Adjustment*), you can define two values for each of these key figures:

- One value for the calculation at the end of the day (key figures `FFV`, `AC`, `FVFL`, `HFV`, `HAC` and `HAD`).
- One value for the start of the day, before the business transactions were posted (key figures: `FFV_EARLY`, `AC_EARLY`, `FVFL_EARLY`, `HFV_EARLY`, `HAC_EARLY`, `HAD_EARLY`)

To price the financial instruments for the individual business transactions, you have to set up the Fair Value Server for the import of values via a secondary data source. The system provides the secondary data source with information about the position and the scaling for the individual business transactions. You store the normed key figures in the generic market data or positions.

> **Note**
>
> For more information, see the documentation in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings*.

Consistency Check for Imported Key Figures

You can use the detail log [page 652] to check the key figures that were imported. The detail log gives you an overview of each step in the calculation process in the Fair Value Server. If there are errors in the key figures that were imported, this may be because the valuation results were incorrect. For example, when you import key figures, you need to make sure that the system treats the full fair value as a clean price.

You can use the detail log in analyses to call the Fair Value Server.

4.3.13.2.4 Detail Log

Definition

Detailed information about the valuations in the Fair Value Server (FVS).

Use

You use the detail log to see how the program calculated values in the Fair Value Server [page 628]. During the valuation process, the Fair Value Server, the cash flow refinement methods called by the method schema, and the price calculator write the main information about their processing steps to the detail log. When the valuation process is complete, you can call up the detail log from the reporting function of the respective application, such as the reporting function for the fair value effectiveness test.

When you start online reports, the system also records the processing steps of the derivation tool in the detail log. It logs the derivation of the market data set and valuation rule [page 490]. For performance reasons, this function is not available for reports run in batch processing.

You can display the detail log in text format, HTML format, or PDF format. You can then print the file or store it outside of the SAP system for documentation purposes. If the fair value effectiveness test is run as a batch job, the system saves the detail log along with the results of the run.

Structure

The system creates the detail log in a tree structure that contains multiple levels. These levels contain, for instance, the object ID of the financial transaction or instrument that was analyzed, the market data scenario that was used, and the horizon or evaluation date. You can display the results of the evaluation for any level in this hierarchy.

Hierarchy Levels

When you call the detail log, in the left-hand part of the screen the system displays the uppermost node in the structure. To display the individual levels in the hierarchy of the detail log, open the uppermost node, or the folder just below the respective level.

If you branch to the detail log from the reporting function for the fair value effectiveness test [page 1036], the system displays the following hierarchy levels, for example:

Hierarchy Level	Values and Comments
Hedging relationship ID	Identifier of the hedging relationship that was analyzed
Object ID	Identifier of the financial transaction or instrument that was analyzed
Key figure	• Hedge amortized cost (HAC) • Hedge fair value (HFV)
Application in the FVS	• Detail display • Display of final results • Version management (of target values)
Valuation method	• Standard valuation • Residual method
Horizon date	Horizon of the valuation, or the evaluation date
Market data scenario	Scenario that was used for the analysis

Valuation Results

In order to display the results for a particular hierarchy level, in the navigation structure choose a folder by double clicking. In the right-hand part of the screen the system displays the valuation results for that particular hierarchy level. If you double click on the uppermost node in the navigation structure, the system displays the entire detail log. If you select the nodes on lowest level of the hierarchy, the system displays the relevant section of the detail log for each node. The lowest nodes in the hierarchy level are individual documents. If you choose a node on a higher hierarchy level, the system aggregates the information contained in the documents below this level.

If the documents are large, we recommend that you use the text version or the PDF version of the detail log. To do so, select the relevant file and in the context menu choose the option *Display Log as Text* or *Display Log as PDF*.

The detail log is divided into sections that contain information about the underlying Customizing settings, the transaction data and market data, and the subtotals of the calculation. The following table contains the main sections. Since the Fair Value Server normally calls the price calculator more than once, each section can occur more than once in the detail log.

Section in the Detail Log	Information
Calculated Key Figure	• Key figure name (For the full fair value, the system displays *FFV*, for example.) • Valuation Rules [page 490] (If you use cash flow splitting, the system displays the valuation rule for the covered and uncovered parts.)
Target Value	• Target value • Validity date of the target value
Market Data Segment	• Market data area • Market data set • Securities type and category • Yield curve type and category • Exchange rate category
Transaction Data	• Transaction form • Object ID • Security ID number
Valuation	• Pricing model • Price calculator environment • Horizon date • Forward date • Market data scenario
Foreign Currency Translation	• Amount in base currency • Amount in target currency • Exch.Rate Cat. • Quotation type • Rate • Fixing indicator • Validity date
Calculation of Effective Interest Rate and Spread	• Cash flow ID • Cash flow indicator • Validity period • Number of days in validity period • Interest rate • Nominal amount • Cash flow amount • Interest rate fixing date • Spread or effective interest rate

Section in the Detail Log	Information
Calculation of Accrued Interest	• Key date • Cash flow ID • Cash flow indicator • Validity period • Number of days in validity period • Interest rate • Nominal amount • Cash flow amount • Interest rate fixing date • Calculation method • Accrued interest
Discounting of Cash Flows	• Cash flow ID • Date of the cash flow • Cash flow amount • Present value of the cash flow • Discount factor of the cash flow • Sum of the present values of the discounted cash flows

Note the following when you interpret a step in the detail log that belongs to a catch-up adjustment:

> **Note**
>
> The system behavior described below occurs if the value 'Y' (Target Value at Start of Day (New Logic)) has been set for the `I_FLG_EARLY_VAP` (Target Value at Start of Day) indicator. In addition, the `X_NPV_FIX_EIR` (Fix Effective Interest Rate) checkbox has been set to indicate that it is a catch-up adjustment calculation. The system provides both settings as part of the calculation base data.

The value AC_{early} (Amortized Cost at Start of Day) has a different meaning than usual in this context. The value AC_{early} is a new target value here that is required for determining the catch-up adjustment. For the amortization, the system discounts all payments to the start of the day; for the cash flow discount it does not take any payments into account on this day. The new target value is displayed after the *Discounting of Cash Flows* section, and copied to the AC_{early} field at the end of the detail log.

Action

You can change the order of the levels in the hierarchy by using Drag&Drop in the navigation structure, and so change how the valuation results are sorted. You could, for example, place those characteristics that are of particular relevance at the start of the hierarchy. This would allow you to display the valuation results for these characteristics all together, and to sort the valuation results by market data scenario, for instance.

You can use the following functions.

Action	Function
	The system saves the detail log locally on your PC. The file is in HTML format.
	The system displays the market data that it used in the calculation process.
	The system prints the detail log displayed.
	The system refreshes the hierarchy displayed in the navigation structure. You use this function if you have used Drag&Drop to change the order of the levels in the hierarchy.
	The system closes the hierarchy tree in the navigation structure, and only the uppermost node is displayed.
	The system opens the next level of the hierarchy tree in the navigation structure.
	The system displays the detail log for the selected node in HTML format.
	The system displays the detail log for the selected node in text format.
	The system displays the detail log for the selected node in PDF format.
Using Drag&Drop to change the hierarchy	You can use the Drag&Drop function to change the order in which individual levels appear in the hierarchy. You do so by choosing a level in the display at the top of the navigation structure, and dragging it to the required place. The system displays the new position of the level by highlighting it in red. To update this in the navigation structure, choose  .

4.3.14 Accrual Methods (FS-BA-PM-GM-AC)

Use

You can use this component to read accrual results supplied from Results Data Layer (RDL), or to calculate accrued items.

Integration

Accruals are calculated in the accounting processes in Accounting for Financial Products. Accounting for Financial Products uses the accruals to assign profits and losses to the correct posting periods. The system differentiates between accruals and deferrals.

The *Accrual Methods* component in the Fair Value Server receives the cash flows used to calculate accruals.

Features

The component can read accrual data transferred from Results Data Layer (RDL).

The component can also calculate the following types of accrual for cash-flow-based financial products:

- Accruals
  If the cash flow is in the new fiscal year even though it belongs (at least proportionately) to the old fiscal year from the point of view of profit, the cash flow has to be accrued. This is called 'accrual' as the profit of the cash flow is anticipated in the old fiscal year although the payment transaction has not yet occurred.
- Deferrals
  If the cash flow is in the old fiscal year even though it belongs (at least proportionately) to the new fiscal year from the point of view of profit, the cash flow has to be deferred. This is called 'deferral' because the cash flow is preliminarily posted in the old fiscal year until the profit is made in the new fiscal year.
- Time effect of contractual cash flow (CCF)
  The system determines the time effect of the contractual cash flow for a period as follows: The accrual at the start of the period is subtracted from the accrual at the end of the period. All accruals in the period are added to this amount.
  You can calculate the time effect of the CCF using the straight-line method only. You cannot import data.

You can use the following methods to calculate accruals:

- Straight line accrual
  Here, the amount to be accrued is distributed linearly across the accrual period and is proportionate to the calculation key date.
- Accrual weighted by the outstanding principal amount
  Here, the amount to be accrued is weighted by the outstanding principal amount for the accrual period, and up to the calculation key date.
- Effective interest method
  Here, the accrual amount is the difference between the effective capital at the beginning of the period and the effective capital on the calculation key date.

You can also create additional accrual methods in Customizing under *BAdI: Calculation Methods for Accruals*.

> **Note**
>
> You can use the program `/BA1/RAL_M3_ACCRUALS_TEST` to test the calculation of accruals.

Constraints

The system calculates accruals only for products with cash flows.

4.3.15 Book Value Determination

Use

You can use the book value determination function to calculate the book value of a financial position for defined book value categories according to accounting system. The system can also read the following values and can transfer the results to the calling application:

- Free line
- Balance offsetting amount
- Scaling factors (position nominal or position amount of financial instruments)

The book value determination can determine book values for aggregation objects that were created by source data aggregation. Although there is no book value available for individual financial transactions of an aggregation object, the book value determination function can read balances of individual transactions from the Source Data Layer and return them to the calling process (for example, calculation of the percentage expected loss).

Integration

Book value determination belongs to the application component *Accounting Methods* (FS-BA-PM-GM-AM) and is called by various SAP Bank Analyzer applications.

For example, the Impairment Server needs the free line to calculate the risk provision for the off-balance-sheet part of a receivable, and the balance of the book value components before valuation to calculate the risk provision for the on-balance-sheet part. The book value determination provides these values, and based on these the Impairment Server executes the calculations.

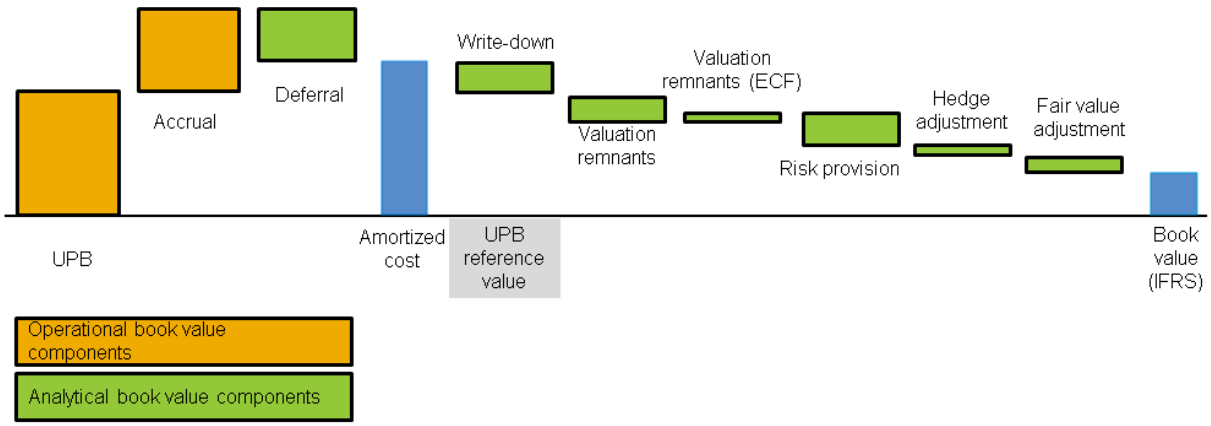
Prerequisites

You have carried out the following Customizing activities:

- *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Define Accounting Systems*
- *Processes and Methods* > *Accounting for Financial Products* > *Before Generation* > *Financial Positions* > *Define Financial Positions Types*
- *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Processing Categories* > *Assign Key Figures to Processing Categories*

Features

The book value according to IFRS is composed of **book value components** as illustrated in the following figure:



The unpaid principal balance (UPB) and accruals/deferrals are operational book value components; all other book value components are analytical.

Each book value component is assigned to a **book value category** with a sequence number. You can view the standard sequence of the book value categories in Customizing for book value determination:

Book Value Component	Book Value Category (ID)	Name of Book Value Category	Book Value Category Number
UPB	UPB	Book Value Components Before UPB	100
Accruals	ACC	Book Value Components Before Accrual	200
Deferrals	DEF	Book Value Components Before Deferral	300
Write-down	VR	Book Value Components Before Valuation Remnants	400
Valuation remnants	VR	Book Value Components Before Valuation Remnants	400
Valuation remnants (ECF)	VRE	Book Value Components Before Valuation Remnants (Expected Cash Flow)	450
Risk Provision	VAL	Book Value Components Before Valuation	500

Book Value Component	Book Value Category (ID)	Name of Book Value Category	Book Value Category Number
Hedge adjustment	VAL	Book Value Components Before Valuation	500
Fair value adjustment	FFV	Book Value Components Before Fair Value	700
--	WD	Book Value Components Before Write-Down	1

The book value as the total of the book value components is assigned to the book value category `BV` with the number 900.

The book value category is the unit with which the calling application requests the book value determination. It determines which book value components are balanced for the book value determination. For this, the system includes all book value components that are assigned to a book value category with a lower number than the number of the requested book value category.

Example

The balance for determining the deferral is requested with the book value category `DEF` (no. 300). In this case, the book value determination returns the balance of the UPB and the accrual to the caller because both of these book value components are assigned to the book value categories `UPB` (no. 100) and `ACC` (no. 200).

Note

Write-down is the exception in the determination of the balance of book value components: The reference value for the write-down is the UPB only, not the balance of UPB, accrual and deferral. The write-down is therefore requested with the separate book value category `WD` to which no processing categories can be assigned.

The book value category `UPB` (book value component *UPB*) cannot be requested because no corresponding balance can be determined.

Activities

The system determines the book value and the amount of individual book value components from the values of the key figures of selected processing categories. The processing categories `1BF_RECPAY` and `1OF_RECPAY` (Receivables/Acquisition Value) are always included. The system also includes the values of the key figures that you have defined using the processing categories in Customizing.

To do so, you make the following settings in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Book Value Determination*.

Assign Processing Categories to Book Value Category

You assign one or more processing categories to a book value category here. A processing category can be assigned to one book value category only.

Example

The book value category `ACC` (Accrual) is assigned to the processing category `1BF_ACCR`.

Caution

In general, no changes to the predefined assignment are required. If you need a customer-specific book value because you are using a customer enhancement, use book value category `BVC` (no. 950). In this case, you assign the processing categories to the book value category `BV` (no. 900).

We do not recommend that you create and assign customer-specific processing categories.

Make Settings for Book Value Determination

Here you define according to accounting system which key figure values the system uses for the book value determination. The system includes key figures that are the result of the combination of the following parameters:

- Financial position type (optional)
  For entries without a financial position type, the combination of accounting system, processing category and key figure applies for transactions of all financial position types. However, if you create an entry with financial position type, the system does take any other entries without a financial position type into account for transactions of this financial position type.
- Processing category (mandatory)
  Processing categories are assigned to a given accounting system The system derives the corresponding book value category and its number from the processing category.
- Key figure (optional)
  If you specify a key figure, the system includes all key figures of the selected processing category.

Example

You only want to include the key figure `_KAIMRP` (Impairment: Risk Provision (Default Risk)) of processing category `2BF_IMPRP` ([A] Risk Provision). In this case, you create one entry for each accounting system with the processing category `2BF_IMPRP` and key figure `_KAIMRP`.

4.3.16 Impairment Server

Use

The Impairment Server belongs to the *Impairment Methods* component (FS-BA-PM-GM-IM). It determines the write-down, the risk provision (after write-down) and the unwinding for individual contracts and transfers these impairment results to the calling applications. The following applications use the results from the Impairment Server:

- Update of Secondary Business Transactions [page 786]
- Key Date Valuation [page 934]
- Simulation of a risk provision calculation from a UI using enterprise services
- Custom risk reporting

The Impairment Server calls *BAdI: Calculation of Impairment Results*, for which SAP provides the following standard implementations:

- **Calculation Approach: Percentage Expected Loss (%EL Approach)**
  This approach determines the impairment results based on the percentage expected loss (%EL).

  > **Note**
  >
  > If you use **source data aggregation**, the system always derives the %EL approach for aggregation objects. In this case the Business Add-In (BAdI) reads the results for the aggregation of expected loss amounts and forwards these to the calling applications. For more information, see Aggregate Key Date Results [page 322].

- **Calculation Approach: Lifetime Expected Loss**
  This approach determines the impairment results based on the percentage expected loss, the exposure amount (EAD) and the effective interest rate. While in the %EL approach the system analyzes the expected loss over a one-year period, here the Impairment Server calculates the expected losses over the entire lifetime of the contract.
- **Calculation Approach: Expected Cash Flow (ECF Approach)**
  This approach determines the impairment results based on the expected cash flow (ECF).
- **Calculation Approach: Write-Down to Collateral Value (WDA Approach)**
  In the event of a foreclosure, for example, this approach determines the amount of the write-down to the fair value as the difference between the exposure amount and the net fair value of the collateral (minus the costs).

Implementation Considerations

The following **prerequisites** apply for implementing the Impairment Server:

- You have made the settings required in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Impairment Methods*. The following parameters and settings are relevant for the Impairment Server:
  - Impairment Server Environment
    Groups together all the settings that the system needs to calculate impairment events.
  - Risk Type
    Classifies the risk in risk provision calculation (for example, counterparty risk or transfer risk). The Impairment Server is called for each risk type.
  - Impairment Exposure Category
    Controls whether risk provision is to be calculated for the on-balance-sheet or off-balance-sheet part of a contract.
  - Impairment Calculation Approach
    The *Determine Impairment Attributes* process derives the calculation approach to be used. You can set the calculation approach manually in an external workplace (UI).
    If you want to use custom approaches in addition to the calculation approaches provided by SAP, you need to create the relevant implementations of *BAdI: Calculation of Impairment Results*.
- You have defined for which products you want the system to calculate impairment. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Impairment Processes* > *General Settings* > *Define Impairment-Relevant Products*.

For products that are not defined as impairment-relevant, the Impairment Server always returns the result “0”. The system checks the impairment-relevance before it calls the BAdI for the calculation of impairment results. Therefore, you cannot override this check.

- Impairment attributes are available in the Results Data Layer (RDL). You can either import these using an enterprise service or determine them using the *Determine Impairment Attributes* process.
- Exposure Category “On-Balance-Sheet”
  - To determine the write-down for the on-balance-sheet part of a contract, you have supplied the balance of the book value components for the determination of the write-down to the Impairment Server.
  - To determine the risk provision for the on-balance-sheet part of a contract, you have supplied the balance of the book value components for the determination of the risk provision to the Impairment Server.

    Example

    You can use book value determination to determine the respective balance of the book value components, for example. You make the settings required in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Book Value Determination* >.For more information, see Book Value Determination [page 658].

    Note

    For transactions that are to be offset, the balance offsetting amount is also included in the balance of book value components for the determination of risk provision.

  When you simulate a risk provision calculation, enter the balance of book value components for the determination of risk provision, the expected cash flow, the write-down amount, and the effective interest rate using an external workplace (UI). You deliver these values to the Impairment Service using enterprise services.
- Exposure Category “Off-Balance-Sheet”
  No write-downs are created for the off-balance-sheet part of a receivable, for example the free line of an account. To determine the risk provision for the off-balance-sheet part, the Impairment Server needs the free line and the credit conversion factor (CCF).
  Credit Conversion Factor (CCF)
  You can import the CCF to the Impairment Server. If you want the Impairment Server to determine the CCF itself, define a module environment and a module for the determination of the CCF in the Impairment Server environment. You edit the module and the module environment in Customizing for the general secondary data sources.
  Free Line
  The book value determination [page 658] can transfer the free line to the Impairment Server.
  - If you use the *Percentage Expected Loss* or *Lifetime Expected Loss* calculation approaches, the free line calculator [page 391] determines the free line.
  - If you use the *Expected Cash Flow* calculation approach, you have to import the free line.

  For more information, see the documentation for the individual calculation approaches.

Integration

The Impairment Server belongs to the Financial Instruments Calculation and Valuation process component within the Process and Method Layer. You can also use it to create custom risk reporting, for example, using the Calculation and Valuation Process Manager (CVPM).

More Information

Calculation Approaches

- Calculation Approach: Percentage Expected Loss [page 664]
- Calculation Approach: Lifetime Expected Loss [page 666]
- Calculation Approach: Expected Cash Flow [page 668]
- Calculation Approach: Write-Down to Collateral Value [page 671]

For more information about the methods of *BAdI: Calculation of Impairment Events*, see the system documentation for the individual methods.

Processes

- Calculation of the Percentage Expected Loss [page 970]
- Determination of Impairment Attributes [page 968]
- Calculation and Valuation Process Manager (CVPM) [page 1622]

4.3.16.1 Calculation Approach: Percentage Expected Loss

Use

The *Percentage Expected Loss* calculation approach (%EL approach) is a standard implementation of *BAdI: Calculation of Impairment Results*.

In this calculation approach, the system determines the impairment results based on the percentage expected loss (%EL) and analyzes a one-year period.

Implementation Considerations

Note also the following for the %EL approach provided by SAP: Since the key date valuation is a periodic process that calls the Impairment Server, for performance reasons we recommend that you schedule the *Calculate Percentage Expected Loss* process before the key date valuation. This ensures that the Impairment Server can access the results from this process.

Alternatively, you can provide the expected losses to the Results Data Layer (RDL) using enterprise services.

If you have entered a module from the application `RIEP` in the fixed step sequence *EL* (Calculate Expected Loss) for the process `/BA1/RRI_EL_PROCESS` in the Calculation and Valuation Process Manager (CVPM), the Impairment Server can calculate the expected loss on the fly instead.

> **Note**
>
> When secondary business transactions are updated, the Impairment Server calls the process *Calculate Percentage Expected Loss* on the fly if a module has been entered in the fixed step sequence. If a module has not been entered, the Impairment Server only uses the RDL results.

If you use **source data aggregation**, the aggregation of key date results must have been completed, in other words the expected loss and the write-down amount are available for each aggregation object.

Also see the general implementation considerations in the documentation for the Impairment Server (see *More Information*).

Features

This BAdI implementation calculates the impairment results for the exposure category "On-Balance-Sheet" as follows:

- Write-down
  In the %EL approach, the write-down (WD) on the key date (t) is the percentage write-down of the balance of the book value components for the determination of the write-down ($BBVC_{WD}$):

  $$WD_{\%EL}(t) = \%WD(t) \cdot BBVC_{WD}(t)$$

  The percentage write-down of the balance of the book value components for the determination of the write-down is the receivable. The system determines the analytical write-down without including the operational waiver, since the receivable has already been reduced by the operational waiver.
- Risk Provision
  In the % EL approach, the risk provision (RP) on key date (t) is the percentage expected loss of balance of the book value components for the determination of risk provision ($BBVC_{RP}$):

  $$RP_{\%EL}(t) = \%EL(t) \cdot BBVC_{RP}(t)$$

  Since the balance of the book value components for risk provision already includes the write-down, the percentage expected loss is the quota *after* write-down. If the Impairment Server does not find a percentage expected loss in the RDL during risk provision calculation, it calls the *Calculate Percentage Expected Loss* process on the fly.
- Unwinding
  No implementation of *BAdI: Calculation of Impairment Results* is delivered for calculating the unwinding in the *Percentage Expected Loss* approach. If you want to calculate the unwinding in the % EL approach as well, you need your own implementations.

The Impairment Server calculates the risk provision in the % EL approach for the **off-balance-sheet** part of a contract in the same way as for the on-balance-sheet part, where the %EL must already contain the credit conversion factor.

More Information

Impairment Server [page 661]

4.3.16.2 Calculation Approach: Lifetime Expected Loss

The *Lifetime Expected Loss* calculation approach is a standard implementation of *BAdI: Calculation of Impairment Results*.

In this calculation approach, the system determines the risk provision for a contract for all time buckets of a predefined maturity band that make a contribution that is not equal to zero. This contribution is the expected loss per time bucket, which is discounted on the reporting key date using the effective interest rate. The contributions across all time buckets are then added up to calculate the risk provision for the contract.

Implementation Considerations

Note the following for the *Lifetime Expected Loss* calculation approach provided by SAP: Since the key date valuation is a periodic process that calls the Impairment Server, for performance reasons we recommend that you schedule the *Calculate Percentage Expected Loss* process before the key date valuation. This ensures that the Impairment Server can access the results from this process.

Alternatively, you can import the expected losses in time buckets (rates or amounts) for each risk type and exposure category to the Results Data Layer (RDL). However, you cannot do this using enterprise services.

If you have entered a module from the application `RIEP` in the fixed step sequence `EL_LT` ("Calculate EL in Time Buckets") for the process `/BA1/RRI_EL_PROCESS` in the Calculation and Valuation Process Manager (CVPM), the Impairment Server can calculate the expected loss on the fly instead.

> **Note**
>
> When secondary business transactions are updated, the Impairment Server calls the process *Calculate Percentage Expected Loss* on the fly if a module has been entered in the fixed step sequence. If a module has not been entered, the Impairment Server only uses the RDL results.

Before you determine the expected loss in time buckets, you need to define the structure of the maturity band. To do so, in Customizing for *Bank Analyzer* choose *Processes and Methods* > *General Calculation and Valuation Methods* > *Impairment Methods* > *Create Maturity Band*.

To discount the expected losses in the time buckets, the Impairment Server needs the contractual effective interest rate. If the effective interest rate is available in accounting, the system takes it from there. Alternatively, you have to import it from a feeder system and store it in the Results Data Layer (RDL).

In Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Impairment Methods* > *Define Impairment Server Environments*, you assign to an impairment sever the storage of the expected loss in time buckets and of the contractual effective interest rate, and which maturity band is to be used.

To calculate the risk provision for the **off-balance-sheet** part of the contract, the free line must first have been determined and stored in the RDL by the *Determine Free Line for Financial Products* process (transaction `/BA1/RU_FREELINE`).

Also see the general implementation considerations in the documentation for the Impairment Server (see *More Information*).

Features

This BAdI implementation calculates the impairment results for the exposure category "On-Balance-Sheet" as follows:

- Risk provision

$$RP_{LEL}(t) = \sum_{i=1}^{n} EAD_i \cdot LGD_i \cdot PD_i \cdot e^{-EIR(t_{i-1}-t)}$$

  In the *Lifetime Expected Loss* calculation approach, the risk provision (RP) is the sum of the expected loss amounts (EAD_i * LGD_i * PD_i) of all time buckets (i) that are discounted on the current key date t using the contractual effective interest rate (EIR). Where the ith time bucket of the maturity band is defined as t_{i-1} to t_i, and where point in time t_i already belongs to the next time bucket.
  - For cash-flow based transactions, the Impairment Server can calculate the **exposure at default (EAD $_i$)**, in other words the net present value of the future cash flows for each time bucket. The Impairment Server receives the **contractual cash flows** from the cash flow generation or reads them from the Source Data Layer (SDL) if they were imported beforehand.

    > **Note**
    >
    > The EAD of the first time bucket is a special case because it not only comprises the net present value of the future cash flows but also further book value components that need to be taken into account when the risk provision is calculated (for example, analytical write-downs and valuation remnants).

    > **Note**
    >
    > If the contractual cash flow is in a different currency as the valuation currency, the system translates it at the rate valid on the valuation day. For financial instruments, the system scales the contractual cash flow to the relevant position nominal value or based on the number of units.

  - For a given maturity band, the *Calculate Percentage Expected Loss* process determines the **percentage expected loss** for each time bucket as the product of loss given default$_i$ and the probability of default$_i$ for each time bucket, and stores this value in the RDL. Alternatively, you can import the percentage expected losses for each time bucket from another system.
  - For each time bucket, the Impairment Server determines the **expected loss amount** from the percentage EL_i and the EAD_i, and discounts these amounts using the **contractual effective interest rate (EIR)** on the current key date. Finally, the Impairment Server adds up the discounted expected loss amounts for all time buckets. This sum is the risk provision amount.

If you import expected loss amounts for time buckets, you can also valuate contracts that are not based on cash flows and for which an EAD_i cannot be calculated. When you import the amounts, the maturity band structure in the feeder system must match the structure in Customizing. The Impairment Server then only discounts the imported amounts.

The Impairment Server determines the risk provision for the **off-balance-sheet** part of a contract by taking only the values in the first time bucket of the maturity band into consideration:

$$RP_{LEL(t)} = PD \cdot LGD \cdot CCF \cdot FL$$

The *Calculate Percentage Expected Loss* process determines the percentage EL for the off-balance-sheet part as the product of the risk parameters (LGD, PD and credit conversion factor/ CCF) in the first time bucket. The Impairment Server multiplies this amount by the free line to determine the risk provision amount.
More specifically, the CCF is typically calibrated for banks for a period of one year. In light of this, the system calculates the risk provision for a free line in this approach using an approximation by restricting the calculation to the first time bucket. Due to the availability of the CCF, this time bucket must have a duration of one year.
For more information, see the Business Content documentation.

- Write-down
  The determination of the write-down is not defined in Business Content, but you can set this up according to your specific needs.
- Unwinding
  The unwinding is not calculated for this calculation approach. The system returns the value “0”.

More Information

SAP Library: Impairment Server [page 661]

Business Content Guide:

4.3.16.3 Calculation Approach: Expected Cash Flow

Use

The *Expected Cash Flow* calculation approach (ECF approach) is a standard implementation of *BAdI: Calculation of Impairment Results*.

In this approach, the system determines the impairment results based on the expected cash flow (ECF).

Implementation Considerations

When using the ECF approach provided by SAP, you need to ensure that the expected cash flows and the write-down amounts are provided to the RDL by means of enterprise services. For financial instruments, both the ECF and the write-down have to be scaled to the nominal value of the financial instrument. When calling the Impairment Server, you need to provide the relevant nominal value of the position or the quantity to scale the result accordingly.

You need to either import the effective interest rate for the expected cash flows with the impairment attributes or it needs to be calculated when these attributes are imported in the service inbound layer, provided the ECF has been imported beforehand.

If you want to calculate the unwinding for accruing contracts with the standard ECF approach, you need to make the relevant settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings*.

Expected cash flows for financial transactions that are subject to the balance offsetting for risk provision have to be created taking balance offsetting into account.

> **Example**
>
> For a syndicated loan on the asset-side with a receivable of EUR 100 and the corresponding loan taken with a total of EUR 60 payable, the ECF has to be created for the asset-side syndicated loan, using only a receivable (after offsetting) of EUR 40.

Also see the general implementation considerations in the documentation for the Impairment Server (see *More Information*).

Features

This BAdI implementation calculates the impairment results for the exposure category “On-Balance-Sheet” as follows:

- Write-Down
  In the ECF approach, the system uses the write-down amount (WD) that is defined for a transaction in the RDL:

$$WD_{ECF} = WD$$

  This write-down does not include the operational waiver. This is already reflected in the receivable amount.
- Risk Provision

In the ECF approach, the risk provision is calculated as the difference between the balance of the book value components for the determination of risk provision ($BBVC_{RP}$) and the net present value of the expected cash flow (ECF) that is discounted on the key date (t) using the effective interest rate (EIR):

$$RP_{ECF}(t) = BBVC_{RP}(t) - \sum CF_i e^{-EIR(t_i - t)}$$

- Unwinding
  In the ECF approach, the system calculates the time effect of an ECF for a specific period [t_0, t_1] as follows:

$$TE_{ECF}(t_0; t_1) = \sum_{t_i > t_1} CF_i e^{-EIR(t_i - t_1)} - \left(\sum_{t_i > t_0} CF_i e^{-EIR(t_i - t_0)} - \sum_{t_1 > t_i > t_0} CF_i \right)$$

  The time effect (TE) is calculated as the difference between the risk provision at time t_0 (last key date valuation) and time t_1, (current key date valuation), and the sum of the payments received within this period. The key date valuations are usually run on the last day of the month.
  For non-accruing contracts, the unwinding corresponds to the time effect described.
  If the ECF approach is applied to accruing contracts, the system deducts both the accruals (interest accrual) and the deferrals for the contractual cash flow from TE_{ECF} because these are recorded separately in profit and loss. This value is the unwinding adjustment.

> **Note**
>
> If the expected cash flow is in a different currency as the valuation currency, the system translates it at the rate valid on the valuation day.

> **Note**
>
> For financial instruments, the system scales the ECF and the write-down to the relevant position nominal value or based on the number of units.

In the ECF approach, the Impairment Server determines the risk provision for the **off-balance-sheet** part of a contract in the same way as for the balance sheet part, and then multiplies the result with the credit conversion factor (CCF).

More Information

Impairment Server [page 661]

4.3.16.4 Calculation Approach: Write Down to Collateral Value

Use

The *Write Down to Collateral Value* calculation approach (WDA approach) is a standard implementation of *BAdI: Calculation of Impairment Results*.

In this approach, in the event of a foreclosure, for example, the system determines the amount of the write-down to the fair value as the difference between the exposure amount and the net fair value of the collateral (minus the costs).

Implementation Considerations

When you use the WDA approach provided by SAP, you need to ensure that the collateral values are imported to the RDL as expected cash flows by means of enterprise services. The collateral values contain the expected costs (for example, for collection). The write-down amount and the date are ignored in the calculation.

For financial instruments, the collateral values must be scaled to the nominal value of the financial instrument. When calling the Impairment Server, you need to provide the relevant nominal value of the position or the quantity to scale the result accordingly.

Also see the general implementation considerations in the documentation for the Impairment Server (see below).

Features

This BAdI implementation calculates the impairment results for the exposure category "On-Balance-Sheet" as follows:

- Write-down
  In this calculation approach, the write-down to the fair value (WD_{WDA}) is calculated as the difference between the balance of the book value components for the determination of the fair value adjustment ($BBVC_{FVA}$) and the sum of the net fair values of the collateral (FV_{Coll}):

$$WD_{WDA}(t) = BBVC_{FVA}(t) - \sum FV_{COLL}(t)$$

- Risk provision and unwinding
  There is no risk provision and no unwinding in this calculation approach. The system returns the value "0".

No write-down is calculated for the **off-balance-sheet** part of the contract.

More Information

Impairment Server [page 661]

4.3.17 Balance Offsetting

Use

The balance offsetting function enables you to offset transactions in the calculation of risk provision and for the determination of the asset/liability status.

The system groups relevant contracts logically in offsetting groups and determines offsetting amounts or the offsetting group balance, which it stores in the Results Data Layer (RDL). These results then serve as an additional calculation base for the determination of risk provision and as base for the determination of the asset/liability status for offset transactions.

You start the balance offsetting process for all contracts in balance offsetting within the specified legal entity and for the specified accounting system.

Prerequisites

- You make the settings for balance offsetting in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Balance Offsetting*.
- In addition, you edit special link types and bundling types in the bundling service under *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Bundling Service*.
- You set up a template for master contracts in the Customizing for the Source Data Layer under *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Master Data*.
- You set up object relationship categories for master contracts in the Customizing for the Source Data Layer under *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Master Data* > *Settings for Attributes* > *Object Relationships* > *Edit Object Relationship Categories*.
- You assign the offsetting run type for the asset/liability determination to an accounting system in Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Transfer Methods for Assets and Liabilities* > *Assign Balance Offsetting Run Type for Asset/Liability Determination*.
- You assign the offsetting run type for risk provision to an accounting system in Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Accounting Processes* > *Valuation* > *Risk Provision* > *Assign Offsetting Run Type for Risk Provision*.
- Before you execute the *Calculate Balance Offsetting Result* transaction on the *Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Balance*

Offsetting, you need to make the settings for the authorization for the *Accounting System*, *Legal Entity* and *Run Type* using authorization object `F_BADK_CL`.

Features

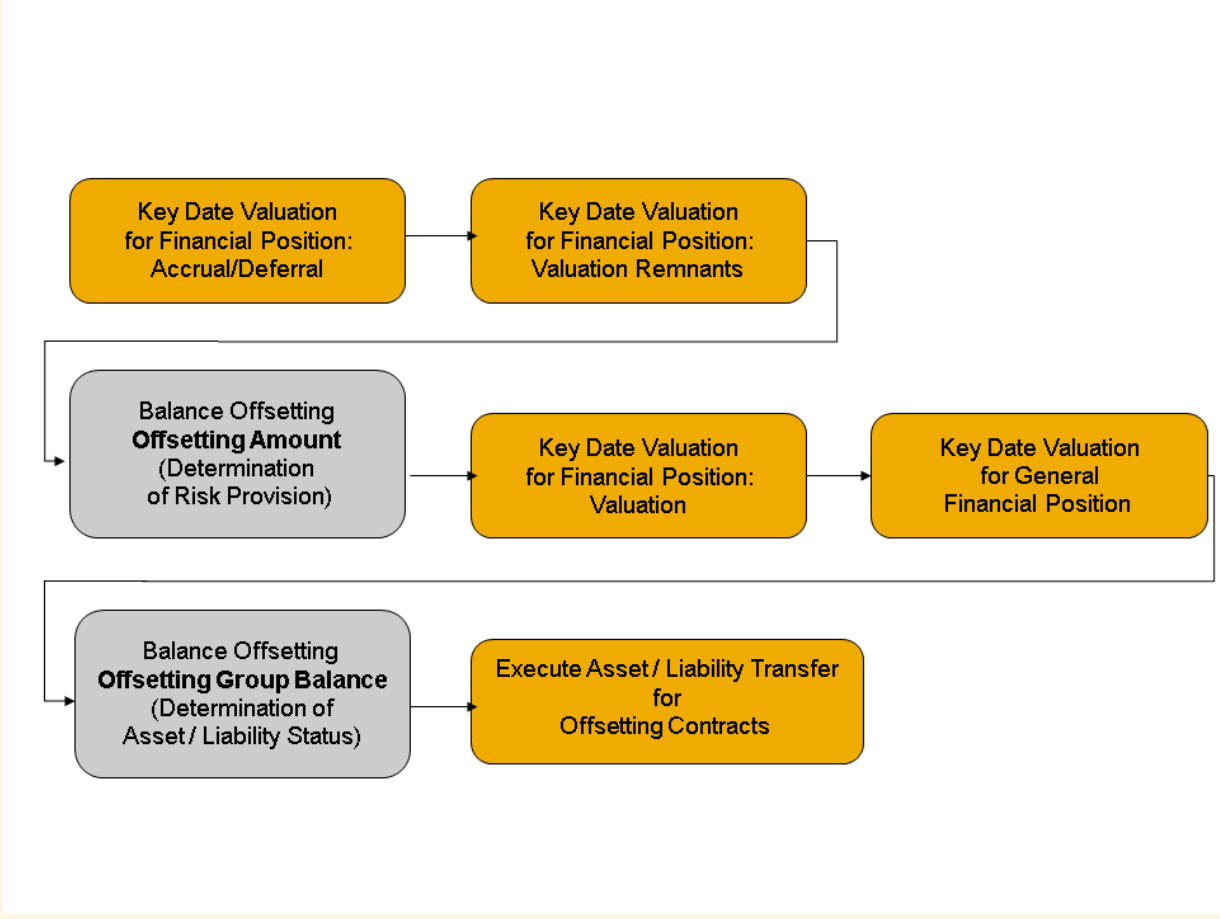
You execute balance offsetting at the end of a period within the tasks for period-end closing. If necessary, you can also run the offsetting during a period.

The system uses master contracts to group the financial transactions. You provide master contracts as financial transactions in the source data layer (SDL) with a template from the category *Financial Transaction* and the production control settings that you have made in Customizing. You assign the financial transactions from a group to the master contract using object relationships. You can use hierarchies of master contracts, in other words, master contracts are also assigned to one master contract. The system determines all of the individual contracts that are bundled under the top hierarchy node, irrespective of how deep the hierarchy is nested.

If a master contract is changed in a current period, for example if individual contracts for a master contract are added or removed, the system only considers this change in the next offsetting run.

Example

You use balance offsetting for the determination of risk provision and the asset/liability status. You need to embed the balance offsetting in the chain of process runs as follows:



The results from *Accrual/Deferral* and the *Release Valuation Remnants* are included in the calculation of the balance offsetting amount for risk provision. Therefore, you need to run the key date valuation with the key date valuation types *Accrual/Deferral* and *Release Valuation Remnants* before you calculate the offsetting amount for risk provision. After the offsetting run you execute the key date valuation with the key date valuation type *Valuation* to determine the risk provision, taking the current offsetting amounts into account.

The results of the key date valuation are used in the determination of the asset/liability offsetting group balance. For this reason, you need to run the key date valuation with the key date valuation type *Valuation* before you calculate the asset/liability offsetting group balance. The offsetting group balances determined are the basis for the *Asset - Liability Transfer for Offset Contracts*, which you execute immediately afterwards. This changes the asset/liability status of the individual contracts according to the offsetting group balance determined by the offsetting.

Note

If you want to use your own functions, you can implement the BAdIs *Balance Determination for Balance Offsetting* and *Determination and Distribution of Offsetting Amount* in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Balance Offsetting*. You can use implementations of the BAdI *Enrichment of Balance Offsetting Results* to add additional values to the balance offsetting results data in the Results Data Layer (RDL) for reporting purposes. For example, you can also define result amounts in functional currency.

Activities

Calculate Balance Offsetting Result

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Balance Offsetting* > *Calculate Balance Offsetting Result*.
2. You enter the necessary data and execute the run.

If you have corrections for special master contracts, you have the option of specifying the master contracts. In this case you have to specify a key date on which you have already executed an offsetting run.

Ensure that the processes always only read the offsetting results from the last complete run. These are the results for the date of the last complete run, for which the key date is on or before the process key date. In this way the system also includes offsetting results that were corrected after the last complete run.

If you want the system to process current offsetting results in the key date valuation, you must schedule the offsetting run accordingly. Otherwise the system considers the results from the last completed run.

You can execute several offsetting runs for one key date. The results from the last offsetting run overwrite all of the existing results for the same key date.

Asset - Liability Transfer for Offset Contracts

You transfer the assets and liabilities at the level of master contracts for a key date at the end of the period. The transfer changes the asset/liability status of the individual contracts in which the asset/liability status differs from the asset/liability status of the offsetting group balance.

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Automatic Processing* > *Execute Asset-Liability Transfer for Offset Contracts*.

Determine Master Contract for Contract

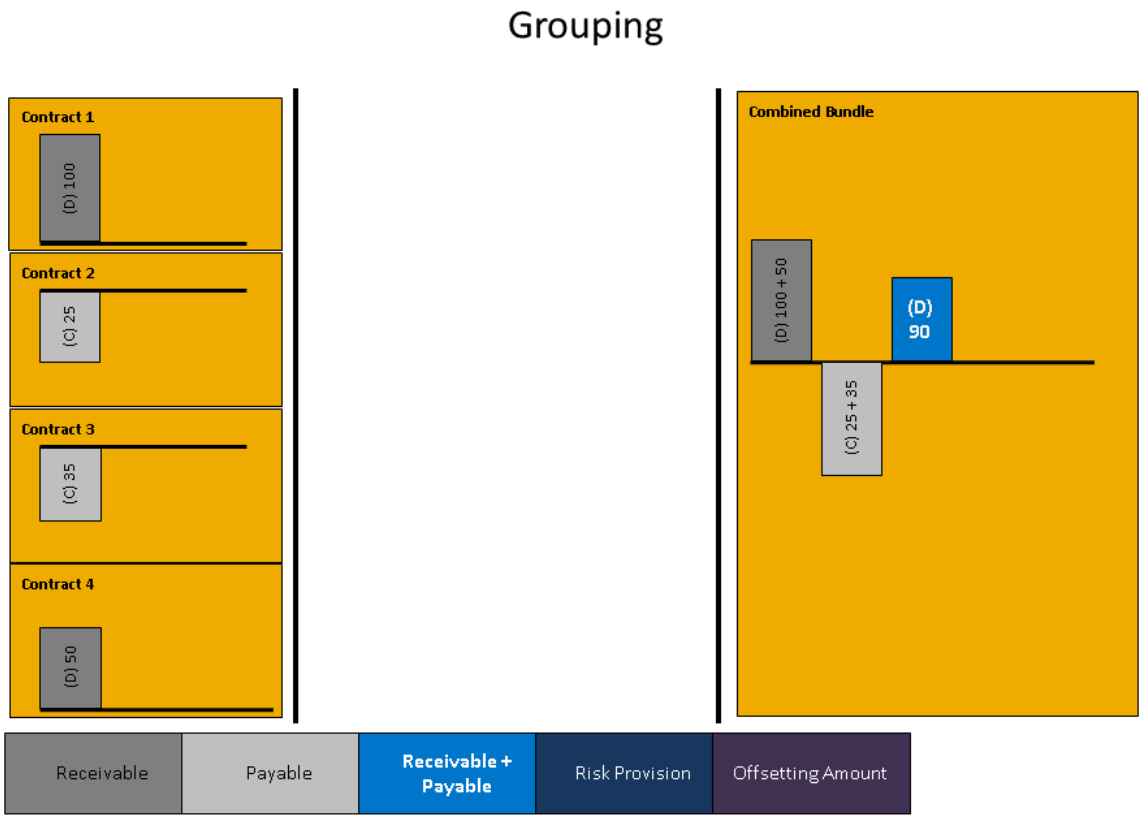
If you want to update specific balance offsetting results after correcting errors in a specific contract, you need the relevant uppermost master contract and the list of all the individual contracts in the relevant master contract. To do this you can use the following search function:

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Balance Offsetting* > *Determine Master Contract for Contract*.

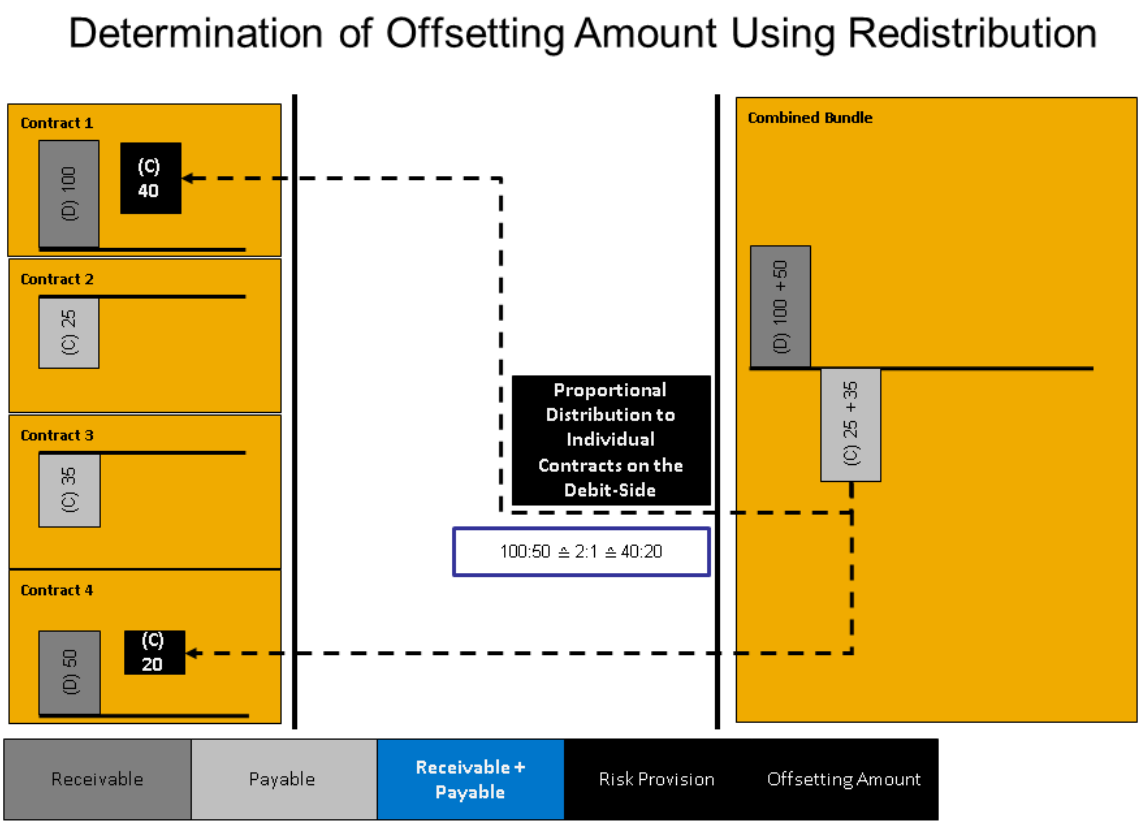
Example

The following example explains balance offsetting in four steps.

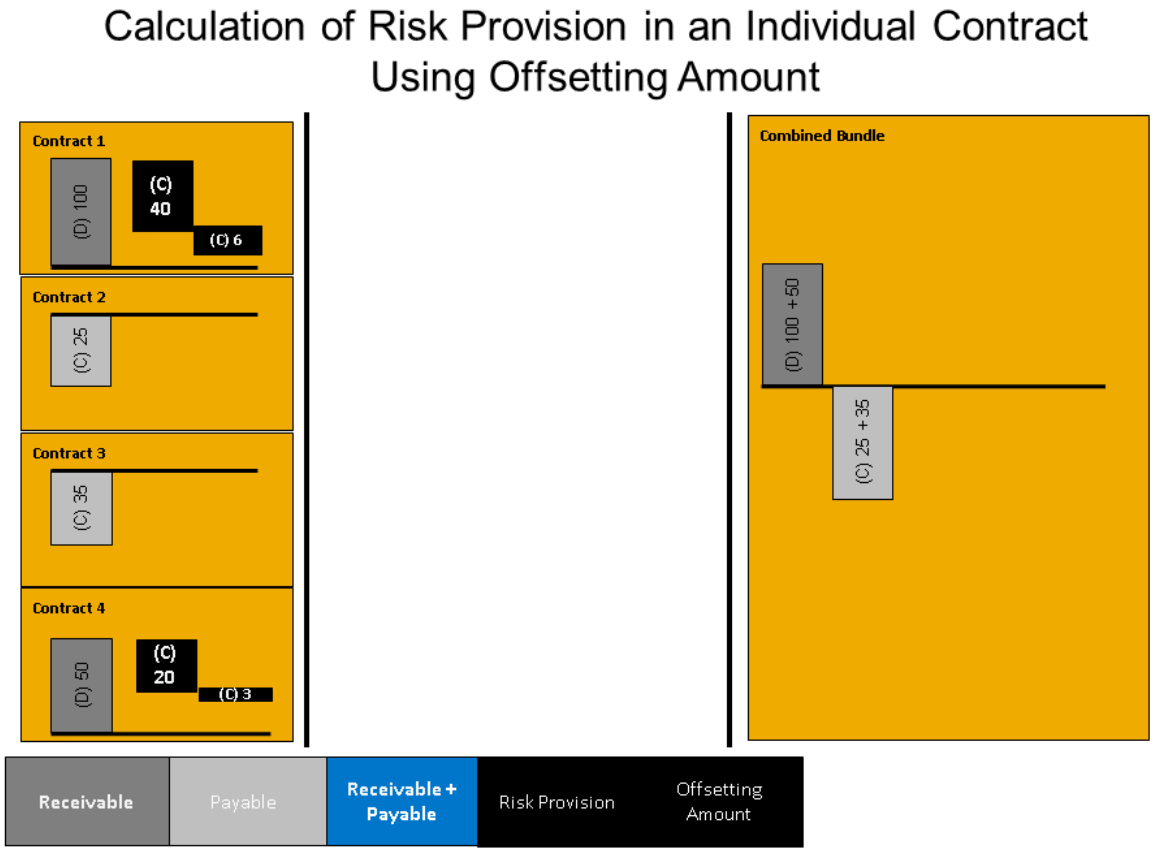
The first step shows the grouping of the different individual contracts in a master contract.



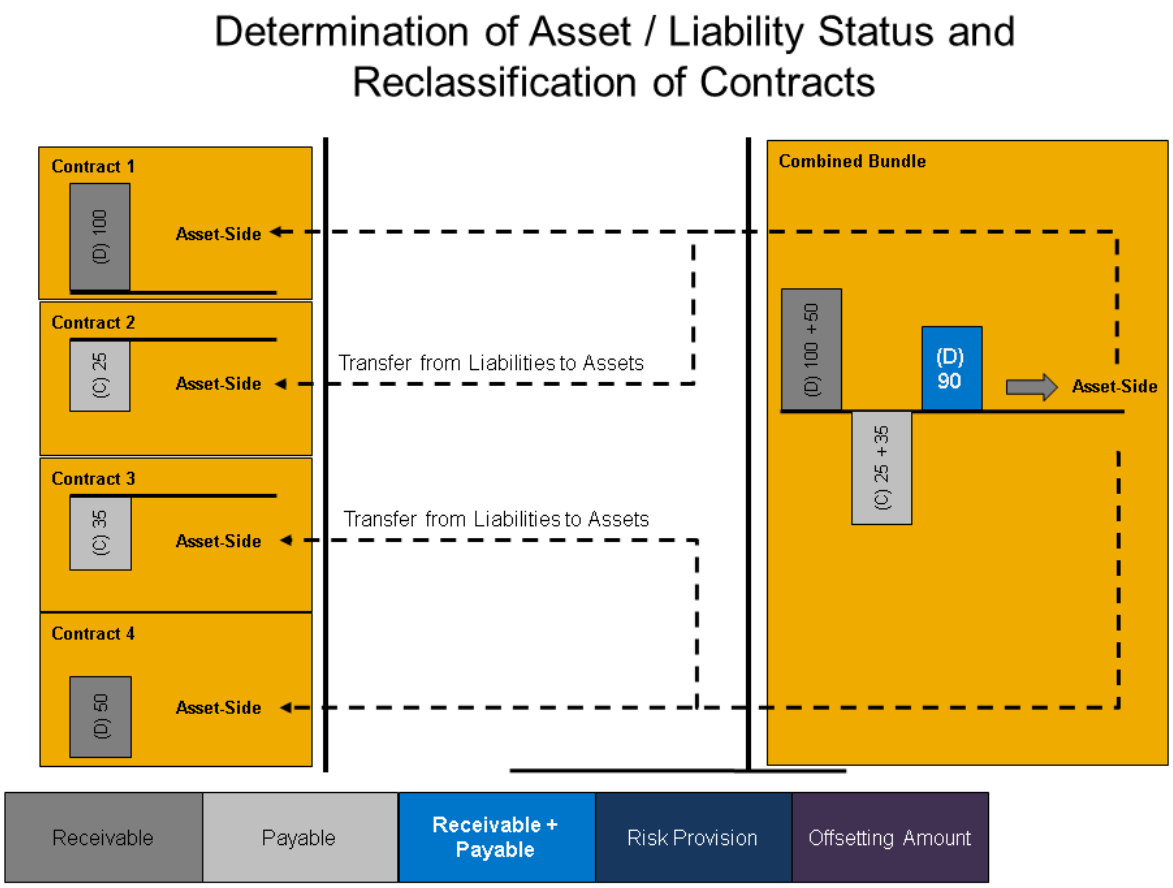
In this step the system determines the offsetting amount and the proportional distribution to individual contracts on the debit-side.



In this step the system calculates the risk provision in the individual contract.



In this step the system determines the offsetting group balance, which determines the asset/liability status of the individual contracts and, if necessary, posts the asset-liability transfer.



4.4 Accounting for Financial Products

This component provides you with functions for accounting (external accounting) for financial transactions and financial instruments.

It comprises the Accounting Processes [page 693] component.

4.4.1 Processes in the AFI Scenario

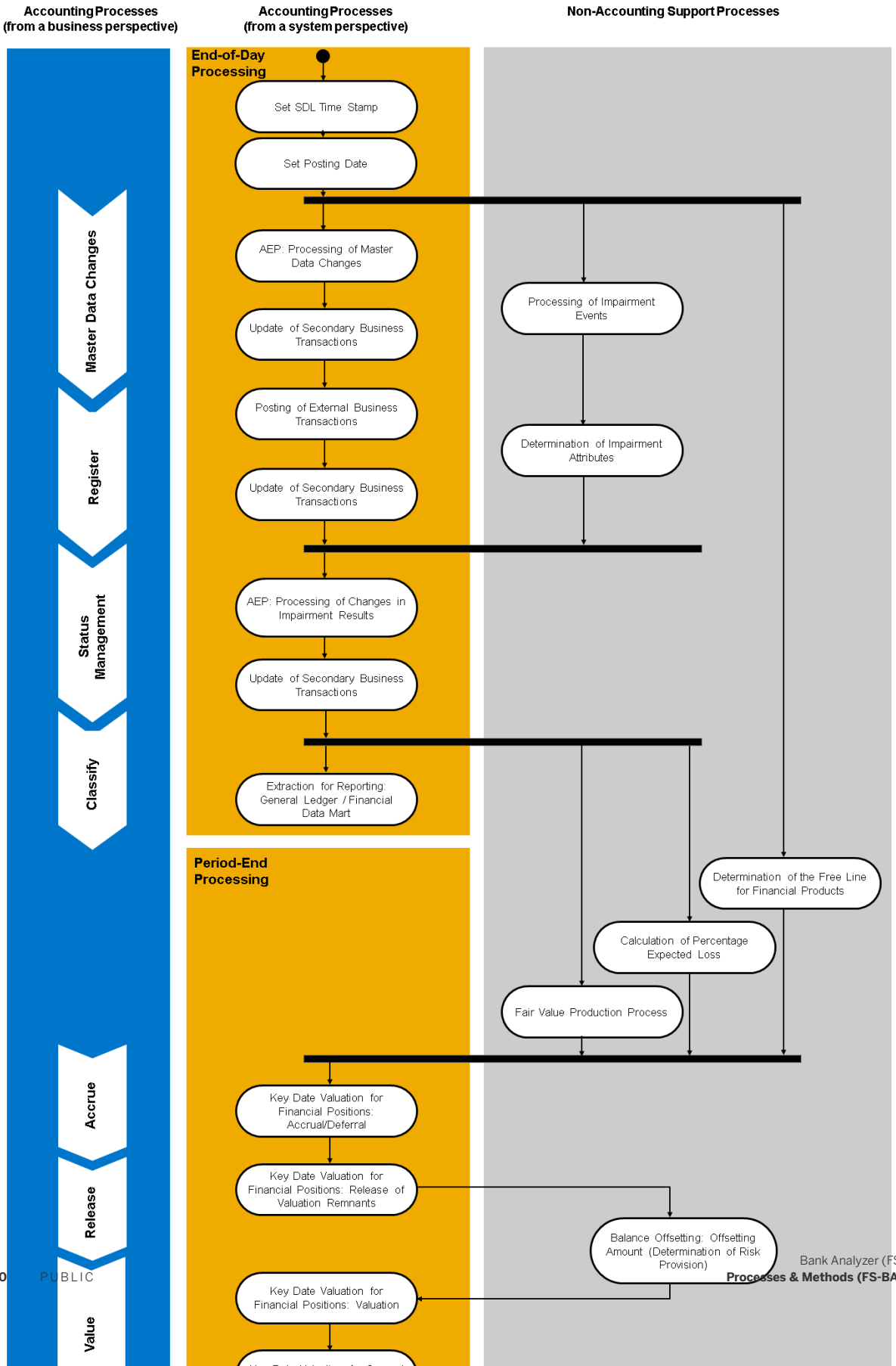
Use

The processes for the *Accounting for Financial Instruments* (AFI) scenario determine accounting results at individual contract level. They periodically forward these results to the general ledger (general ledger level) and to Reporting (subledger level) so that a financial statement (balance sheet, income statement, and notes) can be created for each legal entity.

> **Note**
>
> This process description does not include the Processes for Source Data Aggregation [page 298] and Hedge Accounting [page 904]. The option of executing processes by source system is also not covered.

Process

The figure below illustrates the business accounting processes and how these are implemented in the Bank Analyzer system, as well as non-accounting support processes:



The accounting processes in the AFI scenario are divided into the following steps:

- Register
- Write Down
- Accrue/Defer
- Release
- Value
- Classify
- Manual Posting
- Close
- Carry Forward

For more information about these steps and the accompanying accounting processes (master data changes, for example), see Accounting Processes (Subledger Scenario) [page 693].

In the system these business steps are implemented by various processes that are part of **End-of-Day Processing**, **Period-End Processing**, and **Year-End Processing**. In parallel, you execute support processes, which include the calculation of values required by accounting.

End-of-day processing

1. Set SDL Time Stamp [page 367]
   You begin end-of-day processing by setting a time stamp for access to data in the Source Data Layer (SDL) and to market data.
2. Set Posting Date [page 693]
3. Processing of Accounting Events [page 769] (Accounting Event Processing AEP)
   During the processing of accounting events, the system transfers master data changes as operational events at this stage in the process chain. (A loan rollover, for example). The process generates event documents.
4. Update Secondary Business Transactions (Subledger Scenario) [page 786]
   The process for updating secondary business transactions processes operational event documents for each posting date in a fixed sequence. The process updates the characteristics in the financial position accordingly and generates the documents required for accounting. From a business perspective, this process is used to process changes to master data.
5. Post external business transactions (subledger scenario) [page 774]
   This process is used for registration, in other words the transfer and processing of operational flow transactions.
6. Update secondary business transactions
   The process for updating secondary business transactions processes flow transactions for each posting date in a fixed sequence. It also posts flow transactions from suspense accounts to the relevant book value components. From a business perspective, this process is used for registration.
7. Processing of accounting events (AEP)
   Here the process uses application events that have been derived from analytical events related to changed impairment results. The process generates event documents. From a business perspective, this process is used for status management.

> **Note**
>
> If you calculate **impairment** in Bank Analyzer, you execute the following processes first. These process information about write-downs, risk provision, and the accrual status of contracts:
>
> 1. Impairment Event Processing [page 966]

2. Determination of Impairment Attributes [page 968]

8. Update secondary business transactions
   The process for updating secondary business transactions processes analytical event documents and is used for status management.
9. Extraction for Reporting (Classification)
   All of the resulting posting documents are saved in the Results Data Layer (RDL).
   The periodic data extraction to the general ledger on the basis of the posting documents takes place in the General Ledger Connector. The General Ledger Connector aggregates the subledger documents and sends them to the general ledger using the service outbound layer. For more information, see General Ledger Connector (FS-BA-AN-GL) [page 1411].
   You can also perform periodic data extraction into SAP Business Warehouse or another financial data mart.

Period-end processing

1. Preparation for Key Date Valuation
   To prepare the valuation of the individual positions, non-accounting support processes determine values such as the fair value and the risk provision. Some of the processes can also be called spontaneously during the key date valuation:
   - Fair Value Production Process [page 536]
     This processes calculates fair value by calling the Fair Value Server.
   - Determination of the Free Line for Financial Products [page 388]
     The key date valuation requires the free line when it calls the Impairment Server to determine the risk provision for the off-balance sheet part of a receivable, for example.
   - Calculation of Percentage Expected Loss [page 970]
     The Impairment Server uses the percentage expected loss to calculate the risk provision for contracts that are not assigned to the calculation approach *Expected Cash Flow*.
   - Balance Offsetting [page 672]
     The results of the balance offsetting are used as an additional calculation base for the determination of risk provision for offset contracts.
2. Key Date Valuation [page 934]
   The key date valuation type (such as *[A[Accrual /Deferral*) controls which process steps are executed during a key date valuation run. The following key date valuation runs are available:

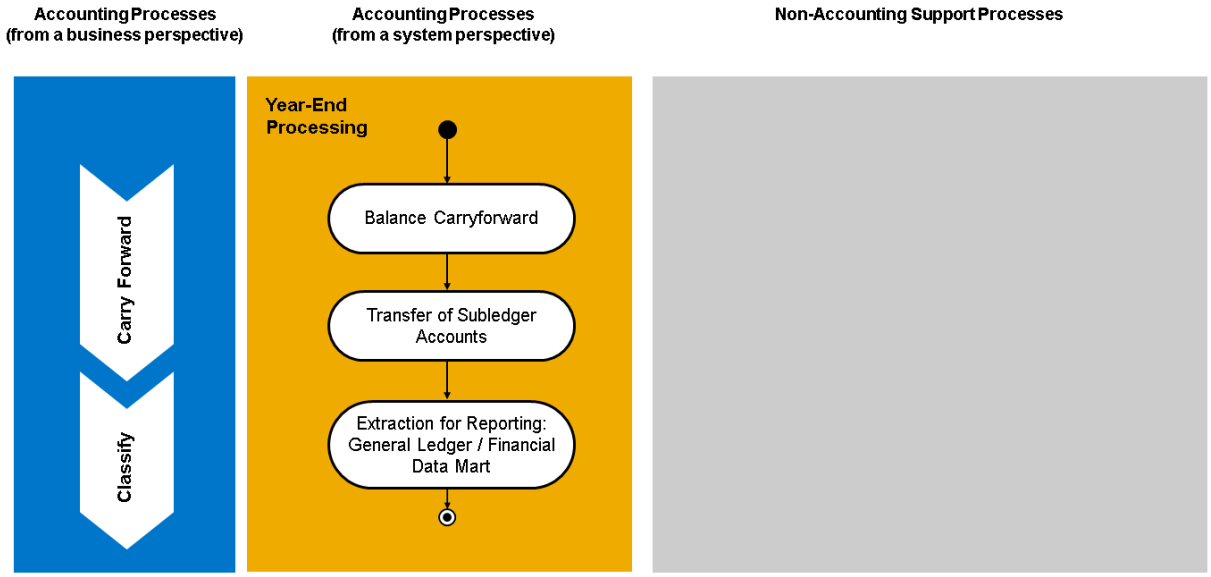
Sequence	Key Date Valuation Run	Business Process
1.	Key date valuation for financial positions: Accrual	Accrue/Defer
2.	Key date valuation for financial positions: Release of Valuation Remnants	Release
3.	Key date valuation for financial positions: Valuation	Value
4.	Key date valuation for general financial positions	

3. Balance offsetting of book value for the determination of the asset/liability status
   The results of the balance offsetting are also used as an additional calculation base for the determination of asset/liability statuts for offset contracts.

4. Asset - Liability Transfer for Offset Contracts (Reclassification)
   When the asset/liability status changes, the system reclassifies the entire position in terms of the balance sheet. For more information, see Asset/Liability Status [page 863].
5. Maturity Grouping [page 543]
   The maturity grouping is not an accounting process but a support process which determines how the cash flows or book value of contracts is to be distributed over the remaining term of the contract, so that these can be explained in the notes to the financial statement.
6. Extraction for Reporting (Classify)
7. Corrections
   You can enter postings to individual key figures of a financial position manually, which will be reset automatically on the next posting date. For more information, see Manual Postings (Subledger Scenario) [page 944].
8. Closing the posting period
   For more information, see Opening and Closing Posting Periods (Subledger Scenario) [page 717].

Year-End Processing

Year-end processing comprises the creation of the balance carryforward [page 947] and, optionally, the transfer of subledger accounts [page 950].



> **Note**
>
> The system does not transfer the balance carryforward data to the general ledger.

More Information

Processing of Events in the AFI Scenario [page 684]

4.4.2 Processing of Events in the AFI Scenario

Use

The processing of events in the *Accounting for Financial Instruments* (AFI) scenario reacts to changes to master data and impairment results in Bank Analyzer that users make or that are based on internal analytical decisions. You can import specific changes as external business transactions. The derived events inform the requesting applications about changes that need to be processed further by the respective application, for example, an address change for a business partner.

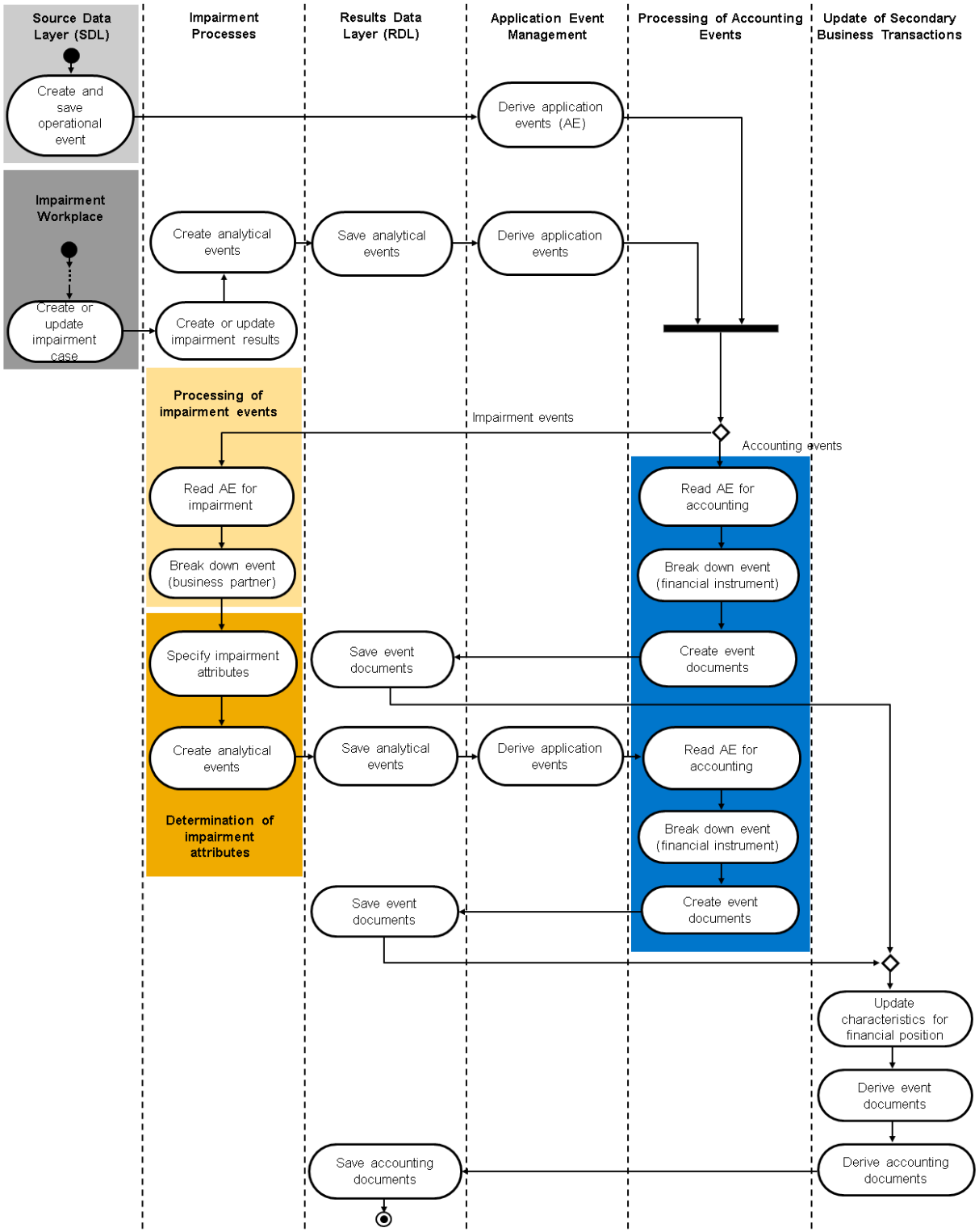
In Bank Analyzer, a distinction is made between the following events:

- Operational Event [page 258]
- Analytical Event [page 689]
- Application Event [page 369]

> **Note**
>
> This process description does not include the following components and applications:
>
> - Hedge processes
> - Reclassification
> - Source data aggregation
> - Event-driven internal cost calculation (EICC) / determination of net present values and calculation bases
> - Calculation of indirect costs for periods (PICC)

Process

The figure below illustrates how Bank Analyzer processes events in the Bank Analyzer AFI scenario.



Processing of Events in the AFI Scenario

1. **Derive application events**

Application event management derives the application events at the same time as creating or importing the following events:

- Operational events
  If the relevant Customizing settings have been made, operational events are generated in the business partner, in financial transactions, financial instruments and securities accounts (including contract creation and contract termination) synchronously to **master data changes**.

  > **Example**
  >
  > Deterioration of the rating for a business partner, rollover of a loan

  Operational events are also generated in the event of **payment arrears**, but have to be imported as external business transactions.

  > **Example**
  >
  > Delinquency of a borrower

- Analytical events
  If a user creates or adjusts an impairment case from an external UI, Bank Analyzer creates analytical events based on the following triggers:
  - Creation or change of the expected cash flow (ECF) and of the write-down amount, if required
  - Creation or change of the value of an impairment attribute

    > **Note**
    >
    > Analytical events are also derived from the results of the *Determine Impairment Attributes* process (see step 2).

    Analytical events that the system derives from imported impairment attributes do not lead directly to the creation of accounting events but to impairment events. In this case, you need to schedule the *Edit Impairment Results* and *Determine Impairment Attributes* processes. It is not until the impairment attributes are redetermined that the creation of analytical events is triggered, from which the system derives accounting events.

For more information, see Application Event Management [page 368]. For more information about the renewed derivation of application events from previously derived operational events, see Regeneration of Application Events [page 374].

2. **Read and process application events**
   - **Impairment**

The processing of application events for impairment involves the following processes and steps:

Process	Step	Additional information
Process impairment events	**Read application events** The process reads all application events relevant for impairment that have not yet been imported and that have the SDL time stamp of the selected process partition.	Impairment Event Processing [page 966]
	Break down application events (business partner) For events that are generated by changes to the business partner data, the process determines which financial transactions and financial instruments for the business partner are affected by the event.	
Determine impairment attributes	**Create worklist** The process creates a worklist of financial transactions and financial instruments based on the data that the preceding process has transferred.	Determination of Impairment Attributes [page 968]
	Derive impairment attributes The impairment attribute values for the selected contracts are redetermined and saved in the RDL.	
	Create analytical events The system derives one analytical event from each changed impairment attribute, which is based on characteristics delivered by SAP, at single transaction level and saves this in the RDL.	
-	**Derive application events** Application event management derives application events that are to be processed in accounting from the analytical events.	Application Event Management [page 368]

Process	Step	Additional information
Process accounting events	**Create worklist** The process reads these application events with the current process time stamp. This means that the worklist for the accounting processes can also contain application events that were derived after the SDL time stamp was set from the determination of impairment attributes. **Note** If the *Determine Impairment Attributes* process is not completed before the *Process Accounting Events* process, some application events from the impairment attribute determination may not be selected until accounting events are next processed.	Processing of Accounting Events [page 769]

- **Accounting**
  When you execute the **Process Accounting Events** process, the system reads all the accounting-relevant application events and creates a worklist for further application-specific processing. The system only selects application events that have not yet been delivered to this process:
  - The system reads application events that were derived from operational events with the SDL time stamp of the selected process partition.
  - The system reads application events that were derived from analytical events with the current process time stamp of the selected process partition.

  The process adjusts the granularity of the read application events so that the subsequent process *Update Secondary Business Transactions* can process them. This means that the system breaks down the events into individual financial positions by breaking down the financial instruments into securities accounts. The process stops processing the contract creation events for financial instruments if there is not yet a financial position available. If the system creates an impairment-relevant financial position for a financial instrument (for example, in the event of a purchase), it reads the descriptive characteristics of the position using a secondary data source from the RDL. If impairment results do not yet exist, the system uses standard values.

For more information, see Processing of Application Events [page 375].

3. **Prepare accounting events for update of secondary business transactions**
   Before the *Update Secondary Business Transactions* process can process an accounting event, the system needs to convert it to a business transaction format. The **Process Accounting Events** process creates an event document for each financial position and accounting event. The event documents, like the business transactions, are saved in the results view for posting documents.

An event document is purely a technical document that contains no amounts, but a business transaction type and item type. The *Update Secondary Business Transactions* process uses this information to derive further actions:

- Update of the characteristics in the financial position
- Creation of documents required for accounting

4. **Update characteristics in the financial position**
   The **Update Secondary Business Transactions** process processes the accounting events for each posting day in a set sequence of process steps. Which process step the accounting events are processed in depends on the business transaction category of the accounting event. This applies both to updating the characteristics in the financial position and to creating the documents required for accounting.
   For example, an accounting event for the operational event *Characteristic Change* is considered before the operational business transactions. Impairment events are not processed until after the processing of operational business transactions.
   The process updates the characteristics of all the relevant financial positions. During the processing of accounting events, which have been derived from analytical events from the impairment process, the process updates all of the relevant financial positions, regardless of whether their holding category is relevant for impairment.
   For more information, see Reaction to Analytical Events During the Update of Secondary Business Transactions [page 793].
5. **Derive accounting document from accounting event**
   Calculation Management creates the documents required for accounting, for example, the update of risk provision based on the new characteristic values.

Result

The created accounting documents are saved in the Results Data Layer (RDL) as secondary business transactions.

More Information

Processes in the AFI Scenario [page 678]

4.4.3 Analytical Event

An analytical decision or the updating of a result that is relevant for analytical components.

The creation and processing of analytical events is implemented in Bank Analyzer as follows:

Trigger	Mapping in the System	Saved in	Example	Additional Information
Reclassification decision	An analytical event of this type is mapped by a business transaction, which you need to import.	Source Data Layer (SDL)	Change of holding category from *Fair Value* to *Amortized Cost*	Reclassification [page 955]
Creation or change of impairment events	This kind of analytical event is only created in Bank Analyzer when an impairment result changes. The category of the analytical event (for example, *Change in Calculation Approach*) is predefined in the system and defines how the event is processed.	Results Data Layer (RDL)	Change to impairment status; import of an expected cash flow	Determination of Impairment Attributes [page 968] Processing of Events in the AFI Scenario [page 684]
Designation or change of a hedging relationship	The system maps this kind of analytical event using an event document from accounting or an event document created from the Hedge Accounting Workplace (HAW). These event documents are created only in Bank Analyzer. You can view details for the events in HAW.	Results Data Layer (RDL)	Dissolution of a hedging relationship due to ineffectiveness (event within accounting) or management decision (HAW)	Hedge Accounting [page 904]
Decision: Major Modification	The system maps an analytical event of this type using a business transaction, which you need to import.	Source Data Layer (SDL)	A contractual cash flow change is classified as a major modification according to IFRS 9.	

Create/Import

- Reclassification
  To trigger a reclassification, you import an analytical event as an external business transaction using the Business Application Programming Interface (BAPI) for business transactions.

- Impairment
  If impairment-relevant events are created or changed, the system creates analytical events. The system does this when the updated impairment results are imported, and also when the impairment attributes are updated by the *Determine Impairment Attributes* process. This process creates analytical events only for impairment attributes that are based on standard SAP characteristics.
  For each accounting system, you define the results data area and results type under which you want to save the results. You do so in Customizing for *Bank Analyzer* under *Results Data Layer* > *Settings for Flow Results* > *Make Settings for Storage of Analytical Events in RDL*.
- Hedge Accounting
  Analytical hedge events are created internally due to the following triggers:
  - A management decision is implemented manually in HAW, for example a designation.
  - The results of effectiveness tests show that a hedging relationship is ineffective.
  - The impairment status of a product changes in such a way that the financial position can no longer be used in a hedging relationship.

    > **Note**
    >
    > In the Customizing activity *Define Impairment Status in Hedge Accounting*, you define which impairment status values are not allowed for financial positions in hedge accounting.

  Analytical hedge events can lead to the designation, dissolution or redesignation of a hedge or the dedesignation of hedging instruments. Analytical events are passed on to accounting so that the effects of the analytical events can be documented there.
  The event documents are stored in the same location as the posting documents in the RDL.

Derive

Application event management can derive one or more application events from one analytical reclassification or impairment event. These events trigger analytical applications.

See also: Application Event Management [page 368]

- Reclassification
  It is not possible to reclassify by transferring a position or by changing master data and processing application events.
- Impairment
  You can activate or deactivate the derivation of application events for selected combinations of analytical event categories, applications, and application event categories in Customizing for *Bank Analyzer* under *Processes and Methods* > *Application Event Management* > *Derive Application Events from Analytical Events*.
- Hedge Accounting
  When an event document is processed in accounting, accounting sends the corresponding notification to the Hedge Accounting Workplace, which leads to a change in the status of the hedging relationship.

Delete

- Reclassification
  In urgent cases, you can delete reclassification events in the data import buffer; in other words, if the events have been created, but not yet been through the derivation process. However, the feeder system and Source Data Layer can no longer be reconciled after this.
- Impairment and Hedge Accounting
  You cannot delete analytical events that relate to impairment and hedge accounting.

Reverse

- Reclassification
  For more information about reversing a reclassification, see Reversing Business Transactions [page 156]. The reversal business transaction and the original business transaction must have the same posting date. In other words, you cannot reverse a reclassification business transaction with a posting date that is higher than that of the original business transaction.
  The *Post External Business Transactions* and *Update Secondary Business Transactions* processes reverse all the documents produced by the reclassification, and the characteristic change.
- Impairment
  If an application event is reversed, the impairment results of the original event are deleted and the relevant analytical events are reversed in the RDL. Impairment attributes are not derived for reversal events.
- Hedge Accounting
  You can reverse a hedge event manually in the Hedge Accounting Workplace and can also enter a new posting manually.

Archive

- Reclassification
  You can archive reclassification events and reversals of reclassification events in the SDL using archiving objects.
  See also: Archiving of Primary Objects [page 279]
- Impairment and Hedge Accounting
  You can archive analytical events or an RDL table with event documents by setting the RDL connection for archiving.
  See also: Archiving and Destroying Results Data Using the Archiving Engine [page 1395]

4.4.4 Setting the Posting Date

Use

In this transaction you define an upper limit for the posting date for each legal entity. The AFI processes then only process business transactions and application events up to this limit. This may be useful if you want to perform a settlement on a key date, and want to prevent later postings from being included, for example.

> **Note**
>
> In the case of application events, the system uses the business record date as the posting date.

1. On the SAP Easy Access screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Daily Processing* > *Automatic Processing* > *Set Posting Date*.
2. Enter the following on the selection screen:
   - Legal entity (mandatory)
   - Posting date (mandatory)
     Enter the latest posting date required.
   - Technical settings (optional)
     You can call up filter variants or specify that the SDL time stamp is to be set in a test run.
3. Choose *Execute*.

Result

The system only processes business transactions and application events for the legal entity up to the specified posting date. You can analyze the run in the Calculation and Valuation Process Manager (CVPM) [page 1622].

> **Note**
>
> You can call up posting dates set for legal entities by choosing *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Tools* > *Display Posting Cut-Off*.

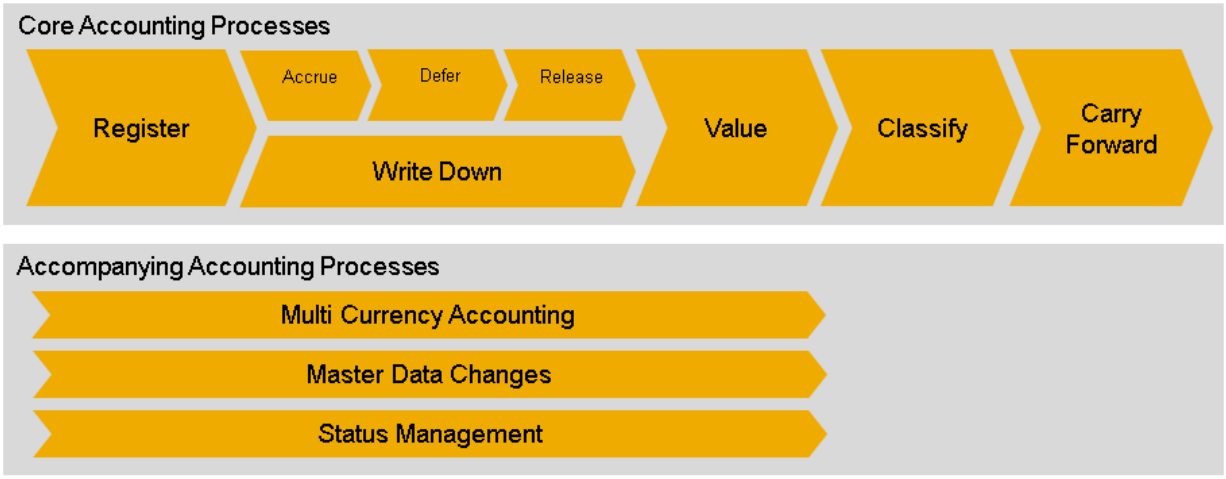
4.4.5 Accounting Processes (Subledger Scenario)

Use

The accounting system functions as a complete subledger for all financial instruments and financial transactions. The subledger scenario is based on the Integrated Finance and Risk Architecture (IFRA).

The subledger is supplied with business transactions from the Source Data Layer [page 100]. The financial position processes [page 743] of the Processes and Methods Layer (PLM) write their results to the Results Data Layer [page 1364]. Analytics [page 1410] has a connection to a general ledger, for example, SAP-FI.

The figure below illustrates the subprocesses into which the accounting processes are divided:



- **Register**
  - Transfer and processing of operational flow transactions (business transactions).
  - Transfer and processing of operational events, including:
    - Master data changes
    - Changes to the cash flow of the current interest period (such as condition change, interest fixing)
    - Changes to the cash flow for a future interest period (such as rollover)
- **Write Down**
  - A depreciation amount and a position relevant for this type of write-down must be available for a write-down to take place.
  - Update of write-down
- **Accrue/Defer**
  - Before the accrual/deferral run can be started all the operational flow transactions must have been transferred and processed.
  - Update of accrual items (such as accrued interest).
  - Update of deferral items (such as accrued discount).
- **Release**
  - One of the prerequisites from a business perspective for performing this step is that valuation remnants must exist. These are created when the valuation method of a position is changed (by a reclassification or when a hedging relationship is dissolved, for example).
  - Update of valuation remnants (valuation remnant hedge adjustment, valuation remnant fair value adjustment, for example)
- **Value Financial Products**
  - The accrual/deferral run must be completed before the valuation run can be started.
  - Transfer and processing of analytical events (such as hedge designation or dissolution, and reclassification decisions).
  - Update of valuation components (such as hedge adjustment, fair value adjustment, and risk provision)
- **Value Foreign Currency Positions**
  - The valuation run for financial products must be completed before foreign currency positions can be valued.
  - Update of amounts in functional currency

- Update of amounts in local currencies

- **Classify**
  - Determination of the relevant balance sheet reporting item (such as G/L account).
  - Reclassification of financial positions due to a change in characteristics relevant for balance sheet reporting (for example, due to a change in asset/liability status or a change in holding category)
- **Carry Forward**
  - Carryforward of balances to determine the year-opening balance of the subledger for financial instruments
  - Transfer of subledger accounts during year-end activities

Operational or analytical events can change characteristics. This is relevant for the following:

- Accrual method
  For example: According to the non-accrual practice, no further adjustments are made to accrual items; any accrual items up to this date are to be dissolved.
- Valuation method
  For example: After the holding category has been changed from *FV (P & L)* to *Amortized Cost*, the fair value adjustment created up to this date must be updated and entered against a fair value adjustment to be deferred.
- Balance sheet reporting
  For example: If the holding category is changed, positions are no longer reported in the G/L account *Receivable F & L* but in the G/L account *Receivable Amortized Cost*.
- Reporting of results
  For example: If the holding category is changed, positions are no longer reported in the G/L account *Interest Income FV P & L* but in the G/L account *Interest Income Amortized Cost*.

Implementation Considerations

Customizing

1. If you want to implement the subledger scenario, you must make an explicit decision to do so. This decision is made for each **client** and cannot be changed.
   - In Customizing, choose *Bank Analyzer* > *Basic Settings* > *Settings for Accounting* > *Choose Accounting Scenarios for Clients*.
2. Since the subledger scenario is based on the IFRA architecture and thus requires activities in several layers within Bank Analyzer, you first have to configure **general settings** in Customizing for *Bank Analyzer*:
   - To edit the **legal entities** and **accounting systems**, choose *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities*.
   - To make Customizing settings for the **application log**, choose *Basic Settings* > *Accounting Settings* > *Subledger Scenario: General Settings* > *Edit Application Log*.
   - To make Customizing settings for **archiving**, choose *Basic Settings* > *Accounting Settings* > *Subledger Scenario: General Settings* > *Archiving*.

   For more information, see General Settings/Applications (Subledger Scenario) [page 696].
3. The activities for **controlling the accounting processes** are available in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products*.

4. A report is available that you can use to check the **consistency** of the settings made in Customizing for the accounting processes. The system generates an application log that contains information about any inconsistencies, their causes, and how to resolve them.
    - In Customizing, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Tools* > *Check Consistency of Customizing Settings* . For more information, see the Customizing documentation.

SAP Easy Access

You can find the activities for the accounting processes in the *SAP Easy Access* menu. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* .

More Information

Processes in the AFI Scenario [page 678]

4.4.5.1 General Settings/Applications (Subledger Scenario)

Basic settings and processes that relate to both the accounting processes and balance processing in the subledger scenario are described here. These include general settings in Customizing, which you have to configure before you can define the Customizing settings for the accounting processes and balance processing. The general activities on the *SAP Easy Access* screen for the subledger scenario are also described.

Consistency of Customizing Settings for Accounting Processes

SAP provides a report that you can use to check the consistency of the settings made in Customizing for the accounting processes. When you run this report, the system generates an application log that contains information about any inconsistencies, their causes, and how to resolve them.

In Customizing, choose *Financial Services* > *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Tools* > *Check Consistency of Customizing Settings* . For more information, see the documentation for this Customizing activity.

4.4.5.1.1 Legal Entity

Use

The legal entity is a defining characteristic at master data level in Financial Position Management. It is predefined by SAP. The legal entity is also an input field for most transactions. You must assign at least one accounting system to every legal entity. Assigning multiple accounting systems to a legal unit ensures that business transactions and

financial positions can be processed using different accounting standards. If an accounting standard applies to several legal entities, you must assign the same accounting system to the corresponding legal entities.

Prerequisites

Before you can post to a legal entity, you must make the following settings in Customizing:

Procedure

1. To define a legal entity, choose *Financial Services* > *Bank Analyzer* > *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Define Legal Entities*
2. You must assign a legal entity to at least one accounting system. To do this, choose *Financial Services* > *Bank Analyzer* > *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Define Legal Entities* (view: *Assign Accounting Systems*)
3. To determine the exchange rate in the currency translation in Accounting, you have to assign a market data area to a legal entity. Choose *Bank Analyzer* > *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Assign Market Data Areas to Legal Entities and Accounting Systems*.
4. To assign document numbers, choose *Bank Analyzer* > *Results Data Layer* > *Settings for Flow Results* > *Edit Intervals for Document Numbers*

4.4.5.1.2 Financial Position Class (Subledger Scenario)

Use

Each financial position class has a key with a maximum of five digits. Every financial position belongs to a financial position class. This class is used to group a particular combination of defining and descriptive characteristics of the financial position.

Defining characteristics are divided into **defining characteristics at master data level** and **defining characteristics at valuation level**. Some of these are predefined by SAP, such as the legal entity (/BA1/C55LGENT). You do not need to create these fixed, defining characteristics in Customizing. They are automatically included in the business transaction structure and the document structure. Some of the defining characteristics can be freely selected. These include the ISIN and the securities account number, for example.

Some of the **descriptive characteristics** are also predefined by SAP, such as the financial position type (/BA1/C55AOTYPE). You do not need to create these fixed, descriptive characteristics in Customizing. Some of the descriptive characteristics can be freely selected, such as the production control.

General Financial Positions

General Financial Position

Every financial position class has a product category. The product categories are specified by SAP. Unlike the financial positions in the merge scenario, the **general financial position** product category cannot be assigned to individual financial instruments.

You have to define a special financial position class for the *General Financial Position* product category. The system automatically adds the legal entity to this financial position class as a defining characteristic at master data level.

The system adds the following characteristics as defining characteristics at valuation level:

- Accounting System
- Financial Position Currency
- General financial position category /BA1/C55GFPCAT

> **Note**
>
> If you use financial statement entities for the accounting processes that are different from the legal unit, the system adds further characteristics at valuation level. These can include the *Business Segment* and *Profit Center* characteristics, for example. The characteristics that are added depend on your settings for the dimensions of financial statement entities in the Results Data Layer (RDL).

The financial position type is the default descriptive characteristic.

You can add further defining characteristics at valuation level to restrict the general financial positions to the dimensions that you want to analyze separately later.

> **Note**
>
> You cannot add descriptive characteristics.

After you have defined a special financial position class for general financial positions, you must define a special financial position type for general financial positions and assign it to the financial position class for general financial positions.

In the case of general financial positions, we use the general financial position category *Default Category for AFI Scenario* in the procedure templates for Accounting. See Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Control* > *Display Procedure Templates*.

Other general financial position categories are obsolete and are used only to display old positions.

Activities

- You define a financial position class and assign defining and descriptive characteristics in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *Before Generation* > *Financial Positions* > *Define Financial Position Classes*.
- You define a financial position type in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *Before Generation* > *Financial Positions* > > *Define Financial Position Types*.

4.4.5.1.3 Financial Position Type (Subledger Scenario)

Use

The financial position type is a refinement of the financial position class. This means that multiple financial position types can be assigned to a financial position class. The financial position type is independent of the accounting system. It determines the splitting strategy and is also used to determine derivation rules in Accounting.

Customizing Example

Bond and *stocks* could be financial position types for a *securities* financial position class

Product type	Financial position class
Bonds	Securities
Stock	Securities

Activities

To edit the financial position type, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Before Generation* > *Financial Positions* > *Define Financial Position Types* in the Implementation Guide (IMG).

4.4.5.1.4 Data Required from the Source Data Layer (SDL)

Use

If the components of a structured product are related to each other within Accounting (for example, they are to be disclosed together in reporting), you must map this product in the SDL as a template hierarchy. The only exception to this rule is the balance method *Pooling*.

When you configure the templates in the SDL, you must activate the *Financial Position Management* tab page.

Features

No data is stored in the positions. For this reason, all business transactions must be supplied for products, that is, also those that comply with GAAP, such as dividends and interest payments.

Cash flows or conditions for interest accrual/deferral must exist in the financial instruments or transactions.

4.4.5.1.5 Mapping of the Valuation Structure in Accounting

Use

The mapping of SDL templates in the Risk Basis and Accounting forms the basis for modeling bank products. To ensure that the financial positions are correctly assigned, the configuration of the generated valuation structure must also be correct for Accounting.

Integration

Data Structures

The SDL stores its data in a generic structure that you can define to suit your needs. Risk Basis, on the other hand, needs a distinct, interpretable structure to calculate theoretical prices, for example. Therefore, in Risk Basis, the SDL objects are displayed in a separate data structure (**valuation structure**) that can be interpreted by Risk Basis. Die SDL objects have to be mapped in the valuation structure. This is performed by a **mapper**.

Features

Basis

In accounting, the financial positions required for accounting and valuation that is in line with IAS are created the moment the first business transaction for a financial transaction or financial instrument is posted to an account or a securities account. These financial positions are a data that consists of defining characteristics.

In accounting, "plain vanilla" financial transactions or financial instruments such as stocks, floaters or fixed-interest bonds can usually be represented from an IAS perspective by a single financial position.

Structured financial transactions, however, usually need several financial positions in accounting to enable them to meet the IAS requirements for correct accounting and valuation. For example, IAS requires a separate key date valuation of the derivative transaction and the main transaction for embedded derivatives [page 900], which means that the financial positions have to be processed separately for these particular transaction parts (see Key Date Valuation (Subledger Scenario) [page 934]).

When you post the first business transaction that has been transferred from the SDL to Accounting for a financial transaction or financial instrument, you decide how the structure of such a product is to be mapped to financial positions for structured products . Accounting is closely linked to the structures of the valuation structure that is generated from the SDL transaction by Risk Basis. In some cases, these structures can differ significantly from the structures delivered in the SDL. The basic rule is that the level of granularity of the financial position management and valuation in Accounting can never be higher than that of the transaction mapping in the valuation structure.

If the valuation structure exists, Accounting has to decide which node in the valuation structure should also be made into financial positions. Each resulting financial position must have a unique reference to the valuation structure node. If possible, it should also have a unique reference to the SDL subobject. These unique references ensure that, for each financial position, the system can determine the necessary key figures by means of the fair value server, which in turn accesses Risk Basis functions. You do not store the references directly in the financial position, but in a separate table (*/BA1/TB0_BARIMAP*) in Accounting. This helps to avoid dependencies and

changes to the characteristics of the financial position if, over time, the potentially volatile nodes of the valuation structure change or refer to other subobjects in the SDL.

The accounting-specific mapping is implemented using a BAdI with a default implementation (*/BA1/B1_MAPPING_FPO*). This default implementation carries out the mapping without any template-specific information by means of a generic algorithm. In exceptional situations, you can also implement your own product-specific mapping. In Risk Basis, you can also use BAdIs in the valuation or in the SDL Risk Basis mapping.

Mapping a Valuation Structure to Financial Positions

The information from the SDL service module *Financial Position Management* that originates from an SDL subobject is also passed on to the valuation structure nodes, and can be accessed from these nodes. The data in the service module contains the *Generate Financial Position Object* field. The use of the algorithm presupposes that you generate a financial position for each valuation structure node, and that the **Generate Financial Position Object** field is set in the associated SDL subobject in the service module *Financial Position Management*.

If the valuation structure node does not have a reference to the SDL subobject (because it is a synthetic node), the system has to read all of the nodes below it. If the *Generate Financial Position Object* field is set in one of the nodes below, a financial position must be generated for the synthetic node without a reference to the SDL subobject. This means, of course, that no more separate financial positions can be generated for the nodes below this one.

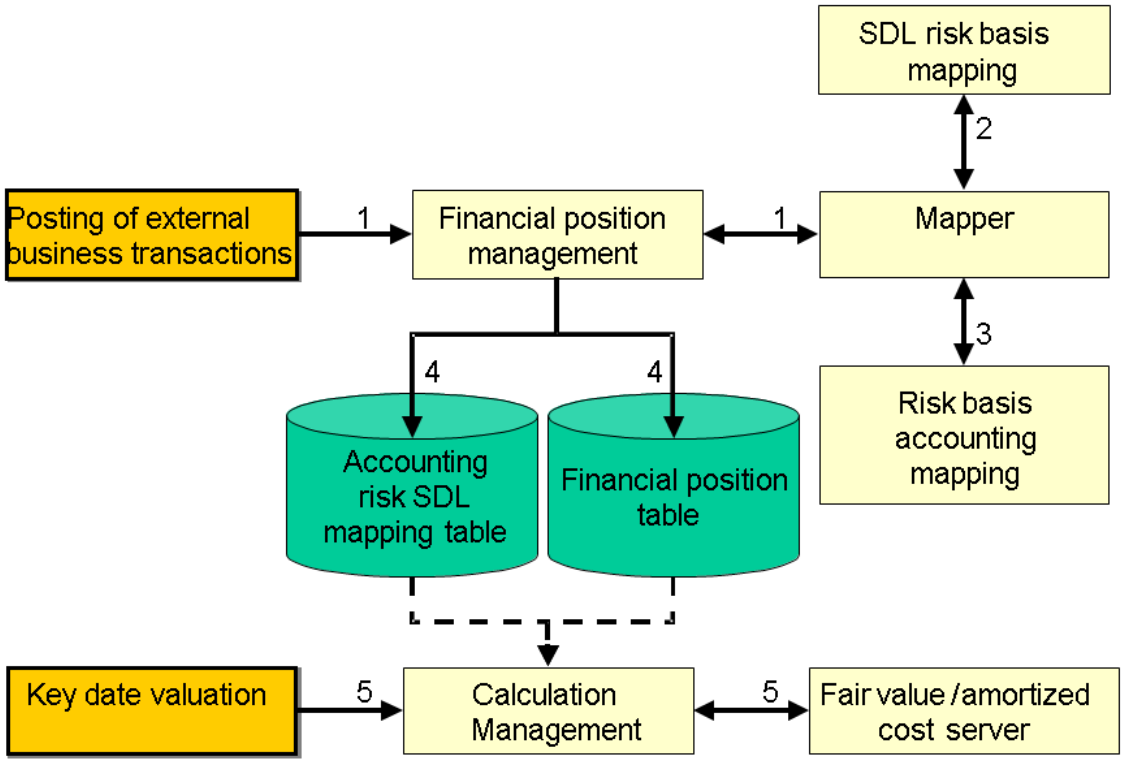
Several SDL subobjects can belong to one valuation structure node. The restriction for customers with this procedure is that they can only deliver data at the level of the SDL structures (subobjects) and not at the level of the nodes in the valuation structure. You can solve this problem by choosing the SDL structure that most closely resembles the valuation structure from the start. You can deliver only external information to the SDL objects.

In the Accounting processes that require the key figures from the fair value server (for example, Update Secondary Business Transactions [page 786] and key date valuation), you can look up which valuation structure nodes and SDL financial transactions and instruments have been assigned to a financial position when querying fair values and so on in the BARIMAP table. Using this information, you can make a request to the fair value server for this financial position.

For more information, see Generation of Financial Position Objects for Structured Products [page 750].

Internal Display and Generation of Financial Positions

The generation of financial positions in Accounting is controlled by Financial Position Management.



The figure above shows the generation process from the accounting perspective and consists of the following steps:

1. Reading the business transaction from the SDL
   The business transaction is imported with all of its items from the SDL.
2. SDL Risk Basis Mapping
   The system reads the appropriate accounting standard from the *Accounting Standard* tab page in the Customizing activity *Define Accounting System* under *Bank Analyzer* > *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities*. The system later uses this accounting standard for processing the *Financial Position Management* tab page.
   Using the financial position GUID, the function module can access the defining characteristics of the financial position and the IDs of all of the relevant SDL primary objects.
   The BAdI */BA1/B1_MAPPING_FPO* is called with the IDs of the financial transaction or instrument that belongs to a financial position. This BAdI contains the default implementation of the mapping for Accounting, as defined by SAP. In this BAdI, the complete template, including all of the subobjects and object relationships, is imported into a CKFC (complex key figure for object) by means of the financial transaction or financial instrument ID.
   The cash flow record for the accounting system is then read from the Customizing activity *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Basic Settings* > *Assign Global Environments to Accounting System*.
3. Risk Basis - Accounting Mapping

Using the result of the SDL Risk Basis mapping, an internal table is built that stores essential data for every financial position to be created. This internal table */BA1/B1_TTY_MAPPING_FPO* consists of the following fields:

Internal Table with Data for Financial Position Generation

Field Name	Component Type	Data Type	Length	Dec.Pl.	Short Description
CLIENT	MANDT	CLNT	3	0	Client
ID	/BA1/B1_DTE_MAPPING_FPO_ID	CHAR	32	0	ID of mapping line
ACC_SYSTEM	ACC_SYSTEM	CHAR	8	0	Accounting system
EXT_CONTRACT	/BA1/F1_DTE_EXT_CONTRACT	CHAR	40	0	External number
PRODTYPE	/BA1/F1_DTE_PT	CHAR	2	0	Template category
ROBJID	/BA1/R0_DTE_OBJECT_ID	CHAR	42	0	Object ID of the valuation structure
ROCONTID	/BA1/R0_DTE_OBJECT_ID	CHAR	42	0	Valuation structure: ID of subobject - part 1
ROSUBNID	/BA1/R0_DTE_ROSUBN_ID	CHAR	30	0	Valuation structure: ID of subobject - part 2
PRODUCT	/BA1/F1_DTE_PROD	CHAR	10	0	Template of the SDL product
VERSION	/BA1/F1_DTE_PRODV	CHAR	10	0	Template version of the SDL product
FATHER_PRODUCT	/BA1/F1_DTE_FPROD	CHAR	10	0	Template hierarchy parent template
NODE_NO	/BA1/F1_DTE_NODE_NO	NUMC	5	0	Node number
FPO_COMPONENT_CLASS	/BA1/F1_B0_DTE_FPO_COMP_CLASS	CHAR	1	0	Component class for financial position

Field Name	Component Type	Data Type	Length	Dec.Pl.	Short Description
FPO_RESIDUAL_ACQUISITION	/BA1/F1_B0_DTE_FPO_RES_ACQUISI	CHAR	1	0	Indicator for determination of residual acquisition value
CURRENCY	/BA1/B1_DTE_MAPPING_FPO_CURR	CUKY	5	0	Financial position currency
IS_ROOTNODE	BOOLEAN	CHAR	1	0	Is set to X for the root node of a product; otherwise set to BLANK

As a result, the values entered on the *Financial Position Management* tab page are passed through to Accounting. This means that the system generates a separate financial position for every node of a structured financial instrument or financial transaction where the *Financial Position Object* field is set. However, if you have not made any entries on the *Financial Position Management* tab page for a financial transaction or financial instrument, the system automatically sets the *Financial Position Object* characteristic for the top node. Due to different currencies in the cash flow, a financial position is created for every currency for financial instruments or financial transactions that are depicted in the SDL without a template hierarchy, but in Accounting as part of a structure. This occurs when you use the SDL sample templates for forward exchange transactions and multicurrency bonds. Financial instruments and transactions that are not structured in the SDL or in Accounting do not require entries on the *Financial Position Management* tab page. In this case, the system implicitly transfers a setting to the internal table. The system returns the internal table as a result of the mapping process.

1. Generating Financial Positions
   The internal table that is returned together with the data from the business transaction gives you all of the information needed to create a financial position. The relevant information is either stored in the financial position as a defining characteristic, a descriptive characteristic or as a status. Alternatively, it is stored as index information in table BARIMAP for future requests to the fair value server. You cannot regenerate or change financial positions at a later date. If you need to directly address components of structured templates in a business transaction, the following characteristics in the external business transaction row must be configured and populated with values:
   - /BA1/C11FPROD
     Parent template (fixed field)
   - /BA1/C11SPROD
     Child template (fixed field)
   - /BA1/C11NODENO
     Node number (fixed field)
   - /BA1/C55USELEG
     Currency leg (optional field)

   For products that contain a structure in the SDL (for example, warrant bonds or swaps), you only need to enter the first 3 characteristics. For products that are unstructured in the SDL but structured in Accounting (for example, forward exchange transactions and multiple currency bonds), you only need to enter */BA1/C55USELEG = X* to call up a currency leg of the transaction. This finds the financial position in

the business transaction item along with the transaction currency (usually transferred from the Position Currency field).

2. Valuating Financial Positions
   The risk-basis mapping data used when generating master data for the financial position is stored in Accounting. If a valuation of the financial positions occurs, the system reads the Risk Basis mapping data stored in the BARIMAP table and transfers it to the fair value or accounting server. Thus, a valuation on the basis of the granularity level of Risk Basis is possible.

4.4.5.1.6 Modeling of Products (Subledger Scenario)

You can use two categories of financial product to model financial positions: Financial transactions and financial instruments. A financial position can be a stock in a securities account, for example.

An example is provided of how the following products are modeled for the Source Data Layer and for Accounting:

- Unstructured Product [page 705]
- Structured Product [page 707]
- Multiple Currency Product [page 709]
- Partly Hedged Product [page 711]

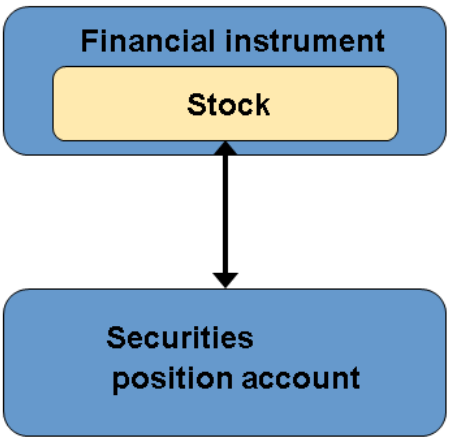
The emphasis here is not on describing the individual examples in great detail, but rather on demonstrating and explaining the basic modeling principles.

4.4.5.1.6.1 Unstructured Product (Subledger Scenario)

The unstructured product is a transaction that is not split into individual components during modeling, such as a stock, bond, or loan. The following example shows how a stock can be modeled.

Modeling in the SDL

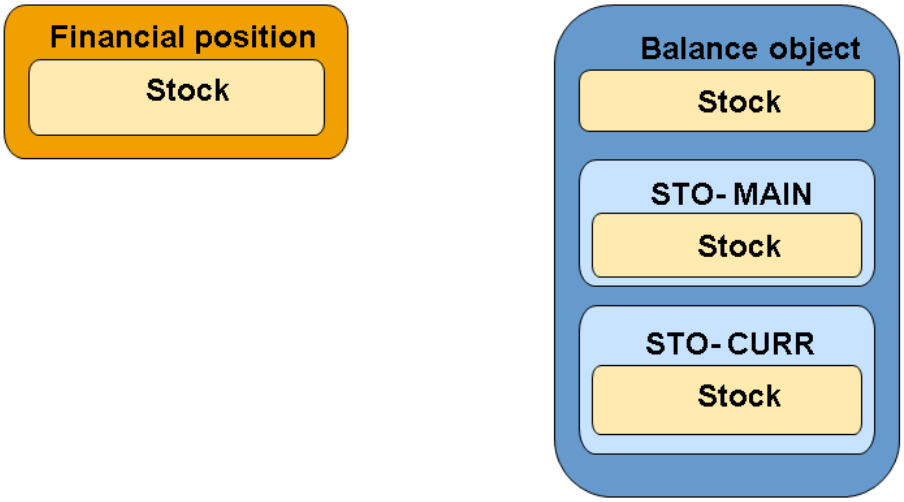
You can model a stock in the SDL using an SDL primary object *financial instrument* and a *securities position account*. The *financial instrument* does not have a hierarchical structure in this case. The model described is shown in the figure below:



Modeling of a Stock in the SDL

Modeling in Accounting

You model a stock in Accounting by using **one** financial position. In balance processing, there is a balance object with balance subobjects of the balance subobject category MAIN and CURR. The model described is shown in the figure below:



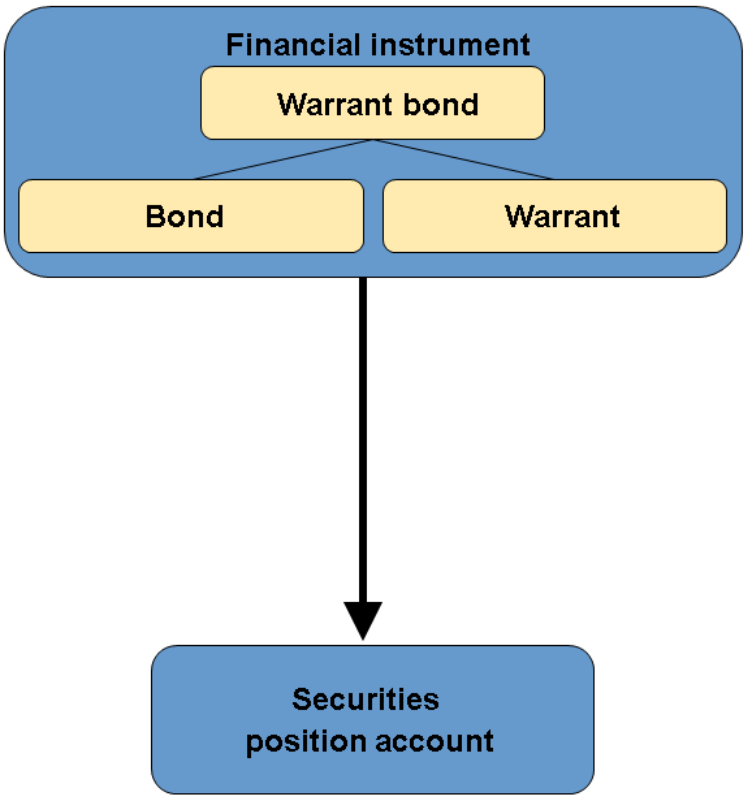
Modeling of a Stock in Accounting

4.4.5.1.6.2 Structured Product (Subledger Scenario)

A structured product is a transaction that must be divided into its component parts either for valuation reasons or to allow the parts to be displayed separately in reporting. The example below involves a warrant bond.

Modeling in the SDL

You model the warrant bond in the SDL using a hierarchically structured financial instrument. This financial instrument consists of a parent node (warrant bond) and two subordinate nodes (bond, option). You also require a securities position account. The model described is shown in the figure below:

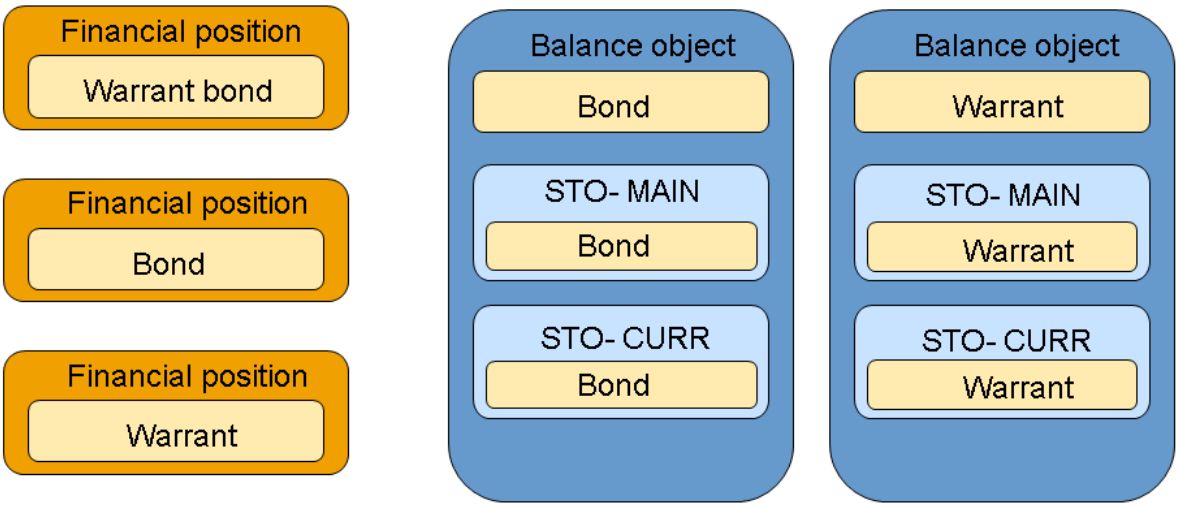


Modeling of a Warrant Bond in the SDL

Modeling in Accounting

Because the option and bond have different valuations, three financial positions for the warrant bond are created in Accounting. There is one financial position for the bond, and one for the option. A separate financial position is also required for the warrant bond to include external business transactions. In balance processing, there are

balance objects each with a balance subobject of the balance subobject category MAIN and CURR. The model described is shown in the figure below:



Modeling of a Warrant Bond in Accounting

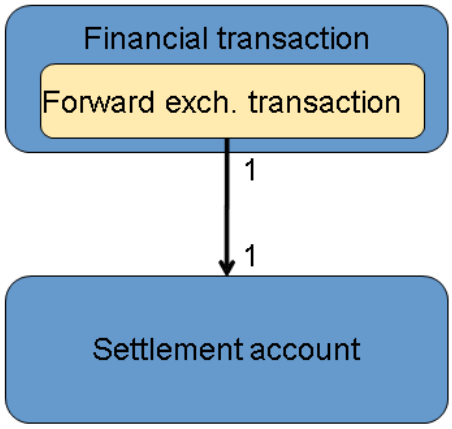
> **Note**
>
> Use a custom characteristic with the values *main part* and *derivative* as an additional defining characteristic for the balance object. You derive this characteristic from the fixed characteristic *Component Class* (/BA1/C55COMPCL) in Accounting.

4.4.5.1.6.3 Multiple Currency Product (Subledger Scenario)

A multiple currency product is a transaction that has at least two object currencies. The modeling example provided here involves a forward exchange transaction.

Modeling in the SDL

You model the forward exchange transaction in the SDL using an unstructured financial transaction. You also require a settlement account. The model described is shown in the figure below:

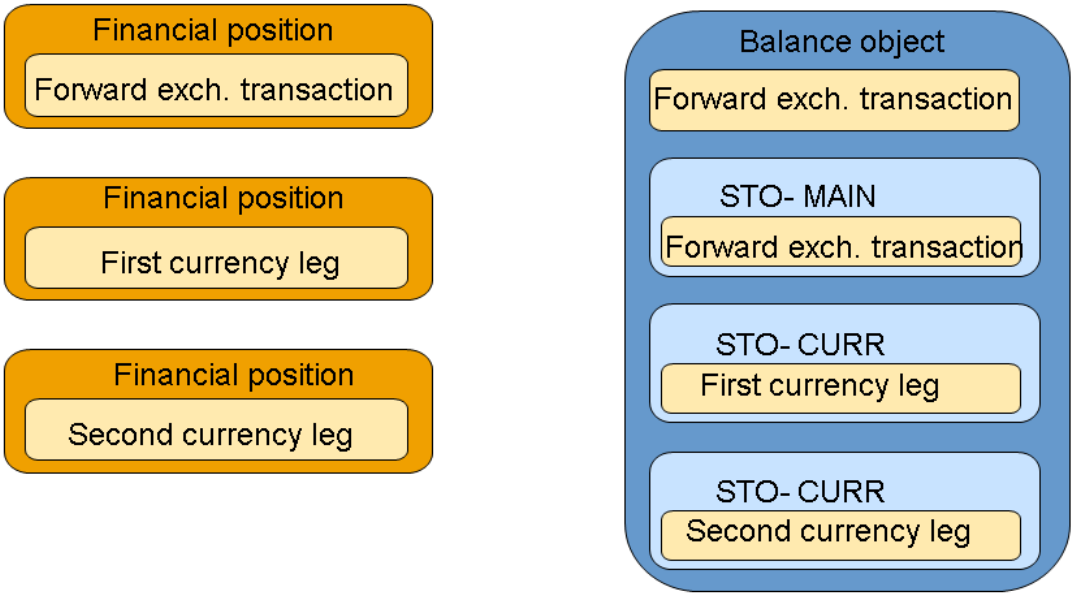


Modeling of a Forward Exchange Transaction in the SDL

Modeling in Accounting

A financial position for the entire forward exchange transaction is created to include the external business transaction; and a further financial position is created for each currency. In balance processing, there is a balance object with a balance subobject of the balance subobject category MAIN and with a balance subobject of the

balance subobject category CURR for each currency leg. The balance objects of the balance subobject category CURR have the same balance subobject type. The model described here is shown in the figure below:



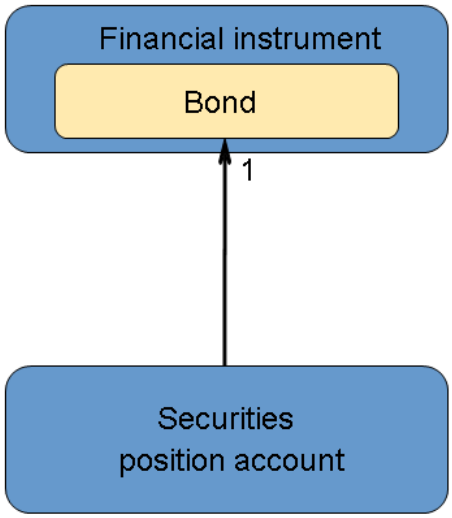
Modeling of a Forward Exchange Transaction in Accounting

4.4.5.1.6.4 Partly Hedged Product (Subledger Scenario)

A partly hedged product is a transaction that is partly involved in a hedging relationship. In the example below, the partly hedged product in question is a bond.

Modeling in the SDL

In the SDL, a partly hedged bond is modeled in the same way as an unhedged bond. You use a financial instrument and a settlement account. The model described is shown in the figure below:

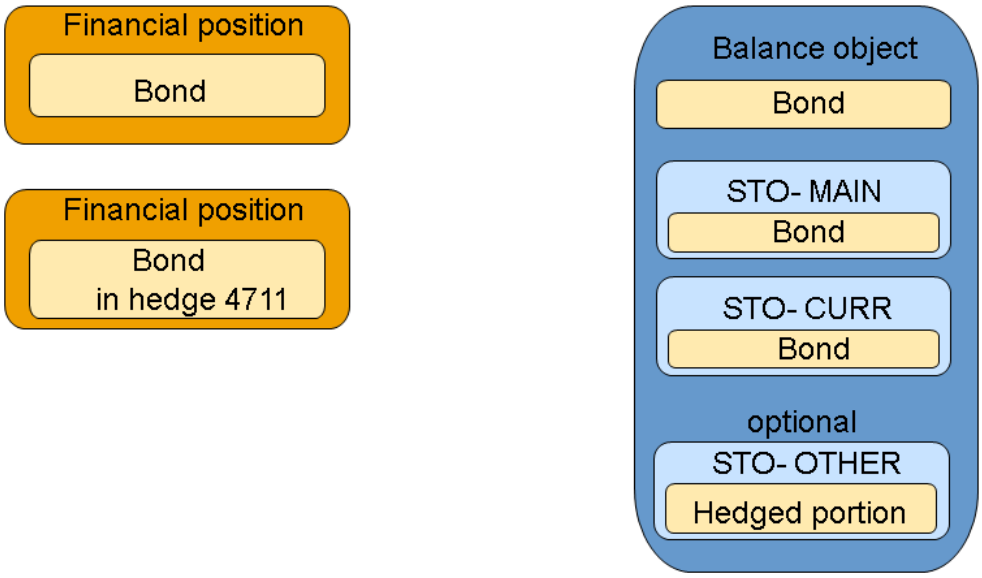


Modeling of a Bond in the SDL

Modeling in Accounting

In addition to the standard financial position in Accounting, you can use other financial positions (one per hedging relationship). In balance processing, there is a balance object with balance subobjects of the balance subobject category MAIN and CURR. The distinction between hedged and unhedged products is not usually important when displaying bonds on a balance sheet (not including the subitems or additional data for reporting). To be able to

display the hedged part in the notes, however, you also need a balance subobject from the balance subobject category OTHER. The model described is shown in the figure below:



Modeling of a Bond in Accounting

> **Note**
>
> If you want to display the hedged and unhedged parts separately in reporting, you have to create an additional balance object for the hedged part.

4.4.5.1.7 Generation (Subledger Scenario)

Use

The purpose of the generation function is to read characteristics and key figures and generate database tables and field structures.

Customizing for the accounting processes consists of the activities *Before Generation* and *After Generation*. You can start the generation process once you have made all of the Customizing settings contained in the *Before Generation* activities.

> **Note**
>
> Note that this function does not generate the document structure in the subledger scenario. You create the document structure in Customizing for the Results Data Layers (RDL).

Prerequisites

Before generation you have edited all the Customizing activities under the following nodes:

- *Bank Analyzer* > *Basic Settings*
- *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Before Generation*

Features

Creation of a Business Transaction Structure

At the start of generation, characteristics and key figures are collected from different origins and then used to create the Accounting business transaction structures.

The header and the item structure form the basis of the structure of the business transaction in Accounting. These structures are created as user-specific. Characteristics and key figures can have the following sources:

- You can specify if the characteristics or key figures are generated in the header or item of the Accounting business transaction in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *Before Generation* > *Business Transactions*:
  - *Edit Additional Characteristics for Business Transactions*
  - *Edit Additional Key Figures for Business Transactions*
- You can use the defining and descriptive characteristics of the financial position.
- You can use the characteristics (InfoObjects) that you have entered for the following rows in Customizing for the SDL under *Register OTS/DataStore Objects*:
  - Financial transaction
  - Financial instrument
  - Settlement account
  - Securities position account
- The following characteristics that have been predefined by SAP are automatically configured in the Accounting business transaction and Accounting document. For this reason, you do not need to specify them as customer characteristics:
  - Business transaction type
  - Accounting system
  - Legal entity
  - Text
  - Reset business transaction reference
  - Sender ID and sender key as the keys for the business transaction

- Item type
- Amount and currency in financial reporting currency
- Amount and currency in object currency
- Amount and currency in nominal currency
- Quantity and unit of measure

> **Caution**
>
> If the Customizing activity *Register OTS/DataStore Objects* in the SDL has not been processed in full or in part, the system cannot find characteristic names for the SDL fields *Financial Transaction, Financial Instrument, Settlement Account* and *Securities Position Account*. This inevitably leads to errors in the generation process.

Other Generated Structures

In addition to the business transaction structure, the generation process is still responsible for the creation of other entities. For example, the following structures are generated:

- Segmentation service structures that are required for the persistence of financial position data
- Structures containing characteristics that are available for derivations in the posting rules

Generation

You can either edit the Customizing settings directly for a client or have these settings transported. Regardless of the strategy chosen, you must work consistently; you cannot combine the direct processing of settings with the use of transports. It is possible to switch from the transport strategy to directly editing the Customizing settings once. This also applies to the Customizing settings for the accounting processes, particularly the activities under *Before Generation*. The following conditions apply to both processes:

- **Editing the Customizing settings directly**
  We recommend that you edit the Customizing activities in the sequence in which they are arranged in Customizing for the accounting processes under *Before Generation*. If you change any Customizing settings you must restart the generation process.
- **Using a transport to supply the Customizing settings**
  You start the generation process in the source system. The generation process is triggered in the target system when the corresponding transport requests are imported. If you have configured the automatic recording of the transport for the client, an entry is automatically inserted into the queried transport request when the generation process is run. This then triggers the generation process in the target system when the request is imported.
  Alternatively, you can insert an entry into a transport request in the Customizing activity *Start Generation* using the *Transport* function. This entry then triggers the generation process in the target system when the request is imported. If different transport requests are used to transport Customizing activities, they must be imported in the same order in which they appear in Customizing. If, for example, three different requests are used (*Before Generation*, *Generation*, and *After Generation*), you must import these in this order.

> **Note**
>
> It only makes sense to transport the generation (that is, release a corresponding request and import it into the target system) if the generation process in the source system was successful and the log does not contain any error messages.

If you want to delete SGS fields from segmentation service structures, select the corresponding checkbox in the subledger scenario. If you do not select this checkbox the system does not display the deleted fields but these

fields still exist in the segmentation service structures. After you have deleted these fields you have to manually regenerate the segmentation service structures (in the case of transports, you have to do this in the target system too).

Changes After Generation

When the system is in operation, you may have to change the Customizing settings made in the *Before Generation* activities. For example, if you require additional characteristics or key figures and thus create new financial position classes [page 697] or financial position types. In this case, you have to restart the generation process. The generation process affects subsequent postings only. Documents that have already been posted are not affected by the generation process.

> **Note**
>
> If the Customizing settings were transported and changes have been made in the source system, you must transport the Customizing changes *Before Generation* and the generation in the target system.

The following conditions apply when you add new characteristics or key figures:

1. If the characteristic you are adding appears on the inbound interface of the business transactions (Customizing activity *Edit Customer Characteristics for External Business Transactions*) or appears as a defining or descriptive characteristic for the financial position (Customizing activities under *Financial Positions*) then after you have restarted the generation process you have to check where the characteristic is used:
    1. If the missing characteristic is a defining characteristic for the financial position, adding it later means you have to adapt a wide range of Customizing settings for the financial position processes. Changing a characteristic within a fiscal year means that you have to repost all the business transactions for that year in a new accounting system, because the previous postings are incomplete.
    2. If the new characteristic is simply a source characteristic for derivations, you can simply include it as an optional source characteristic. You then only have to adapt the derivation rules. If it is a target characteristic (such as a flow type), you also have to revise the settings in the subsequent Customizing activities.
2. When you add a key figure the information under 1.2. applies because key figures must not be used as defining characteristics.

Activities

1. Start the generation process by choosing Customizing for *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Before Generation* > *Start Generation*.
2. Edit the activities under Customizing for *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation*.

4.4.5.1.8 Opening and Closing Posting Periods (Subledger Scenario)

Use

You use this transaction to determine whether postings can be made in periods of the fiscal year. This applies to both automatic and manual postings. When posting, the system uses the posting date to automatically assign the document to the corresponding posting period.

Prerequisites

You have defined at least one legal entity, one fiscal year variant, and one accounting system and assigned them to one another in the following Customizing activities under *Bank Analyzer* > *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities*:

- *Define Accounting Systems*
- *Define Fiscal Year Variant*
- *Define Legal Entities*

Creating special periods when defining a fiscal year variant, for example, for year-end postings, is presently **not** supported by Bank Analyzer.

Procedure

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Results Data Layer* > *Periodic Tasks* > *Open and Close Periods*.

1. Choose an accounting system and a legal entity.
2. Specify a fiscal year and a period.
3. Use the indicator to specify whether the posting period is open or closed.
4. Save your entries.

> **Note**
>
> You should reopen closed posting periods in exceptional cases only. This can arise if you have reversed posting after a posting period has closed. To correct the balances in the posting period, reverse the late reversal, open the posting period, and repeat the reversal. The balances are then correct from the reversal posting date. You can close the posting period again. For more information, see Reversal [page 795].

4.4.5.1.9 Initial Balance Transfer for External Accounting

Use

The initial balance transfer for external accounting (creation of the annual opening balance sheet when going live) describes the necessary steps that you need to carry out before going live with the scenario **Bank Analyzer Accounting for Financial Instruments**. You carry out this process once. In the solution, you can follow the standard approaches that are used in production operation.

Activities

There are two different cases:

- Transfer of balances from operational source systems and generation of accounting operational contributions for accrual items within an accrual run in Bank Analyzer.
- Generation of analytical contributions
  - Through transfer from an analytical source system
  - Through generation as part of a valuation run in Bank Analyzer

Case 1: Transfer and Generation of Value Date-Based (Operational) Contributions for Initialization Balance Sheet

Step 1: Transfer of Balances

In this step, the system uses *initialization business transactions* to transfer balances from operational source systems to generate the basis for the annual opening balance sheet. You use the business transaction category `1006` for this.

The following transactions are required for this step:

- Post External Business Transactions
- Update Secondary Business Transactions

Operational source systems define position management of financial products (such as account management systems or loans management); the balances transferred from here are always value date based.

> **Note**
>
> 1. To separate the effect of the offsetting entry from the effects of production operation in processing, entry is made against the processing category *Equity* `1BG_EQUITY`. For this purpose, create a separate key figure that you use in this context only.
> 2. Since the balances to be delivered in this context carry different information requirements, you should define separate business transaction classes for transferring operational balances and for transferring analytical balances.
> 3. You can initialize general financial positions in this step also using separate item classes. Different to item classes for the initialization of product-specific financial positions, general positions carry different characteristics. To use the standard method for business transactions, use a dummy object and a dummy account in the SDL business transaction.

Step 2 (technical): Setting of target values for processing operational (value date-based) key figures

This step is carried out by the following processes:

- Perform key date valuation for single financial positions
- Perform key date valuation for all financial positions

A separate key date valuation type controls the key date valuation.

With this step, the system initially sets the calculation bases for the deferrals to be calculated for deferral items in live operations. This step does not generate any posting documents but instead records the necessary entries in the results category *HFBAS Calculation Base of Fin. Pos.*. The results category defined through configuration is specified in the following settings in Customizing for Bank Analyzer:

Basic Settings > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Define Accounting Systems* When you have selected the accounting system, assign the relevant results category in the *Result Types* submenu by specifying *Calculation Base for Fin. Pos., AFI, Posting Date, Value Date* in the *Purpose* column.

For this step, use the procedure templates *1290 Initialization of Value Date-Based Calculation Bases*.

Step 3: Generation of Accrual Results in Bank Analyzer

In Bank Analyzer, accrual items can be transferred by means of business transactions, calculated (or transferred from shipped key date results), and updated. This is the standard procedure in production operation. Since no operational balances are shipped with business transactions for accruals and since there are no operational flow transactions for this, these can be generated as part of the normal accrual step.

Case 2: Transfer and Generation of Non-Value-Date-Based (Analytical) Contributions for Initialization Balance Sheet

Step 1: Transfer of Balances

In this step, the system uses initialization business transactions to transfer balances from analytical source systems to be replaced, in order to generate the basis for the annual opening balance sheet. You use the business transaction category `1006` for this.

The transactions required for processing these business transactions are defined as follows:

- Post External Business Transactions
- Update Secondary Business Transactions

When an analytical source system is replaced, the system transfers balances of analytical key figures in this step. The balances transferred from an analytical source system are always non-value-date-based.

> **Note**
>
> 1. To separate the effect of the offsetting entry from the effects of production operation in processing, entry is made against the processing category *Equity* `2BG_EQUITY`. For this purpose, create a separate key figure that you use in this context only.
> 2. Since the balances to be delivered in this context carry different information requirements, you should define separate business transaction classes for *transferring operational balances* and for *transferring analytical balances*.
> 3. You can initialize general financial positions in this step also using separate item classes. Different to item classes for the initialization of product-specific financial positions, general positions carry different characteristics. To use the standard method for business transactions, use a dummy object and a dummy account in the SDL business transaction.

Step 2 (technical): Setting of target values for processing analytical (non-value-date-based) key figures (only when an analytical source system is being replaced)

This step is relevant if the analytical key figures transferred by the system from an analytical source system to be replaced are not those that Bank analyzer initially generated in step 3. In step 3, the system updates the calculation bases to be used in live operations to calculate the following analytical key figures:

- Valuations to be dissolved
- Reserves to be dissolved
- Hedge adjustment
- Fair value adjustment

This step does not generate any posting documents but instead saves the necessary entries in the results category *HFBAS Calculation Base of Fin. Pos.*. The results category defined through configuration is specified in the following settings in Customizing for Bank Analyzer:

▶ *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Define Accounting Systems* ▶.

Once you have selected the accounting system, assign the relevant results category in the *Result Types* submenu by specifying *Calculation Base for Financial Positions, AFI, Posting Date* in the *Purpose* column.

For this step, use the procedure templates *2290 Initialization of Non-Value-Date-Based Calculation Bases*.

Step 3: Generation of Valuation Results in Bank Analyzer

Provided that no analytical source system is being replaced or that no balances are shipped for analytical key figures, the system generates these key figures using standard functions for key date valuation. The next step for both cases is defined as follows:

Since the balance sheet generated by the preceding processes (which includes the contributions for key date valuation) logically represents an annual opening balance sheet, you can use it to create an annual opening balance sheet.

To do so, also use the standard annual financial statement procedure for Bank Analyzer (Initialize Profit and Loss Accounts (Start of Fiscal Year)). This standard procedure ensures that the system generates the currency positions that logically exist through P&L contributions of the procedure and uses them in the annual opening balance sheet.

⚠ Caution

- You cannot import production business transactions with a posting date that is before or the same as the posting date of the initial balances.
- You cannot import production business transactions with a value date that is before or the same as the posting date of the initial balances.
- For processing business transactions for the initial balance transfer, the same statements apply as for processing production business transactions with regard to:
  - Reversibility
  - Error handling
  - Configuration
  - Processing logic

4.4.5.1.10 Schedule Manager: Task List

Use

You can use Schedule Manager to define task lists. A task list consists of individual steps and each step represents, for example, a program with a variant or a user action.

You can use the following programs in Accounting in a task list:

- Set SDL time stamp [page 367]
- Post external business transactions [page 774]
- Update secondary business transactions [page 786]
- Process results of effectiveness test (see Micro Fair Value Hedge [page 905])
- Start key date valuation [page 934] for all or individual financial positions
- Initialize profit and loss statement [page 951] (start of fiscal year)
- Generate simulated financial statement items [page 1453]

Features

Variants in Task Lists

You assign a variant to every program in a task list. You can give this variant any name that does not begin with *SAP&*. This variant determines the input field values to be used to run the program when the task list is scheduled. You can assign variables as well as fixed values using a variant to the input fields. For example, an input field can be assigned to a TVARV variable.

Schedule Manager Variables

You can enter the values for the input fields in the programs in a task list only when you start scheduling. To do this, a Schedule Manager variable must be assigned to the input field in the variant for a program. When the task list is scheduled, you can enter a value for each Schedule Manager variable. You can assign multiple input fields to the same Schedule Manager variables in the variants of the programs.

> **Example**
>
> For example, if the task list contains multiple programs with an input field for the Legal Entity [page 696], you can assign the Schedule Manager variable for the *legal entity* to each of these input fields. As soon as you schedule the task list and enter the value for the Schedule Manager variable for the *Legal Entity*, each of these input fields is filled with the corresponding value.

The list of Schedule Manager variables is predefined by SAP and cannot be changed. You can display this list using the transaction *sm34*, view cluster *VSMANTVARV*. The names of the Schedule Manager variables that are assigned to the *Balance Analyzer* application start with the prefix *SCMA/BA1/B*.

When you schedule the task list you cannot enter more than one value for each Schedule Manager variable. There are, therefore, no complex selection conditions. You can use this single value to fill the following categories of input fields:

- **Parameters**

A parameter is a simple input field. If you want to use a Schedule Manager variable to supply the parameters with the relevant data, you must enter this in transaction *STVARV* under *Parameters*. You do not have to specify a value here.

- **Selection Criteria**
  Selection criteria are represented by two input fields used to specify an interval, and a button for multiple selection. If you want to use a Schedule Manager variable to supply the selection criteria with the relevant data, you must enter this in the transaction *STVARV* under *Selection Criteria*. In this case, you have to enter at least one selection option for each variable. For example, *Option "="* and *initial values* for the upper and lower limits.

You can enter a Schedule Manager variable under both Parameters and Selection Criteria. The entries in the transaction *STVARV* are client-dependent and must, therefore, be repeated in every client where the Schedule Manager variables are used.

If you want to assign a Schedule Manager variable to an input field in the variant for a program, you must set the *Selection Variable* attribute for this input field in the variant. You can then make the assignment. Use the input help (F4) to display the variables from transaction *STVARV*.

> **Note**
>
> If you have assigned a value in transaction *STVARV* for this sort of variable, this value is only used if no value was entered for this variable when you scheduled the task list.

4.4.5.1.11 Parallel Processing Tool

Use

This function processes mass data in parallel. In Accounting, the parallel processing tool is used for the different background processes, such as financial reporting, posting external business transactions, and key date valuation.

Features

For more information about the parallel processing tool, including how to configure and start it, see .

Activities

To define the package size for parallel processing, use the Customizing activity *Define Custom Settings for Processes* under *Bank Analyzer* > *Basic Settings* > *Settings for Accounting* > *Technical Settings for Processes*.

To control the number of parallel jobs and how these are distributed to the existing servers, use the Customizing activity *Maintain Job Distribution* under *Bank Analyzer* > *Basic Settings* > *Parallel Processing*.

4.4.5.1.12 Application Log

Use

The system generates application logs for all application processes, such as updates of secondary business transactions and posting of external business transactions. These logs contain information about whether the processes were successful or whether errors occurred. For each application log, the system also generates statistics for the processed business transactions, financial positions, and balance objects.

Procedure

You can use the following functions in the **subledger scenario**:

- Display Processing Statistics [page 723]
- Delete Processing Statistics [page 724]

4.4.5.1.12.1 Displaying Processing Statistics

Use

You can use this transaction to delete the statistics for processed *business transactions, financial positions,* and *balance objects*. The statistics are generated at the same time as the application log [page 723].

Statistics are generated during the following processes:

- For processed business transactions:
  - Post External Business Transactions
  - Update Secondary Business Transactions [page 786]
- For processed financial positions and balance objects:
  - Key Date Valuation
  - Balance Processing

Prerequisites

The system has generated an application log and thus statistics for processed business transactions, financial positions, and balance objects.

Procedure

Subledger Scenario

To display the statistics for processed business transactions, financial positions, and balance objects, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Tools* > *Application Logs* > *Display Processing Statistics* on the SAP Easy Access Menu.

4.4.5.1.12.2 Deleting Processing Statistics

Use

You can use this transaction to delete the statistics for processed business transactions, financial positions, and balance objects. The statistics are generated at the same time as the application log [page 723].

Statistics are generated during the following processes:

- For processed business transactions:
  - Post External Business Transactions
  - Update Secondary Business Transactions [page 786]
- For processed financial positions and balance objects:
  - Key Date Valuation
  - Balance Processing

Transactions for deleting and displaying have been separated so that authorization can be issued for each individual transaction, but not necessarily for both.

Prerequisites

The system has generated an application log and thus statistics for processed business transactions, financial positions, and balance objects.

Procedure

Subledger Scenario

To delete the statistics for processed business transactions, financial positions, and balance objects, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Tools* > *Application Logs* > *Delete Processing Statistics* from the SAP Easy Access Menu.

4.4.5.1.13 Foreign Currency Concept (Subledger Scenario)

The foreign currency concept in *Bank Analyzer* is called Multi Currency Accounting (MCA).

MCA is based on an internationally valid system that depicts currency-specific scenarios in accounting. The basic principle of MCA is to maintain all accounts and positions in the original currency and to valuate central foreign exchange position accounts in all currencies (instead of individual position accounts) so that foreign exchange results are determined in a standardized way.

This ensures that financial reporting across all foreign exchange exposures is transparent and that the valuation of profits and losses related to foreign currency transactions is comprehensible.

Foreign Currency Types

The following foreign currency types are available in MCA in *Bank Analyzer*:

Foreign Currency Type	Definition
Transaction Currency	Currency of a business transaction, transaction or cash flow. This currency type is predefined by SAP.
Functional Currency	Central currency for all currency translations. You have to specify the functional currency for every legal entity because currency translations can only be carried out if the functional currency is involved.
Local Currency	Local currency of your legal entity in which you maintain your national cash journals. If the local currency and the functional currency are identical, you do not need to maintain the local currency for the relevant legal entities. .
Group Currency	General currency used for consolidation at group level.
Index-Based Currency	Fictitious currency used in countries with high inflation rates as a means of comparison for reporting.
Hard Currency	Used country-specifically as a second currency in countries with high inflation rates.

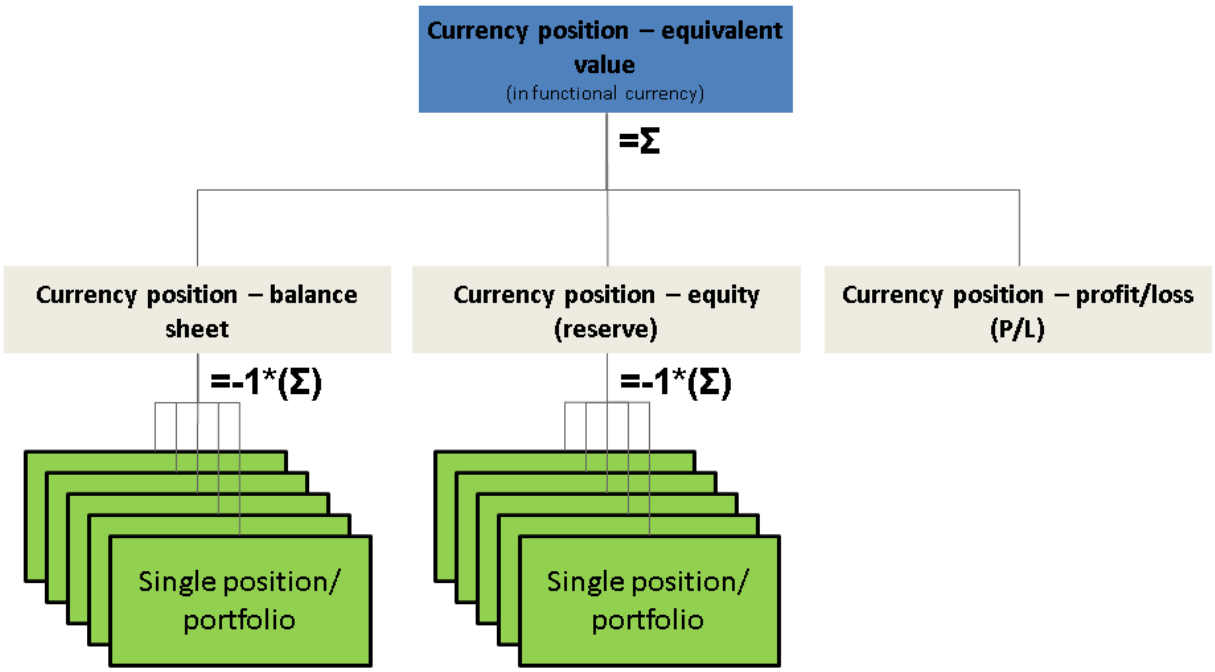
You assign foreign currency types to your legal entity in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Define Legal Entities* > (*Define Currency Translation* view).

Entering a functional currency is mandatory. Assigning all other currency types to your legal entity is optional.

Currency Positions in a Foreign Currency

The figure below shows the different foreign currency positions and how they are managed for each currency that occurs as a transaction currency.

The equivalent value currency position is equal to the total value of a particular currency in functional currency. In the balance sheet structure the equivalent value currency position is distributed between the balance sheet currency position, the equity currency position, and the income statement currency position.



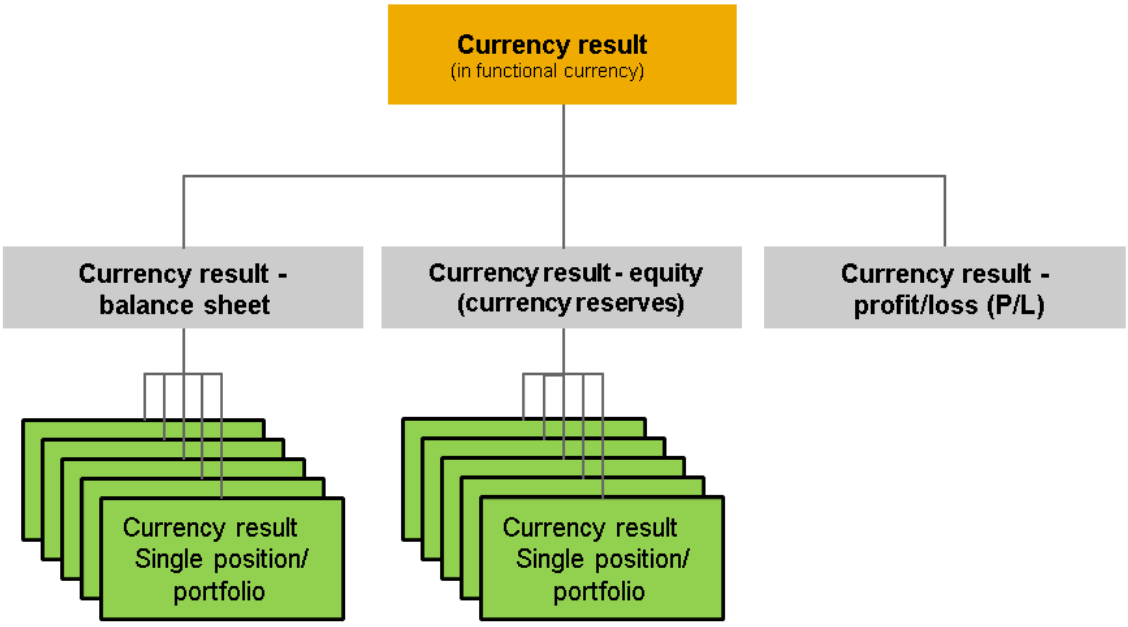
The balance sheet currency position and the equity currency position are based on the corresponding single positions or portfolios. The sum of the balance sheet currency position and its corresponding single positions is zero.

The sum of the equity currency position and its corresponding single positions is also zero.

As with all processing categories, a strict distinction is made between operational and analytical processing categories here to ensure a clear division between operational and analytical currency positions. This means that the tree structure in the figure above exists once for the operational currency position (for example, interest payment), and once for the analytical currency position (for example, result of fair value adjustment).

Foreign Exchange Result

The overall foreign exchange result comprises the balance sheet foreign exchange result, the equity foreign exchange result , and the income statement foreign exchange result. The figure below shows how this is put together along with the origin of these foreign exchange results from the relevant single positions or portfolios:



Posting Key Figures without Currency Reference

Posting key figures without currency reference are key figures that have no direct currency information. You can use them to post amounts in different currencies with a single line item.

The document below shows how a posting key figure is simultaneously posted in transaction currency (CHF), in functional currency (USD), and in local currency (EUR):

Document	Posting Key Figure	Amount in Transaction Currency	Transaction Currency	Amount in Functional Currency	Functional Currency	Amount in Local Currency	Local Currency
#1	In-Transit Account Payment Transactions	130.00	CHF	140.00	USD	100.00	EUR
	Receivable	-130.00	CHF	-140.00	USD	-100.00	EUR

Currency Translation

Every currency translation from a source currency to a target currency is performed based on the functional currency. The following conditions apply:

- If the source currency and target currency are identical, the system transfers the amount to the target currency without changing it (without translating). This applies irrespective of whether the two currencies are the same as the functional currency or not.
- If the source currency and target currency are not identical, the system automatically performs the following activities:
  1. The system translates the source currency amount into the functional currency.
     If the source currency is the same as the functional currency, the system transfers the amount unchanged.
  2. The system calculates the amount in target currency from the amount in functional currency.
     If the source currency is the same as the functional currency, the system transfers the amount without changing it.

Currency translations are only carried out if the functional currency is involved.

Currency translations are performed in this sequence in the following process steps:

1. When a business transaction is posted, the system calculates all currency amount fields that have not been filled (if the currency differs from the transaction currency).
2. When internal business transactions are derived, currency translations take place if an amount cannot be transferred from the document and if it is the result of foreign currency restatement. The system then determines all foreign currency information from the functional currency and offsets any rounding differences on the largest line item (based on the absolute amount).
3. For key date valuations, currency translation is required in the following process steps:
   - Determine currency gains
   - Balance sheet item valuation
   - Revaluation of balance sheet items and currency positions in all additional currency types

Foreign currency valuation

The actual foreign currency valuation has three steps that are always performed in the following sequence:

1. **Determine currency gains**
   The system performs the following activities:
   - Valuates the foreign currency positions in functional currency
   - Posts the valuation in transaction currency which corresponds to the functional currency
   - Translates all additional currency types from the amount in functional currency at the current rate of exchange and enters this amount in the document

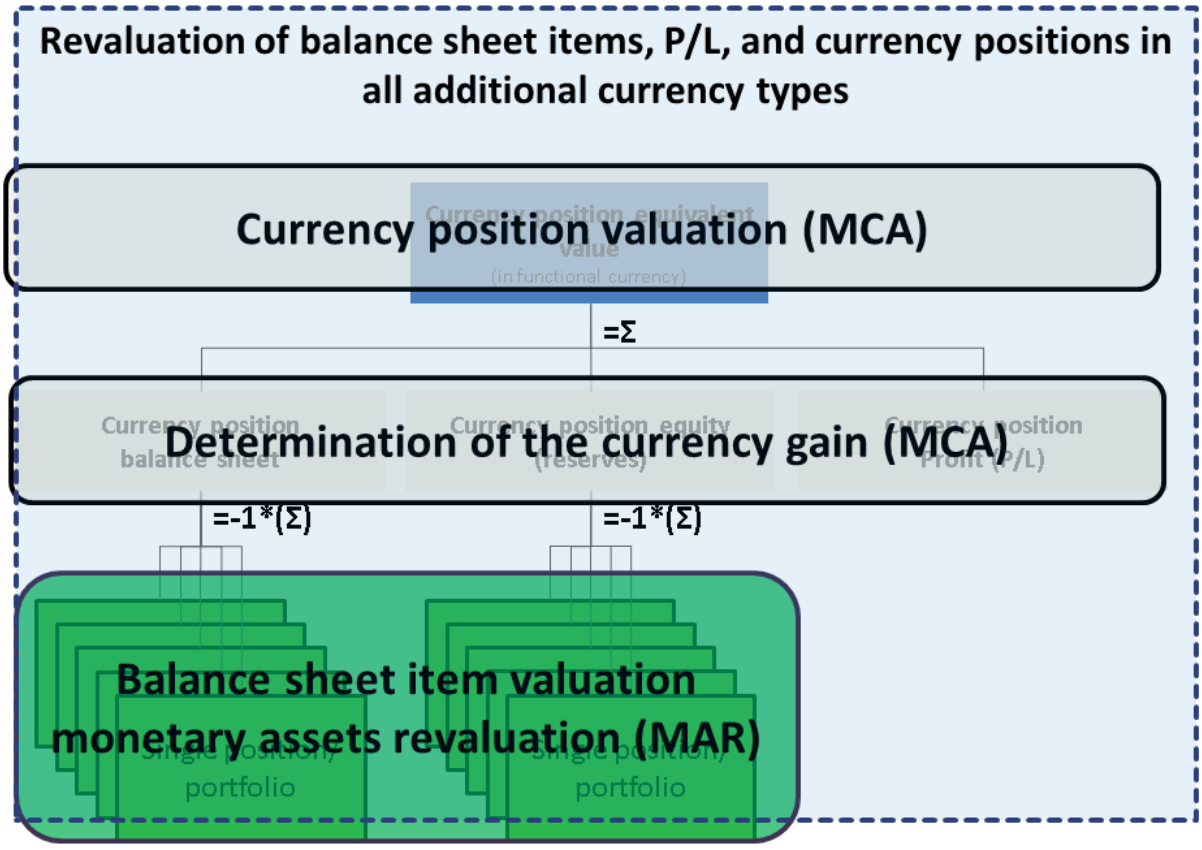
   This can be represented in a formula as follows:
   Currency gain
   = FX position (TC) x FX rate (TC:FC)
   ./. FX position (FC)
2. **Balance sheet item valuation**
   The system performs the following activities:
   - Updates the balance sheet items in functional currency The amount in transaction currency is always equal to zero; the amount in functional currency is not equal to zero.
   - Translates all additional currency types from the amount in functional currency at the current rate of exchange and enters this amount in the document This step is called monetary assets revaluation (MAR).

   This can be represented in a formula as follows:
   Balance sheet item valuation

= balance sheet item (TC) x FX rate (TC:FC)
./. FX position (FC)

3. **Foreign currency restatement of balance sheet items, P/L items and currency positions in all additional currency types**
   The system updates the balance sheet items, P/L items and currency positions in all additional currency types. In these documents, only the amount for the relevant currency is filled. This amount is also calculated from the amount in functional currency at the current rate of exchange.
   This can be represented in a formula as follows:
   Restatement of balance sheet items, P/L items, and currency positions in additional currency
   = position (FC) x FX rate (FC:additional currency)
   ./. position (additional currency)

The figure below illustrates the positions for which the process steps are performed:



The following positions are included in these valuation steps:

- Determination of the currency gain (applies only to foreign currency positions)
- Balance sheet item valuation (applies to single positions or portfolios)
- Foreign currency restatement (performed for balance sheet items, P/L items, and currency positions in all additional currency types for all positions and items)

For more detailed information about the foreign currency valuation process, see Posting Process Steps for Foreign Currency Valuation (Subledger Scenario) [page 730].

4.4.5.1.13.1 Posting Process Steps for Foreign Currency Valuation (Subledger Scenario)

Use

Process

In Bank Analyzer, foreign currency valuation, also called Multi Currency Accounting (MCA), comprises the following posting process steps:

1. Post external business transactions
2. Update secondary business transactions
3. Run key date valuation

In production systems the system executes these steps automatically.

1. **Post external business transactions**
   1. The system determines the business transactions from the Source Data Layer (SDL).
   2. The system compares the object currency of the financial transaction or financial instrument with the functional currency assigned to the legal entity. If these are different, the system performs a currency translation to determine the amount in functional currency. If the currencies are the same, the amount is transferred.
   3. The system also performs a foreign currency translation to determine which optional currency categories (see foreign currency categories) you have entered in Customizing. In this case, the amount in functional currency is the amount that is translated.
      When it translates currencies, the system reads the exchange rates valid for the process time stamp from the market data (fixed exchange rate).
   4. It then posts the corresponding position key figures to the corresponding clearing key figures. This posting step also derives defined characteristics.

   You call the *Post External Business Transactions* transaction on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Daily Processing* > *Automatic Processing* > *Post External Business Transactions*.

   Posting Example
   Transaction currency: USD
   Functional currency: EUR
   Local currency: CHF

	Processing Category					
	1BF_SUSP Operational Suspense Account (Position Clearing)			1BG_TRANS Operational Anonymous In-Transit Account		
	Transaction Currency	Functional Currency	Local Currency	Transaction Currency	Functional Currency	Local Currency

	Processing Category					
	1BF_SUSP Operational Suspense Account (Position Clearing)			1BG_TRANS Operational Anonymous In-Transit Account		
Business transaction 1 (disbursement)	USD 10,000	EUR 8,000	CHF 10,369	USD -10,000	EUR -8,000	CHF -10,369

	Processing Category					
	1BF_SUSP			1BF_TRANS		
	Transaction Currency	Functional Currency	Local Currency	Transaction Currency	Functional Currency	Local Currency
Business transaction 2 (interest payment)	USD -10	EUR -8	CHF -10	USD 10	EUR 8	CHF 10

2. **Update secondary business transactions**
   The system first determines whether external business transactions have been triggered by the process *Post External Business Transactions*. If they have, the system derives internal business transactions from the external business transactions, such as inflows and outflows and accruals and deferrals. The system transfers the translated currencies from the external business transaction. This ensures that there are no rounding effects.
   You call the *Update Secondary Business Transactions* transaction on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Daily Processing* > *Automatic Processing* > *Update Secondary Business Transactions*.
   Immediate Transform and Move
   In your system you have configured transfer postings for profit and loss balances in the foreign currency using key figures with the following processing categories:
   - `1IX_PL` "[O] Profit and Loss"
   - `2IX_PL` "[A] Profit and Loss"

   This means that you are working without unrealized profits and have entered the corresponding procedure in which the profit or loss is transferred immediately to the same document (not for the first time at period-end).
   System Settings for Immediate Transform and Move
   We provide the calculation procedure template 2601 – *Move & Transform* (class 0005; procedure class *M&T Result*) for transferring postings for profit and loss balances with foreign currencies.
   In Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Procedures* > *Display General Overview of All Templates*, you can view the possible calculation step categories, the order in which they are carried out, the calculation methods for each step category and - for credit and debit - the item templates and the processing categories that can be posted.
   Posting Example

Update of secondary business transactions using the example of a loan in foreign currency:
Transaction currency: USD
Functional currency: EUR
Local currency: CHF

	Processing Category					
	1BF_SUSP Operational Suspense Account (Position Clearing)			1BF_RECPAY Operational Payables/Receivables (Acquisition Value)		
	Transaction Currency	Functional Currency	Local Currency	Transaction Currency	Functional Currency	Local Currency
Business transaction 1 (disbursement)	USD -10,000	EUR -8,000	CHF -10,369	USD 10,000	EUR 8,000	CHF 10,369

	Processing Category					
	1BF_SUSP Operational Suspense Account (Position Clearing)			1IX_PL Operational Profit/Loss		
	Transaction Currency	Functional Currency	Local Currency	Transaction Currency	Functional Currency	Local Currency
Business transaction 2 (interest payment)	USD 10	EUR 8	CHF 10	USD -10	EUR -8	CHF -10

	Processing Category								
	1BG_FX Operational Currency Position (P/L)			1BG_FXEV Equivalent Value of Operational Currency Position			1IX_PL Operational Profit/Loss		
	Transaction Currency	Functional Currency	Local Currency	Transaction Currency	Functional Currency	Local Currency	Transaction Currency	Functional Currency	Local Currency
Transform and move	USD -10	EUR -8	CHF -10				USD -10	EUR -8	CHF -10
				EUR 8	EUR 8	CHF -10	EUR -8	EUR -8	CHF -10

3. **Run key date valuation**
   Transform and Move at Period-End

In your system you have configured transfer postings for profit and loss balances in the foreign currency using key figures with the following processing categories:

- `1BF_UNREAR`"[O] Unremitted Earnings"
- `2BF_UNREAR` "[A] Unremitted Earnings"

This means that you are working with unrealized profits that are reposted at period-end using *Move & Transform*.

In the Customizing activity *Define Key Date Valuation Type*, you have selected the *Transform and Move* checkbox for your key date valuation type for the period-end. At period-end the system then reposts those profits and losses that were not posted as unrealized profits. For more information, see Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Accounting Processes* > *Define Key Date Valuation Type*.

System Settings for Transform and Move at Period-End

We provide the calculation procedure template 2602 – *Move & Transform Result (Unremitted)* (class 0005; procedure class *M&T Result*) for transferring postings for profit and loss balances with foreign currencies.

You can use the Customizing activity *Display General Overview of All Templates* to view the possible calculation step categories, the order in which they are carried out, the calculation methods for each step category and - for credit and debit - the item templates and the processing categories that can be posted.

Foreign Currency Valuation

The following valuation methods are available during key date valuation:

- Currency position valuation
  The currency position valuation determines the currency gain. The system also translates the foreign currency positions into the functional currency and then calculates the equivalent value.
  In the first posting step the currency position is posted to the currency gain. In the second posting step the value of the currency gain is posted to the equivalent value of the foreign currency position concerned.
- Balance sheet item valuation (valuation of monetary assets)
  The balance sheet item valuation translates all the balance sheet items into the functional currency. In this posting step the value of the balance sheet item is posted to the currency gain.
- Foreign currency restatement
  This valuation method translates all amounts into the local currency. In this posting step the currency position of the balance sheet item is posted to the currency gain.

In the Customizing activity *Define Key Date Valuation Type* you must enter the *Category of Foreign Currency Valuation* for each key date valuation type. This category specifies which foreign currency valuation procedures are executed in the key date valuation. You can choose from the following categories:

- Category 0: A foreign currency valuation is not executed.
- Category 1: The system valuates the foreign currency positions and determines the currency gain.
- Category 2: In addition to the valuation procedures of category 1, the system updates those balance sheet items whose transaction currency is different from the functional currency.
- Category 3: In addition to the valuation procedures of category 2, the system creates restatement postings. In addition to the valuation procedures of category 2, the system creates restatement postings. This means that all the balance sheet items are updated in all the additional currencies entered in Customizing (local currency, group currency, index-based currency, and hard currency).

> **Note**
>
> You must make sure that a key date valuation is executed with a key date valuation type with the *Category of Foreign Currency Valuation* 2 or 3 at the end of each period at least.

You call the *Carry Out Key Date Valuation* transaction on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Automatic Processing* > and then:

- *Start Key Date Valuation for Single Financial Positions*
- *Start Key Date Valuation for All Financial Positions*

It is not necessary to distinguish between these two transactions for foreign currency valuation.

System Settings for Foreign Currency Valuation

We provide the following calculation procedure templates for foreign currency valuation:

Template	Description	Class	Procedure Class
2211	FX Valuation - Currency Positions	0013	FX Valuation - AFI
2212	FX Valuation - Balance Sheet Items	0013	FX Valuation - AFI
2213	Restatement - Balance Sheet Positions	0013	FX Valuation - AFI

You can use the Customizing activity *Display General Overview of All Templates* to view the possible calculation step categories, the order in which they are carried out, the calculation methods for each step category and - for credit and debit - the item templates and the processing categories that can be posted.

Posting Logic of Key Date Valuation Using the Example of a Loan in Foreign Currency

Transaction currency: USD
Functional currency: EUR
Local currency: CHF

Currency position valuation:

A profit of EUR 3 has been made on the equivalent value of the operational currency position (document row 1). The foreign currency position was valued in functional currency EUR and a profit of EUR 3 EUR was posted to the equivalent value in transaction currency EUR (document row 2).

	Processing Category								
	1BG_FX Operational Currency Position (P/L)			1BG_FXEV Equivalent Value of Operational Currency Position			1IX_PL Operational Profit/Loss		
	Transaction Currency	Functional Currency	Local Currency	Transaction Currency	Functional Currency	Local Currency	Transaction Currency	Functional Currency	Local Currency
Currency position valuation	USD 0	EUR -3	CHF -3.86				USD 0	EUR 3	CHF 3.86
				EUR 3	EUR 3	CHF 3.86	EUR -3	EUR -3	CHF -3.86

Balance sheet item valuation (valuation of monetary assets):

The acquisition value (operational payables/receivables) has been adjusted due to a change in exchange rates for each balance to EUR 2,000 in functional currency (posting row 1). The key figure `1BG_TRANS` has been adjusted due to a change in exchange rates for each balance to EUR 1,997 in functional currency (posting row 2).

	Processing Category								
	1BF_RECPAY Operational Payables/Receivables (Acquisition Value)			1BG_TRANS Operational Anonymous In-Transit Account			1IX_PL Operational Profit/Loss		
	Transaction Currency	Functional Currency	Local Currency	Transaction Currency	Functional Currency	Local Currency	Transaction Currency	Functional Currency	Local Currency
Balance sheet item valuation	USD 0	EUR 2,000	CHF 2,592				USD 0	EUR -2,000	CHF -2,592
				USD 0	EUR -1,997	CHF -2,588	USD 0	EUR 1,997	CHF 2,588

Foreign currency restatement

The change to CHF (local currency) is revalued for each balance sheet key figure for EUR (functional currency) at CHF 1:

	Processing Category								
	1BF_RECPAY Operational Payables/Receivables (Acquisition Value)			1BG_FX Operational Currency Position (P/L)			1IX_PL Operational Profit/Loss		
	Transaction Currency	Functional Currency	Local Currency	Transaction Currency	Functional Currency	Local Currency	Transaction Currency	Functional Currency	Local Currency
Revaluation	USD 0	EUR 0	CHF				USD 0	EUR 0	CHF 1
				USD 0	EUR 0	CHF 1	USD 0	EUR 0	CHF -1
	Processing Category								
	1BF_RECPAY **Operational Payables/Receivables (Acquisition Value)**			**1BG_FXEV** **Equivalent Value of Operational Currency Position**			**1IX_PL** **Operational Profit/Loss**		

	Processing Category								
	1BF_RECPAY Operational Payables/Receivables (Acquisition Value)			1BG_FX Operational Currency Position (P/L)			1IX_PL Operational Profit/Loss		
	Transaction Currency	Functional Currency	Local Currency	Transaction Currency	Functional Currency	Local Currency	Transaction Currency	Functional Currency	Local Currency
Revaluation	USD 0	EUR 0	CHF -1				USD 0	EUR 0	CHF 1
				USD 0	EUR 0	CHF -1	USD 0	EUR 0	CHF 1

For more information about foreign currency valuation, see Foreign Currency Concept (Subledger Scenario) [page 725].

4.4.5.1.14 Archiving in Accounting

Use

Data archiving helps to optimize system performance by removing application data that is no longer needed in the operational system. For more information about using the archiving tool, see .

Depending on the accounting scenario being used, archiving functions are available for the following application data:

Flow Data

- Documents
- Balance objects and financial statement items
- Aggregated transactions (subledger scenario)
- Balance processing postings
- General ledger documents (subledger scenario)

Master Data

- Financial positions
- Balance objects (subledger scenario)

The increment of flow data is usually higher than that of the master data. The main part of the data volume to be archived is therefore usually the flow data.

> **Note**
>
> Data archiving has not been designed as an auditing tool. It is simply to be used as a supporting process (for the provision and conservation of data over a longer period of time) when you fulfill auditing requirements. As part of the auditing process, you can also retrieve archived data if needed. To simplify the auditing process, you should, however, try to meet all the requirements while the data is still in the database.

Prerequisites

You can only archive application data if you no longer need them in operational business. The criteria for this are as follows:

- The data no longer needs to be processed (for example, to create an annual balance sheet).
- The data does not need to be changed again.
- It is likely that you will only need to display the data on rare occasions.

The residence time determines how long the data remains in the system until it can be archived. You define this for the respective archiving object in Customizing. It is important to note that the residence time for master data must be longer than the residence time for flow data. The following dependencies (checked in Customizing for residency times) must also be taken into consideration:

- The residence time for documents must not be longer than that for master data.
- The residence time for results data must not be longer than that for master data.

Before archiving, you must check whether archiving a specific object requires other objects to be archived first or whether other objects need to be archived with it. You can find this information in the network graphic for the respective archiving object (transaction *AOBJ*).

Features

Archiving Flow Data

You define the residence times for all flow data in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Archiving* > *Assign Residence Times for Flow Data*.

This residence time is specified in *Number of Periods* for every accounting system and every legal entity.

The system writes the flow data for an accounting system and a legal entity to archive files for all archivable posting periods. When the archiving process begins, the system determines all periods that can be archived and that fall at least as many periods before the current period as the residence time.

Archiving Aggregated Transactions (Subledger Scenario)

Aggregated transactions are archived in archive administration (transaction *SARA*) for the archiving object Bank Analyzer: Aggregated Transactions (BA1_B7_AT) [page 1765].

Archiving Balance Processing Postings

Balance processing postings are archived in archive administration (transaction *SARA*) for the archiving object Balance Processing Postings (BA1_B2_BPA) [page 1812].

Archiving General Ledger Documents (Subledger Scenario)

Balance processing postings are archived in archive administration (transaction *SARA*) for the archiving object Bank Analyzer: GL Connector (BA1_B8_GL) [page 1766].

Archiving Master Data

The financial positions are based on every process in Accounting. For this reason, they are archived less frequently. This means that master data has longer residence times, usually several years.

You define the residence times for all master data in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Archiving* > *Assign Residence Times for Master Data*.

The residence time is specified for the legal entity in terms of the number of months.

You cannot reload archived master data at any chosen time. You can only do this if the same objects have not been recreated in the system. You can only reload this data if you have archived it incorrectly (see above). Consequently, archiving master data is essentially the same as deleting it from the system.

> **Note**
>
> You cannot archive data at the same time as running modifying processes in Accounting within the same legal entity (for example, the transactions *Master Data Transfer*, *Posting Business Transactions* or *Start Financial Reporting Process*).

Archiving of financial positions

Financial positions are archived in archive administration (transaction *SARA*) for the archiving object Bank Analyzer: Financial Position (BA1_B0_FPO) [page 1763].

The following conditions apply to the archiving process:

- You have specified the residence time for the master data.

You can display the archived financial positions using the read program for the archiving object.

For more information, see Archiving of Financial Positions [page 738],

Archiving Balance Objects (Subledger Scenario)

Balance objects are archived in archive administration (transaction *SARA*) for the archiving object Bank Analyzer: Balance Objects Manager (BA1_B6_BO) [page 1813].

4.4.5.1.14.1 Archiving of Financial Positions

Use

- You can use archiving object *BA1_B0_FPO* to archive financial positions (table */BA1/B1_FP_MD*, */BA1/B1_FP_MDH*, */BA1/B1_FP_STAT*). The objects and items are archived in archive administration (transaction *SARA*). The archiving object offers the following options:
- The object includes a program for writing archives and a program for deleting archives.
- It is possible to read data that has been archived.
- A reload program is also provided, but is intended to be used as an "emergency" function only.

You can display and reload archived runs only after you have carried out both the write process and the deletion process in productive mode.

For more information about archiving, see Archiving in Accounting [page 736].

Prerequisites

You have configured the following activities and settings.

- You have archived the corresponding financial position balances (object *BA1_B1_FPT*).
- You have made sure that the size of the archive file has been configured correctly. We recommend the following settings:
  - Maximum size in MB: 50-100
  - Maximum number of data objects: 0
- You have made sure that you have selected the *Test Mode* option in the test run variant and the *Productive Mode* option in the productive run variant.
- You have used the *Detail Log* option in the deletion variants to determine whether the system lists the deleted financial position balances.

Features

You start an archiving run for exactly one accounting system and one legal entity. If you are using the merge scenario (Balance Analyzer), the system activates all financial position objects for the accounting system and the legal entity whose accounting objects have been inactive for at least as long as the specified residence time.

You can use the input help to display information about the general processing options (test mode, productive mode, deletion with test variant) on the selection screen.

Activities

Archiving Run

1. Create a variant by choosing *Maintain*.
   The system displays the variant processing menu.
2. Select the following fields:
   - Required fields:
     - Accounting system
     - Legal entity
   - Optional fields:
     - Flow control (test mode/productive mode)
     - Detail log
     - Comment for archiving run (useful for identifying run later)
3. Save your variants and schedule them.

Deleting Archived Data

1. Select an archiving run by choosing Archive Selection.
2. If you run the deletion process in test mode, the system does not remove any data from the database.
3. Schedule your deletion variant and choose *Execute*.

> **Note**
>
> If you want to archive data records and delete archived data records in two separate steps, you can start the deletion program separately for each archive file.

Reloading Financial Positions

> **Caution**
>
> Use reloading only under exceptional circumstances.

1. Create a variant.
2. Edit this variant (test or productive run) and save your entries.
3. Select an archiving run by choosing *Archive Selection*.
   If you start the reload process as a test run, the system does not reload the data to the database.
4. Schedule your reload variant and choose *Execute*.
   The system reloads financial positions from the selected archiving run.

> **Note**
>
> It may be that the system cannot reload financial positions. In this case, a new archive file is opened for each financial position. For this reason, do not set the *No New Run when Reloading* checkbox.

4.4.5.1.14.2 Archiving Balance Objects and Financial Statement Items

Use

Archiving object *BA1_R4_055* archives balance objects and financial statement items (Result Database runs and packages). The objects and items are archived in archive administration (transaction *SARA*). The archiving object offers the following options:

- The object includes a program for writing archives and a program for deleting archives.
- A reload program is also provided, but is intended to be used as an "emergency" function only.

You can display and reload archived runs only after you have carried out both the write process and the deletion process in productive mode.

For more information about archiving, see Archiving in Accounting [page 736].

Prerequisites

You have configured the following activities and settings (we recommend the following settings):

- You have made sure that the size of the archive file has been configured correctly. We recommend the following settings:
  - Maximum size in MB: 50-100 50-100
  - Maximum number of data objects: 0
- You have selected the *Test Mode* option in the test run variant and the *Productive Mode* option in the productive run variant.

Features

You can choose the following selection screens:

- Selection screen 1000
  This selection screen starts the process for archiving, according to residence times, for one accounting system and one legal entity.
- Selection screen 2000
  This selection screen starts the process for one of the archiving types.

You can use the input help to display information about the general processing options (test mode, productive mode, deletion with test variant) on the selection screen.

Activities

Archiving Run

1. Create a variant by choosing *Maintain*.
   The system displays the variant processing menu.
2. Choose a selection screen.
   - On selection screen 1000, you find the following fields:
     - Required fields:
       - Object
       - Accounting system
       - Legal entity
       - Processing option (test mode or productive mode)
     - Optional fields:
       - Comment for the archiving run (this is an entry to help identify the run at a later date; it is initially filled by the system with RD for balance objects and FSI for financial statement items)
   - On selection screen 2000, you find the following fields:
     - Required fields:
       - Object
       - Archiving type (according to residence time (default), invalid data, obsolete data)

- Accounting system
- Legal entity
- Fiscal year (for the archiving types "invalid data" or "obsolete data" only)
- Period (for the archiving types "invalid data" or "obsolete data" only)
- Processing option (test mode or productive mode)

- Optional fields:
  - Comment for the archiving run (this is an entry to help identify the run at a later date; it is initially filled by the system with RD for balance objects and FSI for financial statement items)

3. Save your variants and schedule them.

> **Note**
>
> For technical reasons, an archiving run exists for each RDB run and RDB package. This means that one start can contain more than one archiving run. After executing the write phase, you can therefore save the same comment for multiple archiving runs in archive management.

Deleting Archived Data

1. Select an archiving run by choosing *Archive Selection*.
2. If you run the deletion process in test mode, the system does not remove any data from the database.
3. Schedule your deletion variant and choose *Execute*.

> **Note**
>
> If you want to archive data records and delete archived data records in two separate steps, you can start the deletion program separately for each archive file.

Reloading Archived Data

> **Caution**
>
> Use reloading only under exceptional circumstances.

1. Create a variant.
2. Edit this variant (test or productive run) and save your entries.
3. Select an archiving run by choosing *Archive Selection*.
   If you start the reload process as a test run, the system does not reload the data to the database.
4. Schedule your reload variant and choose *Execute*.

4.4.5.1.15 Connection Between Accounting and the Garbage Collector

Use

The Accounting objects mentioned below can be used in the .

Generating Object	Generated Objects
General derivation	DDIC structures
Balance data display	DDIC structures
Balance data segmentation	SGS structures
Accounting object (merge scenario only)	SGS structures DDIC structures
Accounting balances (merge scenario only)	DDIC structures RDB package types
EACC configurator (merge scenario only)	DDIC structures
Results data	RDB package types
Financial statement items	RDB package types
Financial position master	SGS structures DDIC structures ABAP subroutine pool
Common derivation of characteristics (merge scenario only)	DDIC structures

Activities

Start the garbage collector on the SAP Easy Access screen under *Bank Analyzer* > *Tools* and select the corresponding object type. For more information, see .

4.4.5.2 Financial Position Processes

Use

The financial position processes are carried out after generation [page 713]. They include processing the business transactions [page 771] and generating the financial positions [page 744] and posting rules [page 761]. In Calculation Management [page 799], the financial positions are valued using calculation procedures [page 821]. The financial position processes also include Hedge Accounting [page 904]. The results of the financial position processes are stored in the Results Data Layer (RDL).

Implementation Considerations

This documentation describes the business background for the basic settings and the control settings for the financial position process. You cannot make the Customizing settings described in the following documentation until you have made the general settings [page 696] for Accounting in the subledger scenario.

More Information

For more information about the financial position processes in the *Accounting for Financial Instruments* (AFI) scenario, see Processes in the AFI Scenario [page 678].

4.4.5.2.1 Financial Positions

Use

Financial positions are units for which position values and quantities are managed in Accounting. Financial positions are used to differentiate data that is used in accounting processes for measuring and posting transactions. For example, a loan that is hedged by multiple hedging relationships has multiple financial positions (one financial position per hedging relationship). This is necessary because, for the financial mathematical methods (hedge amortized cost, hedge fair value), values must be maintained at portion level for each hedging relationship.

Every financial position belongs to a financial position class [page 697].

Features

Defining characteristics of a financial position

SAP ships certain defining characteristics for each financial position type. However, you can also specify your own defining characteristics.

- **Defining characteristics specified by SAP**
  These can differ depending on the financial position type. The following defining characteristics can be specified, for example:
  - `/BA1/C55ACCSY` (accounting system)
  - `/BA1/C55AOCURR` (financial position currency)
  - `/BA1/C47HEDGE` (hedging relationship identifier)
  - `/BA1/C55CP_NO` (component number)
  - `/BA1/C55SP_NO` (single position number)
  - `/BA1/C55LGENT` (legal entity)

There are also further defining characteristics for certain objects:

- `/BA1/C69TMBCKT` (time bucket: for portfolio fair value hedging relationships, for example)

The system uses the following defining characteristics to model the dimensions of financial statement entities:

- `/1BA/_BA1C_BUSSG` (business segment)
- `/1BA/_BA1C_PBUSG` (partner business segment)
- `/BA1/CP0PRFCTR` (Profit Center)
- `/BA1/CP0PARTPC` (Partner Profit Center)
- `/BA1/CP0PARTLE` (partner legal entity)

> **Note**
>
> Note the following:
>
> - Dimensions of financial statement entities are stored in the Results Data Layer (RDL). For general financial positions, these are defining characteristics; for product-related financial positions, these are descriptive characteristics (path in Customizing for *Bank Analyzer*: *Results Data Layer* > *Basic Settings* > *Edit Results Data Area*).
> - You must always use the business transaction to supply the financial position currency. The *Accounting System* (`/BA1/C55ACCSY`), *Hedging Relationship Identifier* (`/BA1/C47HEDGE`), *Component Number* (`/BA1/C55CP_NO`), and *Single Position Number* (`/BA1/C55SP_NO`) characteristics, on the other hand, are filled by the system during processing in accounting.

- You must always use the business transaction to supply the financial position currency. The *Accounting System* (`/BA1/C55ACCSY`), *Hedging Relationship Identifier* (`/BA1/C47HEDGE`), *Component Number* (`/BA1/C55CP_NO`), and *Single Position Number* (`/BA1/C55SP_NO`) characteristics, on the other hand, are filled by the system during processing in accounting.
- Custom defining characteristics
  You can specify your own defining characteristics. To do so, execute the Customizing activity *Define Financial Position Classes* in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *Before Generation* > *Financial Positions*. Choose *Defining Characteristics at Master Data Level* or *Defining Characteristics at Valuation Level*.
  If you also want to allow initial values for these characteristics when creating the financial positions, enter the respective characteristics in the Customizing activity *Define Optional Fields of Financial Position* in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *Before Generation* > *Financial Positions*.
  You must add the characteristics that define a transaction hedge as custom defining characteristics in the financial position.

  > **Note**
  >
  > The system copies the values of the defining characteristics directly from the business transaction header and item. They are not derived from the financial instrument/financial transaction, the securities position account/settlement account, or the position.

Descriptive characteristics

In addition to defining characteristics, financial positions have descriptive characteristics. SAP also ships fixed and custom descriptive characteristics.

- Descriptive characteristics specified by SAP
  These can differ depending on the financial position type. The following descriptive characteristics can be specified, for example:
  - /BA1/C11HEDCAT (hedging relationship category)
  - /BA1/C55ACPCAT (period category)
  - /BA1/C55ACPLEN (period)
  - /BA1/C55EVALPR (alternative calculation procedure)
  - /BA1/C55HSIMPR (hedge adjustment procedure)
  - /BA1/C55TXRATE (tax rate)
  - /BA1/C55VALTIL (valid to)
  - /BA1/C69HRLCAT (item category of hedge)
  - /BA1/C55AOTYPE (financial position type)
  - /BA1/C55COMPCL (component class of financial position)
  - /BA1/C55RESACQ (indicator for determination of residual acquisition value)
  - /BA1/C55ASSLIA (asset/liability financial position)

  The characteristics *Hedging Relationship Category* (//BA1/C11HEDCAT), *Period Category*(/BA1/C55ACPCAT), *Period* (/BA1/C55ACPLEN) and *Item Category for Hedge* (/BA1/C69HRLCAT) are filled by hedge business transactions.
  You can change the characteristics *Non-Standard Calculation Procedure* (/BA1/C55EVALPR), *Hedge Adjustment Procedure* (/BA1/C55HSIMPR) and *Tax Rate* (/BA1/C55TXRATE) manually or by using business transactions.
- **Custom descriptive characteristics**
  You can specify your own descriptive characteristics. To do so, execute the Customizing activity *Define Financial Position Classes* in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *Before Generation* > *Financial Positions*. Choose *Descriptive Characteristics*.
  If you also want to allow initial values for these characteristics when creating the financial positions, enter the respective characteristics in the Customizing activity *Define Optional Fields of Financial Position* in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *Before Generation* > *Financial Positions*.

The following description applies to both defining and descriptive characteristics.

To be able to generate a financial position or change descriptive characteristics, the following defining characteristics specified by SAP must be filled in the business transaction:

- At *master data level*: *Legal Entity* (/BA1/C55LGENT)
- At *valuation level*: *Accounting System* (/BA1/C55ACCSY) and *Financial Position Currency* (/BA1/C55AOCURR)

To make sure that the defining characteristics *Legal Entity*, *Accounting System*, and *Object Currency*, which are predefined by SAP, as well as the custom characteristics in the business transaction have been filled in, check the assignments made from Source Data Layer (SDL) fields to the business transaction fields. In Customizing for *Bank Analyzer*, choose *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of External Business Transactions* > *Edit Characteristic Mapping for Header Fields of SDL Business Transactions* and *Edit Field Mapping for Item Fields of SDL Business Transactions*.

Granularity

The granularity of the risk mapper governs the granularity of financial positions. This means that the granularity of the financial positions cannot be greater than that of the risk mapper.

Forward exchange transactions, for example, have an unstructured object in the SDL. The risk mapper analyzes the cash flows of the SDL object and generates three single currency valuation structure nodes (one root node and two swap legs) for this forward exchange transaction. The system uses this information when it generates financial positions. Structural changes in the SDL templates cannot be processed by financial position management.

> **Note**
>
> Ensure that a change in the number of the SDL node corresponds to a change in the object structure.
>
> Also ensure that the relevant cash flow generations types can be used in the valuation.

Generation

A financial position is generated by posting external business transactions [page 774] or by updating secondary business transactions [page 786]. If a financial position does not exist with the corresponding defining characteristics, the system creates a new financial position. When the financial positions are created, the financial position class is derived from the master data in the SDL.

When the system generates a financial position, it copies the values for the descriptive characteristics from the business transaction header and item or derives them according to the rules defined in the Customizing settings for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Positing Rules* > *Characteristics of Financial Positions* (see following Customizing activities):

- *Edit Secondary Data Source for Descriptive Characteristics*
- *Derive Descriptive Characteristics of Financial Position and Funding Center*

> **Note**
>
> You can enter as optional fields the defining and descriptive characteristics that are predefined by SAP for the financial position. The required-optional property is predefined by SAP.

Using the Customizing settings, the system determines which additional financial positions are required. In doing so, the system can access information in the SDL. For example, several financial positions can be generated internally for a structured financial transaction following an external *purchase* business transaction. The system does this by reading the structure of the transaction (object or subobject structure) from the SDL as the external business transaction is being processed. Depending on this structure and additional control characteristics in the SDL (currency ID, relevancy status of financial position), it determines the characteristics of the required financial position and, where there are new characteristic combinations, creates new financial positions. For more information, see Generation of Financial Position Objects for Structured Products [page 750].

You can generate multiple financial positions in various currencies for a product. When you post an external business transaction, the system determines the currency of the financial position from the currency of the financial transaction/instrument. If you want to create financial positions in different currencies for unstructured products (such as stocks) that are traded in different currencies, you can configure the Customizing settings in such a way that the system determines the currency for the financial position from the currency of the associated account supplied in the external business transaction. You can also configure the system to generate a financial position with the currency that is specified in the business transaction. The system does not check the currency of the financial position against the currency in the business transaction.

To specify the origin of the currency for each financial position type, choose Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Basic Settings* > *Assign Origin of Currency*.

Change

The system fills the descriptive characteristics only for the first business transaction for a financial position. You can then only change a financial position by making a manual posting with an item type of the item templates `0J40` and `0J41`, or by means of an external business transaction with an item type of item template `1I40`.

You have the following options:

- Change to descriptive characteristics including financial position transfer posting
  The business transaction should contain the following items:
  - Characteristic change (item template `1I40`)
  - Clearing to the same position
  - Posting to the same position
    In this case, the system clears the entities that are dependent on the changed market value, such as the general ledger account, and reposts them. This is also possible for hedging financial products. You can also import the related transfer postings without nominals. The system uses the elementary calculation procedure with the template `2150` for this function. You can use the distribution transfer procedure with the calculation method `0N16` to also process hedging products.
- Change to defining characteristics including update in the Hedge Accounting Workplace (HAW)
  If a business transaction with a characteristic change is supplied (such as a securities account transfer), the system creates a new financial position with the changed defining characteristic. You can also do this for defining characteristics for hedging financial products, without having to dissolve the hedging relationship. The system uses the elementary calculation procedure with the template `0P40` for this function. You can use the distribution transfer procedure with the calculation method `0N17` to also process hedging products.

Recommendations

Note the following general recommendations for financial positions:

- You should use custom descriptive characteristics for financial positions only if they are also required for processing in Accounting, for example, in the derivation rules.
  For more recommendations for the *Holding Category* characteristic, see Characteristics in Calculation Management (Subledger Scenario) [page 878].
- To improve system performance, create secondary indexes for the following characteristics in the tables of the defining characteristics of the financial position:
  - *Financial Transaction Identifier (/BA1/C40FTRAN)*
  - *Securities Position Account Identifier (/BA1/C43CLACC)* and *Financial Instrument Identifier (/BA1/C41FINST)*
  - *Aggregation Object (/BA1/C4AAGO)*
  - You can fill the remaining fields of the index with dimensions of financial statement entities, such as Profit Center (`/BA1/CP0PRFCTR`) or Business Segment (`/1BA/_BA1C_BUSSG`). Use fields that have high selectivity.
- If you do not use aggregation objects, or use only one dimension of a financial statement entity, you can include the settlement account (`/BA1/C42SACC`) in the index.
  At this point, if the index is not yet full, you can also include the legal entity (`/BA1/C55LGENT`) and the accounting system (`/BA1/C55ACCSY`).
- You can find the tables for the financial position objects in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Basic Settings* > *Performance Optimization* > *Edit Segmentation Structures for Accounting*.
  You can add the necessary entries on the *Indexes* tab page and regenerate them by selecting a structure.

Status information

To facilitate processing, the system manages the following status information for a financial position:

- Time of the next business transaction to be updated (posting date)
  Data must be updated as of this time. This time is determined by the next business transaction that updates data. For some business transactions, the resulting position values and quantities cannot be calculated in isolation, but are dependent on other business transactions. In these cases, the system generates posting items in a subsequent process that contain the resulting values and quantities.
- Time of the initialization business transaction (posting and value date)
- Time of the first operative business transaction (posting and value date)
- Time of the last manual business transaction (impairment) (posting and value date)

Activities

- You display a financial position on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Analysis* > *Display Financial Position*.
- For more information about changing descriptive characteristics manually, see Editing Financial Positions [page 749].
- The SDL contains various object relationships that must not lead to financial positions in Accounting (for example, the underlying for options). You can filter out these object relationships by making settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *SDL Connection* > *Assign Object Relationship Categories*. Assign one of the following risk basis relationship categories to the object relationship categories:
  - `NOTREL` (Not relevant for risk basis)
  - `SIMIN` (B is an instrument similar to A)
  - `UNDLY` (B is the underlying of A)

  The system then ignores the object relationships with the above object relationship categories in Accounting.

4.4.5.2.1.1 Edit Financial Positions

Use

You can use this report to edit descriptive characteristics for selectable financial positions, for example, the accrual period.

Procedure

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Tools* > *Financial Positions* > *Edit Financial Positions*.

4.4.5.2.1.2 Generation of Financial Positions for Structured Products

Use

Financial positions are generated by business transactions. For structured products, the system also takes master data from the Source Data Layer (SDL) to generate financial positions (based on mapping rules). The *Financial Position Management* tab page is relevant for accounting when you create financial instruments and transactions. The specifications made here determine how financial positions are created in Accounting for structured products. If you do not complete the specifications on this tab page, the system cannot process business transactions in Accounting.

> **Note**
>
> Later changes made on the *Financial Position Management* tab page are not transferred automatically to Accounting. To transfer these changes, choose either the Edit Financial Positions [page 749] report or post external business transactions using the item template 1I40.

Process

Fields on the Financial Position Management Tab Page

The individual fields on the *Financial Position Management* tab page form a thematic group and together control how financial positions are generated and valuated in Accounting. For this reason, you always view them as a whole. Here you need to bear in mind the following business-related issues:

- The *Accounting Standard* field indirectly specifies the accounting systems in accounting for which the row is evaluated when external business transactions are posted. It is possible, for example, that in the local GAAP a structured product is shown using a single financial position in Accounting and is thus valuated as a complete unit. According to IAS, however, the individual parts that constitute the structured product have to be represented by several financial positions on the balance sheet to enable separate key data valuations and, possibly, reporting.
- The *Generate Financial Position Object* field can contain the value *Generate* for one accounting standard and the value *Do Not Generate* for another. In the latter case, you can also provide no row at all for the accounting standards that do not require a financial position in Accounting.
  The remaining fields on the tab page are relevant only if the *Generate Financial Position Object* field contains the value *Generate*.
- The *Component Class* field, as part of the IAS solution, plays an important role in controlling the valuation of embedded derivatives. It can be filled with the following values: *M Main Part*, *D Derivative Part*, or blank. For this reason, you configure this field as an optional entry field in the Customizing settings for the template version. When you update secondary business transactions [page 786] or valuate a key date, separate calculation methods are used for embedded derivatives. The method selected depends on whether you are dealing with a main part or a part of a derivative.
- The *Determination of Acquisition Value* field is, likewise, important to the valuation of embedded derivatives in the IAS solution and can contain the following values:
  - *Not residual*

- *Residual*

  In principle, you can determine the acquisition value of one part of a contract both by determining the fair value of the individual parts of a product and by carrying out a residual valuation. For more information about distributing the acquisition value on the basis of a residual valuation, see Embedded Derivatives [page 900].

- The *Valuation Determination* field specifies whether the key date valuation of a single contract/object part should take place by means of the fair value calculation of the individual parts of a product or by means of a residual valuation. This field can have the following values:
  - *Not residual*
  - *Residual*

  The default implementation of Accounting valuation structure SDL mapping ignores this field. However, you can read this field in the secondary data source on the fair value server and use it to determine valuation rules.

Data Input and Output

Data can be supplied to and from the tab page in three ways.

1. Data can be entered manually or by the system from the source system. In this way, the tab page data is filled for each financial transaction and financial instrument on an individual basis.
2. You can define default values for this tab page for each product (that is, each primary object category) in the Customizing settings for the SDL.

   To do so, in Customizing for *Bank Analyzer*, choose *Source Data Layer* > *Primary Objects* > *Master Data* > *Templates* > *Edit Template Version*.

   If data is supplied automatically without transferring values for the tab page, the system takes the default values last saved in Customizing and enriches the relevant primary object.
3. A standard option is used when data is read from the tab page if the financial transaction/instrument involves a non-structured product or a main contract/instrument consisting of one of more subcontracts/instruments. In this case, the system automatically adds the following information to enable the financial transaction/instrument to be processed in Accounting.
   - Field*Generate Financial Position Object*: Value*Generate Financial Position*,
   - Field *Component Class*: Field is empty.
   - Field *Determination of Acquisition Value*: Value *Not Residual*

> **Note**
>
> These are the correct settings for unstructured templates. Hence, you do not need to use the *Financial Position Management* tab page for unstructured templates.

Processing in Risk Basis

The transformation of a product in the SDL to a risk object is described under Valuation-Relevant Fields in the SDL Templates [page 581].

Risk basis then transfers the relevant calculation bases for the product concerned, along with its subcontracts and further products assigned for each object relationship (including the balance position management data), in the form of a risk object to accounting. Depending on the type of template, the risk object consists of several risk object nodes that represent the subobjects or object relationships. The risk object represents the most detailed financial position management and key date valuation granularity used (from the perspective of accounting). For more information, see Mapping of the Valuation Structure in Accounting [page 700].

Business Transaction Access to Components of Structured Products

If you want to call up the components of a structured product directly in a business transaction, you must add the following characteristics to the external business transaction row and fill them with values:

- /BA1/C11FPROD (Parent template)
- /BA1/C11SPROD (Child template)
- /BA1/C11NODENO (Node number)
- /BA1/C55USELEG (Currency leg)

For templates that contain a structure in the SDL (for example, warrant bonds or swaps), you only need to enter the three C11 characteristics. For products that are unstructured in the SDL but structured in Accounting (for example, forward exchange transactions and multiple currency bonds), only /BA1/C55USELEG = X needs to be filled to call up a swap leg of the transaction.

Valuating Financial Positions

The risk-basis mapping data used when generating master data for the financial position is stored in Accounting. If financial positions need to be valuated, this data is read and transferred to the FV/AC server to enable the objects to be valuated based on the granularity of the risk basis. If you create a risk mapper for customized new products, you must ensure that the risk mapper data is stored in accounting.

4.4.5.2.1.3 Flag Financial Positions for Updating

Use

This report enables you to flag financial positions for the update of secondary business transactions [page 786] process. This is necessary if incorrect key figures were calculated for a financial position when secondary business transactions were updated (for example, due to incorrect Customizing settings).

> **Note**
>
> This report is not run in parallel and is, therefore, not suitable for mass data. It is primarily intended for supporting tests or selective corrections.

Procedure

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Tools* > *Financial Positions* > *Flag Financial Positions for Updating*.

4.4.5.2.1.4 Calculating Financial Position Balances

Use

Due to the large number of documents that accrue in accounting, calculating and displaying financial position balances each time would have a negative impact on system performance. It is therefore important for the system

to have access to precalculated balances for as many key dates as possible (for instance, every week, every month, or at least every quarter).

It calculates all the necessary balances (such as book value, instrument valuation, and currency reserves) for a given quantity of financial positions and a key date, and stores them in the Results Data Layer (RDL) where they can be accessed to display financial position balances and calculate internal documents.

Recommendation

We recommend that you calculate the balances of financial positions for the greatest possible number of key days only if you rarely make postings to previous months. Otherwise, we advise that you calculate the balances in an interval, whereby only a few new postings are made to previous months. This will improve system performance.

Caution

You must calculate the financial position balances for the first day **after** a period that is to be archived. If there are no balances for the day following the period to be archived, the system cancels document archiving and issues an error message. This ensures that new documents can be posted to the period to be archived **after** the financial position balances have been created but **before** the documents are archived.

Example: The year 2005 is the period to be archived. The financial position balances must therefore be saved for 01/01/2006 and not for 12/31/2005.

Procedure

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Results Data Layer* > *Pre-Aggregation* > *Run Preaggregation*.

For more information about calculating financial position balances, see Aggregation [page 1374] and Preaggregation [page 1408].

You have the following entry options:

1. Specify the legal entity and the accounting system.
2. Enter a financial position type. The input fields of the *position* screen area change according your selection.
3. Enter data to select a financial position.
4. Enter a key date for calculating the financial position balances.
5. If you do not want to save the calculated financial position balances to the database, choose *Simulation Run*.

You can execute this transaction directly or include it in the flow definition of a financial reporting process.

4.4.5.2.1.5 Deactivation of Financial Positions

Use

You can use this process to deactivate and reactivate financial positions. This allows you to optimize system performance in cost accounting processes and accounting processes. You can also archive the deactivated financial positions so that the system only contains current and relevant data.

The system does not have to take financial positions that have reached their maturity into account in certain calculations and processes. It therefore makes sense to deactivate these financial positions. For example, the system does not take inactive financial positions into account in the key date valuation, which reduces the data volume and thereby also the amount of time the process takes to run.

> **Note**
>
> Note that the function for deactivating financial positions is only available in the subledger scenario.

> **Note**
>
> When the contract is terminated, you can directly trigger deactivation of financial positions using application event management.

Prerequisites

Customizing

You have made the settings for the verification of balance-relevant key figures in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Deactivation of Financial Positions* > *Assign Processing Categories for Balances*. These settings are optional since SAP ships relevant processing categories. However, you can create your own custom processing categories.

You have made settings for a secondary data source in respect of the maturity of financial products in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Deactivation of Financial Positions* > *Edit Module Editor Data for Maturity*.

System Checks Applied Before Financial Positions Are Deactivated

The system performs a number of checks prior to deactivation to establish whether the financial positions can be deactivated. When you choose *Individual Display* in the Execution of Deactivation [page 759] function, the system lists all relevant financial positions. Otherwise, the system deactivates all relevant financial positions in a mass run. For more information about the checks that establish whether financial positions are relevant for deactivation, see Checks Before Deactivation [page 757].

Process

You can choose the following functions to deactivate or reactivate individual financial positions or all relevant ones:

- Execution of Deactivation [page 759]
- Execution of Reactivation [page 760]

You can display the status of financial positions (active or inactive) in the analysis of the financial position. The system generates a two-dimensional graphic with the *System Time* and *Valid-From Date* of the financial position. To call up the financial position analysis, proceed as follows:

1. From the *SAP Easy Access* screen, choose *Processes and Methods* > *Accounting for Financial Products* > *Analysis* > *Display Financial Position*.
2. Choose in the results list.

Result

This section describes the effect that the deactivation of financial positions has on certain processes. For each process, it is stated whether the status of a financial position is relevant for the process to be executed. It also stated whether each process supports the automatic reactivation of financial positions. This is necessary, for example, when a business transaction results in a posting for a financial product whose corresponding financial position is flagged as inactive in the system. In this case, the system must be able to reactivate this financial position automatically.

Accounting Processes

Accounting Process	Relevance of the Status of a Financial Position	Automatic Reactivation Function
Post external business transactions	No: Active and inactive financial positions are processed	Yes: The reactivation date is the earliest of the value date or posting date.
Update secondary business transactions	No: Active and inactive financial positions are processed	Yes: The reactivation date is the earliest of the value date or posting date.
Key date valuation	Yes: Only active financial positions are taken into account when generating the worklist	No
Key date valuation of time bucket objects	Yes: Only active financial positions are taken into account when generating the worklist	No
Year-end processing	No: Active and inactive financial positions are taken into account	No
Processing of Accounting Events	No	Yes: The reactivation date is the earliest of the value date or posting date.

Accounting Process	Relevance of the Status of a Financial Position	Automatic Reactivation Function
Asset/Liability Process	Yes: Only active financial positions are taken into account when generating the worklist	No
Process results of effectiveness tests	No: Only hedged financial positions are deactivated	Yes: The reactivation date is the earliest of the value date or posting date.
Flag financial positions for updating	No: Active and inactive financial positions are taken into account	No

Note that in the case of the *Flag Financial Positions for Updating* process, the following situations may occur for inactive financial positions:

- The new processing date for the first business transaction to be updated (`CALC_FROM`) falls in an inactive phase of the financial position. In this case, revaluation or posting is not necessary. The financial position remains inactive.
- The new processing date for the first business transaction to be updated (`CALC_FROM`) falls in an active phase of the financial position. In this case, revaluation is required. The system reverses and generates business transactions.

Cost Accounting Processes

Cost Accounting Process	Relevance of the Status of a Financial Position	Automatic Reactivation Function
Calculate indirect costs for periods	Yes: Only active financial positions are taken into account when generating the worklist	No
Update calculation of indirect costs for periods	No	Yes: The reactivation date is the earliest of the value date or posting date.
Process cost accounting events	No: Cost accounting events for the calculation of indirect costs for periods must be processed on the basis of active financial positions.	Yes: The reactivation date is the earliest of the value date or posting date.
Post currency selldown	Yes: Only general financial positions that have not been deactivated are taken into account.	No

Example

The following example of a backdated posting shows the different status values that a financial position can have.

You purchase a financial product worth EUR 100,000 on January 7, 2008. You sell the same amount on January 14, 2008.

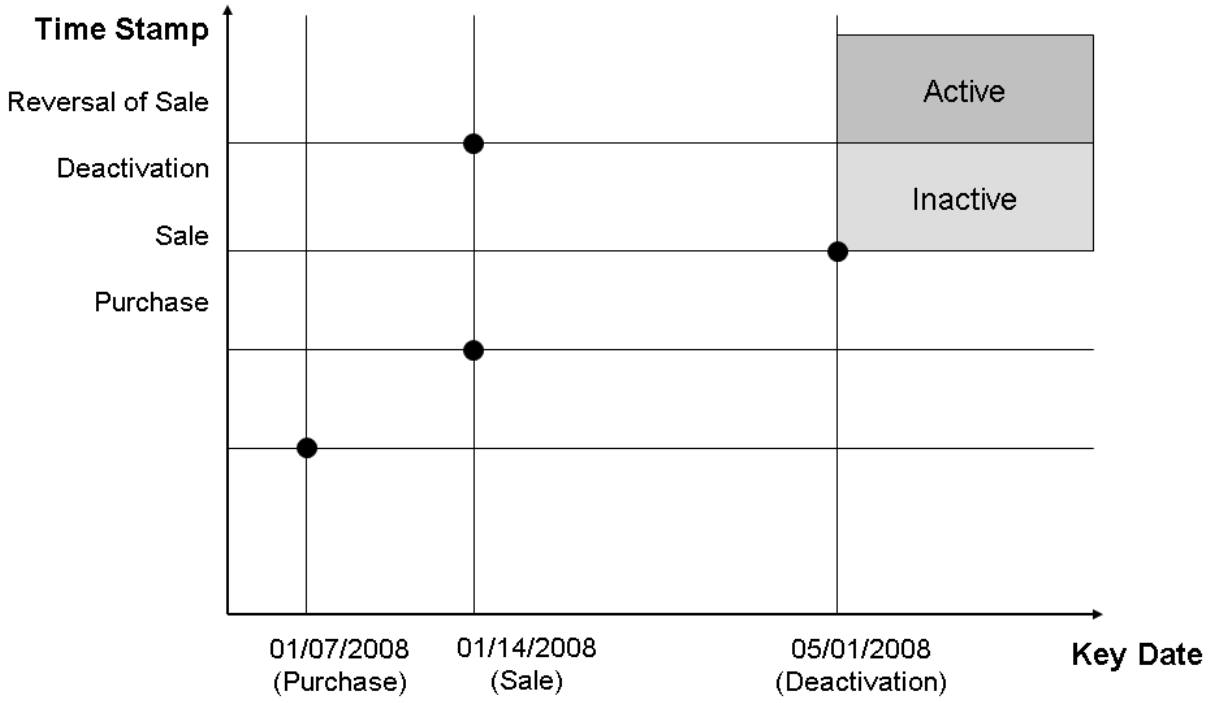
The financial position is deactivated on May 1, 2008.

The sale from January 14, 2008 is then reversed.

The financial position cannot remain inactive because the balances of the financial position are no longer zero. Inactive financial positions would be excluded from certain processes, such as the key date valuation.

To prevent this, the system reactivates future inactive versions of this financial position from the deactivation date of May 1, 2008.

The following graphic provides a two-dimensional overview of this example. You can track the moments at which the purchase, sale, and deactivation take place. In the end, the backdated posting of the reversal takes place, automatically activating the versions from May 1, 2008.



4.4.5.2.1.5.1 Checks Applied Before Financial Positions Are Deactivated

The following checks determine whether a financial position is relevant for deactivation:

1. The financial position must be current. This means the following fields must be initial:
   - Processing date of the first business transaction (`CALC_FROM`) to be updated.

- Date of the first calculation of indirect costs (`PICC_CALC_FROM`) to be updated.

2. The date on which the data is checked must be after the posting date for positions (PD_START_POSTING) and after the value date (PD_START) for positions.
   The check date is calculated by subtracting the *period since maturity* from the *deactivation date*. You enter both these values on the selection screen of the deactivation function. The period since maturity is the wait time from the end date before a financial position can be deactivated (we recommend at least a few days).

   > **Note**
   >
   > The `CALC_FROM`, `PICC_CALC_FROM`, `PD_START`, and `PD_START_POSTING` fields can be found in table `/BA1/B1_FP_STAT`.

3. You are not allowed to post any business transactions in the period between the check date and the deactivation date either in position date or posting date dimension. This means that you cannot deactivate the financial position if there is a business transaction for a financial position that has a position or posting date in this period.
4. In module editor, you can enter your own checks in the Customizing activity under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Deactivation of Financial Positions* > *Edit Module Editor Data for Maturity*.
   Depending on the availability of an end date for a financial product, the following options are available:
   - The end date for the financial product is saved in the Source Data Layer (SDL). You can define a secondary data source so that the system can use the key date and maturity to establish whether the financial position can be deactivated.
     We recommend that the financial product has reached maturity at least a few days previously.
   - The end date for the financial product is **not** saved in the SDL. In this case, you can define the secondary data source such that the system can use certain criteria, for example the financial position type, to establish whether the financial position should be deactivated.
5. All of a financial position's balance-relevant key figures must be zero on the deactivation date.
   The relevant date for this check is:
   - For one-dimensional processing categories: the posting date
   - For two-dimensional processing categories: a combination of posting date and value date, where the value date has the value *Infinity*

   In the case of structured products, the system checks all of the product's financial positions.
   You enter the balance-relevant processing categories in Customizing. All key figures that belong to these processing categories are relevant for the balance and the balance sheet. You can use the processing categories predefined in the system, or your custom processing categories. You can find the Customizing activity under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Deactivation of Financial Positions* > *Assign Processing Categories for Balances*.
6. If the financial position belongs to a financial product that is included in a hedging relationship, this hedging relationship must be dissolved. Otherwise, the financial position cannot be deactivated.

4.4.5.2.1.5.2 Execution of Deactivation

Use

You use this function to deactivate financial positions. This allows you to prevent these financial positions from being included in accounting processes and cost accounting processes.

The system runs different checks before it deactivates the financial positions, for example to ensure that only financial positions that have reached their maturity and whose balance is zero are deactivated. For more information, see Deactivation of Financial Positions [page 754],

Activities

Proceed as follows:

1. On the *SAP Easy Access* screen, choose *Processes and Methods* > *Accounting for Financial Products* > *Tools* > *Financial Positions* > *Deactivate Financial Positions*.
2. Enter the following on the selection screen:
   - Legal entity and accounting system for which you want to carry out the deactivation.
   - Deactivation date
     Key date on which the system will deactivate the corresponding financial positions. The relevant financial positions are inactive as of this date. Enter a date that is the same as the system date or falls before it.
   - Period since maturity
     Financial positions must have reached their maturity at least 90 days previously for deactivation to be possible.
   - Display found objects for selection
     You can make single or multiple selections of financial positions to be deactivated in a results list. If you do not select this individual display checkbox, the system performs a mass run for all relevant financial positions.
   - You can use additional criteria to further restrict the financial positions that are selected for deactivation, for example by specifying the following:
     - Financial position type
     - Financial transaction ID, financial instrument ID, settlement account ID
     - Aggregation object ID
   - You can also call up your own variant, select a layout, and perform the deactivation sequentially or in a test run.
3. Depending on whether you have selected the individual display option, proceed as follows:
   - **Do not display found objects**

     Choose  to perform the deactivation. The system saves an application log in which you can track the results of the deactivation.
   - **Display found objects**

     1. Choose  to open a results list.
     2. Select the rows containing the financial positions you want to deactivate. You can select a number of functions before deactivation takes place, for instance display details or posting results for a financial

position, perform a financial position analysis, or export the list, for example to a local word processing or spreadsheet file.

3. Choose  to perform the deactivation for the selected financial positions.

4.4.5.2.1.5.3 Reactivation of Financial Positions

Use

You can use this function to reactivate financial positions that you have previously deactivated. This may be necessary, for example, if errors occurred during deactivation or if you intentionally want to reverse the deactivation of the financial positions concerned.

> **Note**
>
> In addition to this reactivation function, the system can also reactivate financial positions automatically and as an integral part of certain processes. For more information, see Deactivation of Financial Positions [page 754],

Activities

Proceed as follows:

1. On the *SAP Easy Access* screen, choose *Processes and Methods* > *Accounting for Financial Products* > *Tools* > *Financial Positions* > *Reactivate Financial Positions*.
2. Enter the following on the selection screen:
   - Legal entity and accounting system for which you want to carry out the reactivation.
   - Deactivation date
     Key date on which the system deactivated the corresponding financial positions.
     The system reactivates any financial positions that were deactivated on this deactivation date.

     > **Note**
     >
     > Note that the specific date of deactivation is being referred to here, and not a period from this date in which all financial positions are reactivated.
     >
     > Note also that the system always reactivates financial positions on the basis of a given day. If, for example, you deactivated two financial positions on a given day, the system reactivates both.

   - Display found objects for selection
     You can make single or multiple selections of financial positions to be reactivated in a results list. If you do not select this individual display checkbox, the system performs a mass run for all relevant financial positions.
   - You can use additional criteria to further restrict the financial positions that are selected for reactivation, for example by specifying the following:
     - Financial position type

- Financial transaction ID, financial instrument ID, settlement account ID
- Aggregation object ID

- You can also call up your own variant, select a layout, and reactivate the financial positions sequentially, or start the function as a test run.

3. Depending on whether you have selected the individual display option, proceed as follows:
   - **Do not display found objects**

     Choose to reactivate the financial positions. The system saves an application log in which you can track the results of the reactivation.
   - **Display found objects**

     1. Choose to open a results list.
     2. Select the rows containing the financial positions you want to reactivate. You can select a number of functions before you reactivate the financial positions. For example, you can display detailed data, or analyze the financial positions.
     3. Choose to reactivate the selected financial positions.

4.4.5.2.2 Posting Rules (Subledger Scenario)

Use

The posting rules contain the following derivations, which you can define in Customizing:

- Derivation of the G/L accounts
- Derivation of the flow types
- Derivation of the posting key figures
- Derivation of the document date specifications
- Derivation of the dimensions of financial statement entities
- Derivation of additional document fields

When you derive G/L accounts, you derive the number of the general ledger account for which the system makes a posting in a business transaction.

The flow type is used to summarize amounts of multiple document lines and to divide financial positions, for example, to give a schematic overview.

Posting key figures are standard key figures in Bank Analyzer that are defined on the basis of processing categories.

Posting rules control the determination of the posting key figure for each accounting document line.

Features

Derivation of the G/L Accounts

To assign a general ledger chart of accounts to an accounting system, choose Customizing for *Bank Analyzer* ▶ *Processes and Methods* ▶ *Accounting for Financial Products* ▶ *After Generation* ▶ *Financial Position Processes*

Basic Settings > *Posting Rules* > *Derivation of General Ledger Accounts* > *Assign Chart of Accounts to an Accounting System*.

To derive the G/L account from the fields in the document header and the document items, choose Customizing for *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Derivation of General Ledger Accounts* > *Derive G/L Account*.

You must define the characteristics of the chart of accounts and of the G/L accounts in SAP NetWeaver Business Intelligence (BI).

Derivation of the Flow Types

You use flow types (such as posting, clearing, write up, depreciation) when you generate registers of financial assets, registers of revaluation reserves, and so on. The system stores the flow type for each line in the document.

Edit Characteristics for Derivation of Flow Types

In Customizing, you have to specify a characteristic that acts as the target field for the derivation of the flow types, such as the business transaction type (purchase, sale), for example. In Customizing, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Before Generation* > *Define Characteristic for Flow Type*.

You must also define the flow type characteristic in BI. In the master data for the flow type characteristic in BI, you have to specify the values (such as posting, clearing) with the IDs of the key figures. This can be done manually or automatically.

Derivation of the Flow Types

If the characteristic for the flow type is a document field you derive the flow types from the previously defined characteristics as a step in the Customizing activity *Derive Additional Fields for Documents*.

If the flow type is a document component field, you must define the derivation as a step in the Customizing activity *Derive Additional Document Component Fields*.

Choose Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules*.

One flow type per document row is sufficient. If different registers require different flow types, there can be several flow types per document row.

Derivation of the Posting Key Figures

The objective is to assign exactly one posting key figure to each document row.

1. Defining the posting key figures

The posting key figures can be found in the Bank Analyzer key figure repository. To do this, choose Customizing for *Bank Analyzer* > *Basic Settings* > *Settings for Meta Data* > *Key Figures* > *Edit Key Figures*. You can define any number of posting key figures for the key figure classes provided by SAP. Each key figure must be assigned to a key figure class. A key figure class can contain several posting key figures. Posting key figures are always currency amounts.

2. Assigning posting key figures to processing categories

You must assign each posting key figure to a processing category. Processing categories are assigned to key figure classes. The key figure class of a posting key figure must match the key figure class of the assigned processing category.

You assign the key figures to the processing categories in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Processing Categories* > *Assign Key Figures to Processing Categories*.

Derivation of Posting Key Figures

You derive the posting key figures for debit and credit from the item type of a business transaction. You can add fields from the header and items of the business transaction as further source fields. Choose Customizing for *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Derive Posting Key Figures*.

You can use characteristic `/BA1/C55FXFLOW` to derive different posting key figures for foreign currency documents and non-foreign currency documents. For more information in conjunction with profit and loss, see Key Date Valuation [page 934].

Derivation of the Dimensions of Financial Statement Entities

You can add a further derivation regarding the dimensions of financial statement entities (such as legal units and profit centers) to the documents for business transactions before the system stores them in the Results Data Layer (RDL). Choose Customizing for *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Derivations for Dimensions of Financial Statement Entities* > *Derive Dimensions of Financial Statement Entities*. For more information, see the documentation for the Customizing activity.

Derivation of Additional Document Fields

You can add a derivation of further document fields to the documents for business transactions before the system saves these in the RDL. Choose Customizing for *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Derive Additional Fields for Documents*. For more information, see the documentation for the Customizing activity.

Derivation of Additional Document Component Fields

You can add a derivation of further document component fields to the documents for business transactions before the system saves these in the RDL.

Choose Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Derive Additional Document Component Fields*.

For more information, see the documentation for the Customizing activity.

Handling of Time-Dependent Derivations

If the derivation of G/L accounts or other derivations in the area of posting rules contain time-dependent derivation rules the standard system uses the system date to perform these derivations.

However, in the Customizing activity *Define Chart of Accounts and Date for Time-Dependent Derivations* you can specify that the system use the posting date for derivations.

Choose Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules*.

4.4.5.2.2.1 Posting Date and Value Date

Use

In accounting, you always need to distinguish between the value date and the posting date.

Posting Date

The posting date defines the date on which an activity was registered in the bank's books, for example, by:

- A bank customer at an automated teller machine
- A credit representative
- A bank customer using online banking

When operative master data and transaction data is transferred (for example, from the contract-managing system), this data is initially recorded in these systems. This means that most reliable date information is always in the systems in which an activity is first registered. When data is processed within the subledger, the system uses this date information unchanged.

When analytical events are transferred (such as impairment decisions, reclassification decisions, and so on), the posting date also defines the time at which the activity is recorded. In these cases, the system also uses the posting date unchanged for processing.

In period-end processing, the posting date defines the time at which the system has to update valuation results and accrual/deferral results for financial positions. For these processes, it has to use all the data (such as flow transactions) that is recorded at this point in time.

The posting date controls the following processes in the subledger:

- Period-end processing
  The posting date of the period-end process defines which data is relevant for the valuation and accrual/deferral run, that is, all data that exists up to the time of the period-end process (data was posted before or on the posting date for the period-end process). The system determines the valuation results always based on the posting date (for example, market prices, exchange rates, and so on).
- Balance sheet
  The posting date is the only date relevant for inclusion in the balance sheet.

> **Note**
>
> Based on settlement date accounting logic, the system shows items on the balance sheet as memorandum items (below the line) already on the posting date, while it shows items on the balance sheet (above the line) only once the value date is reached.

Value Date

The value date defines the legally binding value date of an operational flow transaction.

This date therefore defines the time as of which the system has to calculate accrual/deferral items.

The operational (contract-managing) systems define the legally binding value date of operational flow transactions. This definition must be copied from the contract-managing system unchanged.

As part of the subledger period-end processes, the value date defines the start date for calculating accrual/deferral items (such as accrued interest).

The value date controls the following processes in the subledger:

- Accrual/deferral run
  Calculation of accrual and deferral items for a specific time period (such as accrued interest). The value date of the underlying flow transactions defines the underlying start date, while the value date of the accrual/deferral run defines the end date.

Example

For both examples, assume that the current date is 31.12.201.

Example 1

An operational business transaction has the following date information:

- Posting date: 30.12.2011
- Value date: 03.01.2012

Period-End Processing on 31.12.2011

When creating the balance sheet on 31.12.2011, the system has to show the effect of the business transaction on the underlying position since it is already visible (30.12.2011) and must also carry out valuations (for example, calculate the fair value). With respect to calculating accrual and deferral items (accrual/deferral run), this business transaction on 31.12.2011 does not have any effect since its value date is not until 03.01.2012 (for example, the accrued interest is not calculated until 03.01.2012).

Period-End Processing on 31.01.2012

When creating the balance sheet on 31.01.2012, the system has to take into account the effect of the business transaction unchanged. With respect to calculating accrual and deferral items (accrual/deferral run), the effect of this business transaction is that the system has to calculate the accrual/deferral for the period from 03.01.2012 (value date of the inflow) to the value date of the accrual/deferral run (31.01.2012).

Example 2

With respect to posting cut-offs or processing transactions carried out on a non-bank workday, there can generally be cases where the posting date of this transactions is moved to the next bank workday (in the future), already in the operational system.

An operational business transaction has the following date information:

- Posting date: 03.01.2012
- Value date: 30.12.2011

Period-End Processing on 31.12.2011

When creating the balance sheet on 31.12.2011, the system does not have to show the effect of the business transaction on the underlying position since it is not yet known at this time (posting date). This means, therefore, there are no effects with respect to the accrual/deferral calculation.

Processing the Business Transaction on 03.01.2012

On the posting date of the business transaction, it is transferred to the books for the first time and is thus shown on the balance sheet.

> **Note**
>
> We are assuming that the business transaction changes a position that already carried balances (that are calculated based on "earlier" flow transactions not shown here and for which the system has calculated accrual/deferral on 31.12.2011). This means that the system recalculates this calculated accrual/deferral with the posting date of the business transaction (03.01.2012).

It is crucial here that no effects visible at an earlier posting date result due to the posting date being taken into account correctly. The effects do not take effect until the posting date of the business transaction.

Period-End Processing on 31.01.2012

When creating the balance sheet on 31.12.2011, the system has to show the effect of the business transaction on the underlying position since it is already visible (since the posting date 03.01.2012). The system also needs to carry out valuations (for example, calculate the fair value).

With regard to calculating accrual and deferral items (accrual/deferral run), the effect of this business transaction means that the system has to calculate the accrual/deferral for the period from 30.12.2012 (value date of the inflow) to the value date of the accrual/deferral run (31.01.2012) and show these with the posting date of the period-end process (31.01.2012). You can also supply other custom date fields from source systems and transfer these to accounting. However, these are not relevant for controlling processing in the system.

> **Note**
>
> If a business transaction is imported with a posting date that falls within a period already closed from an accounting perspective, the system rejects this business transaction and issues are error message.

4.4.5.2.2.2 Posting to Clearing (Two-Step Posting)

A two-step posting process is used to process business transactions in Bank Analyzer:

- Step 1: Process: Post External Business Transactions
- Step 2: Process: Update Secondary Business Transactions [page 786]

The *Post External Business Transactions* process allows all imported business transactions to be fully transferred to subledger accounting. This process does not use complex processing logic. The processing approach taken in this process is that the amounts to be distributed in the second step are recorded in a suspense account (`1BF_SUSP`). The *Update Secondary Business Transactions* process contains the actual accounting processing logic, which is relatively complex. This means that the amount recorded in a suspense account by the first step is distributed to various accounting processing categories based on rules predefined by configuration.

> **Note**
>
> If a balance exists for the processing category *Suspense Account* (`1BF_SUSP`) when the balance sheet is created, this results in a technical error and the second step is not carried out.

Example

A financial position has five stock shares with an acquisition value of EUR 20.

Business Transaction	Debit/Credit	Key Figure	Amount	Quantity
Initial Balance	D	Acquisition value	EUR 20	5

This position is subject to a business transaction in which the entire position is sold for EUR 23.

Step 1: Process: *Post External Business Transactions*

Business Transaction	Debit/Credit	Key Figure	Amount	Quantity
Sale of Stock	D	Payment transactions	EUR 23	5
Sale of Stock	C	Suspense account	EUR -23	- 5

In the accounting results, the balance for the *Suspense Account* key figure indicates that the business transaction has not yet been processed by step 2.

Step 2: Process: Update Secondary Business Transactions [page 786]

Business Transaction	Debit/Credit	Key Figure	Amount	Quantity
Sale of Stock Derived	D	Suspense Account	EUR 23	5
Sale of Stock Derived	C	Acquisition value	EUR -20	- 5
Sale of Stock Derived	C	Result	EUR -3	

Consequently, the EUR 23 transferred from the suspense account was correctly distributed to the reduction of the acquisition amount and the effect in the profit and loss statement.

As expected, the balance of the suspense account is 0 EUR, indicating that the process was carried out correctly.

If errors occur during processing for either of the processes, the system issues an error message. Viewing the balance for the processing category *Suspense Account* (`1BF_SUSP`) is therefore simply an additional option. It allows you to determine which business transactions were transferred correctly (step 1) but have not yet been or could not be processed.

The component transfer procedure [page 855] is also part of the Update Secondary Business Transactions [page 786] process. When business processes for structured financial positions are processed, the system can split the amount transferred by the process to post external business transactions.

The following procedures are also part of the *Update Secondary Business Transactions* process:

- Component transfer procedure
- Transfer postings for foreign currency profit
- Transfer methods for assets and liabilities

> **Note**
>
> External business transactions from the *direct posting* category are not posted in the two-step process using suspense accounts. However, this business transaction category still requires the two steps to be carried out since the Move and Transform Manager (for postings in foreign currencies to a P/L key figure) or the asset/liability switch of a position must be processed in accounting. For more information, see Asset/Liability Status [page 863].

4.4.5.2.2.3 Long and Short Positions

The following section describes how calculation procedures operate on short and long positions and how they handle the transition from one position situation to another.

Let us assume a business transaction results in a position outflow and the existing long position is thereby not only reduced (reduction part), but a short position is built up (build up part) as well, since the nominal value of the business transaction exceeds the nominal value of the position. In this case, calculation management internally calculates the *transaction reduction factor* as the relative portion of the business transaction nominal value that brings the position to zero. It also calculates the *position reduction factor* as the relative portion of the long position (regarding the nominal value) that the business transaction clears (in our case, the *position reduction factor* is equal to one since the entire long position is reduced).

Before you can modify the position, all the position components must be updated by the calculation procedure. The flows for reducing the long position and the flows for creating the short position are then divided into different calculation steps and executed.

The system runs the calculation steps for reducing the long position first. Since the *position reduction factor* is one, all financial position key figures are reduced to zero. The difference between the transaction amount weighted with the *transaction reduction factor* and the book value of the position is posted as a realization. If the business transaction also has an item with accrued interest, the portion weighted with the *transaction reduction factor* is posted as net income.

The system then runs the calculation steps for creating the short position. The acquisition value for the short position is posted. If the business transaction also has an item with accrued interest, the remaining part of the accrued interest is likewise posted as net income.

A comparable approach also applies to securities account transfers and reclassifications. However, a special feature must be taken into account here. For business reasons, you are not allowed to build up a long or short position by means of a securities account transfer or reclassification on the source position. The source position must always be reduced. The target position can either be reduced or built up. There are two pairs of item templates used to supply the external business transaction for securities account transfers and reclassifications – one for a short source position and any target position and one for a long source position and any target position.

In addition, the realized amount is determined by the residual value on the *general journal key figures: internal clearing of FBF* key figure instead of the *position clearing for value* key figure (as with conventional inflows and outflows).

4.4.5.2.2.4 Pending Positions

Definition

Use

In accordance with IAS 39.30, a regular way contract of financial assets should be recorded using either trade date accounting or settlement date accounting.

If you use the settlement date accounting logic, off-balance-sheet recording (below the line item) takes place for the contract as of the trade date. On-balance-sheet recording for the contract (above the line item) does not have to take place until the settlement date.

You should select which of the two methods is to be used for each category of financial instruments. The chosen method must then be used for all the financial assets of the category:

Holding Categories	Time of Financial Reporting Process
Originated Receivable	Trade date or settlement date (choice optional)
Held for Trading	Trade date or settlement date (choice optional)
Available for Sale (AfS)	Trade date or settlement date (choice optional)
Held to Maturity	Trade date or settlement date (choice optional)
(Other Payables)	Settlement date (only)

Financial Instruments	**Time of Financial Reporting Process**
Derivatives*)	Trade date (only)

*) You must classify derivatives as the HFT holding category under IFRS. The special nature of derivatives is that they are always balanced for the trade date. They are special in that they are always entered on the balance sheet for the trade date. A transaction is considered unsettled if its settlement date is after its trade date. A transaction is said to be in an unsettled status if it is between the trade date and the settlement date.

4.4.5.2.3 Processing of Accounting Events

Use

You use this function to generate a worklist for the accounting processes [page 693]. This worklist is based on application events [page 369] that the system derives from the operational events [page 258] in the Source Data Layer (SDL) and from analytical events [page 689].

The following cases are relevant for the accounting processes:

- Changes to the characteristics of the financial transaction
  For example, a change of borrower.
- Rollover
  For example, a rollover of a loan with a fixed term, but with flexible interest rates and conditions
- Revaluation
  Changes that affect the effective interest rate (such as condition or term changes).
- Contract creation
  If a contract is created in an operation system (settlement of a financial transaction), you can react to the event in accounting by creating the corresponding financial position and posting to it immediately.
- Contract cancellation
  If a contract is cancelled in an operational system, you can clear existing financial position balances for accruals/deferrals and valuations in accounting and subsequently deactivate the financial position. Unremitted earnings that have occurred as a result of the contract are included in this context.

> **Note**
>
> The process suppresses application event processing for financial transactions if the extension of **source data aggregation** is activated, an aggregation view is assigned to the results data area of the accounting system, and the financial transaction is aggregated. If this is the case, the system sets the status of the application event to the value `A` (*In Aggregation, Do Not Supply*).

Features

Accounting events can refer to the following primary transactions in the SDL:

- Financial transaction
- Financial instrument
  The system breaks down the accounting events into individual financial positions by breaking down the financial instruments into securities accounts. To do this, the system accesses the accounting positions using an interface.
- Securities position account

The system updates all financial positions of the financial transactions affected. If the process contains an application event for which there is no financial position, it creates one. This can occur during contract creation, for example.

> **Note**
>
> The system does not support the processing of accounting events for financial instruments without a legal entity, for the combination of financial instrument and securities position account, or for business partners or organizational units. Furthermore, the system does not allow you to change defining characteristics of the financial position in this process.

More Information

Processing of Application Events [page 375]

Application Event Management [page 368]

Processing of Events in the AFI Scenario [page 684]

4.4.5.2.4 Business Transactions

Definition

A distinction is made between external and internal, and primary and secondary business transactions.

Use

External and Internal Business Transactions

An external business transaction is delivered to Accounting from an external system via the Source Data Layer (SDL).

An internal business transaction is created within Accounting, either automatically (by the key date valuation, for example) or manually (by a manual posting, for example).

Most external business transactions are posted in two steps. In such cases, the *Post External Business Transactions* process posts the supplied values to clearing key figures. The *Update Secondary Business Transactions* process generates a business transaction of the derived business transaction category that posts the values to the position key figures.

Internal business transactions include the *key date valuation* and *derived business transaction* categories, which the system processes based on a calculation procedure [page 821].

A distinction is made between external and internal business transactions so that external BTs can be posted separately from the effects they have internally that are posted in financial position management and the income statement, which are dependent on the current financial position values or the processing sequence.

There are several reasons for distinguishing between external and internal BTs:

- While internal BTs are generated for a particular accounting system, operative BTs (except for initialization) are not dependent on an accounting system.
- When a BT is posted, the system determines the relevant position, the value changes of its components, and the net income effects. To do so, it often needs detailed information about the position concerned in addition to the data that is linked directly to the BT.
- The external BTs often have to be put into a sequence, as the processing sequence affects the changes in value and the net income. This sequence is not necessarily identical to the sequence in which the BTs are delivered.

- When an external BT is reversed, the position and net income effects of all subsequent BTs have to be recalculated and reposted, while ensuring that only the subsequent internal BTs are reversed (and reposted) and not the subsequent external BTs themselves.

Primary and Secondary Business Transactions

A primary business transaction is an external or internal business transaction that describes a situation generically.

The following are internal primary business transactions:

- Hedge business transactions
  - Dissolution business transactions
  - Designation business transactions
- Manually posted business transactions
- Forced dissolution

A secondary business transaction is an internal business transaction that is dependent on an accounting system and that must be calculated in a particular order.

The following are secondary business transactions:

- Derived business transactions
- Reversal of designation
- Key date valuation business transactions
- Business transactions that result from a cost accounting process (only for the *Calculation of Internal Costs* scenario)
- Re-derived business transactions
- Resettings
- Year-end closing business transaction
- Business transactions for the reversal of previous year-end closings

Differentiating between primary and secondary business transactions allows the system to update secondary business transactions without affecting the primary business transaction. The amounts posted by primary business transactions to clearing accounts are gradually reduced to zero by secondary business transactions. The partial retirements are reduced proportionally from the balances.

Business transactions		
	Internal	**External**
Primary	• Hedge business transaction • Manually posted business transaction • Forced dissolution	• External business transaction
Secondary	• Derived business transaction • Reversal of designation • Key date valuation business transaction • Created by costing process • (Re-)derived business transaction • Reset business transaction • Year-end closing business transaction • Business transaction for reversal of earlier year-end closing	

4.4.5.2.4.1 Post External Business Transactions (Subledger Scenario)

Use

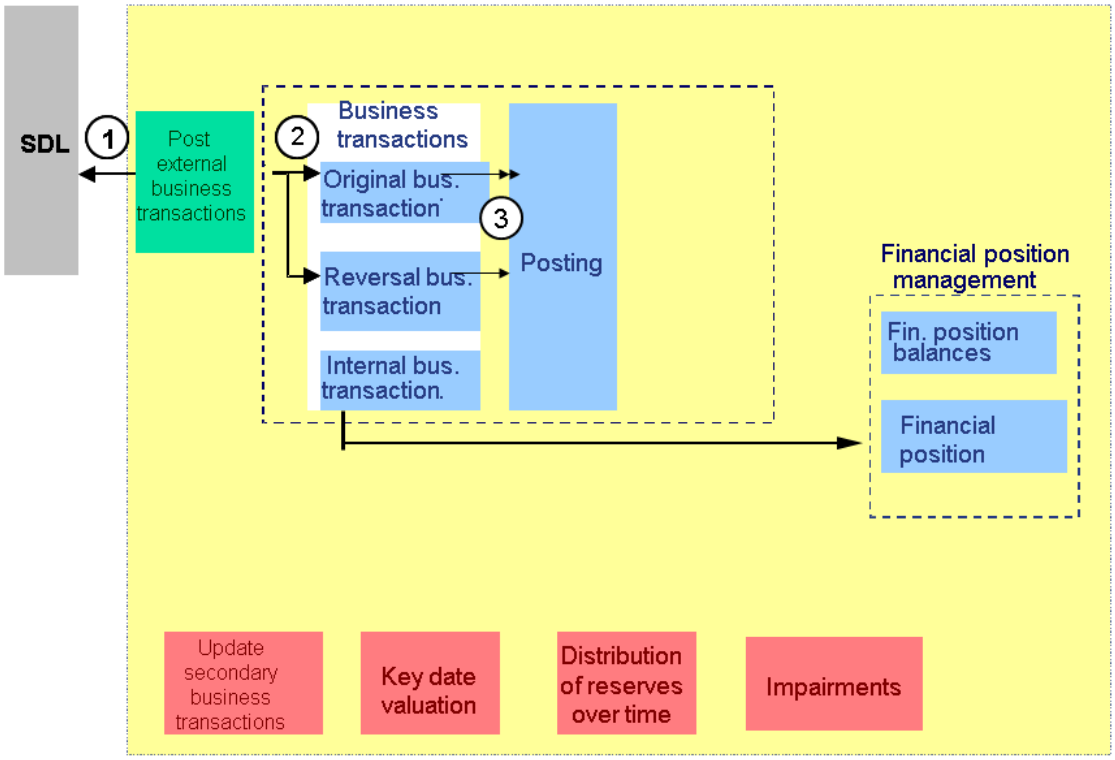


Figure 1: Processes in the Subledger Scenario

Figure 1 provides a rough illustration of the processes that take place in the subledger scenario.

First, the system reads the business transactions from the Source Data Layer (SDL) (step 1). "External business transaction" is the umbrella term given to an initialization and operational business transaction. The system processes the original business transactions first, and then the reversal business transactions. The system reads only those business transactions that are relevant for Accounting. The time stamp can be used as an additional selection criterion. It provides the technical time of generation (system date and time) in the SDL.

When the system reads the business transactions, it converts them (step 2) from the SDL format to the Accounting format.

Then, the system transfers to Accounting (step 3) the business transactions that were read from the SDL and reformatted. In Accounting, the system posts the accounting documents for the specified accounting system. The system posts items from the operational business transaction that are to be processed in financial position management to clearing figures in financial position management.

The system logs the accounting document in accordance with the document items. Logging a posting item for an external business transaction for the financial position triggers the registration function. The registration function generates financial positions [page 744] and records the fact that business transactions have to be processed for them. The financial position balances [page 752] are calculated on the basis of the financial positions.

The descriptive characteristics of the financial position that are not provided with the business transaction can be determined the first time an external business transaction is posted for a financial position.

> **Note**
>
> If you only want the system to edit a business transaction in a specific accounting system (for example, for initializing the merge scenario for the subledger), you can define the characteristic for this accounting system when you create the business transaction on the *SAP Easy Access* screen under *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Business Transaction* > *Create Business Transaction*.
>
> You also need to map the characteristics for the accounting system characteristic in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Processing External Business Transactions* > *Edit Characteristic Mapping for Header Fields of SDL Business Transactions*.

Direct Posting

You can post external business transactions directly to a key figure, that is, without the standard, two-step process. During this single-level direct posting, the external business transactions are also delivered via the SDL. However, the system does not usually derive internal business transactions, and posts the documents directly to the Results Data Layer (RDL). You can use direct posting to correct incorrect postings, for example.

We provide the business transaction category `1006` and item templates (such as `1I22` `1I24`) for direct posting. You must also specify a business transaction type, an item type, and a document type in Customizing. An adjustment business transaction is used to **correct a direct posting** if a characteristic change is entered at a later point in time with an earlier posting date than the posting date of the direct posting. The business transaction type of an adjustment business transaction must be assigned to the business transaction category `1002` (Derived Business Transaction).

We recommend that you avoid deriving internal business transactions for direct postings.

Read Time Stamp for Initialization Business Transactions

There is a read time stamp for initialization business transactions that controls the date from which business transactions of the *Post External Business Transactions* process are selected.

You set the time stamp before the initialization business transactions are generated. By doing so, you prevent the system from re-selecting conventional business transactions that are already included in the generation of initialization business transactions.

Prerequisites

- You have completed generation.
- You have configured the Customizing activities in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of External Business Transactions* (see also Customizing for Business Transactions (Subledger Scenario) [page 779]).

- The SDL time stamp [page 367] must be set before you post external business transactions.
- "External business transaction" is a superordinate term used for an initializing and operational business transaction. Note that you are not permitted to post operational business transactions before you post initialization business transactions. Neither the value date nor the posting date of a purchase can be earlier than the initialization date.

Activities

To post a business transaction, on the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Automatic Processing* > *Accounting Processes* > *Post External Business Transactions*.

If you want to post business transactions that could not be posted in an earlier run, for example, due to incorrect Customizing settings or generation problems, on the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Day Processing* > *Automatic Processing* > *Restart External Business Transactions*.

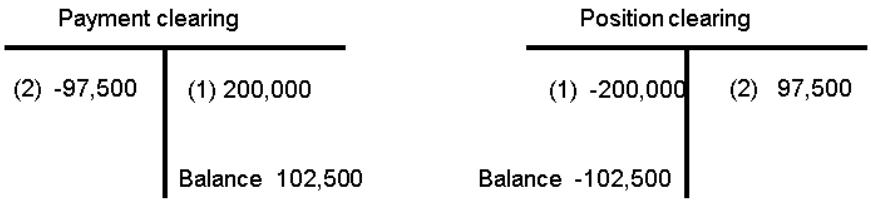
Then enter the following data:

- Enter the date of the program run. This is the date of the last run in which the business transactions either could not be posted or were not posted completely.
- If several runs took place on the same date, enter the sequence number of the program run in which business transactions could not be posted.

Example

1. Initial Position

2. Posting External Business Transactions



(1) 01/20 Purchase of 1,000 units: position clearing to payment clearing : 200,000

(2) 02/15 Sale of 500 units: payment clearing to position clearing: 97,500

"-" = debit

Stock X sec. acct Y AFS			Financial position					FY
No	Date	Item type	*Units*	Clrg acct position Amount	Acquis. value Amount	Instrument valuation Amount	*Book value (IAS)* *Amount*	Payment clearing Amount
1	01/20	Purchase	*-1,000*	-200,000				200,000
2	02/15	Sale	*500*	97,500				-97,500
	02/15	Balances	*-500*	-102,500				102,500

"-" = debit

External business transactions are posted initially to a position clearing account. The position clearing account is cleared later when secondary business transactions are updated [page 786].

4.4.5.2.4.2 Customizing for Business Transactions (Subledger Scenario)

Complete the following Customizing activities:

- *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Before Generation* > *Business Transactions* > *Edit Additional Characteristics for Business Transactions*.
  For more information, see the documentation for the Customizing activity.
- *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Types of Financial Positions and Items* > *Edit Business Transaction Types*.
  Business transaction types are used to sort accounting documents. Each business transaction type must be assigned a business transaction category. When you save the settings, the system checks whether the document type assigned to a business transaction type matches the assigned business transaction category. The business transaction categories are predefined by SAP. Accounting documents are sometimes posted in different ways, depending on the different business transaction categories. For this reason it is important to use the correct business transaction type and assigned business transaction category for each posting.

- *Validate External Business Transactions* and *Validate Internal Business Transactions* , both under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Validation*
  You can use these two Customizing activities to configure separate data checks in the accounting business transaction. You can use the *module editor* tool to do this.
  The following table shows the basic checks on the fields provided by SAP:

Dependent On	Value	Header/Item	Description of Check
All postings		Header	Checks that the business transaction type was specified
All postings		Header	Checks that the legal entity was specified
All postings		Header	Checks that the business transaction key and source system were specified
All postings		Item	Checks that the item type was specified
All postings		Item	Checks that the transaction currency was specified
All postings		Item	Checks that either the quantity or the nominal amount was specified
All postings		Item	If the nominal amount was specified, the nominal currency must also be specified
All postings		Item	If the amount in the transaction currency was specified, the transaction currency must also be specified
All postings		Item	If the amount in the functional currency was specified, the functional currency must also be specified
All postings		Item	If the quantity was specified, the unit must also be specified
Document class	External	Item	Sender/Receiver indicator must have been specified

  Caution

  If you create your own modules for the validation process, you must enter these modules for the item types you want to validate in the Customizing activity *Edit Item Types*.

- *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of External Business Transactions* > *Edit Characteristic Mapping for Header Fields of SDL Business Transactions*
  You must complete this Customizing activity before you can read the business transactions relevant for accounting.
  You assign here all header and item fields that the system transfers into the accounting business transaction and accounting document from the SDL business transaction.

For original business transactions, the system transfers the following standard fields of the SDL business transaction to the accounting business transaction:

- Account category
- Account number
- Template category
- External number

The system uses the field names of the IDs that belong to the following SDL InfoObjects as the target fields:

- Financial transaction
- Financial instrument
- Settlement account
- Securities position account

For items that contain a value (and its currency), in the SDL business transaction this value can be entered either in the *Transaction Amount/Transaction Currency* fields or the *Position Amount/Position Currency* fields (or even in custom fields). However, you must map the fields used to the Accounting fields `/BA1/K55AMOBJ`and `/BIC/OBJ_CURR`. You must map fields that contain the nominal value, nominal currency and quantity to the fields `/BA1/K55AMNOM`, `/BA1/K55AMNOMC` and /BA1/K55QUANT. The same applies for the fields of the functional currency: `/BA1/K55AMBAL` and `/BIC/BIL_CURR`.

One exception to this rule is reversal documents. A fixed structure is used to transfer information for reversal documents. The field names are predefined by SAP. This means that you do not need to map the SDL fields to the Accounting fields for reversal business transactions.

- *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Assign Derived Business Transaction Type to Primary Business Trans. Type*

  For more information, see the documentation for the Customizing activity.

4.4.5.2.4.3 Business Transaction Structure (Subledger Scenario)

Standard Fields in the Source Data Layer (SDL)

The business transaction in the SDL contains some standard fields. These include different key terms, as well as dates in the header or amount fields in the items of the business transaction.

Required Characteristics for Processing Control in Accounting

The following header and item fields from Accounting are required in all business transactions. Therefore, you define these as required fields in each business transaction class (see table below):

Description	Characteristic/Field Name	Found in

Business transaction type	/BA1/C55BTATP	Header
Legal entity	/BA1/C55LGENT	Header
All data fields used to derive the posting date, position date, and document date		Header
Account category	ACCOUNT_TYPE	Item
Account number	ACCOUNT	Item
Template category	PROD_TYPE	Item
External number	EXT_CONTRACT_ID	Item
Item type	/BA1/C55ITMTP	Item
Role in sender/receiver relationship **Note** This is a technical field. It is normally filled with `N`. Therefore, define `N` as the default value. Transfer postings and reclassifications are exceptions to this. For transfer postings and reclassifications, the sender line has the value `S` and the receiver line `R`.	/BA1/C55FPSTA	Item
Object currency	/BIC/OBJ_CURR	Item

Note the following:

- We supply the characteristic /BA1/C55LGENT for the legal entity. You can use this characteristic, or one of your own, for example, SX_LEGENT. The characteristic /BA1/C55LGENT is not used as a fixed, predefined SAP field for the primary object *Account* and *Business Transaction*. This allows you to use your own, user-defined characteristics for the legal entity.
- We do not supply predefined, fixed fields for the dates of the Accounting business transaction. The Accounting business transactions must, however, still contain the date fields in the header because these are used to derive the position, posting, and document dates. You must include these in the header structure of the business transaction as user-defined characteristics.
- The field /BIC/OBJ_CURR is independent of the business transaction. You must always enter a value in the field because it is required to determine the financial position.

Optional Characteristics for Processing Control in Accounting

In addition to the required Accounting characteristics described above, you also need specific optional (depending on their use) characteristics. For example:

Description	Characteristic	Found in	Use
Impairment indicator	/BA1/C55IMPAIR	Header	Used to record an impairment of a financial position if the impairment transaction is to be delivered externally.
External hedge ID	/BIC/TEXTHID	Item	If hedging relationships are to be delivered externally.
Business subtransaction	/BA1/C55SBTNO	Item	If postings are made to lower level nodes for structured products.
Post to currency leg	/BA1/C55USELEG	Item	If postings are made to the component position object of a currency leg directly (instead of the primary position object with the same currency). For example, for forward exchange transactions.

> **Caution**
>
> The characteristics /BA1/C55IMPAIR, /BIC/TEXTHID, /BA1/C55SBTNO and /BA1/C55USELEG from the table above permit empty fields. Therefore, do not define these as required entry fields.

Derivation from Descriptive Characteristics of the Financial Position

You can derive descriptive characteristics of the financial position from the characteristics of the SDL primary object (of the business transaction and also the from the financial transaction or instrument). The following derivation processes available for this)to be run when posting external business transactions):

- Derivation according to the settings in the Customizing activity *Derive Descriptive Characteristics for Fin. Positions + Funding Centers* (under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Characteristics of Financial Positions*
- Derivation according to the settings in the Customizing activity *Edit Secondary Data Source for Descriptive Characteristics* (under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Characteristics of Financial Positions*

This is where you can define a secondary data source that takes characteristics from primary objects.

4.4.5.2.4.4 Error During Posting of External Business Transactions (Subledger Scenario)

Errors that occur when external business transactions are posted can be caused by:

- Missing or incomplete generation
- Incomplete Customizing or Customizing with errors
- Problems that arose when the data was delivered from the source system or from the transformation layer

To identify the cause of the error, analyze all the warnings and error messages in the application log for the report *Post External Business Transactions*. The system saves the business transactions that have not been posted in a worklist in the update run. You can restart these business transactions from this worklist using a special report. In a simulation run, the same business transactions continue to be available to run the report.

Before you can restart the business transactions from the worklist or post in the update run you must resolve any problems that have occurred. The first two problems listed above can be solved by completing and correcting Customizing.

If the data has been incorrectly delivered by the source system, however, the system cannot post the business transactions concerned. A business transaction that is missing important characteristic values that are relevant to posting (for example, type of business transaction) cannot be fixed. Once the system has created a business transaction you cannot change it in the SDL.

Note the following:

- You cannot reverse business transactions with errors in Accounting.
  This is because you can only reverse a reversal business transaction if the original business transaction has been posted. To the system, the reversal business transactions themselves would be incorrect, and they would be, as with the original business transactions, saved in a worklist.
- If the business transactions were not delivered by a source system but were manually created, you can generate a corrected copy of the incorrect business transaction during test operation. You must then enter the missing characteristic values manually.
- Where possible, use specialized business transaction classes. This helps to ensure that business transactions with errors are not delivered.

Error Correction

We recommend the following course of action to resolve errors:

1. If you have run the *Post External Business Transaction* report, note the name of the program run and the date.
2. Analyze the warnings and error messages in the application log, particularly the long texts. In particular, look out for:
   - Missing Customizing or Customizing with errors
   - Corrupted business transactions
3. Edit the Customizing settings where they are either incomplete or incorrect.

4. In the previous simulation run: Start the *Post External Business Transactions* report in the update run.
   In the previous update run: Start the *Restart External Business Transactions* and enter the name of the program run concerned, as well as the date.
5. Check whether all business transactions have been posted.
   If they have, then the problem has been solved.
   If there are still errors, the problems must be related to corrupt business transactions. Proceed as follows:
6. Reverse the corrupt business transactions
7. Generate the corrected business transactions
8. Deliver the corresponding reversal and corrected business transactions to the SDL.
   This ensures that the status in the source system and the status in the SDL are consistent.

Errors in Packages

In Accounting, the external business transaction delivered from the SDL are grouped into packages. When an external business transaction is posted, the following error message may be displayed: *Package #: # business transactions were flagged for restart*.

Note the following:

- The error message can be caused by errors in the business transaction data or in Customizing. Where errors are caused by incorrect BT data, you have to create the BTs in question again correctly.
- The system posts all the **correct** business transactions in the package.
- If the following errors occur, the relevant business transaction is flagged for restart. However, the system still posts the other, error-free business transactions in the package.
  - Important fields have not been filled in or have been filled in incorrectly. In particular, the system checks to see if the legal entity, the business transaction- and item type, the currency-, amount-, and quantity fields have been filled in correctly.
  - Errors have occurred within Accounting, for example, lock errors.
- The system only rejects entire packages if it is unable to determine which particular business transaction is responsible for the error. Possible causes are:
  - Incorrect generation process
  - The necessary fields have not been configured

> **Note**
>
> We would like to explicitly point out that the business transactions causing the errors must not be ignored. In order to ensure that data is kept consistent, all Customizing errors must be resolved before further processing in accounting, and if the business transactions affected still cannot be posted, steps must be carried out to make sure that the data is resupplied correctly from the source system or from the transformation layer.

4.4.5.2.4.5 Deleting Business Transactions from Worklist

Context

Business transactions that cannot be booked due to missing or incorrectly entered key figures are known as corrupt. If a package contains both corrupt and correct business transactions, either the business transactions with errors are deferred or the whole package is rejected depending on the type of error that occurred. The system saves the rejected business transactions in a Parallel Processing Tool [page 722] worklist. You can use this report to identify and delete corrupt business transactions from this parallel processing tool worklist.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Tools* > *Delete Business Transactions from the Worklist*.
2. Enter a legal entity.
3. Enter the name of the program for an application process and press the enter key to confirm.

   The system then fills the *Date of Program Run* and *Sequential Number of Program Run* fields automatically.
4. Enter a package number. When external business transactions were posted and restarted, they were divided into packages.
5. Choose  .

4.4.5.2.4.6 Update Secondary Business Transactions

Use

This process is used to generate and update internal business transactions, for example, for the calculation of price gains or losses realized for a sale.

Prerequisites

- You have executed the Set the SDL Time Stamp [page 367] function.
- If you want to use a log variant, you must create it in advance.
- You have carried out the Processing of Accounting Events [page 769] process.
- To transfer and process operational flow transactions, you have executed the Post External Business Transactions [page 774] process.

- You have configured the Customizing activities in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes*.

> **Note**
>
> The update of secondary business transactions does not include documents from year-end closing. For example, if a business transaction has been reversed after you have already carried out year-end closing, you must repeat this.

Features

The following categories of internal business transaction are available:

- Derived business transaction
  When the external business transaction is posted, items from the operational business transaction that have to be processed in financial position management are posted to clearing accounts in financial position management. The derived business transaction contains the position and net income effects and triggers the clearing postings.
- Accrual/deferral and valuation
- Year-end closing
- Hedge business transactions
  - Designation
  - Dedesignation
  - Dissolution (due to a hedge management decision or the *Process Results of Effectiveness Test* function)

Examples:

1. The revenue from the sale of a security is determined by comparing the amounts received (business transaction data) and the book value of the sold security. In order for the book value to be determined, the financial position value before the outflow took place has to be calculated.
2. If there are several purchases and sales for the same financial position on one day, the revenues assigned to the sales depend on whether the purchases have already been allowed for in the financial position value. If a purchase is processed after a sale has already been posted, the sales postings may have to be reversed and the sales processed again once the purchase has been posted.
3. Position management for securities includes the management of long (asset-side) and short (liability-side) positions. The key figures affected by inflows and outflows often depend on the current financial position situation, the relevant amounts from a business transaction have to be divided into short and long positions.
4. Calculation procedures post the secured and unsecured parts of hedging instruments to different G/L accounts for each financial position component. At the end of the day and based on end-of-day balances, the system transfers the balance of the relevant key figure proportionally to the relationship between the hedged nominal value and the total nominal value.

   For more information, see Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Hedge Accounting* > *Reclassification of Hedging Instruments*.
5. Depending on the calculation procedure, many business transactions require an amortization calculation (seeAmortization (subledger scenario) [page 890]), for example, to determine the value of the retiring part in

the case of sales. Detailed information about the financial position trend has to be made available in addition to the information from the external business transaction.

6. An external business transaction was reversed retroactively. If a key date valuation is rendered invalid as a result, the system automatically reverses the key date valuation and recalculates it.

Error Messages in the Application Log

If errors (inconsistencies) occur in the financial positions when you update the derived business transactions, the following messages appear in the application log:

- Messages about inconsistencies
- Calculation logs for inconsistencies

Messages About Inconsistencies explain the type of inconsistencies that have occurred (for example, when posting to clearing accounts).

The *Calculation Log for Inconsistencies* explains which calculation results caused the inconsistencies. Click *Calculation Log for Inconsistencies* and choose *Details*. In the detail log, lines that belong together have been organized into different color groups in the following sequence:

1. The balances **before** the incorrect, but processed, business transaction.
2. The items from the primary business transaction.
3. The items from the derived business transaction.
4. The totals derived from 1 to 3.

The system does not save the application log in the database. You can, therefore, display the application log only while it is still in the memory.

Posting Date and Value Date Are Not the Same

If the posting date and the value date are not the same, this triggers an analytical adjustment of the two-dimensional derivation in the *Update Secondary Business Transactions* process, that is, a rollback and a recalculation.

The following example explains what this analytical adjustment looks like:

Business transaction A has already been derived by the *Update Secondary Business Transactions* process. The result of the two-dimensional derivation is business transaction A*. There can also be one-dimensional derivations; these are not relevant in this document.

The system processes business transaction B, to which the following applies:

- The posting date of B is after the posting date of A
- The value date of B is before the value date of A

In this case, the results of business transaction A (and derivation A*) are no longer valid. The system performs an analytical adjustment and creates an adjustment for business transaction A comprising the following new business transactions:

- Rollback
  The system creates a rollback for the two-dimensional derivation A* (in this example, this is called rollback A-). Rollback A- reverses the derivation A* at the current exchange rate.
- Recalculation, or a new two-dimensional derivation A**

The new business transactions A- and A** have the value date of business transaction A and the posting date of business transaction B.

Reversals with Later Posting Date

The *Update Secondary Business Transactions* process makes a distinction between the following business transactions:

- Reversal takes place on a posting date that is **identical** to the posting date of the original.
  In this case, the system ignores both business transactions (reversal and reversed business transaction) in further processing.
- Reversal takes place on a posting date that is **after** the posting date of the original
  In this case, the system reverses all derived business transactions that were triggered by the original business transaction and that are two-dimensional (in other words, a value date has been entered) on the later posting date and at the exchange rate valid on the later posting date.
  As a result of analytical adjustments, for example (see the section **Posting Date and Value Date Are Not the Same**), there can be more than one of these two-dimensional, derived business transactions for the original business transaction.
  The system also generates a one-dimensional derivation for the reversal business transaction on the later posting date.
  In the special case of a reversal also being reversed on a later posting date (reversal of a reversal), processing is carried out as follows: In addition to the activities described above, the system generates new one-dimensional and two-dimensional derivations of the business transaction on the posting date of the reversal of the reversal and at the exchange rate in effect on that posting date.
  For more information, see *Reversals with Later Posting Date* in the **Features** section under Reversal [page 795].

> **Note**
>
> Depending on the set-up (in particular the Customizing settings) a one-dimensional derivation only or a two-dimensional derivation only can be generated, even if both one-dimensional and two-dimensional derivation have been triggered.

Activities

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Day Processing* > *Automatic Processing* > *Update Secondary Business Transactions* .

Example

Initial situation

Assets		Liability	
MB	5,000,000	EQ	5,000,000

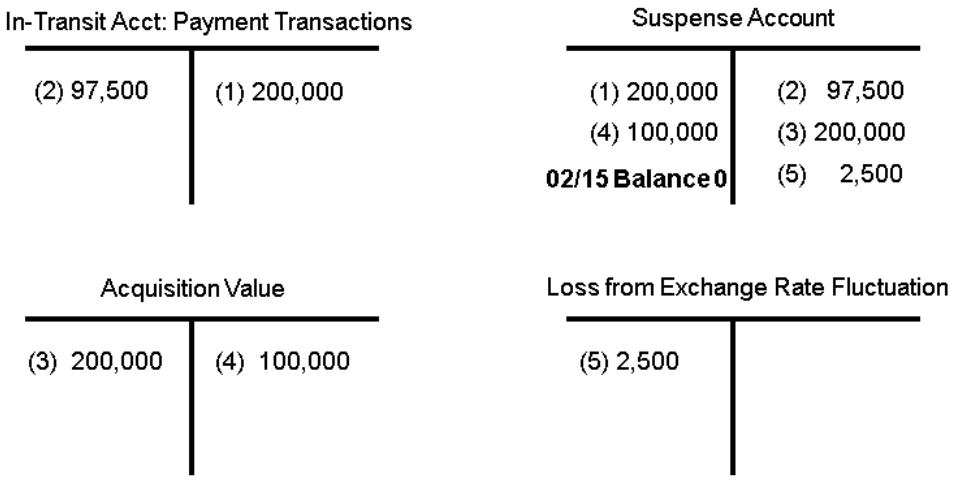
Business transactions

Stock X in sec. acct Y, holding cat. AFS

01/20 Purchase 1,000 units 200,000

02/15 Sale 500 units 97,500

1. Initial Position



(1) 01/20 Purchase of 1,000 units: Position clearing to payment clearing: 200,000

(2) 02/15 Sale 500 units: Payment clearing to position clearing: 97,500

Updating Internal Business Transactions:

(3) 01/20 Acquisition value to position clearing: 200,000

(4) 02/15 Suspense account to acquisition value: 100,000

(5) 02/15 Loss from exchange rate fluctuation to position clearing: 2,500

2. Updating Secondary Business Transactions

Stock X Sec. Acct Y AFS			Financial position					E & R
				Interim account	Acquis. value			Exch. loss
No	Date	Item type	Units	Amount	Amount			Amount
1	01/20	Purchase	1,000	200,000				
2	02/15	Sale	-500	97,500				
	02/15	Balances:	500	102,500				
1a	01/20	Purchase (a)		-200,000	200,000			
2a	02/15	Sale (a)		100,000	-100,000			
		Exch. loss (a)		-2,500				2,500
	02/15	Balances:	500	0	100,000			2,500

Updating secondary business transactions triggers position clearing.

More Information

Processes in the AFI Scenario [page 678]

4.4.5.2.4.7 Update Secondary Business Transactions (Simulation)

Use

You can use this function to update secondary business transactions in simulation mode. The function matches the Update of Secondary Business Transactions [page 786] function in all respects except the following:

- You are able not only to execute the process as a mass run but also for specific financial positions. By entering selection criteria such as the financial transaction ID, you can restrict the selection.

- The system always executes the process in simulation mode.

You can use the simulation of the update of secondary business transactions to perform specific analyses on selected financial positions, for example to analyze errors.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Tools* > *Update Secondary Business Transactions (Simulation)*.
2. Enter the following on the selection screen:
   - *Legal Entity*, *Accounting System*, and *Reversal Reason*
   - Criteria for restricting the financial position selection: *Financial Transaction ID*, *ID of Settlement Account*, *Financial Instrument ID*, *Securities Position Account ID*, *Aggregation Object ID*
   - Temporal restrictions on the update date: *Update from*, *Update to*
   - Technical settings: *Filter variant* and *test run*.

   > **Note**
   >
   > Note that the system **always** executes the process in simulation mode. The setting you make for *Test Run* has no bearing here. If you want to execute the process as an update run, go to the *SAP Easy Access* screen and choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Daily Processing* > *Automatic Processing* > *Update Secondary Business Transactions*.
3. Choose to start the simulation.
4. You can now see a results list. You can select individual financial positions. For example, you can display details of each financial position or the posting results. To process the selected financial positions in simulation mode, choose .

4.4.5.2.4.8 Reaction to Analytical Events During the Update of Secondary Business Transactions

Use

When it updates secondary business transactions, the system processes analytical events that were generated when impairment events were changed.

For more information, see Event Processing in the AFI Scenario [page 684].

Prerequisites

- In the Customizing activity *Derive Application Events from Analytical Events* under *Bank Analyzer* > *Processes and Methods* > *Application Event Management*, you have selected whether the derivation for the assignment of an analytical event category to an application event category is set to active.
- You have defined the relevant business transaction types and item types for the subsequent activities in the Customizing activities under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Business Transaction Types and Item Types*.
- In Customizing activity *Edit Generation Reasons for Business Transactions* under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Application Events* > *Accounting Events*, you have defined how a business transaction type and an item type is linked to the relevant accounting event.

Features

Accounting event processing generates an event document for each impairment event.

You can only use one business transaction type for these event documents. This business transaction type must be assigned to business transaction category `1070`. You can also only use one item type for each item template.

For each posting date, the update of secondary business transactions process processes the impairment events after operational business transactions. More specifically, the system processes operational business transactions based on the impairment characteristic values from the previous day When it processes impairment events, the system updates the descriptive impairment characteristics of the financial position object depending on the item templates of the event document items, and generates the required documents (for example, to adjust the risk provision).

The system adjusts the impairment characteristics of the financial position object only if the event document uses the item template `2I30`, `2I34`, `2I35`, `2I36` or `2I37`. The impairment characteristics of the financial position object are changed by using the values from the Results Data Layer (RDL).

The documents are generated by executing calculation procedures. The system derives a value-date-dependent calculation procedure and a value-date-independent calculation procedure:

- For the value-date-dependent calculation procedure, we provide the procedure template `1117` *Change to Non-Accruing* and `1118` *Change to Accruing*. For an unchanged accrual status, we provide the procedure template `1150` *Reclassification - Value-Date-Based*.
- For the value-date-independent calculation procedure, we provide the procedure template `2170` *Update of Impairment*.

There can be more than one impairment event document for each posting date. Even if there are multiple event documents for one posting date, the system executes only one value-date-dependent calculation procedure and only one value-date-independent calculation procedure. It takes all the event documents into account when deciding which steps are to be executed.

4.4.5.2.5 Reversal

Use

The reversals described here relate to business transactions for financial instruments and financial transactions. Reversals for documents created in accounting for Bank Analyzer also have to be triggered in accounting. Similarly, reversals for documents from external business transactions have to be triggered externally (by the source system). Accounting does not run any validations during a reversal. External business transactions cannot be corrected manually in accounting.

Prerequisites

Plus/Minus Sign Logic

You can choose how the reversal document is determined. The following options are available:

- The system inverts the plus/minus sign and leaves the debit/credit indicator as it is. This means an item that was posted to the debit side in the original business transaction is deducted from the debit side in the reversal document.
- The system inverts the plus/minus sign and the debit/credit indicator. This means an item that was posted to the debit side in the original business transaction is updated on the credit side with inverted plus/minus sign in the reversal document.

Reason for Reversal

You have to enter a reversal reason if you want to process a reversal business transaction.

Features

A reversal completely cancels an original business transaction. An original posting must exist to which the reversal can refer. The system reads the data from the original posting, reverses the document, and posts it again. A reversal is thus the inverse posting of the original posting. The system does not execute the posting rules again. Accounting assumes that the source system delivering the reversal has checked whether the specified original document really is to be reversed. The reasons for this are that:

- Errors have to be corrected in the source system, not in the accounting system.
- There must be no inconsistencies between the source system and accounting system.
- This procedure can be used for periods and posting blocks of less than one year, as well as for those that cross fiscal years.

Manual reversals are possible only for business transactions created within accounting.

Reversals with Later Posting Date

A reversal with a posting date that is after the posting date of the original is not a simple inverse posting of the original business transaction. In this case, the reversal business transaction not only has a different plus/minus sign than the original document but can also have different characteristic values along with a different posting date.

You can also define that the system uses the exchange rate of the later posting date for the currency translation. To do so, in Customizing under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*, you need to add the key figure `&55AMBFX` *Base Amount for Currency Translation* in the result category `HF` for result node category `FLOW`. If you do not add this key figure in Customizing, the system reverses the business transactions using the exchange rate from the original business transaction. In other words, the system creates business transactions with identical values but with reversed plus/minus signs.

For more information, see the section **Reversals with Later Posting Date** under Update Secondary Business Transactions [page 786].

Reversal Business Transaction

The reversal business transaction differs from the original business transaction in the data that has to be delivered. Only the fields *Business Transaction ID*, *Source System of Reversed Business Transaction*, *Business Transaction ID for the Reversal Business Transaction*, and *Legal Entity* are required to process a reversal business transaction correctly in accounting. These SDL fields are transferred directly to the fields *Document Key for the Reversal Document, Sender ID, Document Key for the Document to Be Reversed* and *Legal Entity* of the accounting business transaction. You do not need to make the same Customizing settings as for the original business transaction.

Internal business transactions generated because of a (reversed) external business transaction are also completely reversed in accounting the next time that secondary business transactions are updated [page 786].

Accounting in Bank Analyzer does not support partial reversals. This means a business transaction is either reversed completely or not at all.

Reversal Peg

For reversals, a recognizable link has to be created between the document for the reversal business transaction and the document for the original business transaction (reversal peg). Reversal documents can also be reversed. This in turn triggers a reference to the original business transaction. The reversal peg between the original document and the first reversal document is canceled and a reversal peg is created between the first and second reversal documents. The original document reverts back to the state it was in before the first reversal took place. If you reverse a reversal document, the original document is simply remarked as "not reversed.

A new (corrected) original posting is usually generated as a result of a reversal. This posting is not assigned in any way to the original posting or its reversal.

> **Note**
>
> The applies only to reversals with the same posting date, not for reversals with a later posting date.

Effects of Reversals on Financial Position Management

Reversals affect financial position management because several internal business transactions can already have occurred for a reversed external business transaction. These are also reset the next time secondary business transactions are updated.

Average Exchange Rates and Consumption Sequence Procedure

The weighted average method is used for the consumption sequence procedure. The average exchange rates are recalculated during the valuation. In the case of backdated reversals, the average exchange rates have to be recalculated from the reversal date for the key date valuation and the derived business transactions. The average exchange rates change accordingly in the following periods.

Corporate Actions

As for all other business transactions, reversals of corporate actions are also delivered to the SDL as reversals of business transactions. Related business transactions are reversed together.

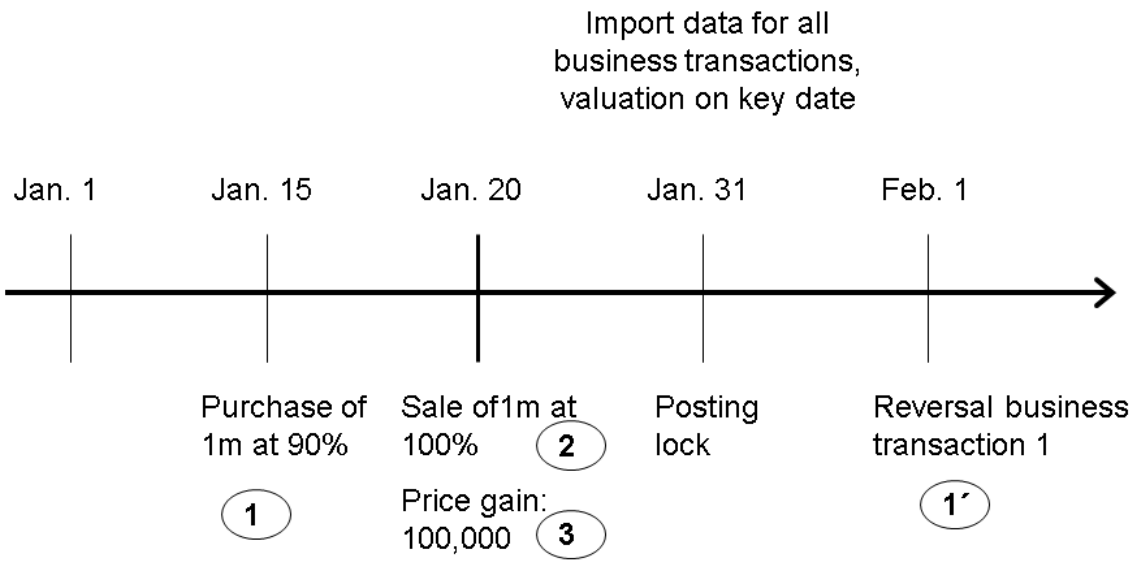
Activities

- You configure the settings for the plus/minus sign in the reversal document in Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Results Data Area*.
- You define reversal reasons in Customizing for *Bank Analyzer* under *Results Data Layer* > *Settings for Flow Results* > *Edit Reversal Reasons*.
- For information about reversing financial products manually, see Entering/Reversing Manual Posting [page 945].

Example

Posting Block at Month-End Closing

Reversal of a purchase from January in February:



In February, it was established that the bond was sold at 80%. A posting error has therefore been made. The report for January remains the same, the reversal posting takes place with a posting date in February because of the posting block.

Business transactions period 1:

BT1: Purchase on January 15

BT2: Sale on January 20

BT3: Derived BT for BT2, gain 100,000

Business transactions period 2:

BT1': Reversal BT for BT1 on February 1

The posting date for the reversal of BT1 is February 1, 2001. This means the data does not change for the period up to January 31. In evaluations for January 31, the status is the same as it was before BT1 was reversed.

Conventional Reporting Using T-Accounts

Period 1: January

Book value

1 900,000 (15/01)	**3a** 900,000 (20/01)

Payment clearing

2 1,000,000 (20/01)	**1** 900,000 (15/01)

Clearing

3a 900,000 (20/01) **3b** 100,000 (20/01)	**2** 1,000,000 (20/01)

Trading result

	3b 100,000 (20/01)

Period 2: February

Book value	
1' -900,000 (02/01)	**2a** -900,000 (02/01)
3' 800,000 (02/01)	**4a** 800,000 (02/01)

Payment clearing	
	1' -900,000 (02/01)
	3' 800,000 (02/01)

Clearing	
2a -900,000 (02/01)	**4b** 200,000 (01.02)
2b -100,000 (02/01)	
4a 800,000	

Trading result	
	2b -100,000 (02/01)
	4b 200,000 (02/01)

Postings in period 1:

1: Purchase on January 15

2: Sale on January 20

3a and 3b: Derived BT for BT2, gain 100,000

Postings in period 2:

1': Reversal posting for BT1 on February 1

2a and 2b: Reversal of derived BT3 on February 1

3': Provision of a new, external BT purchase at 80%

4a and 4b: Generation of a new, derived BT for BT2, gain 200,000

4.4.5.2.6 Calculation Management (Subledger Scenario)

Calculation Management defines the core component of accounting processes within Bank Analyzer.

It generates secondary business transactions as part of processing for operational business transactions, operational events, and analytical events (secondary business transactions update) and also as part of period-end processes (key date valuation).

The main task of Calculation Management is to use calculation procedures to execute the processing logic defined in these procedures and save the results of this processing permanently in posting documents.

Calculation procedures define the following:

- How the system processes (posts) an operational business transaction, an operational event, or an analytical event in the respective accounting standard
- How the system restricts and measures a financial position in the respective accounting standard
- How the system carries out foreign currency valuations

Calculation Management uses the following entities:

- System entity: calculation procedure template
  - Customizing entity: Calculation procedure
- System entity: calculation step category
  - Customizing entity: calculation step
- System entity: calculation method
- System entity: item template
  - Customizing entity: item type
- System entity: processing category
  - Customizing entity: posting key figure

The system entities outlined above are available in the standard system. You can obtain the listed Customizing values in Business Content. You can adjust or redefine these values as part of your implementation project.

Customizing settings are validated by means of the system entities provided by SAP.

Note

The following entities are not unique to Calculation Management since they are also used to post external business transactions, for example:

- Item template
- Item type
- Processing category
- Posting key figure

Note

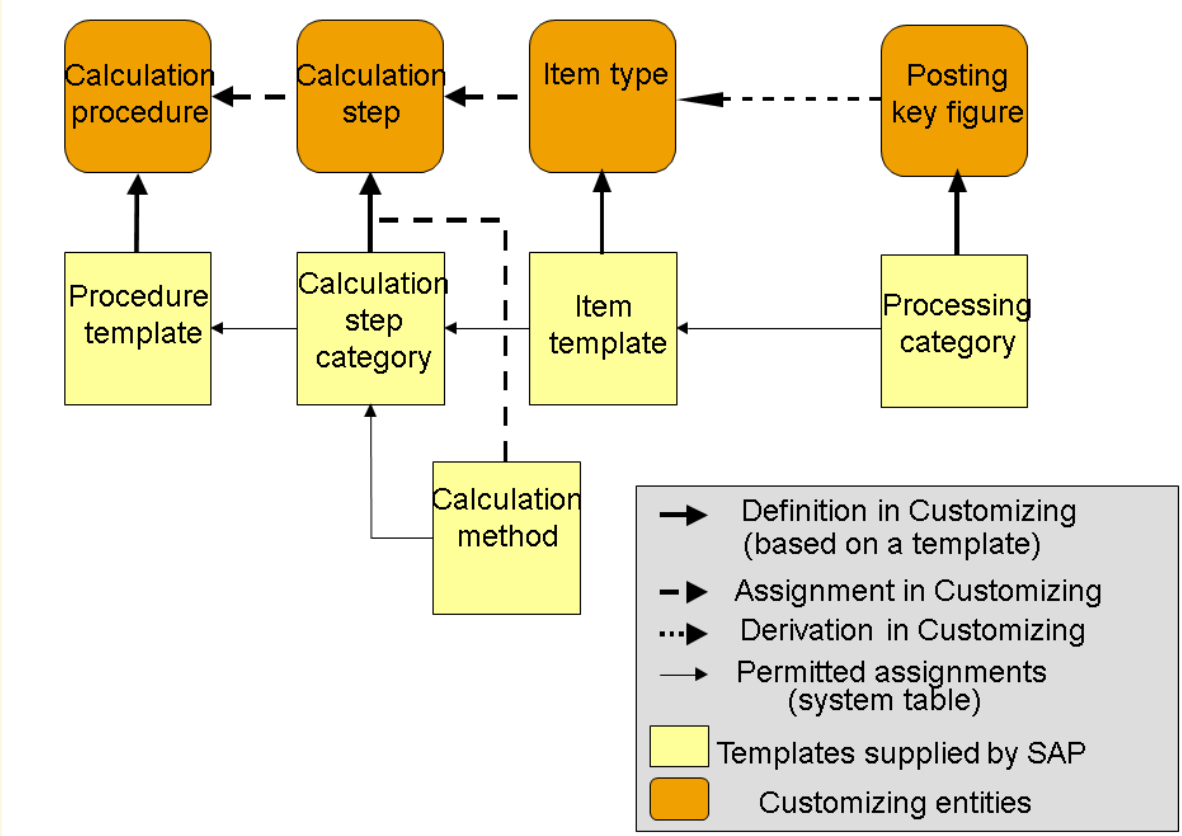


Figure 1: The structure of the calculation procedures in Calculation Management

The following sections provide a detailed description of the relationships depicted in Figure 1:

Calculation Procedure [page 821]

Calculation Steps (Subledger Scenario) [page 867]

Calculation Methods [page 870]

Item Types [page 872]

Customer Enhancements [page 877]

4.4.5.2.6.1 Processing Categories

Use

You have to assign each posting key figure to a processing category in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Processing Categories* > *Assign Key Figures to Processing Categories*.

Features

Some of the standard processing categories provided by SAP are described below.

Operational Processing Categories

These processing categories define the direct effect of the operational business of the company. You can reconcile your balances against operational balances.

Product-Specific Position

Processing Category	Description	Definition
1BF_RECPAY	[O] Receivable/Payable	Defines the legal receivable or payable with respect to a contract partner (counterparty or issuer, for example) resulting from flow transactions. The system shows the balance of this key figure either without incidental acquisition costs (**long-term receivable**) or accumulated with the acquisition costs (**acquisition value**). **Note** In single-currency swap transactions (such as interest swaps), the opposite payments are offset against each other operationally and so do not actually flow. This means that the processing category does not carry a balance in this case, which is correct.

Processing Category	Description	Definition
1BF_DEF	[O] Deferrals	Defines the deferral items (flowed in advance). The system transfers these over a predefined period to the profit and loss statement based on the accounting standard specifications. The balance based on this processing category defines the remaining amount still to be deferred. Methods for clearing the deferral item: • Linear • Weighted by the outstanding principal amount
1BF_DEFDIS	[O] Deferrals - Premium/Discount	Defines the deferral items (flowed in advance), particularly premiums and discounts. The system transfers these over a predefined period to the profit and loss statement based on the accounting standard specifications. The balance based on this processing category defines the remaining amount still to be deferred. Methods for clearing the deferral item: • Effective-interest based • Straight-line method • Weighted by the outstanding principal amount
1BF_ACCR	[O] Accruals	Defines the accrual items (paid in arrears), such as accrued interest and charges.

In-Transit Accounts

Processing Category	Description	Definition
1BG_TRANS	[O] In-Transit Account (Anonymous)	Defines an operational suspense account (for example, payment transactions) at cumulated level. Balances for this key figure can be reconciled against separate opposite operational flow transactions.

Processing Category	Description	Definition
1BF_TRANS	[O] In-Transit Account (Product)	Defines an operational suspense account (for example, short-term receivable) at single-contract level. Balances for these key figures indicate that an operational open item exists.
1BF_TRANSO	[O] Offset In-Transit Account (Prod.)	Defines an offset account for the operational in-transit account (1BF_TRANS) or the receivable (1BF_RECPAY). This processing category is used, based on an analytical decision to handle the individual contract as non-accruing, to offset the operational in-transit account or receivable for interest settlements.

Profit and Loss Statement / Reserves

Processing Category	Description	Definition
1IX_PL	[O] Profit and Loss	Defines the operational expenditure and revenue (such as interest revenue and interest expenditure). **Note** Incoming postings in a transactional currency different from the functional currency are converted by the system immediately by means of *Move & Transform* into a transactional currency that is the same as the functional currency.
1BF_EQUITY	[O] Equity (Product-Specific)	Defines the operational reserve. This can result from the processing of non-monetary financial instruments for which the foreign exchange result has to be entered in the reserve. Besides the use for operational currency reserve, no other use cases are known.
1BG_EQUITY	[O] Equity (Anonymous)	Defines the operational reserve. This processing type is used in year-end closing (**balance carryforwards**) to determine the profit or loss brought forward.

Multi-Currency Accounting

Processing Category	Description	Definition
1BG_FXEV	[O] Currency Position Equivalent Value	Defines the equivalent value of a specific foreign currency (such as CHF) in functional currency (such as EUR) from the operational business activity. All documents entered on this basis always have a transactional currency (such as EUR) that is the same as the functional currency (such as EUR). The system always updates the underlying reference currency (such as CHF) in the financial position currency. The system uses this processing category to update the balance-sheet-relevant foreign exchange result (and the currency reserve) as part of the MCA posting logic. **Note** The currency position (1BG_FX, 1BF_FX) triggers the calculation of the foreign exchange result as part of the MCA posting logic.
1BF_FXEV	Manual Posting: [O] Currency Position Equivalent Value	Unlike processing category 1BG_FXEV,, this is used for postings to product-related financial positions and not to general financial positions. Is used in the foreign currency profit transfer posting for manual postings and during foreign currency valuation in the postprocessing of manual postings. This is because postings associated with manual postings cannot be made to product-related financial positions.

Processing Category	Description	Definition
1BG_FX	[O] Currency Position (Anonymous)	Defines the balance of a specific currency (such as CHF) in transactional currency (such as CHF) from the operational business activity. A currency position always belongs to a well-defined part of the overall financial statement (for example, balance sheet and P&L statement). You can use this position to further refine the total exposure managed at the equivalent value level. The system uses this processing category to trigger the determination of the balance-sheet-relevant foreign exchange result (and the currency reserve) at the equivalent value level as part of the MCA posting logic. The foreign exchange result determined using this processing category is for information purposes.
1BF_FX	[O] Currency Position (Product-Based)	Defines the balance of a specific currency (such as CHF) in transactional currency (such as CHF) from the operational business activity. You can always assign this currency position to a well-defined part of the overall financial statement (reserve) and thus use it to further refine the total exposure managed at the equivalent value level. The system uses this processing category to trigger the determination of the balance-sheet-relevant foreign exchange result (and the currency reserve) at the equivalent value level as part of the MCA posting logic. The foreign exchange result determined using this processing category is for information purposes.

Processing Category	Description	Definition
1BF_UNREAR	[O] Unremitted Earnings	Defines the operational expenditure and revenue (such as interest revenue and interest expenditure) in foreign currency (transactional currency different from functional currency). **Note** Incoming postings are always made in a transactional currency different from the functional currency. You can convert these at the end of the period (as part of the key date valuation) using the exchange rate valid at this time and transfer these to the profit and loss statement (1IX_PL).

Processing within Bank Analyzer Processes

Processing Category	Description	Definition
1BF_SUSP	[O] Suspense Accounts (Product-Specific)	This processing category is used as part of two-level business transaction processing for Bank Analyzer (posting of external business transactions, processing of secondary business transactions). A balance based on this processing category after all the update processes have run though indicates that external business transactions were transferred, but could or were not (yet) processed in the *Update Secondary Business Transactions* process.
1BG_SUSP	[O] Suspense Account (Anonymous)	This processing category is used as part of the component transfer procedure to distribute operational flow transactions between the contract header and the individual contract elements. A balance for this processing category leads to termination, and an error message is issued when secondary business transactions are processed.

Value-Added Tax

Processing Category	Description	Definition
1BG_VAT	[O] Value Added Tax (Anonymous)	Defines the amount of value-added tax to be paid.

Operational Processing Categories (According to Settlement Date Accounting)

These processing categories define the direct effect of the operational business of the company. You can reconcile your balances against operational balances. You use the processing categories in the logic for settlement date accounting to separate predated value date-based activities.

Product-Specific Position

Processing Category	Description	Definition
1OF_RECPAY	[OBS] Receivable/Payable	The definition is the same as the processing category 1BF_RECPAY. This processing category is used in settlement-date accounting only, where it carries the predated value date-based balances.
1OF_DEFDIS	[OBS] Deferrals - Premium/Discount	The definition is the same as the processing category 1BF_DEFDIS. This processing category is used in settlement-date accounting only, where it carries the balances with a value date in the future.

In-Transit Accounts

Processing Category	Description	Definition
1OG_TRANS	[OBS] In-Transit Account (Anonymous)	The definition is the same as the processing category 1BG_TRANS. This processing category is used in settlement-date accounting only, where it carries the balances with a value date in the future.
1OF_TRANS	[OBS] In-Transit Account (Product-Specific)	The definition is the same as the processing category 1BF_TRANS. This processing category is used in settlement-date accounting only, where it carries the balances with a value date in the future.

Profit and Loss Statement / Reserves

Processing Category	Description	Definition
1OF_DEFINC	[OBS] Deferred Income	The definition is the same as the processing category 1IX_PL. This processing category is used in settlement-date accounting only, where it carries the balances with a value date in the future. Since these are off-balance-sheet processing categories, *Move and Transform* is not carried out here.
1OG_EQUITY	[OBS] Reserve (Anonymous)	The definition is the same as the processing category 1BG_EQUITY. This processing category is used in settlement date accounting only.

Processing within Bank Analyzer Processes

Processing Category	Description	Definition
1OF_SUSP	[OBS] Suspense Account	This processing category is used in settlement-date accounting logic to map the carryforward from on-balance-sheet to off-balance-sheet processing when secondary business transactions are processed. A balance for this processing category leads to termination, and an error message is issued when secondary business transactions are processed.

Analytical Processing Categories

These processing categories define analytical enrichments (valuation results) that the analytical system determines for the first time.

Product

Processing Category	Description	Definition
2BF_DEF	[A] Valuation Remnants to Be Released	This processing category is used when the valuation approach is changed (for example, through reclassification, hedge designation, and hedge dissolution). When such a change is made, you can transfer valuation components (such as fair value adjustment, hedge adjustment) to this processing category in order to be "released" over a specific time period, and thereby ensure the continuity of the profit and loss statement.
2BF_VALHAD	[A] Hedge Adjustment	Defines the hedge adjustment.
2BF_VALFV	[A] Fair Value Adjustment	Defines the valuation remnants factor fair value adjustment.
2BF_IMPUNW	[A] Valuation Remnants (Expected Cash Flow)	This processing category is relevant for transactions with a book value that is defined by an expected cash flow, for instance as it is defined in the context of handling impairment. Only the deterministic consequences of the expected cash flow, such as the time effect (the unwinding), are assigned to this processing category.
2BF_IMPRP	[A] Impairment - Risk Provision	Defines the amount of risk provision created for an existing impairment.
2BF_WRTDWN	[A] Write-Down	Defines the write-down. This is defined by analytical decisions.

Profit and Loss Statement / Reserves

Processing Category	Description	Definition
2IX_PL	[A] Profit/Loss	Defines the analytical results (such as the result of the fair value valuation). **Note** Incoming postings in a transactional currency different from the functional currency are converted by the system immediately by means of *Move & Transform* into a transactional currency that is the same as the functional currency.
2BF_EQUITY	[A] Equity (Product-Based)	Defines the analytical reserve. This results from the valuation of financial positions of the category *Fair Value Through OCI*.
2BG_EQUITY	[A] Equity (Anonymous)	Defines the analytical reserve. This processing type is used in year-end closing (*balance carryforwards*) to determine the profit or loss brought forward.
2BF_EQDEF	[A] Equity to Be Deferred	This processing category is used when the valuation approach is changed (for example, through reclassification, hedge designation, and hedge dissolution). When such a change is made, you can transfer reserve components (such as fair value adjustment) to this processing category so that these can be "dissolved" over a specific time period, to ensure the continuity of the profit and loss statement.

In-Transit Accounts

Processing Category	Description	Definition
2BG_TRANS	[A] In-Transit Account (Anonymous)	Defines the non-product-specific analytical suspense account. This processing category is used for the transfer posting of analytical processing categories (such as asset/liability switch; transfer posting and reclassifying a position). Balances based on this key figure can be reconciled against opposite analytical flow transactions.

Multi-Currency Accounting

Processing Category	Description	Definition
2BG_FXEV	[A] Currency Position - Equivalent Value	Defines the equivalent value of a specific foreign currency (such as CHF) in functional currency (such as EUR) from the analytical position valuation. All documents entered on this basis always have a transactional currency (such as EUR) that is the same as the functional currency (such as EUR). The system always updates the underlying reference currency (such as CHF) in the financial position currency. It uses this processing category to update the balance-sheet-relevant foreign exchange result (and the currency reserve) as part of the MCA posting logic. **Note** The currency position (2BG_FX, 2BF_FX) triggers the calculation of the foreign exchange result as part of the MCA posting logic.

Processing Category	Description	Definition
2BF_FXEV	Manual Posting: [A] Currency Position - Equivalent Value	Unlike processing category 2BG_FXEV, this is used for postings to product-related financial positions and not to general financial positions. Is used in the foreign currency profit transfer posting for manual postings and during foreign currency valuation in the postprocessing of manual postings. This is because postings associated with manual postings cannot be made to product-related financial positions.
2BG_FX	[A] Currency Position (Anonymous)	Defines the balance of a specific currency (such as CHF) in transactional currency (such as CHF) from the analytical position valuation. A currency position always belongs to a well-defined part of the overall financial statement (for example, balance sheet and P&L statement). You can use this position to further refine the total exposure managed at the equivalent value level. The system uses this processing category to trigger the determination of the balance-sheet-relevant foreign exchange result (and the currency reserve) at the equivalent value level as part of the MCA posting logic. The foreign exchange result determined using this processing category is for information purposes.
2BF_FX	[A] Currency Position (Product-Based)	Defines the balance of a specific currency (such as CHF) in transactional currency (such as CHF) from the analytical position valuation. You can always assign this currency position to a well-defined part of the overall financial statement (reserve). You can use it to further refine the total exposure managed at the equivalent value level. This processing category triggers the determination of the balance-sheet-relevant foreign exchange result (and the currency reserve) at the equivalent value level as part of the MCA posting logic. The foreign exchange result determined using this processing category is for information purposes.

Processing Category	Description	Definition
2BF_UNREAR	[A] Unremitted Earnings	Defines the analytical results (such as the result of the fair value valuation). **Note** Incoming postings are always made in a transactional currency different from the functional currency. You can convert these at the end of the period as part of the key date valuation using the exchange rate valid at this time and transfer these to the profit and loss statement (2IX_PL).

Processing within Bank Analyzer Processes

Processing Categories	Description	Definition
2BF_SUSP	[A] Suspense Account (Product-Based)	Defines a trigger event for processing analytical events (such as a reclassification decision) while processing internal business transactions.
2BG_SUSP	[A] Suspense Account (Anonymous)	This processing category processes analytical key figure balances as part of the component transfer procedure (for example, entry of an implicit fee for settlement of swap transactions).

Deferred tax

Processing Categories	Description	Definition
2BF_DFTXDF	[A] Deferred Tax on Deferrals	Defines the amount of deferred tax generated due to differing accrual methods in the accounting standards to be compared.
2BF_DFTXHA	[A] Deferred Tax on Hedge Adjustment	Defines the amount of deferred tax generated due to differing collateralization in the accounting standards to be compared.

Processing Categories	Description	Definition
2BF_DFTXFV	[A] Deferred Tax on Fair Value Adjustment	Defines the amount of deferred tax generated due to differing residual valuations in the accounting standards to be compared.
2BF_TXDEF	[A] Deferred Tax to Be Deferred	Defines the amount of deferred tax to be cleared, which always results in cases where a valuation has been changed (for example, through reclassification).

Technical Processing Categories

In a period-end process (for example, accrual/deferral run, valuation run, and so on), the system does not generate any documents for calculation steps that would post an amount of 0 in each currency to reduce the data volume. If none of the steps carried out in this context generate a document to be permanently saved, the system generates a zero document. This zero document is required since the system needs to know that a period-end process was carried out at this point in time that must be triggered when retroactive changes are made to a position through flow transactions; in this case, the calculation steps carried out generate documents containing a balance that the system has to update.

Processing Categories	Description	Definition
1BX_ZERO	[O] Zero Flow	This processing category indicates that there are no amounts to be posted in all steps for an accrual/deferral run.
2BX_ZERO	[A] Zero Flow	This processing category indicates that there are no amounts to be posted in all steps of a valuation run.

Naming Convention

There is a naming convention for the technical name of the processing categories:

- The first character indicates either an operational processing category (**1**) or an analytical processing category (**2**).

  > **Note**
  >
  > This naming convention also applies to the following entities:
  > - Item templates
  > - Step categories

- The second character indicates the following:
  - If the processing category is relevant for the balance sheet, the character **B** is used.
  - If the processing category is relevant for the profit and loss statement, the character **I** is used.
  - If the processing category is off-balance-sheet relevant, the character **O** is used.

- The third character indicates the following:
  - If the processing category is relevant for a financial position, the character **F** is used.
  - If the processing category is relevant for general financial positions, the character **G** is used.
  - If the processing category is relevant for financial positions or general financial positions, the character **X** is used.

4.4.5.2.6.2 Calculation Procedure Classes

Use

Calculation procedure classes group several procedure templates. Each procedure template is assigned to exactly one calculation procedure class. This assignment is delivered by SAP.

The Customizing is structured according to the calculation procedure classes. During the accrual procedure, for example, the system only displays selection options that are relevant for the calculation procedure class *Accrual*.

Activities

To display calculation procedure classes in Customizing for *Bank Analyzer*, choose *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Procedures* > *Display Calculation Procedure Classes*.

Example

The following examples are based on the subledger scenario for accounting:

Calculation Procedure Class - Accrual

The following value-date-based procedure templates belong to this calculation procedure class:

- `1201` Accrual
- `1290` Initialization - Value-Date-Based Calculation Base (system initialization)
- `1295` Re-Initialization - Value-Date-Based (special case re-initialization of accounting as of Bank Analyzer 8.0)

Calculation Procedure Class - Key Date Valuation

The following non-value-date-based procedure templates belong to this calculation procedure class:

- `2201` Valuation
- `2202` Valuation - Time Bucket Objects
- `2290` Initialization - Not Value-Date-Based Calculation Base (system initialization)
- `2295` Re-Initialization - Not Value-Date-Based (special case re-initialization of accounting as of Bank Analyzer 8.0)

4.4.5.2.6.2.1 Deferred Taxes (Subledger Scenario)

Use

Deferred tax liabilities are future corporate tax liabilities resulting from existing taxable temporary differences. Deferred tax claims are tax benefits that can be obtained at some point in the future. These 'benefits' can result from deductible temporary differences, losses carried forward, or unused tax credits. All the financial products (as in IAS 39) that are managed in accounting are subject to the automatic determination and posting process. We differentiate between deferred taxes that have an operational origin:

- Deferred taxes on deferrals – processing category `2BF_DFTXDF`

And two categories of deferred taxes that have an analytical origin:

- Deferred tax on hedge adjustment – processing category `2BF_DFTXHA`
- Deferred tax on fair value adjustment – processing category `2BF_DFTXFV`

Due to this distinction, deferred taxes can be reported separately for the following:

- Accrual/deferral items
- Hedge adjustment
- Fair value

The system always posts deferred taxes on deferrals against the profit and loss account. Depending on the settings made in Customizing, you can post deferred tax on hedge adjustment and fair value adjustment to the P/L or reserve, for example, using the holding category. See Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules*. The system assumes that the differences between the tax balance sheet and IAS balance regarding deferrals, hedge adjustment and fair value adjustment are always temporary for financial instruments.

Since an impairment is a permanent reduction in value, the system does not include it in the calculation of deferred tax. These kind of permanent impairments must, therefore, not have been deducted from the imported tax value before. The system does not make any further automated distinction or carry out any plausibility checks for the temporary and permanent differences.

Business Content does not contain all of the Customizing required for deferred tax. You can refer to the Business Content description for information about which settings you still need to make.

For a detailed description of the calculation of deferred tax, see procedure template 2201 under Not Value-Date-Based Procedure Templates 2101 to 2602 [page 839]. The step categories 2T* can be used for deferred tax in this procedure template. There, you can read the documentation available via the hotspots for the calculation methods MT*. The system calculates the deferred tax during a key date valuation using additional steps in the calculation procedure that is not value-date-dependent.

See also:

- Accounting Processes (Subledger Scenario) [page 693]
- Posting Date and Value Date [page 764]

Prerequisites

Importing Values from the Tax Balance Sheet

Before you perform the relevant key date valuation, you have to import up-to-date values from the tax balance sheet for the key date for which you want the system to calculate the deferred tax.

- The *Tax Value* key figure in local currency (`&55TXVL`)
- The *Amortized Cost (Tax Balance Sheet)* key figure in local currency (`&55TXAC`)

You have to import these key figures scaled with regard to nominal/quantity.

You have to import these key figures to the results data area *SBA* into a results type based on the result category `HKFLG` and a result node type based on result node category `HKFLG`.

To do so, use Business Content or set up your own result type based on the following entities in Business Content:

- Results data area `SBA` Global results data area
- Results type `S_DEFTAX` PML(FA): Calculation Base Deferred Tax
- Result node type `S_DEFTAX`

You have to define the following characteristics as a key in the corresponding result type:

- `/BA1/C40FTRAN` Financial Transaction ID
- `/BA1/C41FINST` Financial Instrument ID
- `/BA1/C43CLACC` Securities Position Account ID
- `/BA1/C55LGENT` Legal Entity
- `/BA1/C80ORGUNI` Organizational Unit
- `/BA1/CR0FCCPCR` Fin. Contract: Component Currency
- `/BA1/CR0FCCPCT` Fin. Contract: Component Cat.
- `/BA1/CR0FCSBCPID` Fin. Contract: Subcomponent ID

You define this result type for the respective IFRS accounting system in Customizing for Bank Analyzer under *Accounting Systems and Legal Entities* > *Define Accounting System*. In the *Result Types* section, enter *Imported Key Date Results* under *Use*.

Note

- If you have not defined a local currency because the functional currency is the same as the local currency, you have to import the key figures in functional currency.
- You have to import the key figures on the basis of the subcontracts. To do this, fill the following key characteristics of the result type used:
  - `/BA1/CR0FCCPCR` Component Currency
  - The `/BA1/CR0FCCPCT` component type as in the Customizing activity *Assign Component Type* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *SDL Connection*.
    The component type is only relevant for structured products.
  - `/BA1/CR0FCSBCPID` Node number of the subcomponent
- The key figures `&55TXVL` and `&55TXAC` are not posting key figures. The system uses them as an internal buffer.
- Since you cannot carry out a key date valuation with a deferred tax calculation on the first that a tax rate change is valid, you cannot import the key figures on this date.

- To improve performance when reading the tax book values in the RDL, you have to create two secondary indexes with the following characteristics:
  1. `/BA1/C43CLACC` and `/BA1/C41FINST` (note: the selected characteristic should come first).
  2. `/BA1/C40FTRAN` and `/BA1/C42SACC`

     You make these settings in Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Technical Settings in Result Data Area*:
     Choose the relevant results data area and the results type for key date results

Features

All the financial instruments (as in IAS 39) that are managed in accounting for financial instruments come under the automatic determination and posting process. The system calculates the difference between the tax balance sheet and IFRS balance sheet for deferrals, hedge adjustment and fair value adjustment on the basis of the tax value and the amortized cost from the tax balance sheet, which have to be imported from feeder systems. The system posts these values to the IFRS balance sheet accordingly.

Accounting for financial instruments allows you to carry out the following:

- Calculate deferred taxes as part of the valuation of financial positions
- Post deferred taxes to either affect net income or not to affect net income, depending on the effect that the tax-relevant difference in value changes between the tax balance sheet and the IFRS balance sheet has on net income.
- Take tax rate changes into account
- Take balances on deferred taxes into account for the following:
  - Position transfer (procedure templates 2140 and 2141)
  - Reclassification (2150)
  - Reclassification (2111), hedge dissolution (2132)
  - Position changes for listed financial instruments (2102)
  - FX valuation (2212) and restatement (2213)

    In Customizing for *Bank Analyzer*, choose *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Procedures* > *Display General Overview of All Templates*.
  - Fulfill reporting requirements

Accounting for financial instruments does not cover the following:

- Inclusion of postings for portfolio fair value hedge
- Continuous updates to deferred tax for inflows and outflows of nominal/quantity in financial transactions.
- Comparison to tax balance sheets that contain hedge adjustments.
- Calculation of deferred tax on the first date that a tax rate change is valid

Note

- During calculation, the system uses the last tax value to be imported or the value for the amortized cost from the tax balance sheet.
  The system does not check how up-to-date the imported results are. Zero is a valid value which means that if no value has been imported, the system uses the value zero for the comparison with the IFRS balance sheet. However, the system writes a warning message to the log.

- Deferred tax is posted at the granularity level of a financial position. The system posts in transaction currency = local currency. If a local currency has not been defined, the system assumes that the functional currency is the same as the local currency.
- For structured products, the system posts the deferred tax at the level of the subobjects. It does not generate postings for deferred tax at the header level of the structured product. Key figures from the tax balance sheet have to be imported in the same way at the level of the subobjects.
- To calculate the deferred taxes, the system uses the currently valid tax rates. If no tax rates have been stored for a financial instrument that has been selected for calculating deferred taxes, the system reports an error (error log). The system saves the tax rate used for creating a posting document for deferred tax as an attribute in the document line item.

For more information, see Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Accounting Processes* > *Valuation* > *Deferred Taxes*:

- *Define Tax Rates*
- *Derive Tax Rates*

Key date valuation, change in tax rate

The system needs a separate run for tax rate changes so that it can display the effects of a tax rate change separately from the effects of other calculation bases for deferred tax. Accounting for financial instruments has the key date valuation classes *End-of-Day Position* and *Start-of-Day Position*.

The key date valuation class *End-of-Day Position* uses the financial position balances at the end of the day; key date valuation class *Start-of-Day Position* uses those at the beginning of the day.

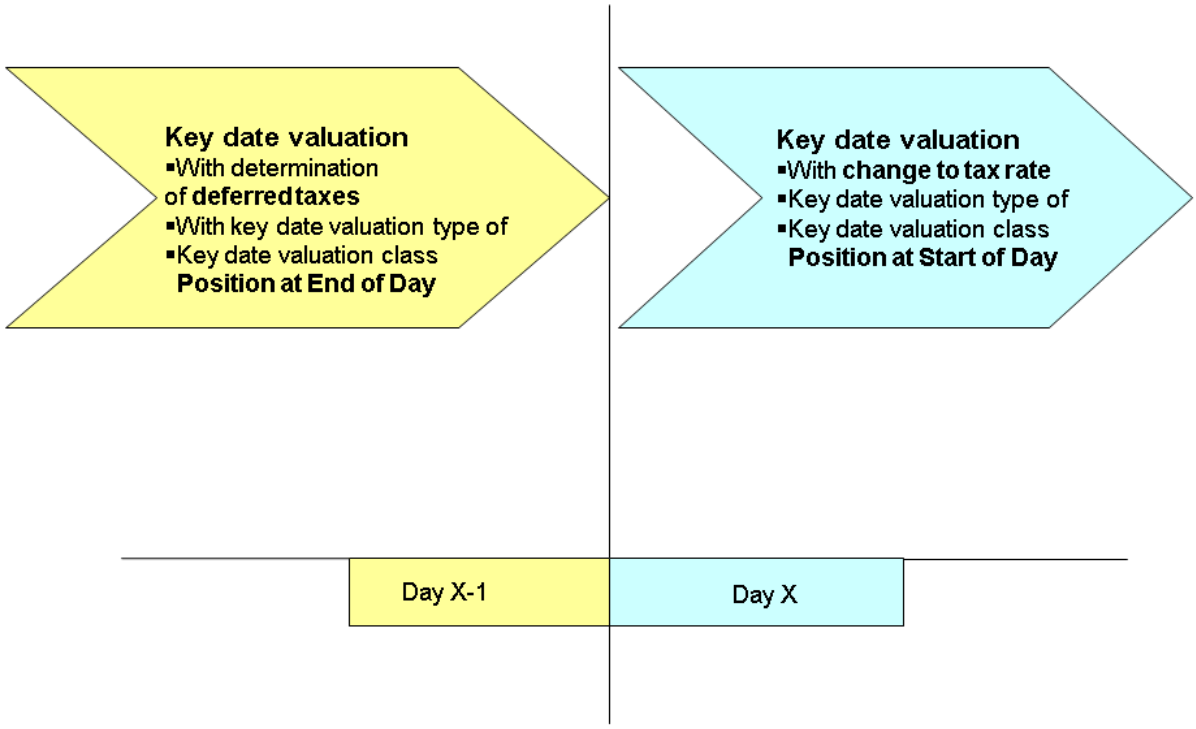
Based on these key date valuation classes, you can define key date valuation types.

For a key date valuation run choose a type of key date valuation. If a tax rate change is to take place within a key date valuation you have to choose a key date valuation type from the Start-of-Day Position key date valuation class. If, however, deferred tax is to be calculated within a "normal" key date valuation, then you must choose a type of key date valuation from the End-of-Day Position key date valuation class.

Procedure for tax rate change on day X:

- Run a key date valuation on day X-1 with deferred tax calculation using a key date valuation type from the *End-of-Day Position* key date valuation class.
- Run a key date valuation on day X with a key date valuation type from the *Start-of-Day Position* key date class. Because the key date valuation type is the *source field* for determining the calculation method, assign different calculation procedures to both the key date valuations as necessary.

You can use the following functions in the IAS solution:



4.4.5.2.6.3 Calculation Procedure

Use

A **calculation procedure** describes how an internal business transaction is generated, in other words, it defines which calculation steps [page 867] the system executes in which order.

The procedures map the business posting logic using their assigned calculation steps and the resulting postings.

Activities

Executing Calculation Procedures

You can trigger calculation procedures using the following transactions:

- Update secondary business transactions [page 786]

- Run key date valuation

Table 1 provides an overview of the events for which the system executes a calculation procedure as part of these transactions.

You can find the Customizing activities under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions*:

Trigger	Event	Customizing Activity	Procedure Class
Primary business transaction	Different payment currency	Procedures for different payment currency	Different payment currency
	Component transfer	Derive Component Transfer Procedure	Component transfer
	Elementary derivation (value-date-based)	Derive Elementary Calculation Procedures (Value-Date-Based)	Elementary Derivation - Value-Date-Based
	Elementary derivation (not value-date-based)	Derive Elementary Calculation Procedures (Not Value-Date-Based)	Elementary Derivation - Not Value-Date-Based
	Dissolution of hedging relationships due to position outflows	Assign Dissolution Strategies to Accounting System	Elementary Derivation - Not Value-Date-Based
Key Date Valuation	Accrual	Derive Accrual Procedures	Accrual/Deferral
	Key date valuation	Derive Calculation Procedures for Key Date Valuation	Key date valuation
	Valuation of foreign currency positions	Assign Foreign Currency Valuation Procedures and Business Transaction Types: • Calculation Procedures • Foreign Exchange Position	FX Valuation - AFI
	Valuation of balance sheet items	Assign Foreign Currency Valuation Procedures and Business Transaction Types: • Calculation Procedures • Balance Sheet Items	FX Valuation - AFI

Trigger	Event	Customizing Activity	Procedure Class
	Generation of restatement documents	Assign Foreign Currency Valuation Procedures and Business Transaction Types: • Calculation Procedures • Restatements	FX Valuation - AFI
	P/L transfer of unremitted earnings	Assign Procedures for Foreign Currency Profit Transfer Postings: Calculation Procedure for P/L Transfer of Unremitted Earnings	Move and Transform Result
P/L posting	GuV transfer	Assign Procedures for Foreign Currency Profit Transfer Postings: Calculation Procedures for P/L Transfer	Move and Transform Result
End of day	Asset/liability transfer	Assign Asset/Liability Transfer Methods to Accounting Systems	Elementary Derivation - Not Value-Date-Based
	Reclassification of hedging instruments	Assign Reclassification Calculation Procedures to Accounting System	Hedge Reclassification

Table 1: Execution of the calculation procedures for the *Update Secondary Business Transactions* and *Run Key Date Valuation* processes

For the *Update Secondary Business Transactions* process, the system first runs a component transfer procedure for each primary business transaction (in the case of structured products), then the value date-based elementary derivation and the non-value-date-based elementary derivation. In addition, hedging relationships are dissolved for position outflows, if necessary, and a dissolution strategy is executed accordingly.

The component transfer procedure, the value-date-based elementary derivation, and the non-value-date-based elementary derivation are performed only if the corresponding business transaction type has been entered as to be derived in the relevant derivation rule. You make this setting in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Accounting Processes* > *Assign Derived Business Transaction Type to Primary Business Trans. Type*.

Which strategy is to be executed is specified in the system in the corresponding Customizing activities, which are also listed in table 1. Some of these Customizing activities are derivations, others are just simple assignments.

When processing a business transaction, a calculation procedure uses the results of the calculation procedures previously executed. For example, a component transfer procedure processes an external business transaction and, in so doing, generates an internal business transaction for each component. These internal business transaction items are processed further by the elementary value date-based calculation procedure. If this is

followed by an elementary non-value-date-based procedure, all business transaction items generated in the component transfer procedure and elementary value date-based calculation procedure are available for this procedure.

Procedure classes are used to structure the Customizing settings for defining the relevant procedures, which is explained further below.

In the *Key Date Valuation* process, the system normally runs the procedures specified in table 1 in the sequence specified, for each position. Whether the relevant procedures are executed depends on the settings for the selected key date valuation type.

If a position is updated on a posting date, and a key date valuation was already run with the same or a later posting date, the system reverses the key date valuation within the process *Update Secondary Business Transactions* and runs it again. It executes the procedures specified under key date valuation in table 1 within the process *Update Secondary Business Transactions*.

If a P/L posting is generated in one of the previously described procedures, the system automatically triggers a P/L transfer, in other words, executes a P/L transfer procedure for each P/L posting. This applies for the *Update Secondary Business Transactions* process and the *Key Date Valuation* process.

In the *Update Secondary Business Transactions* and *Key Date Valuation* processes, a check is carried out at the end of each posting day for which postings have been generated, to determine whether an asset/liability transfer or a hedge reclassification must be carried out to execute the relevant procedures, if required.

As a rule, a secondary transaction is generated when each calculation procedure is executed. You must also assign a business transaction type to each secondary business transaction. This business transaction type is used as information in the document. The assignment to business transaction types is carried out also in the Customizing activities listed in table 1, with the following exceptions:

- In the Customizing activity *Assign Derived Business Transaction Type to Primary Business Trans. Type*, select the business transaction type for the business transactions generated by component transfer procedures and elementary procedures.
- If a P/L transfer procedure is triggered by a P/L posting, an additional business transaction is not created but instead additional items are created in the business transaction generated by the triggering procedure.

Procedure Templates

Technically, you create all the calculation procedures in the same way. These procedures are also handled by the system in the same way. Each calculation procedure is based on a procedure template.

You can use a procedure template to set up a specific calculation procedure in Customizing. These procedure templates contain calculation step categories that you can use. You specify the order in which the system executes the calculation steps and which steps are mandatory in a calculation procedure. You cannot change the procedure templates that we provide.

If you create a new calculation procedure, the system suggests using the relevant calculation steps that are available. This suggestion is based on the calculation step categories. If you have created more than one calculation step for a calculation step category, the system displays all the steps of this category for your selection.

> ⚠ Caution
>
> If you have not created a calculation step for a calculation step category, the system does not take it into account when suggesting a calculation procedure. If the step category is mandatory, an incorrect calculation procedure may be suggested.

The following Customizing activity gives you an overview of all the selections that you can make when defining calculation procedures, and of the interaction between templates for calculation procedures, calculation step categories, and item templates.

- In Customizing for *Bank Analyzer*, choose *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Procedures* > *Display General Overview of All Templates*.

You use the following Customizing activity to display the procedure classes and the calculation procedure templates that contain the step categories.

- In Customizing for *Bank Analyzer*, choose *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Procedures* > *Display Templates for Calculation Procedures*.

Definition of Calculation Procedures

To structure calculation procedures and simplify the definition of calculation procedures in Customizing, procedure classes were introduced.

Table 2 provides an overview of the procedure classes.

You can find the Customizing activities under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions*.

Procedure Class	Procedure Class	Customizing Activity	Prefix for Procedure Templates
0024	Foreign Currency Swap	Edit Procedures for Variant Payment Currency	15
0003	Component Transfer	Edit Component Transfer Procedure	14
0022	Elementary Derivation - Value-Date-Based	Edit Elementary Calculation Procedures (Value-Date-Based)	11
0021	Elementary Derivation - Not Value-Date-Based	Edit Elementary Calculation Procedures (Not Value-Date-Based)	21
0010	Accrual/Deferral	Edit Accrual Procedures	12
0002	Key Date Valuation	Edit Rating Model for Key Date Valuation	22
0013	FX Valuation - AFI	Edit Valuation Procedures for Foreign Currencies	22

Procedure Class	Procedure Class	Customizing Activity	Prefix for Procedure Templates
0005	Move and Transform Result	Edit Procedures for Foreign Currency Profit Transfer Postings	26
0015	Hedge Reclassification	Edit Reclassification Calculation Procedures	21

Table 2: Relevant procedure classes for the *Update Secondary Business Transactions* and *Key Date Valuation* processes

For each procedure class, there is a separate Customizing activity for creating calculation procedures. In these Customizing activities, you define your own calculation procedures and assign the calculation steps.

Each procedure class has a specific number of procedure templates. Each procedure template has a four-character name, where the first two characters are fixed for each class.

In the overview of the events in table 1, these events for which the system executes a calculation procedure are assigned procedure classes, for example. The respective procedures to be executed must be defined in the relevant Customizing activity for the procedure class. For example, the procedure for the asset/liability transfer belongs to the procedure class "Elementary Derivation - Not Value-Date-Based" and therefore must defined in the Customizing activity *Edit Elementary Calculation Procedures (Not Value-Date-Based)*.

4.4.5.2.6.3.1 Value-Date-Based Procedure Templates

Use

The calculation procedure templates based on value date are described below. You can analyze the procedure templates in detail in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings*:

- *Calculation Procedures* > *Display General Overview of All Templates*
- *Calculation Control* > *Edit Procedure Templates*

Features

1101: Banking Book Position Generation

You use procedure template `1101` to generate positions for financial transactions in the banking book.

Examples:

- Loan disbursement

- Conclusion of repurchase agreement

The business transactions can also contain items for premium/discount and charges. Premium/discount and charges can also be provided as separate business transactions.

The procedure template contains the following calculation step categories:

- `1D01` Adjustment - Premium/Discount
  - This step category is always required if an amortization needs to be calculated later (for example, for a repayment or key date valuation) since the step also initializes the relevant calculation base data in this context.
    See also: Amortization [page 890]
  - The assigned methods `MD09` and `ME17` do not play a role in this context.
  - If you want to calculate accruals/deferrals based on the effective interest rate even when no premium/discount exists, you can use method `MD01` instead of method `MD07`.
- `1E01` Inflow to Receivables/Payables
  - This step category is required to generate positions for receivables/payables.
- `1E03` Inflow to Premium/Discount - Implicit
  - This step category calculates a premium/discount if the transaction amount and the nominal amount in the position for position generation do not match.
- `1E04` Inflow to Premium/Discount - Explicit
  - For premium/discount that is supplied separately as a business transaction (position).
- `1E05` Transfer Expense/ Income from Payment
  - For charges
  - You use the method `ME23` for charges that are to be deferred so that the corresponding calculation base data is initialized. Otherwise, you use the method `ME23`.
- `1E98` Update 2D Calculation Base Data (Non-Accruing)
  This step category is used to prevent unscheduled position changes triggering a recalculation of the effective interest rate during the accrual status *Non-Accruing*. This would be relevant if the accrual status changes to *Accruing* at a later point.
- `1D11` Adjustment - Premium/Discount
  You need this step category to make a catch-up adjustment to the accrual position for the premium/discount after a contractual cash flow change for products that are reported at amortized cost.

1102: Banking Book Scheduled Position Reduction

You use procedure template `1102` for scheduled position reductions for financial transactions in the banking book (for example, loan repayment).

Examples:

- Loan repayment
- Repurchase agreement maturity

The business transactions can also contain items for premium/discount and charges. You can supply this premium/discount and these charges as separate business transactions. If you do so, you should also derive these business transactions on the basis of procedure template `1102`.

Calculation Step Categories:

- `1D01` Adjustment - Premium/Discount

This step category is required if you want to calculate amortization for the financial position. See procedure template `1101` for the relevant methods.

- `1E06` Outflow from Receivables/Payables - Planned
  This step category is required to generate positions for receivables/payables.
- `1E04` Inflow to Premium/Discount - Explicit
  As for procedure template `1101`
- `1E05` Transfer Expense/ Income from Payment
  As for procedure template `1101`
  - For charges
  - You use the method `ME23` for charges that are to be deferred so that the corresponding calculation base data is initialized. Otherwise, you use the method <sap-technical-name>ME23</sap-technical-name>.
- `1E98` Update 2D Calculation Base Data (Non-Accruing)
  This step category is used to prevent unscheduled position changes triggering a recalculation of the effective interest rate during the accrual status *Non-Accruing*. This would be relevant if the accrual status changes to *Accruing* at a later point.

1103: Banking Book Unscheduled Position Reduction

You use procedure template `1103` for unscheduled position reductions for financial transactions in the banking book (for example, loan repayments).

An *Unscheduled Loan Repayment* is an example of this. The business transactions can also contain items for premium/discount reimbursement and charges. You can import this premium/discount reimbursement and these charges as separate business transactions. If you do so, you should also derive these business transactions on the basis of procedure template 1103.

Calculation Step Categories:

- `1D01` Adjustment - Premium/Discount
  This step category is required if you want to calculate amortization for the financial position. See procedure template `1101` for the relevant methods.
- `1E02` Outflow from Receivables/Payables
  This step category is required for an unscheduled reduction of receivables/payables.
- `1E38` Pro Rata Outflow - Premium/Discount - Financial Transaction
- `1E10` Result (realization)
- `1E04` Inflow to Premium/Discount - Explicit
  For premium/discount reimbursement
- `1E98` Update 2D Calculation Base Data (Non-Accruing)
  This step category is used to prevent unscheduled position changes triggering a recalculation of the effective interest rate during the accrual status *Non-Accruing*. This would be relevant if the accrual status changes to *Accruing* at a later point.
- `1E05` Transfer Expense/ Income from Payment
  As for procedure template `1101`
- `1D11` Adjustment - Premium/Discount
  You need this step category to make a catch-up adjustment to the accrual position for the premium/discount after a contractual cash flow change for products that are reported at amortized cost.
- `1EA2` Outflow: Offset Interest Receivable
  You can use this step category to post the amount for an unscheduled interest payment as an outflow from the interest receivable offset to the interest income.

1104: Banking Book and Trading Book: Position Change Financial Instrument

Template `1104` is used for the following:

- Purchase or sale of securities
- Repayment of securities
- Purchase or sale of derivatives
- Purchase and sale of stocks
- Change to acquisition value without change to nominal or quantity (for example, pool factors in asset-backed securities, cash settlements for equity instruments or repayment transactions for securities in which only the acquisition value but not the nominal value is to be changed)

The business transactions can also contain items for premium/discount (position template `1I14/1I15`) and charges (position template `1I18/1I19`). You can supply this premium/discount and these charges as separate business transactions. If you do so, you should also derive these business transactions on the basis of procedure template `1104`.

Position changes **with** change to nominal or quantity are represented by a position in the business transaction with item template `1I00` or `1I01`. For position changes **without** changes to nominal or quantity use item template `1I32` or `1I33`.

Calculation Step Categories:

- `1D01` Adjustment - Premium/Discount
  - This step category is required if you want to calculate an amortization.
- `1E08` Pro Rata Outflow - Payables/Receivables - Financial Instrument
  - For credit postings to assets or debit postings to liabilities.
- `1E11` Pro Rata Outflow - Premium/Discount - Financial Instrument
  - This step category is required for amortization in the event of position outflows.
- `1E01` Inflow to Receivables/Payables
  - For debit postings to assets or credit postings to liabilities. For a position change from long to short, both the step category `1E08` and the step category `1E01` generate a posting.
- `1E03` Inflow to Premium/Discount - Implicit
  - As for procedure template `1101`
- `1F19` Transfer Foreign Currency Position (Anonymous/Prod.-Based)
  - Relevant for non-monetary financial instruments; the documents from the step categories `1E08` and `1E01` are the basis for this.
- `1E10` Result (realization)
  - Realized profit/loss from the step categories `1E08`, `1E01` and `1E11`.
- `1E05` Transfer Expense/ Income from Payment
  - For charges
    As for procedure template `1101`

1105: Position Update - Value-Date-Based

Procedure template 1105 is used to update accruals/deferrals due to events for financial transactions. Examples:

- Cash flow changes (for example, for rollovers)
- Change to accrual status
- Change to impairment status

Calculation Step Categories:

- 1A01: Adjustment - Accruals
  Required if the accrual for the non-value-date-based procedure has to be up-to-date. This is the case when you determine the risk provision in the event of impairment, for example.
- 1D01 Adjustment - Premium/Discount
  Required for amortization to close the old effective view and/or create current values for use in the non-value-date-based procedure.
  - Methods MD01 and MD07 calculate based on the effective interest rate.
  - Method MD02 calculates using the straight-line method.
  - Use method MD09 for impairment instead of MD01.
  - Use method ME17 for rollovers instead of MD01/MD07/MD02.
- 1D02 Adjustment Deferrals
  Required if the accrual for the non-value-date-based procedure has to be up-to-date. This is the case when you determine the risk provision in the event of impairment, for example.
- 1E92 Amortization - Premium/Discount
  For a change to non-accrual/deferral, for example, due to operational illness.
- 1E93 Clear Accruals
  For a change to non-accrual/deferral, for example, due to operational illness.
- 1E94 Clear Deferrals
  For a change to non-accrual/deferral, for example, due to operational illness.
- 1E46 Clearing - Offset In-Transit Account (Product-Based)
- 1E07 Start New Effective View
  Required for effective interest rate based accruals for changes in cash flow or operational illness. If the method ME17 is executed in step category 1D01, step category 1E07 is no longer required.
- 1D11 Adjustment - Premium/Discount
  You need this step category to make a catch-up adjustment to the accrual position for the premium/discount after a contractual cash flow change for products that are reported at amortized cost.

1106: BB and TB: Position Change Financial Instrument (w/o Nominal)

Procedure template 1106 is used for the primary external business transaction *Variation Margin Payment* for ETD derivatives (business transaction type DE03 , business transaction category 1001), and for the primary external business transaction *Cash Settlement for Equity Instruments* (business transaction type EQ05, business transaction category 1001). The template's calculation step category 1E69, to which the calculation method ME05 is assigned, enables you to transfer the transaction amount of the external business transaction for the inflow or outflow of the receivable or payable.

For non-monetary financial instruments in the banking book, the calculation step category 1F19, to which the calculation method MF17 is assigned, carries out a transfer between the anonymous and product-based foreign currency position.

> **Note**
>
> If you use functions that offer both the procedure template 1106 and the procedure template 1104, we recommend you use procedure template 1104.

1107: Derivative (OTC): Upfront Payment

You use procedure template 1107 for the primary external business transaction Post Deferrals for OTC and ETD Derivatives. The template's calculation step category 1E04, to which the calculation method ME04 is assigned, enables you to post an upfront payment for OTC swaps (business transaction type DO02, business transaction category 1001), and to post an option premium for OTC options (business transaction type DO02, business transaction category 1001) and ETD options (business transaction type DE04 , business transaction category 1001).

The business transaction *Post Deferral* is posted to the relevant component of the financial transaction. A component transfer is, therefore, not required.

1108: Derivative (OTC): Debit position

You use procedure template 1108 for the primary external business transaction Settlement of Derivative (OTC) (business transaction type DO03, business transaction category 1001). The template's calculation step category 1E09 , to which the calculation method ME13 is assigned, and the calculation step category 1E35, to which the calculation method ME01 is assigned, post the outflow or inflow of receivables and nominal/quantity according to the amounts from the business transaction. Operational expense and income, such as interest payments, can also be posted using the calculation step category 1E05, to which the calculation method ME05 is assigned. The business transaction *Debit Position* is posted to the components of the financial transaction. A component transfer is, therefore, not required.

1109: Derivative (OTC): Close Out

You use procedure template 1109 for the primary external business transaction *Settlement of Derivative (OTC)* (business transaction type DO04, business transaction category 1001). The template's calculation step category 1E09, to which the calculation method ME13 is assigned, posts the outflow of receivables and nominal/quantity according to the amounts from the business transaction. The calculation step category 1E95, to which the calculation method ME11 is assigned, enables the relevant outflow of the option premium for OTC options. The business transaction *Close-Out* is posted to the components of the financial transaction. A component transfer is, therefore, not required.

1110: Derivative (OTC/ETD): Maturity

You use procedure template for the primary external business transaction *Maturity of OTC and ETD Derivatives* combined with the component transfer procedure (business transaction type DO05 or DE05, business transaction category 1001). The template's calculation step category 1E12, to which the calculation method ME14 is assigned, enables you to close out a receivable and the nominal. In addition, the calculation step category 1E92, to which the calculation method ME17 is assigned, can be used to clear the option premium.

1111: Reclassification (value date-based)

In the two-dimensional part of the procedure for reclassification a position update first takes place, during which the system redetermines various accrual/deferral items. Afterwards, you can clear the discount or make it explicit when you change the accrual method. The procedure template contains the following step categories in the sequence as they are listed here:

- 1D05 Deferrals (Reclassification)
- 1D06 Adjustment - Deferrals (Reclassification)
- 1A03 Adjustment - Accruals (Reclassification)
- 1E14 Clear Discount
- 1E13 Make Discount Explicit

When you use a specific procedure, for which you have to make Customizing settings, you can insert steps based on the step categories. Depending on the reclassification direction (for example, from fair value (P&L) to amortized cost), only a subset of the step categories is relevant – for example, only `1E14` or `1E13`.

1112: Spot Exchange Transaction

Procedure template `1112` is used for the following:

- Spot exchange transaction (business transaction type `DO06`, business transaction category `1001`)
- Implicit currency swap (business transaction type `DM01`, business transaction category `1006`)

Calculation Step Categories:

- `1F13` Entry of currency position if transaction currency 1≠ functional currency
- `1F14` Entry of currency position if transaction currency = functional currency
- `1F15` Entry of equivalent value position if transaction currency 2 ≠ functional currency (only relevant for implicit currency swap)
- `1F16` Entry of equivalent value position if transaction currency 2 = functional currency (only relevant for implicit currency swap)
- `1F17` Entry of equivalent value position if transaction currency ≠ functional currency
- `1F01` Clear Operational Profit in Foreign Currency
- `1F02` Post Operational Income in Functional Currency
- `1Z01` Zero Document

Transaction currency 1 corresponds to the "position currency" in the SDL business transaction.

Transaction currency 2 corresponds to the "transaction currency" in the SDL business transaction.

1114: BB: Position Change Off-Balance-Sheet Financial Instrument

The procedure template `1114` is used to for position changes in financial instruments in settlement date accounting. Examples:

- Pre-valued purchase or sale of securities
- Pre-valued purchase and sale of stocks

The business transactions can also contain items for charges. You can supply these charges as separate business transactions. If you do so, you must also derive these business transactions on the basis of procedure template `1114`.

Calculation Step Categories:

- `1E15` Clear Financial Statement
  Transfer from balance sheet suspense account to the off-balance-sheet anonymous in-transit account
- `1E16` Post Off-Balance-Sheet
  Transfer from off-balance-sheet, anonymous in-transit account to off-balance-sheet, product-specific suspense account
- `1E17` Pro Rata Outflow - Receivables/Payables - Off-Balance-Sheet
  For credit postings to off-balance-sheet assets or debit postings to off-balance-sheet liabilities.
- `1E18` Pro Rata Outflow - Premium/Discount - Off-Balance-Sheet
  Required for amortization in the event of off-balance-sheet position outflows
- `1E19` Inflow - Receivables/Payables - Off-Balance-Sheet
  For debit postings to off-balance-sheet assets or credit postings to off-balance-sheet liabilities
- `1E70` Inflow of Premium/Discount - Implicit Off-Balance-Sheet

As for procedure template 1101, for off-balance-sheet position inflows

- 1E71 Unremitted Earnings
  Creates off-balance-sheet unremitted earnings using the steps 1E17, 1E18 and 1E19
- 1E73 Income/Expense/ Income from Payment - Off-Balance-Sheet
  For charges. Since no pre-valued accrual is performed, the method ME23 is not provided here.

1115: BB: Position Change Off-Balance-Sheet FI (w/o Nominal)

You use procedure template 1115 for the primary external pre-valued business transaction *Cash Settlement for Non-Monetary Equity Instruments in the Banking Book* (business transaction type EQ05, business transaction category 1001). The template's calculation step categories 1E15, 1E16 and 1E79, which call the calculation method ME05, allow you to post the pre-valued business transaction and the inflow or outflow of the relevant off-balance-sheet receivable or payable.

1116: Value Added Tax

Calculation Step Categories:

- 1T01 Inflow/ Outflow Deductible VAT
- 1T03 Inflow/ Outflow Non-Deductible VAT

1117: Change to Non-Accruing (Change of Accrual Status to Non-Accruing)

Step categories to close the accrual phase:

- 1A04 Adjustment of Accrual without Analytical Adjustment
  This step category updates the accrual up to the key date on which the status changes to *Non-Accruing*, before the transfer to the profit and loss account in step category 1E93.
- 1D07 Adjustment of Premium/Discount without Analytical Adjustment
  This step category updates the premium/discount up to the key date on which the status changes to *Non-Accruing*.
- 1E99 Start New Effective View (Amortized Cost = Book Value)
  This step category reacts only to item template 2I37 (Imp.: Adj. Mkt Conf. Status of Contract). When there is a change in the *Market Conformity Status of Contract* and the accrual status changes to *Non-Accruing* on the same day, it is used to set the target values for the effective interest based calculation of deferrals in such a way that the recalculation of the effective interest rate is triggered. The step category 1D09 is used for this recalculation. For more information, see the documentation for calculation method ME48.
- 1D09 Adjustment Effective Interest Rate Deferral
  This step category reacts both to the item template 2I37 and item template 2I34. It is used to recalculate the effective interest rate for the deferral after the recalculation has been triggered in a preceding step. This happens when the accrual status changes to *Non-Accruing* on the same day as a condition change or a change in the *Market Conformity Status of Contract*. For more information, see the documentation for calculation method MD21.
- 1D02 Adjustment Deferrals
- 1E93 Clear Accruals

The calculation step categories mentioned above use impairment characteristic values from the previous day because they close the accrual phase.

The following calculation step categories use current characteristics because they open the non-accruing phase:

- 1A05 Adjustment of Accruals - Change to Accruing

This step category is required if accruals were also made for accrual status *Non-Accruing*. These then need to be posted against an offset account (processing category `1BF_TRANSO`). If the corresponding key figure in processing category `1BF_ACCR` is part of the book value, the associated key figure in processing category `1BF_TRANSO` must also be part of the book value.

- `1EA1` Increase: Offset Short-Term Receivable Acc. to Shrt-Trm Rec.
  This step category is used when the accrual status changes to *Non-Accruing* to offset the short-term receivables against the P&L by increasing a position under *Offset Short-Term Receivable*.

Unless otherwise specified above, these calculation step categories react to item template `2I34` (Impairment: Change to Non-Accruing).

Derive a calculation procedure based on this procedure template for the source field `INPUT_FIELDS FLG_ACCR_ST_CHNG` *Accrual Status Change* = X.

1118: Change to Accruing (Change of Accrual Status to Accruing)

Calculation step categories to close the non-accruing phase:

- `1A01` Adjustment - Accruals
  This step category creates the balance using the processing category `1BF_ACCR`. It is required if accruals were also made for accrual status *Non-Accruing*. These then need to be posted against an offset account (processing category `1BF_TRANSO`).
- `1D08` Adjustment of Premium/Discount - Straight-Line Method
- `1D02` Adjustment Deferrals
  The calculation step categories mentioned above use impairment characteristic values from the previous day because they close the accrual phase.
  The following calculation step categories use current characteristics because they open the non-accruing phase:
- `1E46` Clearing - Offset In-Transit Account (Product-Based)
  This step category clears the balance on the offset account. This is usually posted against an accrual item. This step category is not required if accruals were also made for accrual status *Non-Accruing*. In this case, use step category `1E93`.
- `1E93` Clear Accruals
  This step category clears the balance on accruals. It is cleared against the offset account. The step category is required if accruals were also made for the accrual status *Non-Accruing*.
- `1A05` Adjustment of Accruals - Change to Accruing
  This step category creates the accrual against the P&L. It is required if accruals were also made for accrual status *Non-Accruing*.
- `1E99` Start New Effective View (Amortized Cost = Book Value)
  This step category reacts only to the item template `2I37` (Imp.: Adj. Mkt Conf. Status of Contract). When there is a change in the *Market Conformity Status of Contract* and the accrual status changes to *Accruing* on the same day, it is used to set the target values for the effective interest based calculation of deferrals in such a way that the recalculation of the effective interest rate is triggered. The step category `1D09` is used for this. For more information, see the documentation for calculation method `ME48`.
- `1D09` Adjustment Effective Interest Rate Deferral
  This step category reacts only to the item template `2I37` (Imp.: Adj. Mkt Conf. Status of Contract). It is used to recalculate the effective interest rate for the deferral after the recalculation has been triggered in a preceding step. This happens when a change to *Accruing* and a change in the *Market Conformity Status of Contract* occur on the same day. For more information, see the documentation for calculation method `MD21`.

If steps based on the step categories 1E99 and 1D09 are executed when there is a change in the *Market Conformity Status of Contract*, then steps based on the subsequent step categories 1E97 and 1D11 no longer lead to changes. This is because the prerequisite for the step category 1E97 no longer exists due to the updated effective interest rate, and a catch-up adjustment afterwards is "0".

- 1E97 Start New Effective View (Switch to Accruing)
  This step category is required if there is no effective interest for a catch-up adjustment.
  In this case the system sets the amortization costs in the calculation bases to **Amortization costs = Outstanding Nominal Value /Outstanding Capital Amount**. In this case the system does not calculate a catch-up adjustment.
- 1D11 Adjustment - Premium/Discount
  In this step category, the system calculates a catch-up adjustment if there is an effective interest rate.

Unless otherwise specified above, these calculation step categories react to item template 2I36.

Derive a calculation procedure based on this procedure template for the source field INPUT_FIELDS FLG_ACCR_ST_CHNG *Accrual Status Change* = X.

1140: Position Transfer - Value-Date-Based, Debit-Side

Template 1140 is used for complete or pro rata position transfers for financial instruments from a source position to a target position. This changes a defining characteristic of the financial position.

Examples:

- Securities account transfer
- Stock split

The business transaction must contain one item for the source position and one item for the target position. You have to fill the defining characteristics for the positions accordingly, and set the characteristic /BA1/C55FPSTA (role in sender-receiver relationship) to S for sender or R for receiver. Enter only the nominal amount or quantity. The position amount and transaction amount must remain empty.

The system does not support position transfer postings between positions from different accounting systems or different holding categories. The accrual and valuation methods for source and target position must be the same. However, the system does not check this.

The template is structured as follows:

Source position:

- Calculation step categories to update the relevant value-date-based processing categories/key figures
  - 1A01 Adjustment - Accruals
    Relevant for accruals so that these are up-to-date for step category 1E24.
  - 1D01 Adjustment - Premium/Discount
    Relevant for amortization so that premium/discount and 2D calculation base data for step categories 1E21 and 1E22 are up-to-date.
  - 1D02 Adjustment Deferrals
    Relevant for deferrals so that these are up-to-date for step category 1E23.
- Calculation step categories to clear the relevant value-date-based processing categories/key figures on a pro rata basis
  - 1E20 Pro Rata Outflow - Receivables/Payables - Source Position
  - 1E80 Pro Rata Outflow - Receivables/Payables - Off-Balance-Sheet - Source
    Relevant for settlement date accounting

- 1E21 Pro Rata Outflow - 2-D Calculation Base Data - Source Pos.
- 1E22 Pro Rata Outflow - Premium/Discount - Source Position
- 1E81 Pro Rata Outflow: Premium/Discount OBS Source Position
  Relevant for settlement date accounting
- 1E23 Pro Rata Outflow - Deferrals - Source
- 1E24 Pro Rata Outflow - Accruals - Source
- 1E27 Pro Rata Outflow - In-Transit Account (Product) - Source
- 1E59 Pro Rata Outflow - In-Transit Account (Product) - Source
- 1E82 Pro Rata Outflow: In-Transit Account OBS Source Position
  Relevant for settlement date accounting
- 1F19 Transfer Foreign Currency Position (Anonymous/Prod.-Based)
  Relevant for postings from step category 1E20 with a transaction currency that is not equal to the functional currency

Further calculation step categories, for example for new processing categories, can be added in the same way.

1141: Position Transfer - Value-Date-Based, Credit-Side

This procedure template acts in the same way as procedure template 1140. The difference is that the step categories for the source position (sender) react to the item template 1I12 and the step categories for the target position (receiver) react to the item template 1I13 .

1150: Reclassification (value-date-based)

This procedure template updates the value-date-dependent P&L key figures before a reclassification or before an event-based update of the risk provision. In the latter case, you derive a calculation procedure based on this procedure template for the source field INPUT_FIELDS FLG_ACCR_ST_CHNG *Accrual Status Change* = __ (value empty).

The template is structured as follows:

- 1A01 Adjustment - Accruals
- 1D01 Adjustment - Premium/Discount
- 1D02 Adjustment Deferrals

For a description of the step categories, see procedure template 1105.

The calculation step categories mentioned above use characteristic values from the previous day. The following calculation step categories use current characteristics:

- 1E99 Start New Effective View (Amortized Cost = Book Value)
  This step category reacts only to the item template 2I37 (Imp.: Adj. Mkt Conf. Status of Contract). When there is a change in the *Market Conformity Status of Contract*, it is used to set the target values for the effective interest based calculation of deferrals in such a way that the recalculation of the effective interest rate is triggered. The step category 1D09 is used for this. For more information, see the documentation for calculation method ME48.
- 1D09 Adjustment Effective Interest Rate Deferral
  This step category reacts only to the item template 2I37 (Imp.: Adj. Mkt Conf. Status of Contract). It is used to recalculate the effective interest rate for the deferral after the recalculation has been triggered in the preceding step. For more information, see the documentation for calculation method MD21.

> **Note**
>
> You can use this procedure template for structured products only if there are no postings at overall contract level. You then have to derive a component transfer procedure on the basis of procedure template `1401` to apply the procedure template at subcontract level.

1201: Accrual/Deferral

Template `1201` is used in key date valuation for accruals/deferrals.

Calculation Step Categories:

- `1A01` Adjustment - Accruals
  Relevant for accruals
  `1D01` Adjustment - Premium/Discount
  The assigned methods `MD09` and `ME17` do not play a role in this context.
  If you want to calculate accruals/deferrals based on the effective interest rate even when no premium/discount exists, use method `MD01` instead of method `MD07`.
- `1D02` Adjustment Deferrals
  Relevant for deferrals

1290: Initialization - Value-Date-Based Calculation Base

You use this procedure template during key date valuation for the initialization required to set target values for processing operational (value-date-based) key figures.

For more information, see Initial Balance Transfer for External Accounting [page 718] (case 1, step 2).

1295: Re-Initialization - Value Date-Based

For more information, see SAP Note 1593272.

1299: Empty Procedure Without Zero Documents

You use this procedure template to stop the generation of documents - including zero documents - during key date valuation.

Example: You are using only one key date valuation type for which you want only one two-dimensional calculation (accruals/deferrals) for specific products and both one-dimensional and two-dimensional calculations for other products. By deriving this empty procedure, you ensure that no documents (not even zero documents) are generated for the products mentioned first in the one-dimensional calculation.

For more information, see SAP Note 2396025.

1401: Derivative (OTC): Component Transfer Procedure

You use the procedure template `1401` for the component transfer when posting the primary external business transactions *Conclusion* and *Maturity* for structured OTC derivatives. The template contains only the calculation step category `1C01`, to which the calculation method `MC01` is assigned. This calculation method transfers the transaction amount from the external business transaction to the component of the structured product, and thereby derives postings to the individual components.

1402: Structured Financial Instrument: Component Transfer Procedure

Procedure template 1402 is used in the Component Transfer Procedure [page 855] for bonds with split derivative. Examples:

- Issue
- Repayment
- Purchase
- Sale
- Buyback
- Warrant Bonds

The business transactions can also contain items for charges. You can supply these charges as separate business transactions at the contract level. If you do so, you must also derive these business transactions on the basis of procedure template 1402.

Calculation Step Categories:

- 1C02 Clear Contract
  Transfer from product-specific suspense account to anonymous suspense account at contract level
- 1C03 Post Subcontract
  Distribution from anonymous suspense account at contract level to the product-specific suspense accounts of the subcontracts for the position change.
  - The transaction amount is distributed between the residual and non-residual portion based on fair value.
  - The nominal value or the quantity is transferred from the external business transaction to the main component. The nominal value or quantity for the split derivative is determined from the Source Data Layer.
- 1C04 Transfer to Main Part
  Transfer from anonymous suspense account at contract level to the product-specific suspense account of the main component for charges

1403: Component Transfer Procedure: Transfer FI (Struc.)

Procedure template 1403 is used in the Component Transfer Procedure [page 855] for the complete or pro rata position transfer posting from a source to a target position for structured products. For the entries in the business transaction, see procedure template 1140.

Securities Account Transfer: Warrant Bond Position is an example of this.

Calculation step category:

1C05 Transfer to Subcontracts/Subcontract Position Transfer

Transfer from product-specific suspense account at contract level to product-specific suspense account at subcontract level.

- Only the main part is included when you use method MC03.
- When you use method MC04, the nominal value or quantity for the split derivative is also determined from the Source Data Layer.

1404: Component Transfer Procedure: Multicurrency Bond

You use procedure template 1404 to post the legs of the multicurrency bonds that are maintained in accounting as a structured product (business transaction type DI01, business transaction category 1001).

Calculation Step Categories:

- 1C02 Clear Contract

- 1C06 Distribute to Bonds (Multicurrency Bonds)
- 1F27 Increase in Currency Position - Bond Residual
- 1F28 Increase in Equivalent Value Position - Multicurrency Bond
- 1F31 Increase in Equivalent Value Position - Multicurrency Bond
- 1F29 Increase in Equivalent Value Position - Multicurrency Bond Residual
- 1F30 Increase in Currency Position - Multicurrency Bond (Coupon)
- 1C12 Distribute to Coupon (Multicurrency Bonds)
- 1C07 Distribute to Interest Payment (Multicurrency Bonds)

1500: Procedure Template for Different Payment Currency

The procedure template 1500 can be used for all position-relevant business transaction categories.

Calculation Step Categories:

- 1F13 Entry of currency position if amount in contract currency ≠ functional currency
- 1F14 Entry of currency position if amount in contract currency = functional currency
- 1F15 Entry of equivalent value position if amount in transaction currency ≠ functional currency
- 1F16 Entry of equivalent value position if amount in transaction currency = functional currency
- 1F17 Entry of equivalent value position if amount in transaction currency ≠ functional currency and amount in contract currency ≠ functional currency

4.4.5.2.6.3.2 Procedure Templates (Not Value-Date-Based)

Use

The calculation procedure templates that are not based on value date are described below.

You can analyze the procedure templates in detail in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings*:

- *Calculation Procedures* > *Display General Overview of All Templates*
- *Calculation Control* > *Edit Procedure Templates*

Features

2101: Unscheduled Position Change (Not Dependent on Value Date)

This template is used for creating postings that are not value-date-based and are due to unscheduled position changes in financial transactions. Examples of unscheduled position change in financial transactions:

- Disbursements and unscheduled repayments of loans
- Early termination of a derivative (OTC)

The template is structured as follows:

- Calculation step categories to update the relevant processing categories/key figures that are not based on value date
  - `2F20` FX Valuation - Analytical Currency Position (FPO)
    Relevant for fair value OCI valuated positions in foreign currency (transaction currency not equal to functional currency, so that the reserve for step category `2E24` is up-to-date.
  - `2F21` FX Valuation - Analytical Currency Gain (FPO)
    Relevant for fair value OCI valuated positions in foreign currency so that the reserve for step category `2E24` is up-to-date.
  - `2D01` Adjustment - Valuation - To be Deferred
    This step category is relevant only for the release of valuation remnants using the straight-line method. If you want to release valuation remnants using the effective interest rate method (with contractual cash flow or expected cash flow), this step is not relevant.
  - Step categories for updating and adjusting to the position change of target values that are not value-date-based
    - `2D08` Adjust Calculation Base Data for Valuation Remnants After Contractual Cash Flow Changes
    - `2D09` Update EIR for Valuation Remnants for Position Change
    - `2V09` Adjust Target Values of Hedge Adj. for Unscheduled Outflow
      Relevant for hedged positions.
    - `2V05` Adjustment - Target Values - Fair Value for Unsched. Outflow
    - `2V35` Adjustment of Fair Value for Free Line

    The step categories for updating and adjusting target values store both outflows and inflows for changes. An outflow is a credit posting to a liability or a debit posting to an asset.
  - `2D02` Adjustment - Reserve - To Be Deferred
    Relevant after reclassification from a fair value OCI valuated position so that the reserve for step category `2E21` is up-to-date.
- Calculation step categories to dissolve the relevant processing categories/key figures that are not based on value date on a pro rata basis
  - `2E20` Pro Rata Clearing - Valuation to Be Cleared
    Here there is a balance following a previous change to the valuation approach (hedge dissolution, reclassification from a fair value-valued position) or as a result of transferring subledger accounts.
    Use the method `ME11` for this step category. The system also permits use of the method <sap-technical-name>ME11</sap-technical-name>. However, this is only intended to be used in procedure template `2102`.
  - `2E23` Pro Rata Clearing - Fair Value Adjustment
    Here there is a balance for the fair value calculation.
    Relevant for fair value OCI valuated positions in foreign currency so that the reserve for currency is consistent.
  - `2E25` Pro Rata Transfer of Fair Value Adjustment for Free Line
    When the free line is reduced due to a disbursement, you use this step category to transfer the fair value adjustment for the free line to valuation remnants.
  - `2E21` Pro Rata Clearing - Reserve to Be Cleared
    Here there is a balance following a previous change to the valuation approach (reclassification from a fair value OCI-valued position).
  - `2E24` Pro Rata Clearing - Reserve (Analytical)
    Here there is a balance for the fair value OCI-valued positions.

All the step categories for pro rata clearing result in posting documents only for outflows. An outflow is a credit posting to a liability or a debit posting to an asset. As a result of these step categories, the remaining valuation and the reserves are dissolved according to the position outflow.

2102: Position Change Financial Instrument (Not Dep. on Val. Date)

This template is used for creating postings that are not value-date-based and are due to position changes in financial instruments.

Example of position changes in financial instruments:

- Purchase or sale of securities
- Repayment of securities
- Purchase or sale of derivatives
- Change to acquisition value without change to nominal or quantity (for example, pool factors in asset-backed securities, cash settlements for equity instruments or repayment transactions for securities in which only the acquisition value but not the nominal value is to be changed)

The template is structured as follows:

- Calculation step categories to update the relevant processing categories/key figures that are not based on value date
  - `1F20` FX Valuation: Operational Currency Position (Reserve)
    Relevant for fair value OCI valuated financial instruments in foreign currency so that the operational reserve for step category `1E25` is up-to-date.
  - `1F21` FX Valuation: Operational Currency Gain (Reserve)
    Relevant for AfS positions in foreign currency so that the operational reserve for step category `1E25` is up-to-date.
  - `2F20` FX Valuation - Analytical Currency Position (FPO)
    Relevant for fair value OCI valuated positions in foreign currency so that the reserve for step category `2E24` is up-to-date.
  - `2F21` FX Valuation - Analytical Currency Gain (FPO)
    Relevant for fair value OCI valuated positions in foreign currency so that the reserve for step category `2E24` is up-to-date.
  - `2D01` Adjustment - Valuation - To be Deferred
    This step category is relevant only for the release of valuation remnants using the straight-line method. If you want to release valuation remnants using the effective interest rate method (with contractual cash flow or expected cash flow), this step is not relevant.
  - Calculation step categories for updating and adjusting (on a pro rata basis) the position change of calculation base data that is not value-date-based
    - `2D10` Adjustment of Calc. Base Data Valuation Remnants, Pro Rata
    - `2V10` Adjustment - Calculation Base Data - Hedge Adj. Pro Rata
      Relevant for hedged positions.
    - `2V42` Adjustment of CBD for Credit Risk (Pro Rata)
    - `2V06` Adjustment - Calculation Base Data - Pro Rata Fair Value
      Relevant for fair value-valuated positions if FI spreads are used. Otherwise, the next key date valuation would be restarted for the incorrect FI spread.

    All the step categories for updating and adjusting calculation base data (on a pro rata basis) store both outflows and inflows for changes. An outflow is a credit posting to a liability or a debit posting to an asset.
  - `2D02` Adjustment - Reserve - To Be Deferred

Relevant for former AfS positions after reclassification so that the reserve to be dissolved for step category `2E21` is up-to-date.

- `2D07` Adjustment Deferred Taxes to be Deferred
  Ensures that the deferred tax to be dissolved for the step category `2E22` is up-to-date.

- Calculation step categories to dissolve the relevant processing categories/key figures that are not based on value date on a pro rata basis
  - `2E20` Pro Rata Clearing - Valuation to Be Cleared
    Here there is a balance after a hedge has been dissolved, for former fair value-valuated positions after reclassification or when subledger accounts are transferred.
    The calculation methods `ME11` and `ME44` are available for this step category:
    A non-value-date-based calculation procedure is usually executed on the posting date. In this case, you use the method `ME11`.
    A non-value-date-based calculation procedure can also be executed on the value date. It can be used, for example, to reduce the valuation remnants proportionally during settlement date accounting for position-reducing business transactions with a value date in the future on the settlement date (value date). In this case, you use the method `ME44`.
    For more information, see Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Assign Derived Business Transaction Type to Primary Business Trans. Type*.
  - `2E23` Pro Rata Clearing - Fair Value Adjustment
    Here there is a balance for the fair value calculation.
  - `2E21` Pro Rata Clearing - Reserve to Be Cleared
    Here there is a balance for former fair value OCI-valued positions after reclassification.
  - `1E25` Pro Rata Clearing of Operational Reserve
    Relevant for fair value OCI-valuated stocks (non-monetary transaction).
  - `2E24` Pro Rata Clearing - Reserve (Analytical)
    Here there is a balance for the fair value OCI-valued positions.
  - `2E22` Pro Rata Clearing - Deferred Taxes to Be Cleared
    There is a balance here if deferred taxes have been cleared after a hedge dissolution or reclassification.

  All the step categories for pro rata clearing result in posting documents only for outflows. An outflow is a credit posting to a liability or a debit posting to an asset. As a result of these step categories, there is nothing left over after a position has been reduced to zero.

2103: Derivative (OTC): Settlement

You use procedure template `2103` for the primary external business transaction *Conclusion of OTC Derivatives* combined with the component transfer procedure (business transaction type `DO01`, business transaction category `1001`). Calculation step category `2V03` allows you to post an initial fair value adjustment. The calculation method `MV02` (Adjustment - Fair Value) or `MV03` (Adjustment - Fair Value (Main Currency)) are provided for this purpose. If you use the first of these methods, the calculation step category `2E01` allows you to enter an implicit fee that is posted to the main component in structured products. If you use the calculation method `MV03` to post the initial fair value adjustment, you must not enter an implicit fee using the calculation step category `2E01`.

2104: Move & Transform Result

This template is used to transfer profits and losses in foreign currency.

It contains the calculation step categories 2F01 and 2F24 for clearing in foreign currency, which call the calculation method ME07, and the calculation step category 2F02 for posting in functional currency, which calls calculation method ME02.

2106: Impairment: Impairment Decision (Obsolete)

This procedure template is old and should no longer be used.

2107: Impairment: Reversal of Impairment Decision (Obsolete)

This procedure template is old and should no longer be used.

2111: Reclassification - Not Value-Date-Based

In the non-value-date-based procedure for reclassification, a position update first takes place, during which the system redetermines various valuation components.

- 2D05 Adjustment: Valuation to be Deferred (Reclassification)
- 2V40 Adjustment: Valuation Remnants (Expected Cash Flow, Reclassification)
- 2D06 Adjustment: Reserve to be Deferred (Reclassification)
- 2V07 Adjustment - Hedge Adjustment (Reclassification)
- 2V24 Adjustment Risk Provision Time Effect (Old View) (Reclass.)
  With the adjustment of unwinding in step category 2V40, this step has become obsolete.
- 2V25 Adjustment Credit Risk (Old View) (Reclassification)
- 2V04 Adjustment - Fair Value (Reclassification)
- 2V31 Adjustment to the Lower of Cost and Market Write-Down (Reclassification)
- 2V32 Adjustment to the Lower of Cost and Market Write-Up (Reclassification)
- For fair value adjustment before reclassification, you use either step category 2V04 *Adjustment Fair Value (Reclassification)* or you use a combination of the calculation step categories 2V31 *Adjustment to the Lower of Cost and Market: Write-Down (Reclassification)* and 2V32 *Adjustment to the Lower of Cost and Market: Write-Up (Reclassification)*.
- 2V36 Adjustment of Fair Value for Free Line (Reclassification)

Afterwards, you can transfer postings when you change the valuation method:

- 2E10 Transfer of Fair Value Adjustment to Valuation to be Deferred
- 2E11 Transfer of Reserve to Reserve to be Deferred
- 2E12 Transfer Deferred Tax on Fair Value Adjustment to Deferred Tax to Be Deferred
- 2E75 Clearing: Valuation to be Deferred
- 2E09 Clear Fair Value Adjustment
- 2E04 Clearing - Loan Loss Reserve

The system then changes the display during the reclassification process; it clears all the relevant processing categories or key figures with the old holding category and old general ledger account and reposts these with the new holding category and new general ledger account. As a last step, the system can update risk provision or fair value adjustments based on the new holding category.

- 2V27 Adjustment Risk Provision (Old View) (After Reclassification)
- For fair value adjustment after reclassification, you use either step category 2V28 *Adjustment Fair Value (After Reclassification)* or you use a combination of the calculation step categories 2V33 *LCM Adjustment: Write-Down (After Reclassification)* and 2V34 *LCM Adjustment: Write-Up (After Reclassification)*.
- 2V37 Adjustment of Fair Value for Free Line (After Reclassific.)

When you use a specific procedure, for which you have to make Customizing settings, you can insert steps based on the step categories. Depending on the reclassification direction (for example, from fair value (P&L) to amortized cost), only a subset of the step categories is relevant – for example, only `2E10` if dealing with a valuation of amortized costs. You can release the valuation remnants that result from the fair value adjustment using the straight-line method (calculation method `ME15`) or the effective interest rate method (calculation method `ME33`).

2131: Hedge Designation

You use this procedure template for the primary internal business transaction *Designation for Single Positions* (business transaction type `HA01`, business transaction category `1101`). The template contains only the calculation step category `2H01`, to which the calculation method `MH01` is assigned. You use can use this calculation method for both subsequent and concurrent designation. It updates the calculation base data (CBD) for specific financial positions for the hedged position without generating a posting document.

Calculation Step 100: Hedge Designation

2132: Hedge Dissolution (obsolete)

We recommend that you use procedure template `2135` - *Hedge: Dissolution* for the dissolution of hedging relationships.

2133: Hedge: Redesignation Time Bucket Objects

You use this procedure template for the primary internal business transaction *Designation for Time Bucket Objects* during the redesignation of a portfolio fair value hedge (business transaction type `HA01`, business transaction category `1101`). It contains the calculation step categories `2H06` and `2H07`, which call the calculation methods `MH06` or `MH07`. Since a shift in the maturity band during a redesignation results in the reassignment of the hedge adjustment being dissolved between the time bucket objects, if required these calculation methods transfer the postings for the hedge adjustment between the time bucket objects.

2134: Hedge: Reclassification

You use this procedure template for reclassifying hedging instruments.

For more information, see Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Hedge Accounting* > *Reclassification of Hedging Instruments*.

2135: Hedge: Dissolution

You use this procedure template for the following business transactions:

- Primary internal business transaction *Dissolution Due to Management Decision* (business transaction type `HA02`, business transaction category `1102`)
- Primary internal business transaction *Dissolution Due to Ineffectiveness* (business transaction type `HA03`, business transaction category `1106`)
- Primary external business transactions that result in a dissolution

The procedure template contains the calculation step category `2H03`, to which the calculation method `MH09` is assigned. This calculation method resets the financial position-specific calculation base data for the hedged position. In the case of an internal dissolution (position outflow), the calculation method determines which hedged positions are to be dissolved. In the case of a dissolution due to a management decision or ineffectiveness, the system applies the calculation method directly to the hedged position. The calculation step category `2E13` transfers the current balance of the hedge adjustment to the valuation remnants.

If deferred tax for the hedge adjustment has been posted for the hedged position, this must be released when the hedging relationship is dissolved. This occurs with step category 2E05, which posts the deferred tax for the hedge adjustment (2BF_DFTXHA) to the deferred tax to be deferred (2BF_TXDEF).

Optionally, you can include the calculation step categories 2H12 (method MH10) and 2H13 (method MH11) in this procedure template. You use these methods to transfer the balance of a key figure or the balance of all key figures of a processing category from the hedged position to the unhedged position.

2140: Position Transfer - Not Value-Date-Based

This template is used for creating postings that are not value-date-based and are due to (pro rata) position changes in financial instruments.

This template is used for the following business transactions:

- Securities account transfer
- Stock split

This kind of business transaction must contain two items: One item for item template 1I13 for the source position (sender) and one item for item template 1I12 for the target position (receiver). The items in the business transaction must be nominal amounts or quantities only and must not contain any other amounts. In the case of the stock split, you must enter a different quantity for the sender item as for the receiver item.

The template is structured as follows:

Source position

- Calculation step categories to update the relevant processing categories/key figures that are not based on value date
  - 1F20 FX Valuation: Operational Currency Position (Reserve)
    Relevant for fair value OCI valuated financial instruments in foreign currency so that the operational reserve for step category 1E28 is up-to-date.
  - 1F21 FX Valuation: Operational Currency Gain (Reserve)
    Relevant for AfS positions in foreign currency so that the operational reserve for step category 1E28 is up-to-date.
  - 2F20 FX Valuation - Analytical Currency Position (FPO)
    Relevant for non-monetary financial instruments in foreign currency so that the reserve for step category 1E32 is up-to-date.
  - 2F21 FX Valuation - Analytical Currency Gain (FPO)
    Relevant for fair value OCI valuated positions in foreign currency so that the reserve for step category 2E32 is up-to-date.
  - 2D01 Adjustment - Valuation Remnants - To be Released
    Relevant for the release of valuation remnants using the straight-line method
  - 2D10 Adjustment of Calc. Base Data Valuation Remnants, Pro Rata
    Relevant for the release of valuation remnants using the effective interest rate method
  - 2D02 Adjustment - Reserve - To Be Deferred
    The steps 2D01 and 2D02 are only required for hedge adjustments or fair value OCI-valuated positions.
- Calculation step categories for updating and adjusting (on a pro rata basis) to the position change of target values that are not value-date-based
  - 2V42 Adjustment of CBD for Credit Risk (Pro Rata)
  - 2V06 Adjustment - Target Values - Pro Rata Fair Value
    Required for fair value calculation.

- 2V10 Adjust Target Values of Hedge Adj. Pro Rata
  Required for hedge accounting.

- Calculation step categories to dissolve the relevant processing categories/key figures that are not based on value date on a pro rata basis
  - 1E28 Pro Rata Outflow - Operational Reserve - Source
    Relevant for fair value OCI-valuated stocks (non-monetary transaction).
  - 2E30 Pro Rata Outflow - Valuation to Be Deferr. Analytical - Srce
    To release valuation remnants using the effective interest rate method, you use calculation method ME38, otherwise method ME11.
  - 2E31 Pro Rata Outflow - Reserve to Be Deferred Analytical - Srce
    Here there is a balance for former fair value OCI-valued positions after reclassification.
  - 2E32 Pro Rata Outflow - Reserve (Product-Spec.) Analytical - Srce
    Here there is a balance for the fair value OCI-valued positions.
  - 2E33 Pro Rata Outflow - Hedge Adjustment - Source Position
    This step category is only required for hedge adjustment.
  - 2E74 Pro Rata Outflow: Credit Risk from Source Position
  - To adjust the processing category Fair Value Adjustment, you use either 2E34 Pro Rata Outflow - Fair Value Adjustment - Source Position or 2E87 Pro Rata Outflow Source Position (LCM) - Write-Down and 2E88 Pro Rata Outflow Source Position (LCM) - Write-Up.
    This step category is only required for fair value calculation or for valuation at the lower of cost and market.
  - 2E35 Pro Rata Outflow - Foreign Currency Position (Analytical) - Source Position
    Relevant for fair value OCI-valuated financial instruments in foreign currency.
  - 2E36 Pro Rata Outflow - Deferred Tax - Deferrals - Source
  - 2E37 Pro Rata Outflow of Deferred Tax on Hedge Adjustment Source
  - 2E38 Pro Rata Outflow of Deferred Tax on Fair Value Adj. Source
  - 2E39 Pro Rata Outflow of Deferred Tax to Be Deferred from Source
    The step categories relating to deferred tax are only required if deferred tax is posted in this accounting system.

  Further step categories, for example for new processing categories, can be added in the same way.

Target position

- Calculation step categories to update the relevant processing categories/key figures that are not based on value date
  - Corresponds to the step categories for the target position.
    To release valuation remnants using the straight-line method, you use step category 2D03, to release them using the effective interest rate method, you use step category 2D12.
- Step categories for updating and adjusting (on a pro rata basis) to the position change of target values that are not value-date-based
  - Corresponds to the step categories for the target position.
    For the fair value adjustment of the processing category Fair Value Adjustment, you use either 2E44*Transfer Fair Value Adjustment from Source to Target* or 2E89*Transfer from Source to Target Position (LCM): Write-Down* and 2E90*Transfer from Source to Target Position (LCM): Write-Up*.
- Step categories for transferring the source position to the target position for relevant processing categories/key figures that are not value-date-based
  - All the key figures that were reduced in the source position are increased by the same amount in the target position.

2141: Reversal of Pro Rata Position Transfer Not Value-Date-Based

This template is used for creating postings that are not value-date-based and are due to a reversal with a later posting date than (pro rata) position changes in financial instruments.

It is structured in the same way as procedure template 2140. The difference is that the step categories for the source position (sender) react to the item template `1I12` and the step categories for the target position (receiver) react to the item template `1I13` .

2150: Reclassification (Not Based on Value Date)

This template is used for the business transaction "Master Data Change (AEP)" and for the asset/liability transfer method.

The template is structured as follows:

- Calculation step categories for updating value-date-dependent P&L key figures
  For a description of the step categories, see the relevant step categories of the calculation procedure template 2201.
- Calculation step categories for clearing the entire position for a key figure
- Calculation step categories for posting the entire position to a key figure

There is a clearing step category and a posting step category for each product-related processing category, with the exception of interim account and (preliminary) revenues and expenses.

For characteristic changes, the system uses the characteristic values of the previous day for the step categories for clearing, and the characteristics of the current day for the step categories for posting.

> **Note**
>
> You can use this procedure template for structured products only if there are no postings at overall contract level. You then have to derive a component transfer procedure on the basis of procedure template `1401` to apply the procedure template at subcontract level.

2160: Contract Termination

This template is used to clear reserves and valuations when contracts are terminated, and contains the following step categories:

- `2E75` Clearing - Valuation - To Be Deferred
- `2E78` Clearing - Hedge Adjustment
- `2E09` Clear Fair Value Adjustment
- `2E86` Clearing: Valuation Remnants (Expected Cash Flow)
- `2E85` Reverse Risk Provision
- `2E76` Clearing - Reserve - To Be Deferred
- `2E77` Clearing - Reserve (Product-Specific)
- `2E79` Clearing: Write-Down
- `1F07` FX: Clear Operational Unremitted Earnings
- `1F02` FX: Post Operational Income in Functional Currency
- `2F07` Clearing of Analytical Unremitted Earnings in Foreign Crcy
- `2F02` FX: Post Analytical Income in Functional Currency

2170: Update of Impairment

This template is used for elementary calculation procedures to update risk provision. The template is structured as follows:

- Calculation step categories for completing the old view
  - 2D01 Adjustment - Valuation - To be Deferred
  - 2V18 Adjustment - Unwinding (Old View)
    With the adjustment of unwinding in step category 2V39, this step has become obsolete.
  - 2V39 Adjustment of Valuation Remnants (Expected Cash Flow)
  - 2V19 Adjustment - Risk Provision (Old View)
- Calculation step category for clearing the valuation remnants (expected cash flow)
  - 2E86 Clearing: Valuation Remnants (Expected Cash Flow)
- Calculation step category for clearing the risk provision
  - 2E85 Reverse Risk Provision
- Calculation step categories for adjusting the current view
  - 2V22 Adjustment of Write-Down
  - 2V38 One-Time Fair Value Adjustment (Valuation Remnants)
    This step category reacts only to the item template 2I37 (Imp.: Adj. Mkt Conf. Status of Contract). For more information, see the documentation for calculation method MV37.
  - 2D13 Adjustment of Effective Interest Rate for Valuation Remnants
    This step category reacts both to the item template 2I37 and item template 2I34. It is used to recalculate the effective interest rate for the valuation remnants after the recalculation has been triggered in a preceding step. For more information, see the documentation for calculation method MD21.
  - 2E91 Start New Effective View (Amortized Cost VR = Book Value)
    This step category reacts only to the item template 2I36 (Impairment: Change to Accruing). It is required if there is no effective interest for a catch-up adjustment. For more information, see the documentation for calculation method ME49. In this case the system does not calculate a catch-up adjustment.
  - 2D14 Adj. CBD Amortization Incl. Val. Remnants (-> Non-Accruing)
    This step category reacts only to the item template 2I36 (Impairment: Change to Accruing). The system changes target values for a catch-up adjustment if there is an effective interest rate.
  - 2V17 Adjustment - Risk Provision (Current View)

You can derive different calculation procedures based on this procedure template depending on the value of the source field INPUT_FIELDS FLG_ACCR_ST_CHNG *Change in Accrual Status*.

2171: Update of Impairment (Collateral)

This template is used for elementary calculation procedures to update the write-down to the collateral value. The template is structured as follows:

- Calculation step categories for completing the old view
  - 2D01 Adjustment - Valuation - To be Deferred
  - 2V39 Adjustment of Valuation Remnants (Expected Cash Flow)
  - 2V19 Adjustment - Risk Provision (Old View)
- Calculation step category for clearing the risk provision
  - 2E04 Clearing - Loan Loss Reserve
- Calculation step category for clearing the write-down
  2E79 Clearing: Write-Down
- Calculation step category for calculating the collateral write-down to the fair value of the collateral
  2V41 Adjustment of Fair Value (Collateral)

2201: Valuation

This template is used for valuation procedures for the key date valuation. The template is structured as follows:

- 2V22 Adjustment of Write-Down
- 2D01 Adjustment - Valuation - To be Deferred
- 2V39 Adjustment of Valuation Remnants (Expected Cash Flow)
- 2V01 Adjustment - Hedge Adjustment [page 895]
- 2V15 Hedge: Hypo Correction
- 2V16 Adjustment - Unwinding (Current View)
  With the adjustment of unwinding in step category 2V39, this step has become obsolete.
- 2V17 Adjustment - Credit Risk (Current View)
- Either 2V02 Adjustment Fair Value [page 896] or the combination of calculation step categories 2V29*Adjustment to the Lower of Cost and Market: Write-Down* and 2V30*Adjustment to the Lower of Cost and Market: Write-Up*.
- 2V35 Adjustment of Fair Value for Free Line
- 2V41 Adjustment of Fair Value (Collateral)
- 2D02 Adjustment - Reserve - To Be Deferred
  For example, for AfS positions.
- 2D07 Adjustment Deferred Taxes to be Deferred
  For the calculation of deferred tax.

Calculation step categories for calculating deferred tax

- 2T01 Adjustment of Deferred Tax on Deferrals
- 2T02 Adjustment of Deferred Tax on Hedge Adjustment
- 2T03 Adjustment of Deferred Tax on Fair Value Adjustment
  See also: Deferred Taxes [page 817]

2202: Valuation - Time Bucket Objects

The procedure template 2202 is used for the valuation of time bucket objects in a portfolio fair value hedge.

It contains the calculation step category 2H05, to which the calculation method MH05 is assigned. This method is used to reduce the hedge adjustment being dissolved if there are unplanned position outflows, for example. In this case the hedged volume assigned to the time bucket is also reduced. The calculation step category 2V08, assigned the calculation method MV08, determines the hedge adjustment for the time bucket objects. In the case of time bucket objects of portfolio fair value hedges that are part of an AfS portfolio, the calculation method MH08, which is assigned to the calculation step category 2H08, adjusts the fair value to the equity.

2211: FX Valuation - Currency Positions

Template 2211 is used to determine the currency gain.

2212: FX Valuation - Balance Sheet Items

Template 2212 is used to update the balance sheet items in functional currency. This type of valuation is also called “monetary assets revaluation” (MAR).

2213: Restatement - Balance Sheet Positions

Template 2213 is used to update additional currencies (such as local, group, hard, or index-based currency) in balance sheet items.

2214: Transfer Product Financial Position

Template `2214` is used for transferring the balance of subledger accounts for product financial positions, for example, for transferring subledger accounts for currency position (see also Transfer of Currency Positions [page 950]) and for the valuation components risk provision, hedge adjustment, and fair value adjustment.

2215: Transfer General Financial Position

Template `2215` is used to transfer the balance of subledger accounts for general financial positions, for example, for subledger accounts for the Transfer of Currency Positions [page 950].

2290: Initialization - Not Value-Date-Based Calculation Base

You use this procedure template during key date valuation for the initialization required to set target values for processing analytical (non-value-date-based) key figures.

For more information, see Initial Balance Transfer for External Accounting [page 718] (case 2, step 2).

2295: Reinitialization - Not Value Date-Based

For more information, see SAP Note 1593272.

2299: Empty Procedure Without Zero Documents

You use this procedure template to stop the generation of documents - including zero documents - during key date valuation.

Example: You are using only one key date valuation type for which you want only one two-dimensional calculation (accruals/deferrals) for specific products and both one-dimensional and two-dimensional calculations for other products. By deriving this empty procedure, you ensure that no documents (not even zero documents) are generated for the products mentioned first in the one-dimensional calculation.

For more information, see SAP Note 2396025.

2300: Initialization of Impairment

This procedure template is used for the initialization of impairment-specific characteristics and accounting key figures. The template is structured as follows:

- `2E86` Clearing: Valuation Remnants (Expected Cash Flow)
- `2E85` Reverse Risk Provision
- `2V22` Adjustment of Write-Down
- `2V16` Adjustment - Risk Provision Time Effect (Current View)
  With the adjustment of unwinding in step category `2V39`, this step has become obsolete.
- `2V39` Adjustment of Valuation Remnants (Expected Cash Flow)
- `2V17` Adjustment - Risk Provision (Current View)

2301: Initialization of Balance Sheet Entities

The procedure template 2301 is used to initialize balance sheet entities during the processing of general financial positions objects (GFPO).

It contains the calculation step categories `1I80` to `2I84`, which call the calculation method `ME50`, and are used to clear balances for each key figure from a source GFPO, and the calculation step categories `1I90` to `2I94`, which call the calculation method `MI15`, and are used to post these balances to a target GFPO.

2302: Balance Carryfwd: Product Financial Pos.

Procedure template 2302 is used for balance carryforwards for product-related financial positions.

It contains calculation step categories 1I05 to 1I16 and 2I07 to 2I20, which call calculation method ME50.

2303: Balance Carryforward for Gen. Fin. Pos.

Procedure template 2303 is used for balance carryforwards for general financial positions.

It contains calculation step categories 1I18, 1I20 to 1I23, 2I21, 2I23 and 2I24, which call calculation method ME50.

2601: Move & Transform Result

Template 2601 is used to transfer profit in foreign currency.

2602: Move & Transform Unremitted Earnings FX Template 2602 is used to transfer unremitted earnings in foreign currency.

4204: PA Based on Effective Capital

The procedure template is structured as follows:

- 4V02 Funding Contribution
  - Processing category: 4BF_VALFND
  - Calculation method: NV02 or NV03.
- 4V08 Net Interest Margin
  - Processing category: 4BF_VALFND
  - Calculation method: NV08
- 4V04 Funding Contribution Due to Deviation from Expectation
  - Processing category: 4BF_VALFND
  - Calculation method: NV04
- 4V06 Funding Contribution Due to Funding Loss
  - Processing category: 4BG_VALFND
  - Calculation method: NV06
- 4V10 Standard Costs
  - Processing category: 4PF_PLSTDC
  - Calculation method: NV10

4205: PA Not Based on Effective Capital

The procedure template is structured as follows:

- 4V02 Funding Contribution
  - Processing category: 4BF_VALFND
  - Calculation method: NV02 or NV03
- 4V08 Net Interest Margin
  - Processing category: 4BF_VALFND
  - Calculation method: NV08
- 4V04 Funding Contribution Due to Deviation from Expectation
  - Processing category: 4BF_VALFND
  - Calculation method: NV04
- 4V10 Standard Costs
  - Processing category: 4PF_PLSTDC

- Calculation method: `NV10`

4.4.5.2.6.3.3 Calculation Procedures for Different Payment Currency

Use

You use these calculation procedures for business transactions in which the transaction currency of the business transaction (= payment currency) differs from the position currency. The system always executes these procedures as a substep of the Update Secondary Business Transactions [page 786] process.

You can use the procedure template 1500 – *Calculation Procedures for Different Payment Currency* for calculation procedures when the payment currency differs.

Prerequisites

- This is a position-type business transaction.

  > **Note**
  >
  > Multicurrency bonds cannot be mapped using calculation procedures when the payment currency is different.

- You have defined the amount in position currency as well as the transaction currency in the business transaction.
- The business transaction contains one position-type item with a different payment currency.
- You have assigned a business transaction type to the calculation procedure in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Procedures for Different Payment Currency* > *Edit Procedures for Different Payment Currency*.
- The item type of the item with a different payment currency is assigned to the calculation procedure in the event of a different payment currency. You make this setting in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Procedures for Different Payment Currency* > *Derive Procedures for Different Payment Currency*.

Features

The system posts an external business transaction in the transaction currency defined for the business transaction. After posting a business transaction in which the transaction currency differs from the position

currency, the system derives the calculation procedure for swapping the currencies that has been defined for this case. This calculation procedure automatically transfers the amount in transaction currency to the amount in position currency, and updates the foreign currency positions and the equivalent value position.

The document that the calculation procedure creates is passed on in the substeps that follow. This means that in the subsequent component transfer procedures and elementary derivation procedures the amount in position currency is used as the transaction amount.

The figure below illustrates this process:

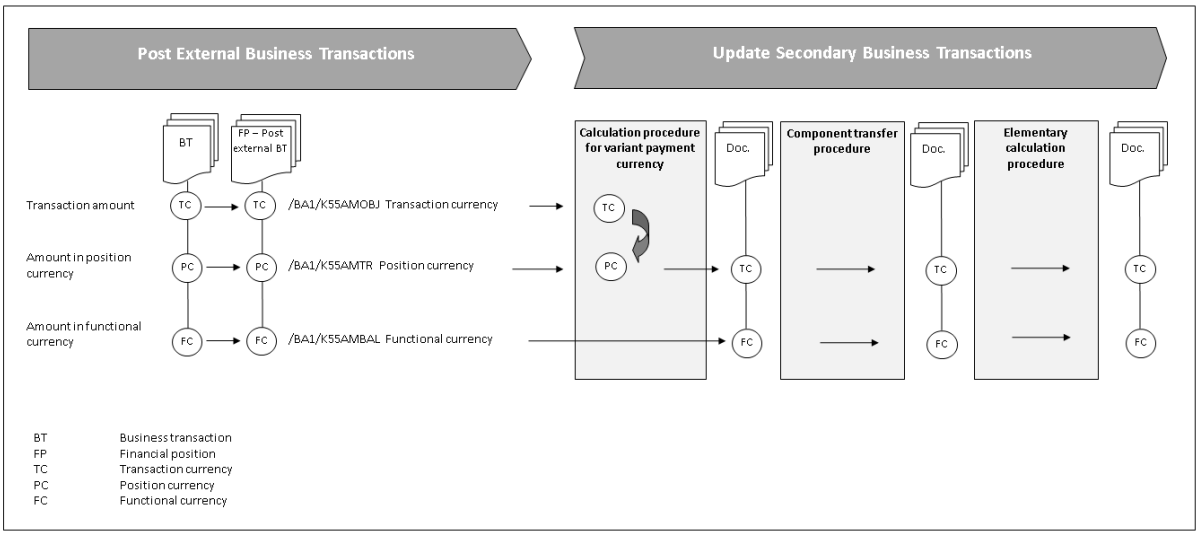


Figure: Processing of Business Transactions with a Different Payment Currency

Example

1. Example
   Functional currency: EUR
   Transaction currency: JPY
   Position currency: USD

	In-Transit Account BG_TRANS	Suspense Account BF_SUSP	Currency Position BG_FX (JPY)	Currency Position BG_FX (USD)	Currency Position - Equivalent Value BG_FXEV (JPY)	Currency Position - Equivalent Value BG_FXEV (USD)
External posting	JPY 1000 (USD 120)	JPY -1000 (USD -120)				
Update currency position JPY		JPY 1000	JPY -1000			

	In-Transit Account BG_TRANS	Suspense Account BF_SUSP	Currency Position BG_FX (JPY)	Currency Position BG_FX (USD)	Currency Position - Equivalent Value BG_FXEV (JPY)	Currency Position - Equivalent Value BG_FXEV (USD)
Update foreign currency position USD		USD -120		USD 120		
Update equivalent value position					EUR 100	EUR -100

Once the procedure for variant payment currency has been executed successfully, the external document is passed on with the transaction amount in US dollars to the component transfer procedure and to the elementary calculation procedures.

2. Example
   Functional currency: EUR
   Transaction currency: EUR
   Position currency: USD

	In-Transit Account BG_TRANS	Suspense Account BF_SUSP	Currency Position BG_FX (USD)	Currency Position - Equivalent Value BG_FXEV (USD)
External posting	EUR 1000 (USD 1200)	EUR -1000 (USD -1200)		
Update foreign currency position USD		USD -1200	USD 1200	
Update equivalent value position		EUR 1000		EUR -1000

Once the procedure for a different payment currency has been executed successfully, the external document is passed on with the transaction amount in US dollars to the component transfer procedure and to the elementary calculation procedures.

3. Example
   Functional currency: EUR
   Transaction currency: USD
   Position currency: EUR

	In-Transit Account BG_TRANS	Suspense Account BF_SUSP	Currency Position BG_FX (USD)	Currency Position - Equivalent Value BG_FXEV (USD)
External posting	USD 1200	USD -1200		
Update foreign currency position USD		USD 1200	USD -1200	
Update equivalent value position		EUR -1000		EUR 1000

Once the procedure for different payment currency has been executed successfully, the external document is passed on with the transaction amount in euros to the component transfer procedure and to the elementary calculation procedures.

4.4.5.2.6.3.4 Component Transfer Procedure

Use

You use component transfer procedures for structured products [page 900]. The system always executes these procedures as a substep of the Update Secondary Business Transactions [page 786] process.

Generally, the financial positions managed in Accounting have the same granularity as the positions in the system in which they are maintained. In certain cases, however, the granularity of a financial position can differ if a certain accounting standard requires a more detailed level of reporting. An example of this is the splitting of embedded derivatives for structured bonds (warrant bonds, convertible bonds) that is required by IFRS.

This greater level of detail means that operative business transactions are supplied with the granularity of the overall instrument but must be split among the individual substructures when they are processed in the subledger.

It is then the task of the financial position processes to distribute the values of the business transaction to the relevant components. The basis for this distribution performed by Calculation Management is the result of the *post external business transactions* process, in which the system posts the transaction amount and/or the nominal value or the quantity to the parent node of the complex financial position.

The component transfer procedure generates postings to the components of a structured product and is triggered by the external posting to the main component of a structured product (processing categories `1BF_SUSP` or `1BG_SUSP`). The component transfer procedure returns *transfer items* as a result (see figure 1). There are one or more mapped transfer items, depending on the posting specifications. The system posts them as part of the derived business transaction. Once the transfer items have been posted, the financial position key figures of the parent node must not contain any values or quantities.

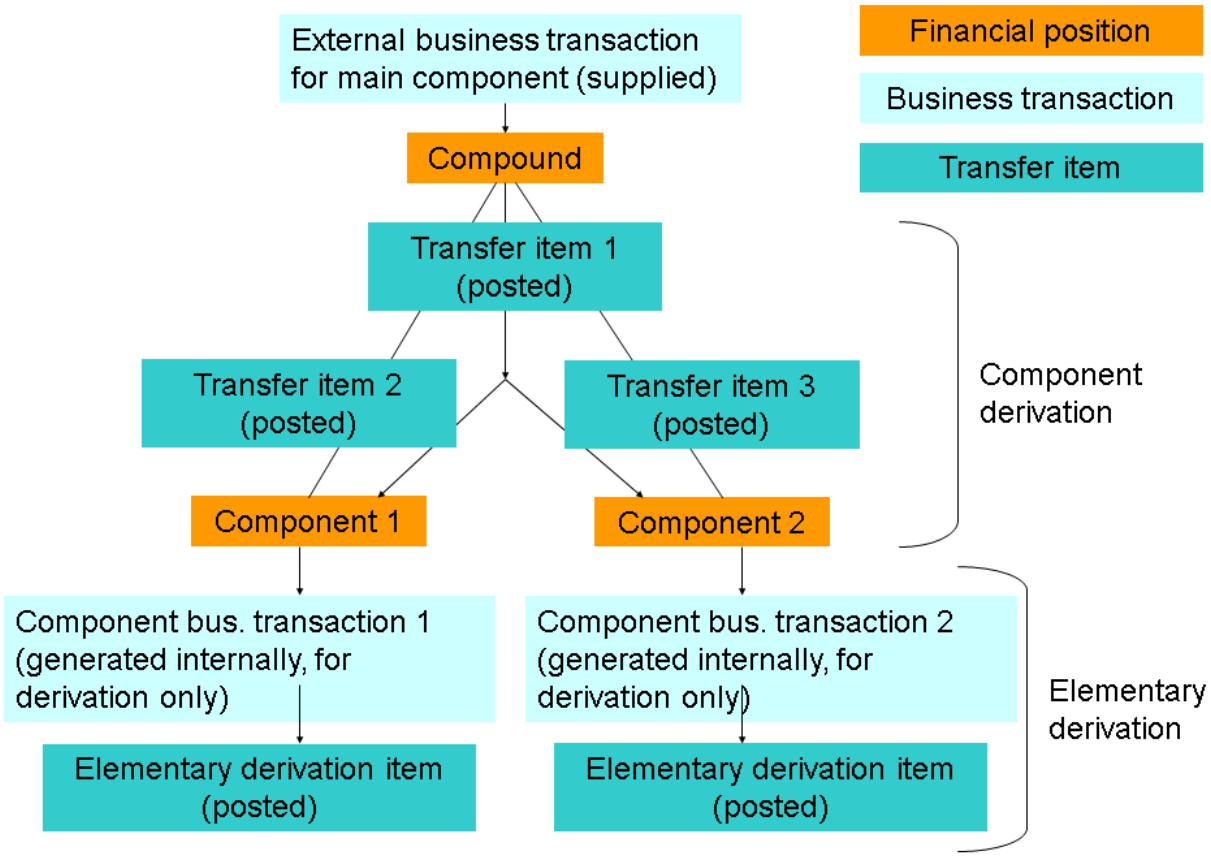


Figure 1: Items Generated by a Component Transfer Procedure

In Customizing for Bank Analyzer, you can assign a component transfer procedure to a business transaction, for example for the purchase or sale of structured products, under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Assign Derived Business Transaction Type to Primary Business Trans. Type*.

Instead of supplying one business transaction for all components, you can supply a business transaction for each component by using the *Parent Template*, *Child Template*, and *Node Number* fields. You must take the distribution of the values to the components and the associated consistency checks into account when supplying the business transactions to the SDL. In some cases (for example, reclassifications where the business transaction does not contain any amounts), you supply the business transactions at subcontract level. Then you do not require a component transfer procedure.

> **Note**
>
> Component transfer procedures are not suitable for initialization postings of complex financial positions. The system has to post each component (using the parent template, child template, and node number fields) directly when a complex financial position is initialized.

Features

The following procedure templates are provided for the component transfer procedure:

- *1401: Derivative (OTC): Component Transfer Procedure*
- *1402: Derivative: Component Transfer Procedure*
- *1403: Component Transfer Procedure: Transfer FI (Struc.)*
- *1404: Component Transfer Procedure: Multicurrency Bond*

For more information, see Value-Date-Based Procedure Templates 1101 to 1404 [page 826].

4.4.5.2.6.3.4.1 Component Transfer Procedure for Structured Products with Split Derivative

Use

The system uses this procedure to process external business transactions for changing quantities and values or the operational profit of a structured product with split derivative. To do so, the system generates an internal business transaction containing items for transfer to the subobjects of the structured product. These items are referred to below as transfer items.

The system generates transfer items to clear the structured product and post or transfer them to the subobjects. In addition, the system executes elementary calculation procedures for the subobjects of the structured product based on the transfer items posted to the subobjects and adds the resulting elementary derivation items to the same internal business transaction.

Features

The following example shows how the system processes a structured product with split derivative for a change in quantity and value.

The example is based on the purchase of a warrant bond. Foreign currency, hedging, and short position aspects are not taken into account. The calculation step for the purchase of a warrant bond splits the transaction amount according to the fair values of the components, whereby the residual part – in this case, the warrant – receives the remaining amount.

In the following example, the fair value for the non-residual component is 8000. There are 10 warrants at a nominal value of 10000.

For an external business transaction for changing the operational profit of a structured product with split derivative, the transaction amount of the relevant business transaction item is transferred to the subobject, which represents the main component of the structured product.

The structure of the two figures is the same as the structure in figure 1 under Component Transfer Procedure [page 855].

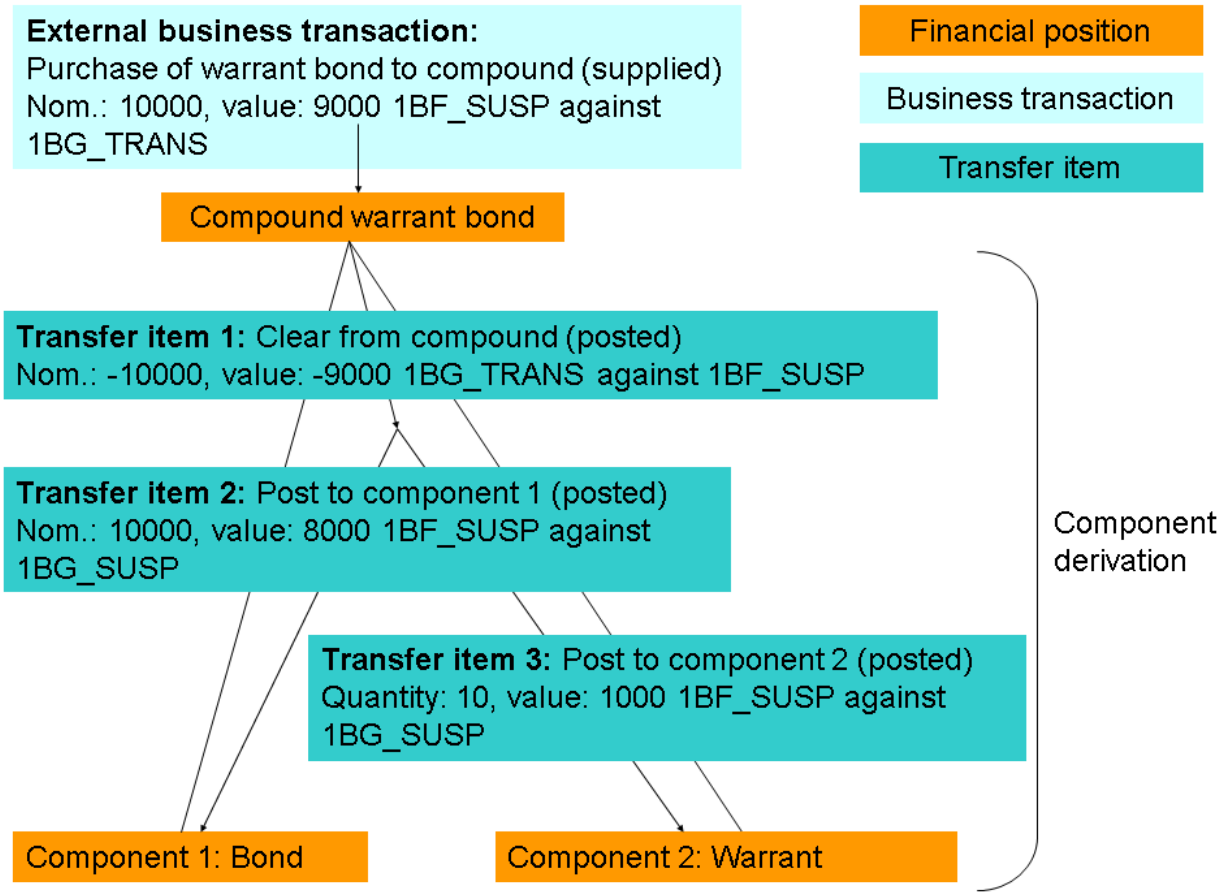


Figure 1: The process flow in the Update Secondary Business Transactions [page 786] transaction for the purchase of a warrant bond (component derivative).

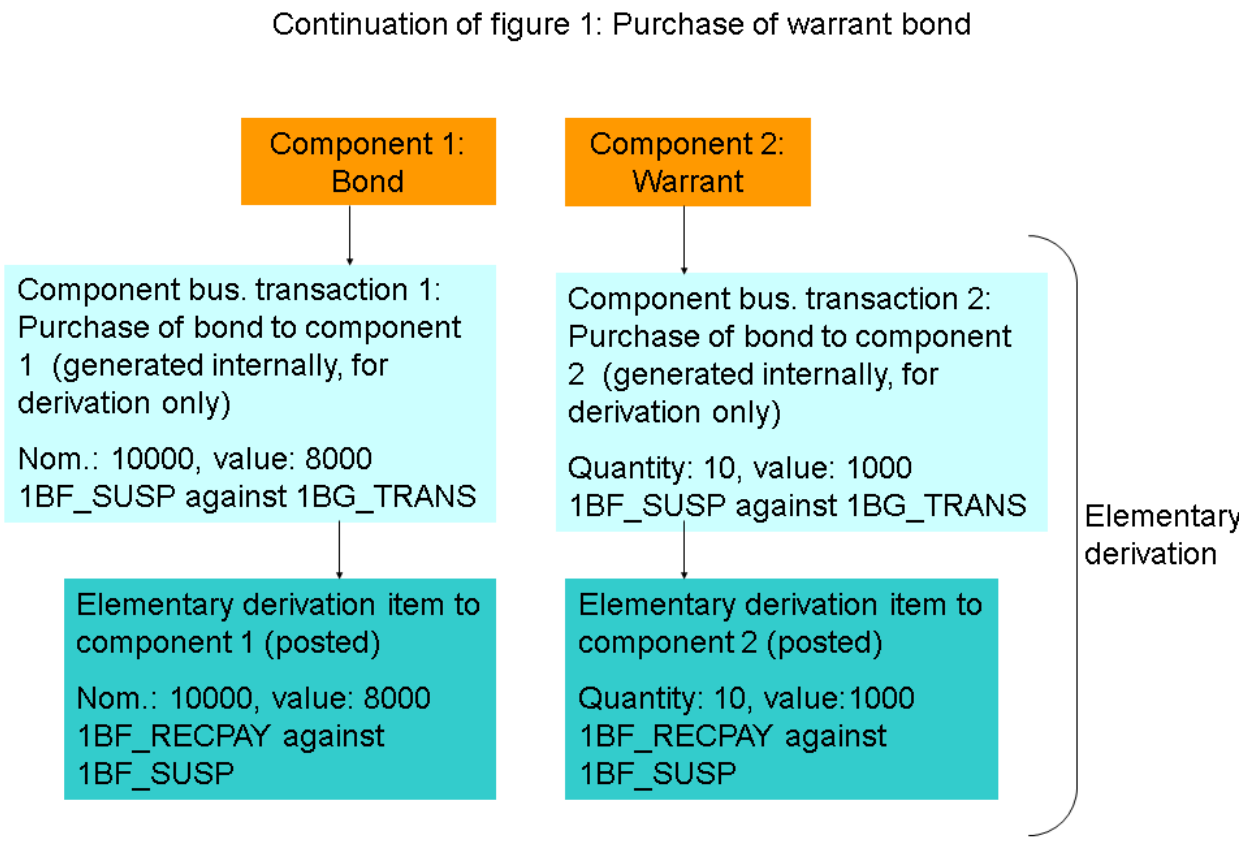


Figure 2: The process flow in Update Secondary Business Transactions for the purchase of a warrant bond (elementary derivation, continuation of figure 1).

Note the special status of the component business transactions in figures 1 and 2. The system generates them internally and derives them during the *Update Secondary Business Transactions* process. It posts all transfer items and elementary derivation items to a single document in this process.

Activities

For a detailed description of the component transfer procedure, see the Customizing activity *Display Calculation Procedures* under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Procedures*.

This activity provides a detailed overview of the calculation procedures and related entities. From here, you can also display an overview of the procedure templates and read the documentation or coding for the individual calculation methods.

You can find the *Display General Overview of All Templates* activity in Customizing for Bank Analyzer under *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal*

Business Transactions > *Basic Settings* > *Calculation Procedures*. This activity provides a detailed overview of the procedure templates and related entities.

You can also view all of the details of the calculation procedure template *1402 Derivative: Component Transfer Procedure* with the relevant steps.

4.4.5.2.6.3.5 Elementary Calculation Procedure

Use

The Update Secondary Business Transactions [page 786] process executes value-date-based and non-value-date-based calculation procedures after the component and distribution transfer procedures.

For example, during a stock purchase (in functional currency) the system executes an elementary value-date-based calculation procedure and a non-value-date-based calculation procedure during the *Update Secondary Business Transactions* process.

When you conclude an OTC derivative, the system executes only a non-value-date-based elementary calculation procedure.

> **Note**
>
> You must assign an elementary calculation procedure (value-date-based or non-value-date-based) to each external business transaction if the corresponding business transaction type has been entered in the Customizing activity *Assign Derived Business Transaction Type to Primary Business Trans. Type* as to be derived in the derivation rule (value-date-based or non-value-date-based).
>
> You find this activity in Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Assign Derived Business Transaction Type to Primary Business Trans. Type*.
>
> If you do not want any derived postings, you must enter *Do Not Derive* in the relevant columns in the Customizing activity *Assign Derived Business Transaction Type to Primary Business Trans. Type*.

Features

Content of Procedures

Elementary calculation procedures can contain a wide variety of calculation steps depending on the product and business transaction to be processed.

For more information, see

- Value-Date-Based Procedure Templates [page 826]
- Procedure Templates Not Based on Value Date [page 839]

4.4.5.2.6.3.5.1 Issue Currency Changeover

Use

In Accounting, processing of a currency conversion is triggered when an external business transaction is posted. This business transaction must contain two items; the first item contains the entire nominal volume of the financial position being cleared in the old currency with the old external number of the financial instrument and the second item contains the entire nominal volume of the financial position being posted in the new currency with the new external number of the financial instrument.

When an external business transaction is posted, a new financial position is created for the new currency and the new external number. The calculation procedure template is based on the template for the financial position transfer posting. In the transfer posting the system transfers all the key figure balances from the source position using the processing category `1BG_TRANS`. The key figure balances in the contract currency of the source position are translated in the transfer posting into the contract currency of the target position. The translation rate is taken from the exchange rate table.

Prerequisites

Specifying the Calculation Steps

In Customizing for *Bank Analyzer* , choose *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Control* > *Edit Calculation Steps* .

We provide the following new **calculation step templates** for the conversion of issue currency.

- `1F32` Issue Currency Changeover: Enter FX Position Not Equal to FC
- `1F33` - Issue Currency Changeover: Enter EV Position ((TC1 Equals FC and TC2 Does Not Equal FC) or (TC1 Does Not Equal FC and TC2 Equals FC))
- `1F34` - Issue Currency Changeover: Enter EV Position (TC1 Does Not Equal FC and TC2

> **Note**
>
> - The issue currency changeover applies to financial instruments only.
> - After the issue currency conversion, business transactions may no longer be provided in the old currency.
> - The system supports same currency products only.

Activities

The procedure for an issue currency conversion can be broken down into the following steps:

1. Change the currency in the SDL.
   Supply the products to the SDL in the new currency. Once you have done this, valuation in the old currency is no longer possible.

2. Posting business transactions
   You can post business transactions to the issue currency conversion only after the currency of the product has been changed. The business transaction for issue currency conversion must contain two items.
   - The first item brings about the clearing of the financial position in the old currency. The first item must contain the entire current nominal amount in the old currency with the old external number of the financial instrument. The item type must have the item template `1I13` (outflow).
   - The second item is for posting the financial position in the new currency. The second item must contain the translated entire nominal amount in the new currency with the new external number of the financial instrument. The item type must have the item template `1I12` (inflow). Neither positions must contain a transaction amount. You must however enter a transaction currency.
3. Derive business transaction
   When you post an external business transaction for the issue currency conversion, the system creates a new financial position. In the case of a structured product, this involves several financial positions. To transfer key figures from an old financial position to a new one, you must perform a derivation for the business transaction. Once the derivation has been performed, the conversion is closed.

4.4.5.2.6.3.6 Transferring Profits and Losses in Foreign Currencies

Use

For each accounting system in the subledger scenario, there is exactly one calculation procedure that you run to transfer profit or loss in a foreign currency. The same applies for management accounting.

The system triggers this procedure automatically at the end of the *Update Secondary Business Transactions* and *Run Key Date Valuation* accounting processes. In the subledger scenario, you can also transfer unrealized profits/losses (both in management and financial accounting) to the foreign currency positions during key date valuation.

Features

Content of Procedures

In accordance with the posting logic of the foreign currency concept, the system transfers profit or loss in a foreign currency to functional currency using two calculation step categories:

Clearing the Net Income in Transaction Currency

The system clears a P&L key figure in the transaction currency against the *Income Statement* currency position or the *Equity* currency position.

Posting the Net Income in Functional Currency

The system posts from the *Equivalent Value* currency position in transaction currency to a P&L key figure in the functional currency.

> **Note**
>
> The system transfers the profit or loss in a foreign currency automatically if a P&L key figure was posted in foreign currency.

Activities

You can create calculation procedures for transferring profit or loss in a foreign currency and assign them to accounting systems for the profit and loss transfer postings. You do this in Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Transfer Postings for Foreign Currency Profit*.

4.4.5.2.6.3.7 Asset/Liability Status

Use

Financial positions can appear on the assets or liabilities side of the balance sheet, depending on whether their A/L book value is positive or negative. The A/L book value of a financial position is the key figure balance from the processing category `1BF_RECPAY` and key figures specified in Customizing.

The system determines the asset/liability status for each accounting system as follows:

A/L book value = key figure balance from the processing category *Receivable* `1BF_RECPAY` plus the values from the key figures entered in Customizing (specified by financial position type, processing category, and key figure).

A financial position on the assets side has a positive A/L book value or an A/L book value of zero; a financial position on the liabilities side has a negative A/L book value.

- A security can represent a receivable or a liability, depending on whether it is long or short.
- Some derivatives, such as swaps, can represent both receivables and liabilities at the same time. You must consider these together in the balance sheet.

Every financial position has a standard SAP **Asset/Liability Status** characteristic (`/BA1/C55ASSLIA`). The value of this characteristic depends on the sign of the A/L book value of a financial position and if required is changed by the A/L transfer. Flows at component level (results of the result node `FLCMP`) also have the fixed characteristic *Asset or Liability Status of a Flow*)`/BA1/C55ALSFL`). This matches the asset/liability status of the financial position after all the accounting processes have been completed at the end of the day.

> **Note**
>
> Assign direct postings to the correct business transaction category 1006 so that the assets and liabilities are transferred without problem.
>
> In the case of direct postings, the system sets the A/L status for the flow using a logic that cannot consider the status at the end of the day because this status is not known. The A/L transfer sets the correct status at the end of the day.

You can use the characteristic `/BA1/C55ALSFL` in the derivation of G/L accounts.

> **Note**
>
> The system creates the characteristic `/BA1/C55ASSLIA` at item level (results of the result type `FLOW`) in the business transaction.

Features

During the life cycle of a financial position, the asset/liability status can change as a result of the following triggers:

- Operational flow transactions
- Updating of accruals/deferrals or a valuation

For this reason, the system determines the asset/liability status after every analytical and operational event. This is illustrated in figure 1, where the A/L book value is specified by the fair value:

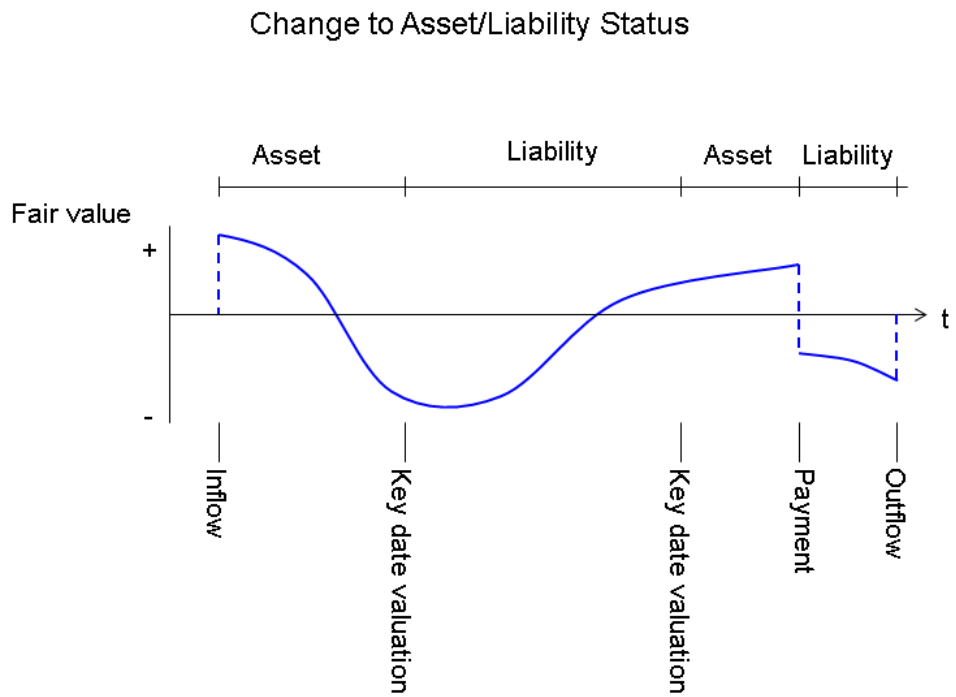


Figure 1: Change to the asset/liability status of a financial position

Explanation: The initial posting changes the asset/liability status from *Undefined* to *Active*. The fair value of the financial position becomes negative. The position still remains an *asset*, however, because no event triggers a change in the liability status. When the key date valuation is carried out, the system updates the A/L book value

with the fair value, and the asset/liability status changes to *liability*. The fair value becomes positive again, which is taken into account by the system during the next key date valuation. A subsequent payment changes the asset/liability status back to *liability*.

Asset/Liability Transfer

When the asset/liability status changes, the system reclassifies the entire position in terms of the balance sheet. This happens in two steps:

1. The system clears all balance-sheet key figures with the characteristic value *Before Change*.
2. The system posts all balance-sheet key figures with the characteristic value *After Change*.

The procedure template 2150 (Reclassification) is the template for the asset/liability transfer.

In the standard system, when secondary business transactions are updated and during key date valuation, the system checks the asset/liability status of the position at the end of each posting date based on the known balances and executes an asset/liability transfer if required.

However, the book value may not be up-to-date at this point in time. For example, in the case of foreign exchange products. The profit or loss in foreign currency may also not be up to date. The only time at which the balances and book value are definitely up to date is after a complete key date valuation with a foreign currency valuation.

You therefore have the option of deactivating the asset/liability determination in the Update Secondary Business Transactions process and in the key date valuation, irrespective of the key date valuation type. However, you should ensure that a key date valuation is run that determines the asset/liability status at least at the end of the period. You make the relevant settings in the Customizing activities *Control Asset/Liability Transfer Method* and *Define Key Date Valuation Type*. For more information, see SAP Note 2012986.

> **Note**
>
> When you use direct postings for positions whose asset/liability status can change, you can only deactivate the asset/liability determination in the Update Secondary Business Transactions process if you have defined that the system generates adjustment business transactions to correct direct postings (see Customizing activity *Assign Derived Business Transaction Type to Primary Business Trans. Type*). For more information, see SAP Note 1942507.

Complex Financial Positions

The system makes the following distinction when it handles the financial positions in the financial position processes:

- Structured products:
  - For products that are structured exclusively for calculation purposes in the financial position processes, the system determines the asset/liability status by the total of the A/L book values for all the financial positions (swaps) rather than by the A/L book value of the individual financial position.
  - For products that comprise groups of financial positions because the groups are reported separately in the balance sheet, the system determines the asset/liability status by the granularity of the individual groups (embedded derivatives). The system determines the asset/liability status of a group by totaling the A/L book value of all financial positions in a group. It then assigns the resulting asset/liability status to each financial position in the group.
- Hedged financial positions
  There is no different between the balance processing of hedged and unhedged positions. The system therefore totals these to determine the asset/liability status. The only relevant book value component for hedged

positions is the Hedge Adjustment component. Note that this is used only if the appropriate Customizing settings have been made for determining the A/L book value.

Customizing Tables

In Customizing for *Bank Analyzer*, choose *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Transfer Methods for Assets and Liabilities*:

- *Control Assets/Liability Transfer Method*
- *Assign Asset/Liability Transfer Methods to Accounting Systems*
- *Define Determination of Asset/Liability Status*

Activities

Customizing Activity: Derive Descriptive Characteristics of Financial Position and Funding Center

You must create a new derivation rule in the Customizing activity *Derive Descriptive Characteristics of Financial Position* to determine the asset/liability status of the financial position group of a complex financial position. You can use any of the document fields as the source field. Use the characteristic `/BA1/C55ALSGRP` as the target field. Positions with the same characteristic value are grouped in one A/L group.

You generate a new derivation rule in Customizing under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Characteristics of Financial Positions* > *Derive Descriptive Characteristics for Fin. Positions + Funding Centers*.

Customizing Activity: Edit Elementary Calculation Procedures (Not Value-Date-Based)

Use the calculation procedure template 2150 (Reclassification) for the transfer of assets and liabilities. You have to use this as a basis for defining a new calculation procedure for asset/liability transfer and then assign it calculation steps. You also have to assign item types to the calculation steps.

You define a new calculation procedure for asset/liability transfer and assign calculation steps in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Elementary Calculation Procedures* > *Elementary Calculation Procedures (Not Dependent on Value Date)* > *Edit Elementary Calculation Procedures (Not Value-Date-Based)*.

Customizing Activity: Assign Asset/Liability Transfer Methods to Accounting Systems

To be able to use your calculation procedure for asset/liability transfer, you have to assign it to an accounting system.

You assign the calculation procedure for asset/liability transfer to an accounting system in Customizing under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Transfer Methods for Assets and Liabilities* > *Assign Asset/Liability Transfer Methods to Accounting Systems*.

Customizing Activity: Define Determination of Asset/Liability Status

You define the relevant key figures in Customizing under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Transfer Methods for Assets and Liabilities*.

4.4.5.2.6.4 Calculation Steps (Subledger Scenario)

Use

Calculation steps are grouped together to form calculation procedures [page 821] and are the most detailed elements of a calculation procedure. A calculation step generates a business transaction item. Each business transaction item then generates two document lines – one for debit and one for credit.

Calculation steps are defined in Customizing on the basis of **calculation step types**. They are assigned item types and the calculation methods [page 870] provided by SAP.

- Calculation methods are used to specify the value and flow direction (debit/credit) of an item if an internal business transaction is generated.
- An item type for debit postings and one for credit postings can be assigned to a calculation step.

Features

An overview of the system table entries for calculation management can be found in the Customizing activity *Edit Procedure Templates* `/BA1/B1_CM_SY_MAINT` under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Control*. These are displayed in the following subdialogs:

- *Define Processing Categories*
- *Define Item Templates*
- *Assign Processing Categories to Item Templates*
- *Define Calculation Step Categories*
- *Assign Item Templates to Calculation Step Categories*
- *Define Calculation Methods*
- *Assign Calculation Methods to Calculation Step Categories*
- *Define Calculation Procedure Classes*
- *Define Calculation Procedure Templates*
- *Assign Calculation Step Categories to Calculation Templates*
- *Assign Relevant Item Templates and Calculation Step Categories*
- *Define Business Transaction Categories*

> **Note**
>
> Descriptions are available for the individual subdialogs. To display these, choose the *Documentation* pushbutton in the respective subdialog.

Consistency Checks

The following consistency checks are carried out for calculation steps:

- Only certain calculation methods can be assigned to a calculation step category.
- Only certain item types can be assigned to a calculation step category.
- A procedure template specifies a particular set of calculation step categories that can be used for the calculation procedure. You cannot insert calculation steps from other calculation step categories. You can omit some of the calculation step categories.
- Some step types have been selected as mandatory step types in the procedure templates. Each corresponding calculation procedure must contain a calculation step from these calculation step categories.

Activities

Creating Calculation Steps

To define calculation steps based on calculation step categories and assign a calculation method [page 870] to them, you make settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Procedures* > *Edit Calculation Steps*

Key Figure Steps

A calculation step is a key figure step if the *Key Figure Step* indicator has been selected for the assigned step category.

A key figure step processes all posting key figures that are assigned to the selected input processing category. You can assign a key figure step a default calculation method that is to be used when these posting key figures are processed. You can also define specific calculation methods for selected key figures.

If no specific method is assigned to a key figure, the system does the following:

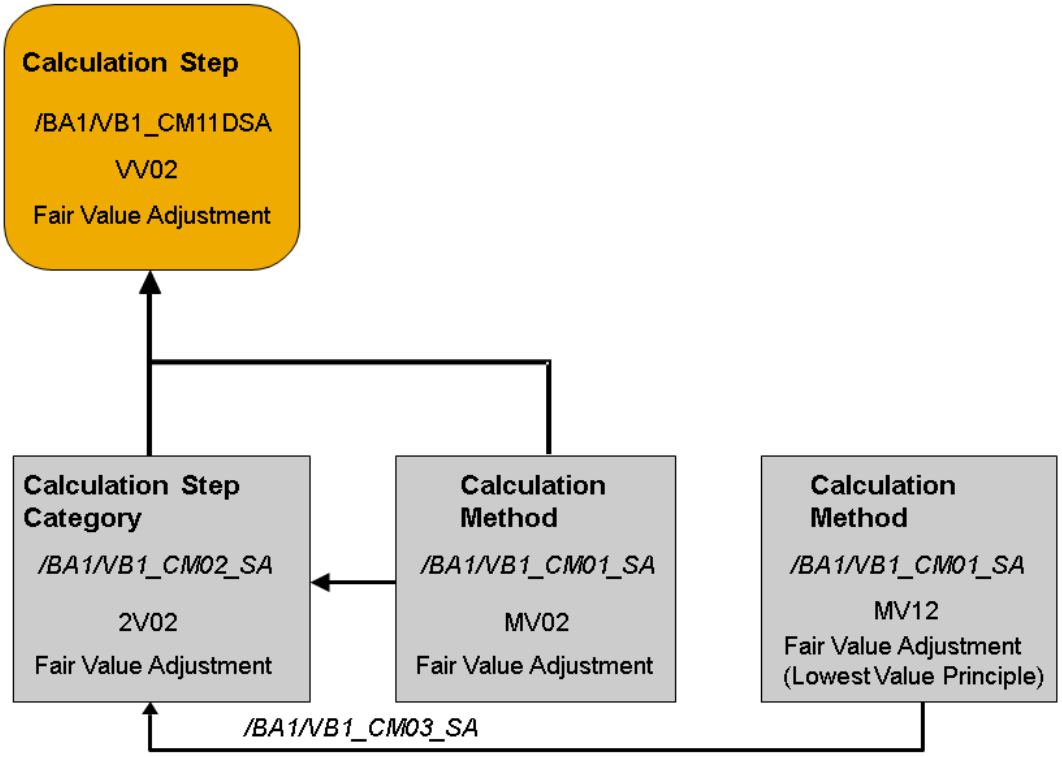
- It uses the calculation method that is defined at the level of the calculation step.
- If no calculation method has been defined here either, the system does not execute a calculation method for this key figure.

Assigning Calculation Steps to Calculation Procedures

You make the settings for the assignment of the calculation steps to the calculation procedures in the same Customizing activities as the definition of the calculation procedure, for example for accrual procedures under *Edit Accrual Procedures*.

Example

The figure below illustrates the definition of a calculation step used to run a fair value valuation. The corresponding calculation step category permits two different calculation methods; you need to choose one of these:

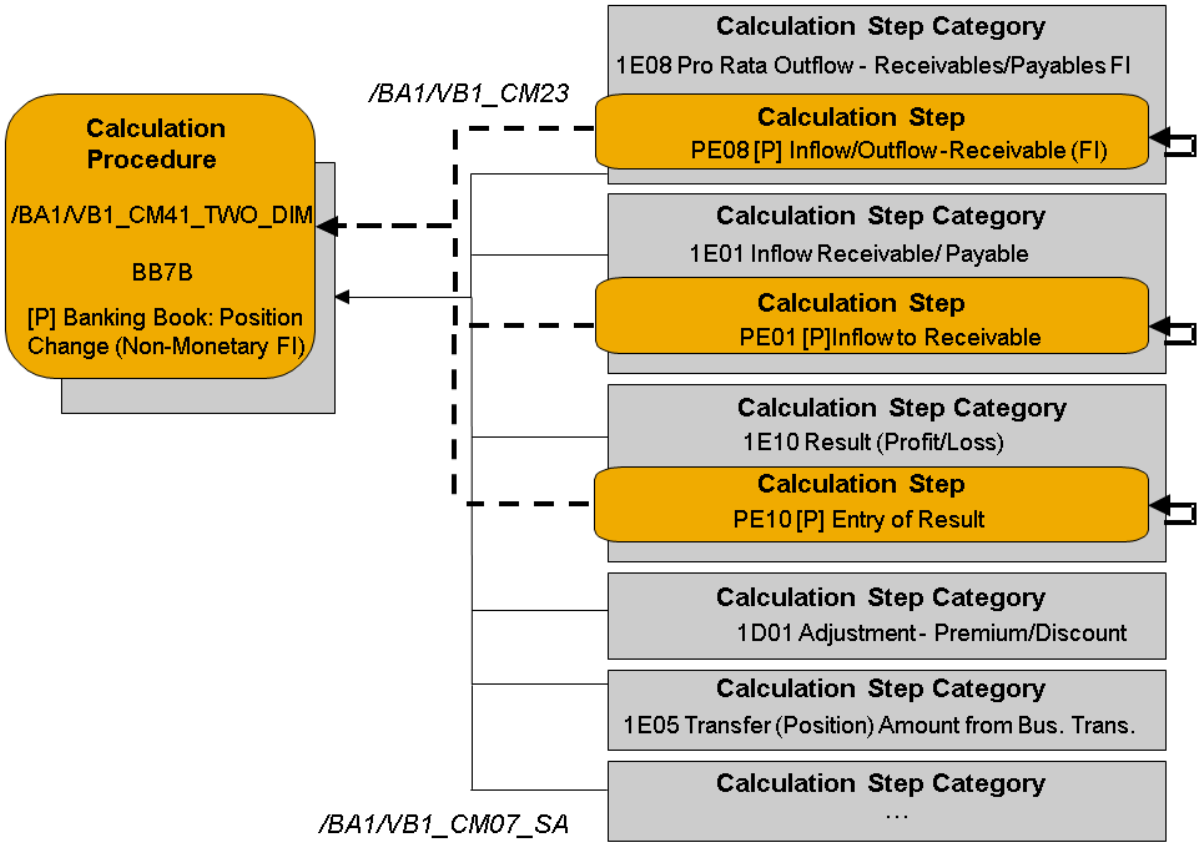


Definition of a calculation step

Only one step of the fair value valuation is created in this example. If you want to differentiate between a valuation against the revaluation reserve and a valuation against the P&L statement, you can do so by deriving the posting key figure type from the item type [page 872].

Alternatively, you could create a second calculation step for the calculation step category `2V02` (one for valuations against the revaluation reserve and one for valuations against the P&L statement) and then assign different item types to the calculation steps (with the result that even different calculation procedures would have to be created with the matching calculation step).

The figure below illustrates the calculation steps assigned to a customer-defined calculation procedure for the purchase of stocks.



Assigning calculation steps to a calculation procedure

Three calculation steps have been assigned. Calculation step `PE08` reduces a potential short position and `PE10` posts a potential realization. Calculation step `PE01` can be used to create a long position.

4.4.5.2.6.5 Calculation Method

Use

The calculation methods for internal business transactions are supplied by SAP, and include the logic required to calculate the values of a posting. The result of a method is generally a quantity or value, and this result affects a position. Some methods also or only provide calculation bases for accruals, valuations, or hedge management.

This section is only relevant if you want to gain a deeper technical understanding.

The calculation methods supplied by SAP are assigned to one or more calculation step categories. Customizing does not let you modify the assignment of a calculation method to a calculation step category.

Features

Logic

All calculation methods available in the system are stored in a system table. From a technical perspective, these entries are used as filter values when the corresponding BAdI method is executed.

The calculation methods are implementations of the BAdI interface `/BA1/IF_EX_B1_STD_CM_METHOD` (enhancement spot `/BA1/B1_STD_CM_METHOD`). This allows you to define a standard interface for all calculation methods. To link the calculation method to a step category, the BAdI implementation must use a filter that corresponds to the method name in the system table.

The user can write additional implementations that can then be used in calculation procedures [page 821]. If the interface does not provide the required data, such as cash flows or market data, you need to request this data from the relevant source.

System Customizing

An overview of the system table entries for calculation management can be found in the Customizing activity *Display Procedure Templates* `/BA1/B1_CM_SY_MAINT` under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Control*.

Activities

There are no special Customizing activities for editing calculation methods. You can edit calculation methods using the Object Navigator (transaction `SE80`). When you define a calculation step, you must assign a calculation method. For more information, see Calculation Steps [page 867].

Example

In Figure 1, you see that a relevant BAdI implementation is stored for calculation method ME01. You can use the method for various calculation step categories.

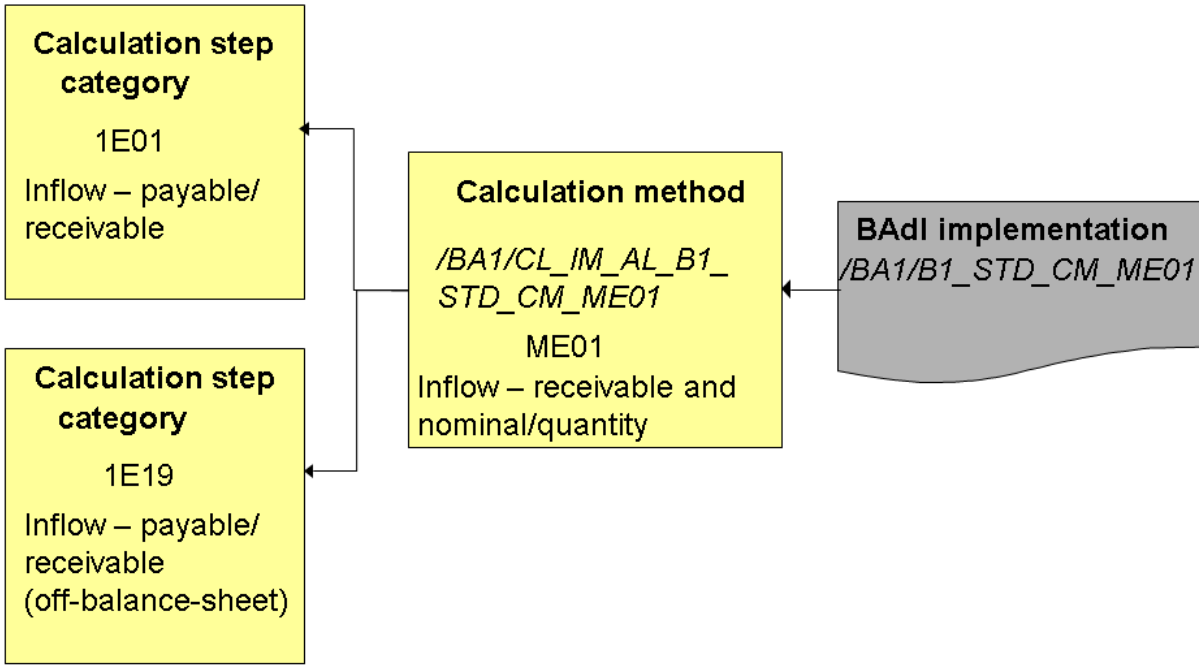


Fig. 1: Definition of a calculation method and its assignment to calculation step categories

4.4.5.2.6.6 Item Types

Use

Item types are used to specify which key figures need to be posted (a debit and a credit key figure for each item type). Item types are defined according to the item templates supplied by SAP. You can create your own item types in their own namespace.

The term *item* refers here to items in both external and internal secondary business transactions.

When it processes an external business transaction, the system transfer each relevant item of the SDL business transaction to an item type in Accounting. This results in two posting lines. For internal secondary business transactions, each calculation step [page 867] can one or more items in the document. You define the item type for an item in Customizing (Assign Item Types to Calculation Steps).

Features

Item Templates

Item templates are used either for **internal** or **external** business transactions, depending on the document type.

Item templates can contain the following information or restrictions. You can configure the following in the Customizing activity `Verfahrensvorlagen anzeigen` under *Define Item Templates*:

- You can assign a flow direction to an item template for internal control purposes.
- You can assign the category of the position value to be changed to an item template for internal control purposes.
- You can configure the check for amounts and quantities.
- You can control whether an item template can change the characteristics of the financial position.
- You can control whether an item type can be internally processed with the original or reversed plus/minus sign.
- You can specify the item types for which the system does not derive a general ledger account.
- You can control the FX translation according to currency swap transactions.
- You can control whether an item template can be used for manual postings.

Item templates also specify which processing categories can be posted on either the debit or credit side (`/BA1/TB1_ACCKF2S` table). For more information, see the Item Templates for External Business Transactions [page 875] documentation.

System Tables

`/BA1/TB1_ITMPAT`: Item Pattern Definition

`/BA1/TB1_CM04_SA`: Item Template for Calculation Step Cat.

`/BA1/TB1_CM05_SA`: Relevant BT Item Templates and Calc. Step Cat

> ⚠ Caution
>
> Table `/BA1/TB1_CM05_SA` is relevant only in conjunction with certain method implementations.

Customizing Tables

`/BA1/TB1_CM12`: BT Item Types for Calculation Steps (customer side)

`/BA1/TB1_ITMTYP`: Definition of Item Types (customer side)

Consistency Checks

You can only derive key figures of specific processing categories from an item type. Further consistencies are ensured by the properties of the item templates described in the introduction.

- You can only assign item types from specific item templates to a calculation step (`/BA1/TB1_CM04_SA` table).
- A calculation step can only process the values of item types from specific item templates (of the external business transaction) (`/BA1/TB1_CM05_SA` table).

Activities

Subledger Scenario

Displaying the Calculation Procedure Template

This Customizing activity provides an overview of the system table entries in Calculation Management. It includes a definition of the item templates and their properties.

In Customizing for *Bank Analyzer* , choose *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Control* > *Display Procedure Templates.*

Defining Item Types

In this Customizing activity, you define all item types (for internal and external business transactions) based on the item templates.

In Customizing for *Bank Analyzer* choose *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Types of Business Transactions and Items* > *Create Item Types.*

Assigning Item Types

- The permitted item types of the external business transaction are assigned to the previously defined business transaction types for validation purposes. This Customizing activity is only mentioned here for the sake of completeness. It is not actually part of Calculation Management since it does not affect the processing of internal business transactions.

  In Customizing for *Bank Analyzer* choose *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Types of Business Transactions and Items* > *Assign Item Types to Business Transaction Types.*
- The item types for **internal** business transactions are assigned to calculation steps. At least one, but no more than two item types, must be assigned to each calculation step (debit/credit). For example, a calculation step only has one item type for the asset-side inflow of the acquisition value (always a debit posting). On the other hand, a calculation step for realization generally has a debit and a credit item type for posting expenditures and revenues.

  You assign item types to calculation steps in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Control* > *Display Procedure Templates.*

Example

In figure 1, you can see the definition of two item types that can be used to post valuations. They are assigned to a calculation step of the corresponding calculation step type for valuations (0S06). You must set the indicator to specify whether you want to make a debit or credit posting.

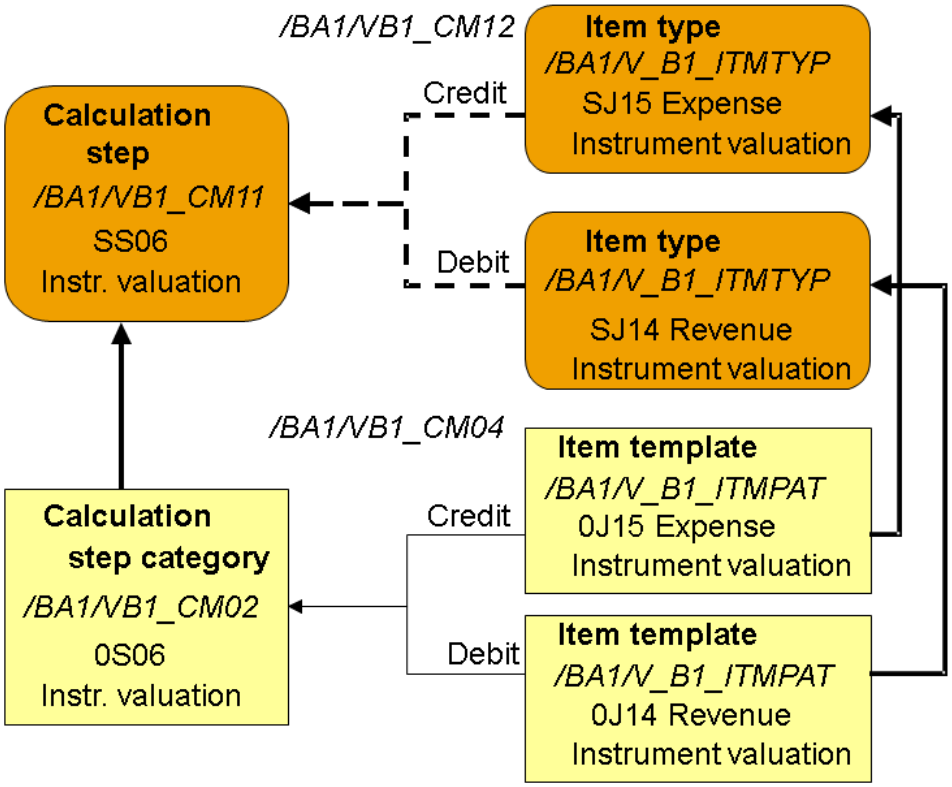


Figure 1: Definition of item types and their assignment to a calculation step

4.4.5.2.6.6.1 Item Templates for External Business Transactions

Use

This document contains detailed information about using item templates in external business transactions.

Validation

Each business transaction item can contain the fields *Amount* and *Quantity*. **Plausibility checks** (validations) are carried out for these two fields, based on the item template. You can see which validations the system carries out in the view **/BA1/VB1_ITMPTSA** in the two columns **AmtCheck** (amount check) and **QtyCheck** (quantity check).

In Customizing for *Bank Analyzer* , choose *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Control* > *Display Procedure Templates* .

Choose the view *Define Item Templates*.

Posting Management

Possible item type postings are defined using the item templates, based on the assignment of the **credit/debit indicator** and the **processing categories** for the item templates. In the Customizing activity *Display Procedure Templates*, you can check the amount and quantity checks under *Define Item Templates* and the assignment of processing categories for item templates under *Assign Processing Categories to Item Templates*.

Example

Item template `1I00` with the *Credit* indicator can post the processing categories `1BF_TRANS` and `1BG_TRANS`. Processing category `1BF_SUSP` posts the amounts on the debit side.

In accordance with the table mentioned above, the item templates can essentially be differentiated on the basis of whether they process a transaction using:

1. Processing category `1BG_TRANS` for *In-Transit Account (Anonymous)*
2. Processing category `1BF_SUSP` for *Suspense Account (Product-Based)*

Process Management for Calculation of Derived Business Transactions

Item templates control the internal processes during the calculation of derived business transactions using:

1. The values in the columns:
   - Position change direction (*FlowDirect* column)
   - Position change category (*FlowExtent* column)
   - Checkbox for inverse of plus/minus sign for quantity (*Inv+/-:Qty* column)

   You can find these values in the Customizing activity *Display Procedure Templates* under *Define Item Templates*.
2. In the Customizing activity *Display Procedure Templates*, you can assign item templates, for example, to a calculation step category of a procedure template under *Relevant BT Item Templates and Calculation Steps*. This allows you to specify that the calculation step assigned to the calculation step category is only to react to a certain external item that has an item type for the assigned item template (particularly in the case of external business transactions with multiple items).
   You can find the assignment of external item templates under *Relevant BT Item Templates and Calculation Steps* in the Customizing activity *Display Procedure Templates*.

Example

`1I00` **Inflow of Quantity and Value to Suspense Accounts**

Item template `1I00` is used in the following calculation procedure templates, for example:

- 1101 Banking Book: Position Generation
- 1102 Banking Book: Scheduled Position Reduction
- 1103 Banking Book: Unscheduled Position Reduction
- 1104 BB&TB: Position Change Financial Instrument
- 2101 Unscheduled Position Change (Not Dependent on Value Date)
- 2102 Position Change Financial Instrument (Not Dep. on Val. Date)

`1I12` **Inflow of Quantity to Suspense Account**

Item template `1I12` is used in the new calculation procedure templates, for example:

- 1140 Pro Rata Position Transfer - Value-Date-Based
- 2140 Pro Rata Position Transfer - Not Value-Date-Based
- 2141 Reversal of Pro Rata Position Transfer Not Value-Date-Based

4.4.5.2.6.7 Customer Enhancements

You can create your own calculation methods and templates. This section provides a brief introduction on how you can enhance Calculation Management with your own implementations. For more information, contact SAP.

You edit the system tables for Calculation Management in the Customizing activity *Display Procedure Templates* `(/BA1/B1_CM_SY_MAINT)` by choosing *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Control* > *Display Procedure Templates*.

Creating Your Own Calculation Methods

Calculation methods [page 870] form the basis of Calculation Management. They contain the program logic for all calculations.

Every calculation method has a four-character name, which you can edit in the Customizing activity *Display Procedure Templates* under *Definition of Calculation Methods*. There is also a BAdI implementation for each method:

BAdI definition:

- Definition name: `/BA1/B1_STD_CM_METHOD`, defined in package `/BA1/AL_B1_STDACC_CALC_MGMT`.
- The BAdI implementations are filter-dependent, and the four-character method name is also the corresponding filter value.
- Interface name: `/BA1/IF_EX_B1_STD_CM_METHOD`, defined in package `/BA1/AL_B1_STDACC_CALC_MGMT`. The actual program logic is implemented in method `EXECUTE`.

The classes for the SAP calculation methods are all stored in package `/BA1/AL_B1_STDACC_CALC_METH` in the SAP namespace. Rather than modifying this package, you should create a new package for your own methods in your namespace. For your own methods, you therefore first create an entry under *Display Procedure Templates* under *Definition of Calculation Methods* and then create a BAdI implementation.

You then still have to assign the new calculation method to a calculation step category. You do this using a new entry in the Customizing activity *Display Procedure Templates* `/BA1/B1_CM_SY_MAINT`.

Creating Your Own Templates

No additional programming is required to create your own templates. To create a new template, you simply make additional entries in the system tables: definition of the templates and assignments of templates. You also do this using the Customizing activity *Display Procedure Templates*.

We distinguish between three types of templates:

1. Procedure templates: You have to assign new procedure templates to a procedure class. You also have to assign calculation step categories to your new procedure template.
2. Calculation step categories: You must assign a method and item templates to the new calculation step category.
3. Item templates: You must assign two processing categories to a new item template; these are for the debit and credit posting.

4.4.5.2.6.8 Characteristics in Calculation Management (Subledger Scenario)

The posting logic in Calculation Management is primarily controlled by the settings made in Customizing, which use the characteristics of the financial position [page 749] and the accounting document. The derivation tool plays a decisive role here. In the derivations, parameters that control the calculation procedures [page 821] (target fields) are derived from the characteristics of the financial position and the accounting document (source fields).

If you select the characteristics for the objects in Accounting (especially business transaction and financial position characteristics), you must take into account the requirements of the calculation procedures and the posting logic. This section describes the origin of the characteristics used to determine the posting logic, and how you can use these characteristics in Customizing. It also gives a more detailed description of the procedures for specific characteristics that are typically used.

Origin of Characteristics

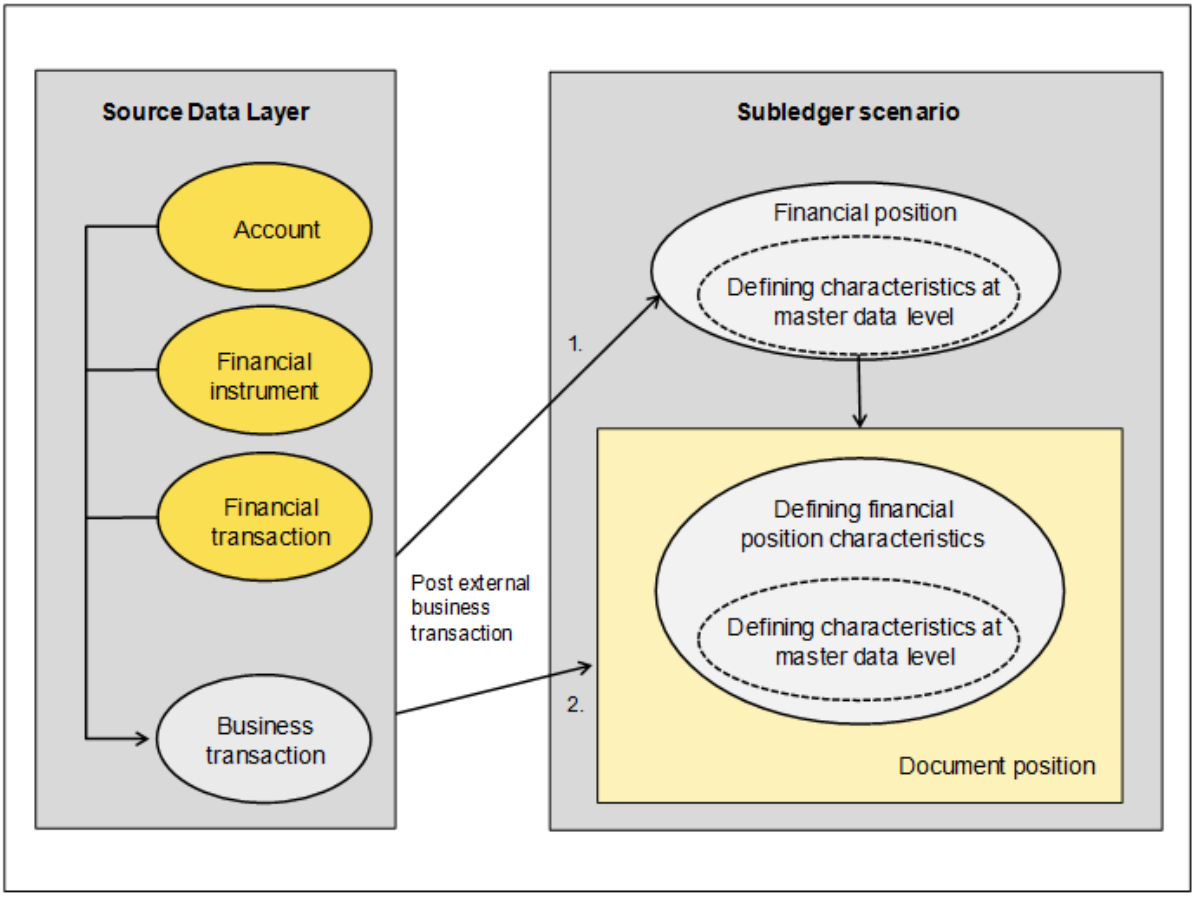


Figure 1: Origin of Characteristics within Calculation Management

1. Financial position: The financial position is identified by the defining characteristics at master data level and the defining characteristics at valuation level. All the characteristics of the financial position are populated in assignment or derivation procedures by means of business transactions.
2. Accounting document: All the defining characteristics of the financial position automatically become characteristics of the accounting document. These characteristics are populated by copying processes from the business transaction.

Financial Position

The financial position is identified by the defining characteristics at master data level and the defining characteristics at valuation level. The defining characteristics of the financial position are populated directly from the business transaction. The descriptive characteristics can either be populated directly by the characteristics in the business transaction or derived from these characteristics.

Accounting Document

The characteristic structure of the accounting document contains fields of different sources, such as the business transaction header and items. There are many predefined characteristics, including the business transaction type or item type, for example. All of the defining characteristics of the financial item are automatically characteristics of the accounting document. You can also define your own characteristics.

The characteristics of the accounting document are populated from the characteristics of the business transaction using either copy or derivation procedures. For structured products, some predefined characteristics are internally populated by interpreting the appropriate financial instrument and flow data in the SDL.

Using Characteristics

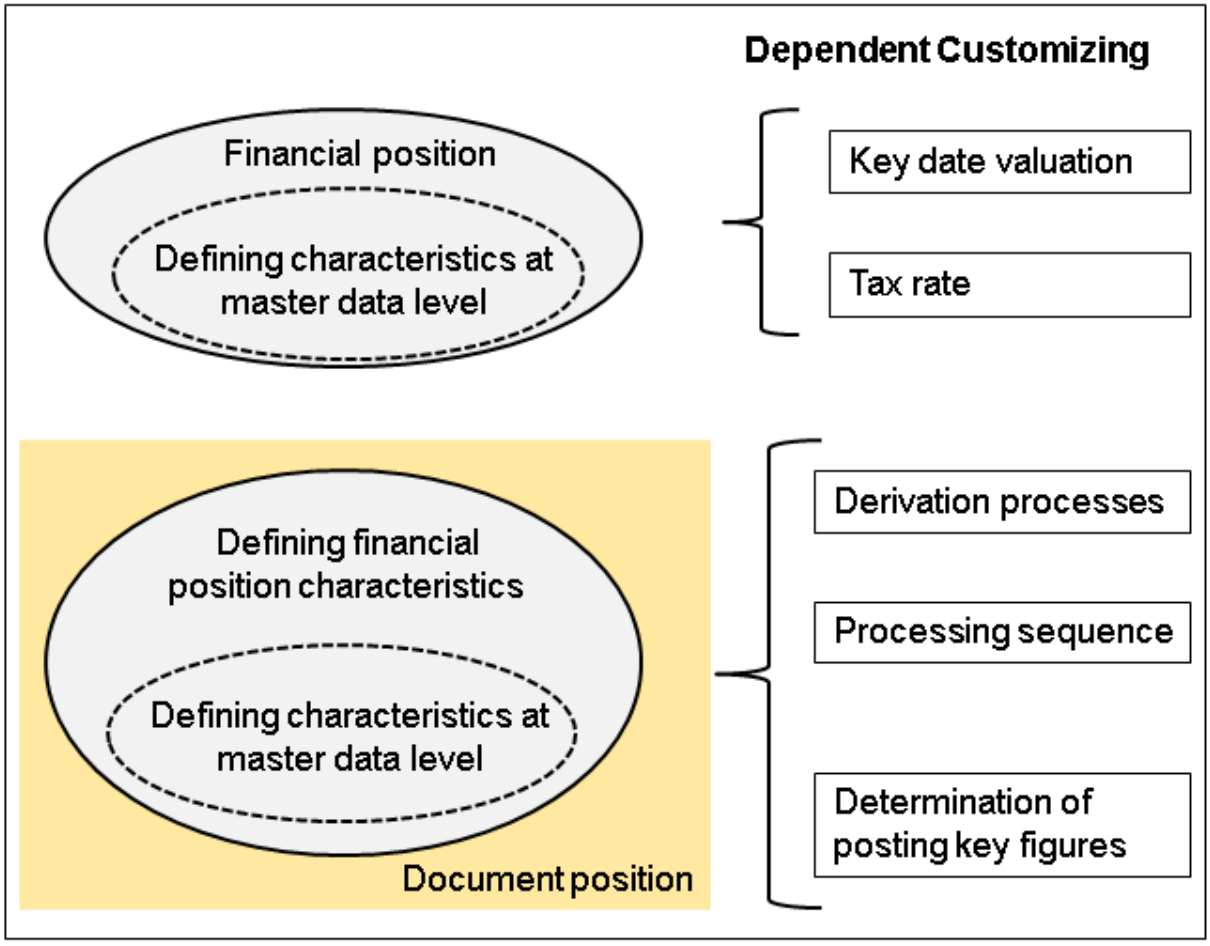


Figure 2: Use of Characteristics in Customizing for Calculation Management

The figure shows the Customizing activities that use the characteristics of the financial position and the accounting document:

Characteristics of the Financial Position

All characteristics of the financial position can be used in Customizing regardless of whether they are defining or descriptive. This also applies to the differentiation between predefined and customer-specific characteristics. When processing the Customizing activities in calculation management, the system provides only those characteristics that are valid for the respective Customizing activity. You cannot use key figures.

- **Key Date Valuation**
  You derive the calculation procedures for the key date valuation from the characteristics of the financial position you want to valuate. You can use the additional source field *+/- Quantity*.
  You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Accounting Processes*.
  - *→ Accruals/Deferrals→ Accrual Methods→ Derive Accrual Procedures*
  - *→ Valuation→ Derive Calculation Procedures for Key Date Valuation*

  Example: Typical characteristics used to derive the calculation procedure include the *key date valuation type*, *financial position type*, and *holding category*.
- **Tax Rates**
  You can derive the tax rate to calculate the deferred taxes [page 817] from the characteristics of the corresponding financial position.
  You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Accounting Processes* > *Valuation* > *Deferred Taxes* > *Derive Tax Rates*.
  Example: The legal entity is a typical source characteristic in this case.

Characteristics of the Accounting Document

- **Derivation of External Business Transactions**
  The characteristics of the accounting document control the calculation procedures (for example, elementary calculation procedures and component transfer). These are used to create derived business transactions. In Customizing for *Bank Analyzer*, choose:
  - *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Elementary Calculation Procedures*:
    - *→ Elementary Calculation Procedures (Dependent on Value Date)→ Derive Elementary Calculation Procedures (Value-Date-Based)* or
    - *→ Elementary Calculation Procedures (Not Dependent on Value Date)→ Derive Elementary Calculation Procedures (Not Value-Date-Based)*

- *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Component Transfer* > *Derive Component Transfer Procedures*

  Example: The typical characteristics used to derive the derivation procedure for a financial position are the *business transaction type* and *holding category*.

- **Secondary Processing Sequence**
  In calculation management, you must specify the sequence in which you want the operational business transactions to be processed for a position in Accounting. You can determine this sequence on the basis of the characteristics of the accounting document. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Edit Rules for Determining Secondary Processing Sequences*.
  Example: If we assume that inflows should be processed before all outflows, you could select the business transaction type as the source parameter in Customizing and assign a higher sort key to the outflows.
- **Effects of Position Changes on Hedged Positions**
  The effects of an operational business transaction on a position that consists of an unhedged part and one or more parts of a hedging relationship can vary. Possible consequences include:
  - Dissolution of a hedging relationships
  - Distribution of the position change to both the positions of the hedging relationships and the position without reference to hedging relationships
  - Overall distribution of the position change to the unhedged positions

  You can define the effects of a position change in Customizing using the accounting document characteristics. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *Accounting Processes* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Hedge Accounting* > *Processing of Position Changes* > *Derive Consequences of Position Changes*.
  Example: An unscheduled position outflow causes the hedging relationship to dissolve. A scheduled position outflow is distributed proportionally to both the hedging relationship position and the position with no reference to hedging relationships, and an unscheduled inflow changes the unhedged position. This is done by defining the business transaction type as the source field in Customizing. You only need to edit those entries that refer to scheduled and unscheduled position outflows, since position inflows do not affect hedged positions by default.
  For more information, see Micro Fair Value Hedges (Subledger Scenario) [page 905].

Selected Characteristics

This section provides you with some selected characteristics that are typically used in Customizing for calculation management:

- **Holding Category (SAP Category /BA1/C55HOLDCT)**
  The *holding category* characteristic lets you classify financial instruments according to IAS 39 §10. How a financial instrument is valuated depends on its assigned category. The holding category must be a characteristic of the financial position. The system ensures this for the *financial instrument*, *financial transaction*, and *aggregated product* product categories. The *holding category* characteristic is supported as a descriptive characteristic only.

The holding category is derived from the master data in the Source Data Layer (SDL) when a financial position object is created. Changes to the holding category (reclassification) can be made by means of a reclassification decision only. A reclassification decision is an analytical event in the form of an SDL business transaction. In contrast, master data changes do not lead to changes to the holding category.
The different rules for valuating financial instruments are reflected in the various calculation steps in Accounting. For example, the valuation of a held-for-trading bond always includes a step for fair value valuation. There is no such step for a held-to-maturity bond. The best way of mapping these differences in the system is to define each different calculation procedure according to the holding category. This is partly reflected in the procedure templates.
Another difference between the holding categories concerns posting net incomes. In terms of financial instruments, the *held-for-trading* category means value changes are posted to the P&L statement, while the *available-for-sale* category means changes in value are posted to the valuation reserve. This difference can be implemented in the system, whereby a separate calculation procedure that includes the corresponding calculation step is created for each holding category. The derivation of the calculation procedure now depends on the holding category.

- **Item Category for Hedge (/BA1/C69HRLCAT)**
  The *Item Category for Hedge* characteristic is a predefined descriptive characteristic of the financial position. It is used to distinguish the hedging instrument (hedge transaction) from the hedged item (underlying transaction) in a hedging relationship.
- **Financial Position Type (/BA1/C55AOTYPE)**
  The financial position type is a predefined descriptive characteristic of the financial position.
- **Procedure for Calculating Hedge Adjustments (/BA1/C55HSIMPR)**
  This characteristic represents a predefined descriptive characteristic of the financial position. The value of this characteristic controls, for example, whether the standard method or the simplified method is used to determine the hedge adjustment for a position. You fill this characteristic manually in the Hedge Accounting Workplace. It is passed on to Accounting by the *designation* business transaction.
- **Component Class (/BA1/C55COMPCL)**
  The component class is a predefined descriptive characteristic of the financial position. It is used to distinguish the main portion (M) from the separated derivative part (D) or non-separated derivative part of a structured product. This characteristic is filled using the characteristics for Financial Position Management with attributes of the financial instrument or financial transaction from the Source Data Layer (SDL). The component class is used to derive different calculation procedures for the components of a structured product.
  Example: The system splits the warrant bond purchase [page 857] business transaction into two business transactions – one for the main portion and one for the warrant. Since the business transaction type is the same for both components, you can use the component class as a second source field to derive different calculation procedures such as the *warrant purchase* and *bond purchase*.

4.4.5.2.6.9 Amortization, Fair Value Valuation, and Hedge Adjustment (Subledger Scenario)

Use

This section describes the calculation steps amortization [page 890], valuation to be released, hedge adjustment (subledger scenario) [page 895] and valuation [page 896] and their processes.

Features

The data basis for the amortization, fair value valuation to be released, fair value and hedge adjustment calculation steps consists of the balances of the accounting key figures that are available in accounting on a key date, and of the values provided by the Fair Value Server.

- The balances of the accounting key figures are determined for a key date and an SDL time stamp using the calculation methods that are assigned to the calculation steps.
- The system determines current values, such as fair value or amortized cost by calling the Fair Value Server for this key date and SDL time stamp.

The following calculation steps relate to amortization, valuation to be released, hedge adjustment, and fair value adjustment.

Common Data Basis for All Calculation Steps/Valuations

- Financial instrument or financial transaction ID of the Source Data Layer (SDL) including information about parent/child node and the node number
- Position nominal value or quantity and its units
  This data is used for scaling.
- Financial position currency
- Weighting factors
  These are used for scaling if hedging relationships exist and are determined internally by SAP.

> **Note**
>
> SAP does not scale the cash flows from SDL financial transactions using the nominal values from accounting. These values are used, however, to scale the cash flows from the SDL financial instruments. The calculated target value (fair value/amortized cost/hedge adjustment) is corrected in the Fair Value Server using the weighting factor. That is why the calculation log of the risk basis shows the cash flows for the entire SDL financial transaction, while the correct fair value/book value is calculated in accounting.

- Key date for which you want to determine a value.
  If you have selected the start-of-day logic for the value date of deferrals in Customizing (start-of-day logic), during key date valuation the system calculates using the key date +1. Since key date valuation is an end-of-day process, this illustrates that even for payments on the same day (start-of-day values) a day has already passed.
- Time stamp to which the SDL data is to be taken into account.
  You use the function Set the SDL Time Stamp [page 367]. If, for example, you created a new version of an SDL primary object or new market data without creating a new time stamp afterwards, the new SDL data is not included in the calculation.
- Global valuation environment/cash flow rate
  You assign these to your accounting system (see the *Activities* section).
- Market Data Area
  You assign this to your accounting system (see the *Activities* section).

> **Caution**
>
> Note that the system can only calculate the amortization values for multicurrency bonds if the currency used for the payment is the same as the currency used for the repayment.

- Fictitious cash flows

Fictitious cash flows are included in the calculation of the amortized cost.
Example: Amortizing Swaps [page 893]
For a leg of a swap, the capital that is swapped in the initial payment and settlement is included as a fictitious cash flow when calculating amortized cost. Therefore, make sure in Customizing for the determination of market data that both legs of the swap have the same market data record (see the *Activities* section). When calculating the fair value, however, the system only includes actual flows of the future cash flow.

Special Data Basis for Amortization

Characteristics and key figures for the key figures of processing category `1BF_DEFDIS` are stored in the results data of accounting as calculation bases at the financial position level. This calculation base data is communicated to the Fair Value Server in a later fair value valuation. The following data, for example, is automatically updated and delivered to the Fair Value Server as calculation base data by the accounting processes:

- Amortized cost at the time of the last update as the target value
  The *Amortized Cost* is communicated to the Fair Value Server instead of the *Amortization* key figure. This amortized cost is equal to the sum of the acquisition value and amortization amounts. For multiple amortizations it is the sum of all amortization amounts.
- Currency
  The amortization, and thus the amortized cost, is always calculated in the financial position currency.
- Posting date and value date of the last amortized cost update
- Indicator: Amortized Cost at Start of Day/End of Day
  Depending on which settings you have made in Customizing for the value date of deferrals, the system interprets the effective capital values and the accrued interest values of the value date either as start-of-day or end-of-day values.
- SDL time stamp that specifies the time until which SDL data was taken into account during the last update of the amortized cost.
- Effective Interest Rate (Amortized Cost)
- Posting date and value date of last effective interest rate update
- SDL time stamp that specifies the time until which SDL data was taken into account during the last update of the effective interest rate.
- Control indicator: Calculation of the effective interest rate as per IFRS 39 AG8
- Control indicator: Calculation of the effective interest rate as per IFRS 39 AG7
- Control indicator: Recalculation of Effective Interest Rate

If multiple key figures are assigned to the `1BF_DEFDIS` processing category, the system distributes the amount to be amortized across the key figures.

Calculation of the effective interest rate as per IFRS 39 AG8

As an alternative to the standard SAP effective interest method, you can also calculate the effective interest rate in accordance with IFRS 39 AG8. Only the effective interest rate that was calculated at the start of the term of the loan on the basis of the **expected** cash flows can be used to calculate the amortized cost according to IFRS 39 AG8. For example, the expected cash flows for a loan are the result of the borrower's payment history. The effective interest (based on the **expected** cash flows) remains constant over the life of the transaction, regardless of whether the expected cash flows are realized or not. You can calculate the effective interest rate on the basis of both the **contractually agreed** and the **expected** cash flows.

In Customizing for the Fair Value Server under *Edit Valuation Settings*, you can use the characteristics of the financial transaction to define a setting that keeps the effective interest at a constant rate, and does not react to business transactions or events (see *Activities* section).

> **Note**
>
> In principle, when dealing with a loan, you should not switch between valuation rules that use the standard SAP method and AG8 valuation rules. However, you could, for example, create groups of loans and amortize one group in accordance with the AG8 regulations, and one with the standard SAP method.

Recalculation of effective interest rate

If, despite the above, you want to calculate a new effective interest rate, you can supply an operational event to do so. This operational event triggers a calculation procedure in the relevant position for the financial transaction (see Calculation Procedure [page 821]). This both starts the amortization calculation based on the previous effective interest rate and sets an indicator that can be used to determine a new effective interest rate for amortization at a later date. The system then uses both the expected cash flows and the last determined amortized cost value to calculate the effective interest rate.

If you have selected the *AG7* checkbox, when the system recalculates the effective interest rate for variable-rate transactions it updates the effective interest rate for every interest fixing between the last amortized cost calculated and the key date.

> **Note**
>
> In the event of an **unscheduled repayment**, you do not need to deliver any additional operational events apart from the unscheduled repayment business transaction if you derive a calculation procedure for the unscheduled repayment based on procedure template 1103.
>
> **If a version is changed in the SDL**, you must specify the operational event for recalculating the effective interest rate on the same day that the version changes. Irrespective of AG8, you need to provide an operational event to inform Accounting of a condition change, for example.
>
> The system will still allow you to import a business transaction instead of an operational event.

The **calculation template** 1105 for the procedure described is available in Customizing. You use the **calculation rule** to determine which financial transactions are the subject of the fair value valuation in accordance with IFRS 39 AG8. For more information, see *Activities*.

> **Note**
>
> If you adjust the future cash flows to match the expected payment behavior of the borrower **following an impairment** (for example, of a loan), these new cash flows are then discounted using the effective interest rate in accordance with IFRS 39 AG8. The system also amortizes the data and adjusts the value of the loan correspondingly.

Calculation of the effective interest rate as per IFRS 39 AG7

Under IFRS 39 AG7, interest rate fixing for variable-rate transactions leads to an update of the effective interest rate. The effective interest rate, however, remains unchanged for other adjustments to expected payments. In the same way as for AG8, you can define processing according to AG7 in Customizing for the Fair Value Server under *Edit Valuation Settings*.

If you have activated the fair value valuation in accordance with AG7, the Fair Value Server recognizes the relevant interest fixing dates based on the cash flow of a transaction and updates the effective interest rate on these dates.

> **Note**
>
> The behavior described only occurs if you set the *Valuation as per AG7* indicator and the *Valuation as per AG8* indicator. AG8 suppresses adjustments to the effective interest rate. AG7 allows the system to adjust the effective interest rate based on interest fixing. If you activate AG7 only, however, every change to the expected cash flows directly affects the effective interest rate.

Special Data Basis for Amortization including Valuation Remnants

Characteristics and key figures for processing category `2BF_DEF` key figures are stored in the accounting results data as calculation bases at the financial position level. This calculation base data is communicated to the Fair Value Server in a later fair value valuation. The following data, for example, is automatically updated and delivered to the Fair Value Server as calculation base data by the accounting processes:

- Amortized cost plus valuation remnants at the end of the calculation date, as a target value
- Currency
  The amortization, and thus the amortized cost including valuation remnants, is always calculated in the financial position currency.
- Posting date of the last update of the amortized cost including valuation remnants
- SDL time stamp that specifies the time until which SDL data was taken into account during the last update of the amortized cost including valuation remnants.
- Effective interest rate (amortized cost plus valuation remnants)
- Posting date on which the effective interest rate including valuation remnants was last calculated
- SDL time stamp that specifies the time until which SDL data was taken into account during the last update of the effective interest rate including valuation remnants
- Control indicator: Recalculation of Effective Interest Rate for Amortized Cost incl. Valuation Remnants
- Control indicator: Effective Interest Rate Calculation in Accordance with IFRS 39 AG8 for Amortized Cost Including Valuation Remnants

If multiple key figures are assigned to the `2BF_DEF` processing category, the system distributes the amount to be amortized across these key figures. You cannot exclude individual key figures from this.

The effective interest rate for amortization including valuation remnants is always recalculated when the effective interest rate is also recalculated for the amortized cost (see the section *Special Data Basis for Amortization*). The effective interest rate is also updated when new valuation remnants are recorded, for example, for a reclassification or when a hedging relationship is dissolved.

Special Data Basis for Hedge Adjustment

Characteristics and key figures for the key figures of processing category `2BF_VALHAD` are stored in the results data of accounting as calculation bases at the financial position level. This calculation base data is communicated to the Fair Value Server in a later fair value valuation. The following data, for example, is automatically updated and delivered to the Fair Value Server as calculation base data by the accounting processes:

- Date when the *Hedge Adjustment* key figure was last changed
- SDL time stamp that specifies the time until which SDL data has been taken into account.
- Hedged nominal amount
- Hedge fair value and hedge amortized cost
  The *Hedge Amortized Cost* and the *Hedge-Fair-Value* is communicated to the Fair Value Server instead of the *Hedge Adjustment* key figure. These amounts are also called target values. The *Hedge Adjustment* key figure is stored in the results data (posting documents) of accounting.
- Hedge Spread

- Hedge effective interest rate
- Market data record for the hedge fair value
- Date when the hedge effective interest rate/hedge spread was last calculated

Special Data Basis for Fair Value

Characteristics and key figures for the key figures of processing category `2BF_VALFV` are stored in the results data of accounting as calculation bases at the financial position level. This calculation base data is communicated to the Fair Value Server in a later fair value valuation. The following data, for example, is automatically updated and delivered to the Fair Value Server as calculation base data by the accounting processes:

- Date when the Fair Value key figure was last changed
- SDL time stamp that specifies the time until which SDL data has been taken into account.
- Fair Value
  The *Fair Value* is communicated to the Fair Value Server instead of the *Fair Value Adjustment* key figure. These amounts are also called target values.
- Financial instrument spread (FI spread)
- Market data record for the uncovered fair value
- Market data record for the covered fair value
- Date when FI spread was last calculated

Calculation Algorithm of the Amortization

Accounting uses the *Receivables/Acquisition Value* (`1BF_RECPAY`) and the *Deferrals - Premium/Discount* (`1BF_DEFDIS`) processing categories to determine the last calculated amortized cost for a financial position and date. In other words it calculates the total of all documents up to a given key date. If you want to determine a new amortization amount for a key date, then the amortized cost for this key date is determined by the Fair Value Server.

> **Note**
>
> Note that the accrued interest is included in the amortized cost calculation. By contrast, the business definition of amortized cost does not include accrued interest.

The difference between the new and the old amortized cost values is the amortization amount that is updated in the documents.

An effective interest rate is determined using the cash flows (disbursement, interest, repayment). The effective interest rate is used to determine the effective capital. The amortized cost is based on the difference of the effective capital and the interest to be accrued. The amortization amount is based on the difference between two amortized cost values ($AC_1$and AC_2 in the example).

In accounting, this would occur in the key date valuation of a loan, for example. In the example, AC_1 is the value that was last stored in accounting for the amortized cost for day t_1. Accounting now requires the amortized cost from the Fair Value Server for the key date t_2 based on AC_1 (for day t_1) and the given cash flows of the financial instrument. The Fair Value Server returns a value for the amortized cost AC_2 for day t_2. The value AC_2 may also have been determined by a transfer.

The system also saves the calculation base data (possibly a new effective interest rate with date see above) in the results data area of the calculation base data.

This document firstly offers a general introduction to the calculation algorithm used by SAP. This can help you to define the calculation procedures correctly, or understand the technical background.

Short Description:

- **Case 1: Effective interest rate is up to date**
  The effective interest rate provided by accounting is used for interest accumulation to calculate the current effective capital. The system calculated this based on the last calculated amortized cost and on the cash flow of the relevant financial instrument or financial transaction in SDL. The accruals are subtracted from the effective capital. This new amortized cost value is transferred to accounting.
- **Case 2: Effective interest rate is no longer up to date** (bond purchase or unscheduled repayment of a loan, for example)
  The Fair Value Server calculates the accruals for the old amortized cost and associated date and, as a result, the effective capital for the old date. The system determines the cash flows, including any required scaling, from the supplied SDL information (financial instrument/transaction). All cash flows that are older than the supplied amortized cost date are no longer included in the calculation. The future cash flows are now discounted in such a way that they yield the effective capital. In this way, the system provides you with the effective interest rate.
  Information about every condition change must be sent as an event so that the Fair Value Server is called to recalculate the effective interest rate. The effective capital is then calculated for the required key date, the accruals are subtracted and, as a result, the current amortized cost is determined.

> **Note**
>
> If you use the method `MD01`, amortization amounts can only be generated if there is a balance on Deferrals (Premium/Discount).
>
> You can use the method `MD07` for certain procedure templates. This method also calculates the target value for amortized cost if there is no balance on Deferrals (Premium/Discount).

Activities

Make the following settings in Customizing for *Bank Analyzer*.

Global Valuation Environment/Cash Flow Rate

You assign the global valuation environment or cash flow rate to your accounting system under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Basic Settings* > *Assign Global Valuation Environment to Accounting System*.

> **Note**
>
> For more information about secondary data sources, see the Customizing documentation.

Market Data Area

You assign the market data area to your accounting system by choosing *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Assign Market Data Areas to Legal Entities and Accounting Systems*.

Determination of Market Data

You make settings for determining market data under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivation*.

Fair Value Server

You make the valuation settings under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration*.

Calculation Procedure Template

For an overview of all the calculation procedure templates provided by SAP, choose *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Procedures* > *Display General Overview of All Templates*.

Valuation Rule

To create a valuation rule to recalculate the effective interest rate, choose *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivation* > *Determination of the Valuation Rule* > *Create Valuation Rules*.

You define the derivation of valuation rules from financial transaction characteristics under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivation* > *Determination of the Valuation Rule*.

To activate the fair value valuation according to IFRS 39 AG 8 or according to AG7 and AG8, choose *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Edit Valuation Settings*.

4.4.5.2.6.9.1 Amortization (Subledger Scenario)

Use

See IAS 39.10. The aim of amortization is to distribute the premium/discount to the maturity in order to determine the amortized cost.

An amortization calculation in Bank Analyzer is generally useful only for cash flow-based transactions. The amortization for the distribution of deferred premiums and discounts can be executed for each financial position using the straight-line method or based on the effective interest rate. The amortization is calculated based on the value date of a relevant business transaction, for example a position change.

For the following business transactions, define the calculation procedures in such a way that an amortization calculation ensues. In other words, the calculation procedure contains a calculation step of the *amortization* calculation step type.

> **Note**
>
> - Only one amortization occurs on a financial position for each value date.
> - No amortization occurs for an initial purchase. However, an amortization step is required in the calculation procedure for the initial purchase (see below).

- The system displays a detailed log of the calculations in the key date valuation [page 934] and Update Secondary Business Transactions [page 786] process logs if you have selected the *Create Detail Log* checkbox on the selection screen.
- If the valuation of an SDL financial transaction is involved, you can find the cash flows for the entire SDL transaction in the price calculator detailed log. A scaling of the cash flows based on single positions is not carried out for SDL financial transactions within the price calculator log.

Features

Certain types of business events trigger the amortization calculation. These events are **key date valuations** and **position-changing business transactions**, such as inflows and outflows (for example purchases apart from the initial purchase, sales, transfer postings, and redesignations). Amortization is triggered when **secondary business transactions are updated** as a result of business transactions that affect the position.

1. **Key date valuation**
   During key date valuation the financial positions are amortized by a corresponding calculation step in the calculation procedure. The difference between the old and new amortized cost values and the amortized costs values determined by the Fair Value Server is the amortization amount to be posted.
2. **Position-changing business transactions (inflows/outflows)**
   For inflows (purchases) and outflows (sales), the amortization is calculated **up to** the value date of the business transaction.

> **Note**
>
> The amortization step determines the amortized cost before the first business transaction of the day. If there are several position-changing business transactions on **one** day (where all business transactions have the same value date), the system calculates only **one** amortization.
>
> The fair value server bases its amortization calculation on the amortized cost that is saved in the value-date-dependent calculation base data for specific financial positions, and not on the *Amortization* key figure. This calculation base data is changed not only in the amortization step but also in the corresponding step for the position change so that the calculation base data is up-to-date for amortization on a later value date.

The most important business transactions relevant to amortization are listed below:

Disbursement/Repayment/Purchase/Sale

Procedure templates: 1101 to 1004

An amortization is usually executed for position changes of financial transactions and financial instruments. The system recognizes an initial purchase as a special case and does not carry out an amortization calculation, but initializes calculation base data in the amortization step. This calculation base data is adjusted accordingly in the step for the position change and is thus available for amortization at a later date.

Unscheduled Repayment

Procedure templates: 1103

The following options are available for processing an unscheduled repayment:

- An unscheduled repayment changes the position nominal value and the associated effective interest rate. You must calculate the amortization of the original financial position (before repayment).
- If the cash flow for a valuation according to IAS 39 AG8 changes, the system posts a catch-up adjustment of the amortized cost when it processes the business transaction. The effective interest rate remains constant here.

> **Note**
>
> The effective interest rate in the calculation base data for financial positions applies to the original position before the unscheduled repayment. Only a subsequent amortization calculation during a further position change or a key date valuation calculates a new effective interest rate based on the new target value (amortized cost amount on the date of the unscheduled repayment). You can control this using the indicator for recalculating the effective interest rate in the calculation base data for financial positions.

> **Caution**
>
> Note that the Source Data Layer (SDL) cash flow generation is essential to determining the future amortized cost. The cash flow generation must reflect unscheduled payments.
>
> If the repayment is supplied as a business transaction, you must ensure that the chosen item class of the SDL business transaction is taken into account in the cash flow generation process.

Scheduled Repayment

Procedure template: 1102

Securities Account Transfer

Procedure template: 1140

You must calculate an amortization of the source position and target position before the transfer posting.

Reclassification

Procedure templates: 1111

Rollover

Procedure template: 1105

The accruals/deferrals for the premium and discount are calculated for the previous interest rate condition period. The balance of the key figure for the deferrals is reduced to zero. The system updates any calculation base data for specific financial positions so that the amortization can be calculated for a subsequent valuation in the new interest rate condition period. This also applies to a valuation according to IAS 39 AG8.

Early Rollover

Procedure template: 1105

Settlement/Partial Close-Out/OTC Derivative

Procedure template: 1107 to 1109

These templates contain amortization steps and are used for the business transactions *Clear Deferral*, *Settlement*, and *Partial Close-Out*.

Condition Change

Procedure template: 1105

For IAS 39 AG8 and for significant modifications according to IFRS 9, processing is the same as for the *Rollover* business transaction.

For insignificant modifications according to IFRS 9, the system posts a catch-up adjustment of the amortized cost when it processes the business transaction. The effective interest rate remains constant here.

Follows the same procedure as the *Rollover* business transaction.

Activities

To define a global valuation environment, in Customizing for *Bank Analyzer*, choose *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Create Global Valuation Environments*.

You can assign a global valuation environment to an accounting system in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Basic Settings* > *Assign Global Valuation Environment to Accounting System*.

To determine the exchange rates for the currency translation in accounting, you must assign a market data area to the accounting system and the legal entity. You do this in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Assign Market Data Areas to Legal Entities and Accounting Systems*.

More Information

Calculation of Amortized Cost [page 636]

4.4.5.2.6.9.1.1 Amortizing Swaps

Use

> **Note**
>
> Before you amortize a swap, the first thing to consider is whether there are *upfront payments* for the swap and whether the swap is to be used as a hedging instrument in a hedging relationship. If this is not the case, amortization is not required.

Amortization is calculated using the effective interest method (see Calculation of Amortized Cost [page 636]). Interest rates are updated for variable transactions/swap legs based on the last fixing or the last known market data. In principle, the upfront payment is amortized using the fixed side of the swap.

The following figure illustrates the components of a swap:

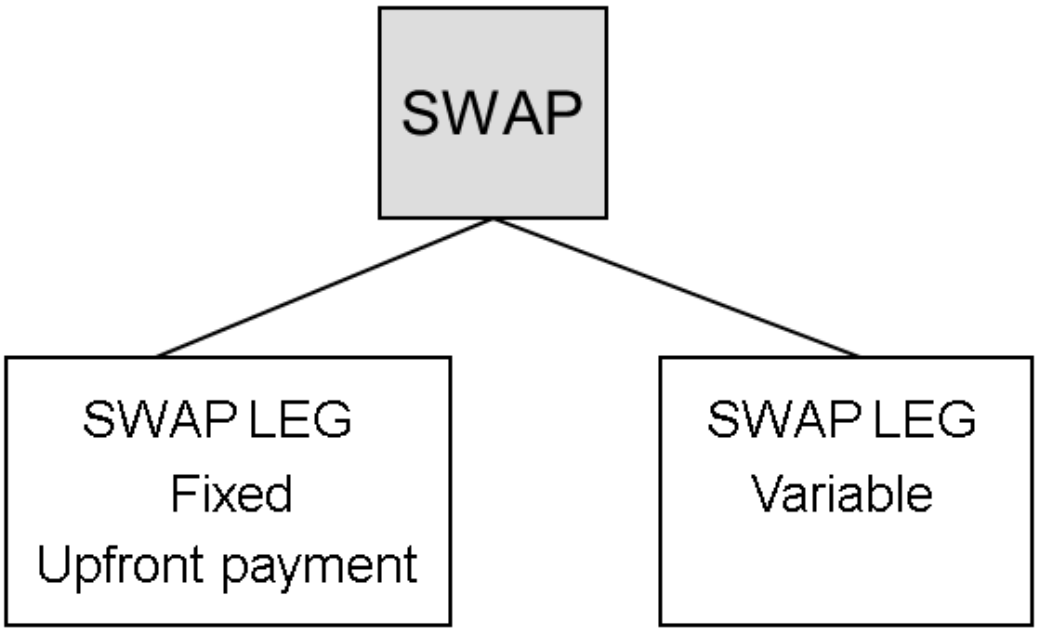


Figure: Components of a Swap

Procedure

Settings Required to Create a Financial Transaction

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Financial Transaction* > *Create Financial Transaction*:

- On the *Financial Position Management* tab page, enter **`Not Residual`** in the *Determination of Acquisition Value* column for both the fixed swap leg and the variable swap leg.
  Example: Accounting standard = **`XY`**; Generate FPO = **`Generate`**; Component class = **`Main Part`**; Determination of acquisition value = **`Not Residual`**
- Both swap legs are valuated. The fixed leg is amortized using the upfront payment.
- For the general swap node, also enter **`Not Residual`** on the *Financial Position Management* tab page.

Settings Required to Create a Business Transaction

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Business Transaction* > *Create Business Transaction*:

The nominal amount is supplied at swap leg level in the item data. In the business transaction, the upfront payment is to be entered at swap leg level in the item data in the position amount.

Forward Swap

The forward swap can be mapped as a regular swap.

4.4.5.2.6.9.2 Hedge Adjustment (Subledger Scenario)

Use

A *hedge adjustment* (book value component) is made for financial positions. The hedge adjustment only contains values if the hedging relationship concerned contains a position. If you generally work without IAS hedges, you do not need to read this document. The hedge adjustment is calculated using the difference between the fair value of the hedge and the amortized cost of the hedge. These two values are stored as auxiliary values and calculation bases for the position in the key date results (Results Data Layer).

Designation

Concurrent Designation

The (pro rata) acquisition value is applied to the fair value of the hedge and the amortized cost of the hedge.

Subsequent Designation

At the time of designation, the fair value is calculated on the basis of the source position. The fair value is transferred proportionally to both the fair value of the hedge and the amortized cost of the hedge (assuming you only hedge part of the position).

The system calculates the amortized cost of the hedge in a similar way to the amortized cost calculations. However, the target value is no longer the last amortized cost value. The value of the amortized cost of the hedge (initially the acquisition value or fair value) is now the last amortized cost value. The effective interest rate for the hedge is saved as a calculation basis in position management for the amortized cost of the hedge. You can check these values in the documents.

When you calculate the fair value of the hedge, the hedge spread is determined as part of the first valuation (as with the FI spread). This hedge spread remains constant over the entire term, and is also saved in position management. You can check these values in the key date results.

Dissolution

Depending on the settings in the Customizing activity *Derive Consequences of Position Changes*, scheduled and unscheduled position outflows can be made to dissolve the hedging relationship or can be distributed proportionally.

Calculation Procedures and Calculation Steps

Designation

To process the *Designation* business transaction for a hedging relationship, define a calculation procedure using the procedure template 2131 (*Hedge: Designation*). For concurrent and subsequent designations, create a calculation step with the category 2H01 and the calculation method MH01 (*Hedge: Designation*). Assign this calculation step to the calculation procedure.

Dissolution

If hedging relationships are dissolved, the system transfers the balance of the posting key figure *Hedge Adjustment* to the posting key figure *Hedge Adjustment to Be Dissolved*. For this purpose, define a calculation procedure with

the procedure template 2132 (*Hedge: Dissolution*). Define a calculation step with the category 2H02 and the calculation method MH02 (*Hedge: Dissolution*). Assign this calculation step to the calculation procedure.

Key Date Valuation

Procedure template: 2201 (*Valuation*)

The above-mentioned procedure template contains the calculation step category 2V01. The calculation method MV01 (*Adjustment - Hedge Adjustment*) is assigned to this calculation step category. This calculation method contains the standard procedure and the simplified procedure for hedge adjustment calculation. The characteristic /BA1/C55HSIMPR (Hedge Adjustment Procedure) determines which procedure is used by the system to calculate the hedge adjustment within this method. The value of this characteristic is specified in the Hedge Accounting Workplace before the designation process.

Activities

You assign one item type for credit and one for debit to a hedge business transaction type in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Hedge Accounting* > *Assign Item Types to Hedge Business Transaction Types*. You do not have to assign an item type to the hedge business transaction *Reject Designation*. The business transaction category *Reject Designation* is required because the required quantity/nominal value to be hedged may have been sold and would, therefore, no longer be available.

You define business transaction types and their business transaction categories in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Business Transaction Types and Item Types*:

- *Edit Business Transaction Types*
- *Assign Business Transaction Categories to Business Transaction Types*

For more information about the Customizing activities for dissolving micro fair value hedges, see Micro Fair Value Hedge (Subledger Scenario) [page 905].

4.4.5.2.6.9.3 Fair Value Calculation (Subledger Scenario)

Use

You use this function to calculate the fair value of an asset. A fair value calculation results in a write-up or a write-down.

Integration

The fair value calculation is triggered during a key date valuation.

Features

A fair value calculation can be configured by a calculation step in the non-value-date-based calculation procedures and can be executed for a valuation run of the key date valuation. Fair Value is calculated in the service functions (see Provide Fair Value [page 633] for more information).

The Price Calculator [page 547] can also be used here. In the templates for calculation procedures that are predefined by SAP, the fair value calculation always **follows** amortization. You can specify whether the valuation rule in the Fair Value Server is to use or the FI spread (retail spread) in the calculation procedure (see Activities).

Digression to Market Interest Rates/Financial Instrument Rates

The fair value is usually determined by discounting cash flows at market interest rates. With SAP, the yield curve/ instrument market data that is essential to determining the value of book value components is determined as follows:

1. The accounting system is assigned to a global valuation environment. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Basic Settings* > *Assign Global Valuation Environment to Accounting System*.
2. The market data area is assigned to a legal entity or to an accounting system. You do this in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Assign Market Data Areas to Legal Entities and Accounting Systems*.
3. You can define a derivation strategy for the valuation rule and determination of the market data set. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Create Global Valuation Environments*.
4. A valuation rule is found by means of the characteristics of the financial position and the derivation strategy valuation rule.
   You make the relevant settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivations* > *Determination of the Valuation Rule* > *Edit Rule Entries*.
   You enter the modules required (but not the derivation strategy) on the following selection screen. You can find them in the *Create Derivation Strategy* Customizing activity at the same level.
5. Using the valuation rule and existing characteristics (including the market data area), the system finds a market data set in Customizing with the help of the derivation strategy used to determine the market data set:
   You make the relevant settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivations* > *Determination of Market Data Sets and Spread Curve Types* > *Edit Rule Entries*.
6. The assigned rate types for the market data set found are available in Customizing for *Financial Services* under *Foundation* > *Market Data* > *Create Market Data Set*.
7. You can edit the market data by entering the market data area and the rate type. On the *SAP Easy Access* screen, choose *Financial Services* > *Foundation* > *Market Data*.

Calculating Without a Financial Instrument-Dependent Spread (FI Spread, retail Spread)

If you calculate without an FI spread, the cash flows are discounted based on the yield curve interest only. You can decide when you want to determine the fair value.

Fair Value Calculation of Loans of Holding Category LaR (Loans and Receivables) for the Notes

IAS32 requires you to calculate a fair value for loans for the enclosed report (notes). IAS39 is based on balance sheet valuations. Fair value valuations are usually not defined for LaR loans, which means no FI spread is determined within the AFI solution. Therefore, when you determine the fair value of loans for the notes, you can only determine the fair value by discounting the cash flows at market interest rates (that is, without an FI spread). If you want to calculate the loans with an FI spread (an internal bank spread), you can define a yield curve in the Source Data Layer (SDL) that corresponds to the market interest rate curve + x base points. By using valuation rule determination in risk basis, you can find this yield curve, and determine a fair value in accordance with it (that is, instead of basing this calculation purely on the market interest rate curve).

The same applies to the designation of LaR loans. The fair value that is calculated for subsequent designations as the basis for the *Hedge Amortized Cost* (HAC) and *Hedge Fair Value* (HFV) key figures when the designations are made is based on this yield curve.

Calculating with a Financial Instrument-Dependent Spread (FI Spread, Retail Spread)

If you want to use the fair value method (*Fair Value Adjustment*) using a financial instrument spread in Accounting, you must execute a calculation step [page 867] to adjust the fair value calculation base data (calculation templates 2101 and 2102) for every position change (including purchase/sale/repayment). If you do not do this, the system does not use the financial instrument spread in the calculation.

Although they are managed in Accounting, there are some products that are valuated using amortized cost, and not the fair value. To nevertheless report a fair value (fair value 32) in the notes of a financial statement, you can determine an FI spread at the start of a product's term. This is done using a calculation step for adjusting the fair value calculation base data (calculation templates 2101 and 2102). This step does not generate any posting document rows.

4.4.5.2.6.10 Update of Risk Provision and Write-Down

Use

The procedure templates, calculation step categories, and calculation methods described here are used to update the following impairment result categories for financial instruments and financial transactions that are carried at acquisition cost and amortized cost:

- Write-down
- Risk provision
  - Default risk (utilization)
  - Default risk (free line)
  - Transfer risk

This kind of update can be performed during a periodic valuation (key date valuation) or as a reaction to an analytical decision (impairment event). Accounting is informed of decisions of this kind by the impairment processes and they are processed in the *Update Secondary Business Transactions* process.

Accounting queries the relevant write-down and risk provision target balances in the Impairment Server.

When the default risk is updated, the following effects can be documented separately:

- Effect due to passage of time (unwinding)
- One-time effects due to a change in expectations or different counterparty behavior

Features

You can make settings for the update of risk provision and write-down by using the calculation step categories in the non-value-date-based calculation procedures delivered by SAP.

- Update during key date valuation
  You can use the relevant calculation step categories of calculation procedure template 2201 - *Valuation* to update the risk provision including unwinding.
- Event-driven update during processing of secondary business transactions
  You can use the relevant calculation step categories of calculation procedure template 2170 - *Update of Impairment* to update the risk provision including unwinding and adjust the write-down. For more information, see Reaction to Analytical Events During the Update of Secondary Business Transactions [page 793].
  You can also update the unwinding using the corresponding calculation steps of procedure templates 2150 Reclassification and 2111 Reclassification.

Activities

In Customizing for *Bank Analyzer*, you need to make the following settings so that Accounting can query values for write-down and risk provision in the Impairment Server:

- *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Accounting Processes* > *Valuation* > *Risk Provision* > *Assign Posting Key Figures to Impairment Results*
  For each accounting system and risk type, you assign a posting key figure to the possible combinations of impairment result category (risk provision or unwinding) and impairment exposure category (on-balance-sheet or off-balance sheet). For more information, see the system documentation for this Customizing activity.
  Note that you only need to make an assignment for risk provision; customers do not need to make an explicit assignment for the write-down.
  In the posting documents, the posting key figures assigned here are displayed by the key date valuation and the update of secondary business transactions as reference key figure `/BA1/C55REFKF` during the calculation of the risk provision and unwinding. The reference key figure can be used as input in deriving the posting key figure to ensure that the posting key figure `/BA1/C55ACCKF` matches the reference key figure.
  The derivation of the posting key figures must be consistent with this Customizing. This means that you derive the reference key figure as the posting key figure for the balance sheet side in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Derive Posting Key Figures*.
- *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Basic Settings* > *Assign Global Valuation Environments to Accounting Systems*
  You assign an Impairment Server environment that groups the required Customizing settings to an accounting system in order to calculate impairment results. This assignment connects Accounting and the Impairment Server so that the relevant key figures are determined in the same way for a given accounting system.
  If one of the calculation methods for updating the impairment results is listed in an accounting process for a specific accounting system, the system provides the assigned Impairment Server environment for the Impairment Server. For the risk provision update, it also provides the risk type and the impairment result category.

More Information

Procedure Templates (Not Value-Date-Based) [page 839]

Impairment Server [page 661]

Balance Offsetting [page 672]

4.4.5.2.7 Structured Products

This function of the accounting solution also covers the valuation for accounting purposes and mapping of structured products. Structured products are to be split in certain circumstances. The individual parts are then to be valuated and reported.

The accounting solution enables the user to create and identify such structured products and the associated business transactions, to value and post the individual components, and to assign the corresponding financial statement items.

You are responsible for delivering the relevant characteristics for structured products. Equally, it is your responsibility to identify the individual parts of the structured product (subcontracts) and to group them into units relevant for accounting (for example, Main Part and Split Derivative).

You can deliver business transactions at the level of the structured product and have the system process them using the accounting solution. To do this, a Business Transaction Split [page 903] is required that splits the individual business transaction (related to the structured product) into accounting-relevant units.

A further delivery scenario involves delivering the business transactions for the individual subcontracts directly from the operational system. It must be ensured that the business transactions directly address the financial positions, that belong to the subcontracts. The *split derivative/main part* characteristic in balance processing is used to group the relevant financial statement and profit and loss items, in line with splitting requirements.

4.4.5.2.7.1 Embedded Derivatives

Use

Embedded derivatives are financial instruments that can be combined with other financial assets or payables to form financial instruments (hybrids or structured products [page 900]) that are commercially distinct. As a result, part of the cash flow is subject to similar fluctuations as an independent derivative financial instrument.

Prerequisites

All data must be delivered to the Source Data Layer (SDL). The financial instruments (in terms of IAS) are split into their separate parts (subcontracts and basis products) using the transformation layer (which you have implemented). To do this, a structure of financial instruments is prepared using the transformation layer. This

structure illustrates how the compound instrument, its subcontracts and its basis products are formed. The subcontracts and basis products, as well as the structure information, are delivered to the SDL. This structure information is used to map the basis products and their relationships to each other in the SDL.

Business transactions are transferred in one of two ways: They are transferred as whole (not split) from the transformation layer to the SDL, whereby the IAS solution functions for splitting a business transaction are used. Alternatively, they can be delivered from the operational systems separately (split). However, the basis products must also be available separately in the SDL. Consequently, you must ensure that the business transactions also address the subcontract and basis product concerned directly.

Features

Distribution of the Acquisition Value

The entire structured product is represented in the system by a financial position. This financial position is addressed using external business transactions. Behind this financial position there are further financial positions that represent the subcontracts (components). The component transfer method is used to transfer the effects of the external business transactions on the individual components. The aim of this is to create positions for the individual components.

You make the settings for the component transfer method in Customizing for *Bank Analyzer*.

Analytics > *Accounting: Merge Scenario* > *Balance Analyzer* > *After Generation* > *Accounting* > *Processing for Business Transactions* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Component Transfer*.

With non-structured products, a position change takes place in clearing when you post external business transactions. When you update secondary business transactions [page 786] therefore, the posting is transferred from clearing to the position.

With structured products however, a position change takes place in clearing when you post external business transactions for the whole product. This position change in clearing is distributed to component clearing when you update secondary business transactions by means of the component transfer procedure. In addition to this, the transfer posting from clearing to the position is carried out for every individual component as part of the secondary business transaction update.

Fair Value Determination for Subcontracts for Initial Valuation

Business transactions for structured financial transactions and financial products can be delivered in Accounting for the compound instrument or for its components directly. If the business transaction is delivered for the compound instrument, the nominal amount and transaction value of the business transaction must be transferred to the components. The following different scenarios arise in the determination of the acquisition value for the residual component:

- **Currency swap transactions**: The acquisition value of the residual components is always determined using the functional currency for business and technical reasons.
- **Non-currency swap transactions**:
  - If the financial position currencies of the compound instrument and the non-residual component are the same, then the acquisition value of the residual component is calculated in this financial position currency, and is then converted into the financial position currency of the residual component.

- If the financial position currencies of the compound instrument and the non-residual component are the same, then the acquisition value of the residual component is calculated in the functional currency, and is then converted into the financial position currency of the residual component.

Fair Value Determination for Subcontracts for Subsequent Valuation

The service function determines the fair value separately for the individual elements (subcontracts and basis products) which form a complex contract (=compound instrument). This can be done by determining the fair value for the individual elements, or by carrying out a Residual Valuation.

Because of these options, fair value is determined as follows, taking the different currencies into account:

1. System check: Do fair values exist for Split Derivative and Main Part?
2. If no fair value is available for a split derivative or main part, the system checks if a fair value is available for the compound instrument. Beforehand, the system also checks if the residual valuation has been activated for a financial instrument in the fair value server.
3. If the fair value is available for the compound instrument and main part or split derivative then the residual amount is calculated for the missing fair value after the above mentioned residual valuation.
4. If no fair value can be determined in steps 1 through 3 the systems reports an error.

For more information about residual valuation, see Residual Method Pricing Model [page 644].

Grouping Subcontracts

The calculation bases for a product and its subcontracts are generated in Risk Basis and are transferred to Accounting in the form of a valuation structure. For more information, see the documentation for the price calculator [page 547]/SDL link mapper.

Managing Units Relevant to the Financial Reporting Process

IAS accounting manages every unit relevant to the financial reporting process as a financial position.

Splitting Business Transactions

You make the settings for splitting external business transactions [page 903] at contract level into business transaction items for units relevant to the financial reporting process. Under certain conditions when splitting, fair value is determined at the level of single position parts (subcontracts and basis products).

Example

When a compound instrument consisting of a basis contract (host) and a split derivative is acquired, the fair value determined is either that of the derivative that must be split, or that of the host part. The basis contract, or host, fair value is the result of the difference of the whole acquisition value (compound instrument) and of the fair value of the derivative.

Transactions in Multiple Currencies

Separate to the topic of splitting embedded derivatives, IAS 21 requires currency gain to be split from other net income components by firstly managing positions and P&L statement amounts in the original currency and then converting them at suitable rates into the functional currency. For transactions involving multiple currencies this means that position and net income components have to be managed separately for each currency. Thus, transactions in multiple currencies have to be broken down into multiple parts. For every currency there is exactly one financial position for the compound instrument. There can, however, be as many financial positions as you want for the main part components and the split derivative components.

4.4.5.2.7.2 Operational Business Transactions Split

Use

This report describes the processes used to determine and post the accounting effects on accounting-relevant entities for business transactions that are delivered for a structured product as a whole. Deriving the effects on individual invoice-relevant entities in this way is termed a business transaction split.

As an alternative to the business transaction split described, business transactions can be supplied in separated form and the corresponding financial positions can therefore be kept and addressed separately. In this way, the operational business transaction requires no special treatment in processing. If you post to a component of the compound instrument, you must specify its defining characteristics in the business transaction.

Activities

You define calculation procedures for the transfer to the components of a structured product (component transfer procedure) and assign this to calculation steps. Derive the component transfer procedure from key figures and characteristics of the external business transactions. To do this, make the following settings in Customizing for *Bank Analyzer*.

Processes and Methods > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Component Transfer* >:

- *Edit Component Transfer Procedure*
- *Derive Component Transfer Procedure*
- *Assign Rules for Residual Valuation*

More Information

Component Transfer Procedure [page 855]

Component Transfer Procedure for Structured Products with Split Derivative [page 857]

4.4.5.2.7.3 Business Transactions for Accounting Purposes

Use

The business transactions for structured products are described in a consecutive manner.

Features

Key Date Valuation

The process of key date valuation for structured products is the same as the usual process. The different units are managed separately for accounting. It is important to ensure that, where necessary, a residual procedure is used to value individual units.

Impairment

No special features are used by the system to react to an impairment because it can only relate to the main part. Derivative transactions (the split derivative in the case of structured products) are valued according to the International Financial Reporting Standards (IFRS) as affecting net income after fair value. This means that there is no need for this split derivative to take the impairment status of the contract into account. A change in value as a result of changes in credit standing are reflected in the balance and the profit and loss.

Hedge Accounting

The following applies for the interaction between embedded derivatives [page 900] and hedge accounting [page 904]: hybrid instruments must first be separated in accordance with IAS 39. The created components are treated like other instruments in hedge accounting.

Thus, hybrid instrument postings and financial reporting are not handled in a special way in hedge accounting. However, when it comes to valuation, some problems arise for fair value hedges with hybrid instruments whose split derivatives do not have to be valued separately (for example, caps/floors or notice rights/swaptions).

In hedge accounting, the change in value resulting from the hedged risk has to be determined for these instruments, as with all other underlying transactions.(hedge adjustment).

4.4.5.2.8 Hedge Accounting

The accounting solution supports hedge accounting for fair value hedging relationships with the following hedge categories:

- Micro Fair Value Hedge (Subledger Scenario) [page 905]
- Portfolio Fair Value Hedge (Subledger Scenario) [page 912]
- (Fair Value) Cross-Legal-Entity Hedging Relationships (XLE Hedges) [page 916]

The following hedge categories can be mapped in the accounting solution for cash flow hedging relationships:

- Macro cash flow hedge [page 920]
  - Hypothetical derivative approach [page 927]

In fair value hedging relationships assets or liabilities are hedged against changes in fair value; however, in cash flow hedging relationships cash flows are hedged against fluctuations. For both types of hedging relationships, the accounting solution supports hedging against the risk related to interest rate fluctuation. The accounting solution does not currently support hedges against other types of risk, such as currency risk.

The changes in value that arise as a result of hedging relationships are reported in the financial statement. IAS 39 requires proof that a hedge actually exists. This proof is provided by means of an effectiveness test.

4.4.5.2.8.1 Micro Fair Value Hedge (Subledger Scenario)

Use

Accounting supports you in hedge accounting for fair value hedges in line with IAS 39 in the phases of a hedging relationship.

Designation of Hedging Relationships

You create a hedging relationship in the Hedge Accounting Workplace [page 976] (HAW). Up-to-date information on available positions is transferred to the Hedge Accounting Workplace to aid the selection of hedged items and hedging instruments. Business transactions are generated from the Hedge Accounting Workplace when a hedging relationship is designated and are processed in Hedge Accounting. Positions for hedged items and hedging instruments that have not been used for hedging purposes until now are transferred into single positions for each hedging relationship. The positions are updated immediately after the business transactions have been transferred, ensuring the available information is up-to-date.

Position Management for Hedged Items and Hedging Instruments

Hedged items and hedging instruments are managed as separate financial positions that contain the SDL identification of a hedging relationship as an additional, defining characteristic. The following are managed as fixed, descriptive characteristics of a financial position: The category of the hedging relationship (for example, the micro fair value hedge), the hedge item category (hedged item or hedge instrument), and an indicator to show if a simplified procedure can be used. This characteristic information is used to control the accounting processes in different ways (for example, posting rules for business transactions, key date valuation).

By partially using positions for hedging relationships, positions that are assigned to hedging relationships and positions that are not used for hedging purposes can exist alongside one another. You can include individual positions in the processing of operational business transactions by making the appropriate settings in Customizing.

Effectiveness Test and Valuation of Hedged Items and Hedging Instruments

The results of the continuous hedging relationship effectiveness check are provided by the *Effectiveness Test* service function for accounting. The *Process Results of Fair Value Effectiveness Test* process retrieves hedging relationships that have been made ineffective for the relevant key dates, and generates business transactions to dissolve the hedging relationship. This process can be included as a step in the financial reporting process and is upstream of the key date valuation. When these business transactions are processed for accounting purposes, the hedged positions for all the financial positions concerned are dissolved. The Hedge Accounting Workplace is informed about all hedging relationships that are no longer effective.

You can reverse the editing of effectiveness test results.

Positions for hedging relationships are valuated in accordance with the hedge accounting rules. To save the valuation results, the additional IAS book value components Hedge Adjustment [page 895] and *Hedge Adjustment to Be Deferred* are managed for each financial position.

Dissolution of Hedging Relationships

The dissolution of a micro fair value hedge relationship can occur for the following reasons:

- A management decision
- Ineffectiveness
- Position outflow

The figure below shows the events that trigger the dissolution of a micro fair value hedge.

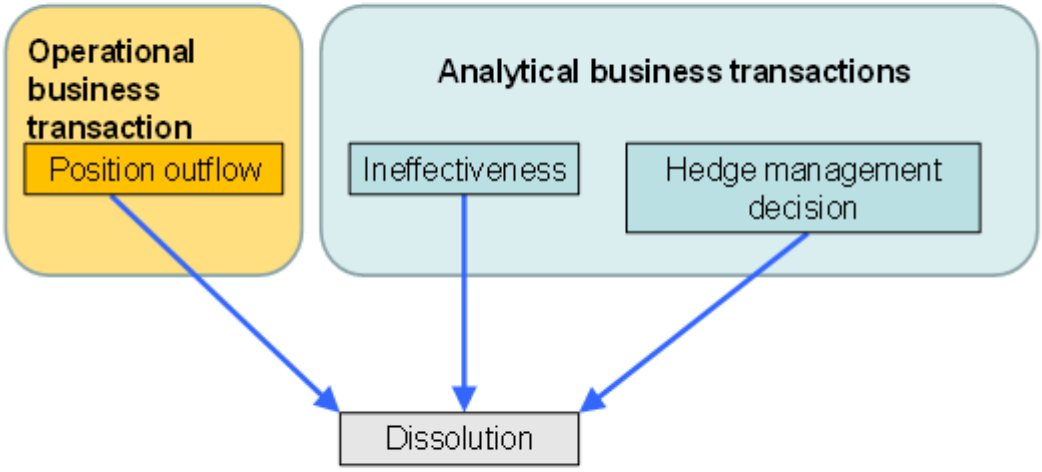


Figure: Events that trigger the dissolution of a micro fair value hedge

A negative effectiveness test result (ineffective) is an analytical event [page 689] within accounting. A hedging relationship can also be dissolved, for example, as a result of analytical events triggered by the Hedge Accounting Workplace, such as a management decision. In contrast, unplanned outflows caused by external business transactions of positions used in the hedging relationship are operational events. These externally triggered events are automatically processed for accounting purposes. In every case, the dissolution is documented by a dissolution business transaction. Operational events involve only secondary business transactions; analytical events also involve primary business transactions.

Irrespective of the triggering event , when these business transactions are processed, the current balance of the *Hedge Adjustment* key figure is tranfserred to the *Hedge Adjustment to Be Deferred* key figure, and the hedge-relevant target values are reset. Both of the updates are carried out for every hedged financial position. The Hedge Accounting Workplace is informed about the hedging relationships that have been dissolved and changes the status of the hedging relationship.

Features

Business Transaction “Designation”

An internal and analytical accounting designation business transaction is used to inform Accounting when financial positions are designated as the hedged items or hedging instruments of a micro fair value hedge.

Analytical hedge business transactions already contain the characteristics of the financial position. The Hedge Accounting Workplace passes the user-defined characteristics defined in Customizing to accounting.

The Hedge Accounting Workplace transfers the designated nominal amount to hedge accounting. In the Hedge Accounting Workplace, the amount is updated to the key figure /BA1/K55HNOM (*Hedged Nominal Amount*) in the hedged financial position. The system does not transfer postings from the unhedged to the hedged financial position. Therefore, the designation hedge business transaction has only one business transaction item type for the hedged financial position. The unhedged financial position remains unchanged.

The designation date of the hedging relationship defines the posting date transferred by the designation business transaction to hedge accounting.

The designation is posted in two stages. In the first step, the system creates a hedging financial position. In the second step, the hedge-relevant key figures, such as the hedged nominal amount, hedge fair value and hedge amortized cost, are updated to the Results Data Layer (RDL). The system updates all the business transactions of the financial positions concerned for the posting date of the designation business transaction.

The system sorts business transactions that have the same posting date as follows:

1. 0100 Changes to Characteristics
2. 0200 Operational Business Transaction
3. 0300 Operational Accruals
4. 0400 Operational Deferrals
5. 0510 Analytical Events - Dissolve/Designate Hedge - Outflow
6. 0520 Analytical Events - Dissolve Hedge - Management Decision/Ineffectiveness
7. 0530 Analytical Events - Designate Hedge - Management Decision
8. 0540 Analytical Events - Designate Hedge
9. 0600 Key Date Valuation
10. 0700 Analytical Deferrals
11. 0800 Transform and Move
12. 0900 Asset/Liability Transfer
13. 1000 Foreign Currency Valuation

Designation business transactions are processed **after** operational business transactions (such as purchases and sales) but **before** key date valuation business transactions. Dissolution and dedesignation business transactions are processed **before** designation business transactions. This means that you can transfer a transaction from a dissolved or dedesignated hedge to a new hedging relationship on the same day.

As part of the update, all the financial positions concerned are subjected to an availability check to ensure that nominal amounts are also available for the hedging business transactions. Further processing is dependent upon the result of this check:

- Rejection of Designation
  If the nominal value or amount to be reposted for at least one financial position is not available, the entire designation is rejected and a corresponding internal business transaction, *Rejection of Designation*, is created. A designation that has been rejected once remains unchanged when it is updated again.
- Designation
  The posted nominal amounts are transferred from hedged nominal amounts to the new hedge financial position.
- Updating the Designation Business Transaction
  If you update the designation business transaction again, the generated nominal transfer posting is reversed and new postings are generated in line with the result of the check.

Concurrent and Subsequent Designations

The system compares the start date of each transaction involved in the hedging relationship with the designation date. The tolerance value stored in Customizing is read, and the system determines the time of hedge designation (*Concurrent/Subsequent Designation* checkbox).

Based on this value, the designation calculation method automatically calculates the hedge amortized cost (HAC) and hedge fair value (HFV) using the following formula:

- If the hedge was designated concurrently: HAC = HFV = acquisition value
- If the hedge was designated subsequently: HAC = HFV = FFV (full fair value at the time the hedge was designated)

The hedge amortized cost and hedge fair value are used to calculate the hedge adjustment during the key date valuation and in the effectiveness test.

Special Case: Subsequent Designation During Initialization

When Bank Analyzer is initialized, it is possible to subsequently designate existing productive hedging relationships, effective hedging relationships, and hedging relationships not yet valuated in accordance with IAS. A subsequent designation is designated after the business transactions have been initialized. This subsequent designation calculates a hedge adjustment of zero. The fair value of the hedge is set as equal to the full fair value, and the amortized cost of the hedge is set as equal to the amortized cost. In the later key date valuation the hedge adjustment is calculated as normal using the formula HFV - HAC = HADJ.

The subsequent designation date is not the designation date from the source system but rather the designation/posting date from the Hedge Accounting Workplace.

Example

If you want to create a financial statement for December 31, you post the initialization business transactions for the financial instruments on December 29 and run the subsequent designation on December 30. On December 31, you can run a key date valuation that calculates the hedge adjustment and posts it to the profit and loss account.

The system evaluates the *Concurrent/Subsequent Designation* indicator value and calls the calculation method MH01 for every financial position, both for subsequent and concurrent designation.

Business Transaction “Reversal of Designation”

A designation business transaction can be reversed by a reversal business transaction in the Hedge AccountingWorkplace. The system reverses the designation in two steps. In the first step, the posting of the primary designation business transaction is reversed. The financial position concerned is selected for updating. In the second step, those hedge-relevant target values that were previously updated by the designation are reset and updated by the business transactions. The reversal postings are made on the original posting date of the designation. The system updates all the business transactions affected by the reversal as of this date.

Business Transaction “Dissolution”

Dissolution by Analytical Events

To dissolve a hedging relationhsip, the system generates an internal dissolution business transaction and forwards this to Hedge Accounting for posting. The business transaction triggers the resetting of hedge-specific calculation base data and the transfer posting of hedge ajustment balances to the remaining valuation items to be deferred (hedge deferral). When the dissolution business transactions are processed for accounting purposes (derivation), the accounting process sends a dissolution event to the HAW, which leads to a change in the status of the hedging

relationship. This event is documented on the *Events* tab page for the hedging relationship. The business status of the hedging relationship changes from *Designated* to *Dissolved*. The technical status makes a distinction between dissolution due to a management decision and dissolution due to position outflow.

The dissolution is posted in two stages.

1. In the first step, the business transaction is posted to clearing key figures. The financial position concerned is selected for updating.
2. By updating the business transactions in the second step, all of the business transactions from the financial positions concerned are recalculated for the dissolution date. For a dissolution due to ineffectiveness, the accounting process *Update Secondary Business Transactions* performs the update for the *Edit Effectiveness Results* process. For a dissolution due to management decision, the Hedge Accounting Workplace performs the update directly.

Dissolution by Operational Events

Position Outflow

A position outflow caused by an operational business transaction reduces the financial position identified by the characteristics in the business transaction. You can confgure the effect on each of the unhedged and hedged single positions in the Customizing activity *Derive Consequences of Position Changes* as follows:

- Distribution of the business transaction to individual financial positions
  The position outflow is distributed to all positions that exist for the financial position on a pro rata basis. Both positions for hedging relationships and the unhedged position are reduced proportionally. You can use this procedure, for example, to map a scheduled repayment. The hedge-relevant posting key figures and calculation base data are updated.
- Dissolve the positions managed for each hedging relationship
  After the outflow of positions from the unhedged position has been posted, the system checks whether a sufficient number of positions (including those in hedging relationships) is available. If the system finds that there are not enough positions, the system continues to dissolve hedging relationships until a sufficient number of positions exist for the position without reference to hedging relationships. The calculation procedure assigned for hedge dissolution updates the hedge-relavant posting key figures and calculation base data. You can determine the sequence in which hedging relationships are dissolved by making the appropriate settings in the Customizing activity *Assign Dissolution Strategies to Accounting Systems*. The LIFO and FIFO methods are offered as standard dissolution strategies.

Even if the setting is *Dissolution of Hedge Positions*, the hedge positions remain unchanged if there are enough unhedged positions. The setting *Ignore Hedge Positions* is therefore irrelevant in the case of position outflows.

Position Inflow

A dissolution cannot be triggered based on a position inflow A position inflow caused by an operational business transaction increases the financial position identified by the characteristics in the business transaction. You define the effects on the hedge positions assigned to the unhedged position in the Customizing activity *Derive Consequences of Position Changes*: The position inflow can either affect only the unhedged position or be distributed equally between unhedged and hedged positions. The hedge-relevant calculation base data is updated.

The reversal of a position inflow that has a later posting date than that of the original document works in the same way as a position outflow. You make settings for the derivation of a reversal with a later posting date in the Customizing activity *Derive Consequences of Position Changes*. For more information, see Reversal [page 795] and Update Secondary Business Transactions [page 786].

Business Transaction “Reversal of Dissolution”

Reversal of the Dissolution by Analytical Event

You can reverse a dissolution business transaction delivered by the Hedge Accounting Workplace using a reversal business transaction. The HAW triggers the generation and processing of the reversal business transaction. The reversal is posted in two stages.

1. In the first stage, the posting of the dissolution business transaction to the clearing key figures is reversed. The financial position concerned is selected for updating.
2. By updating the business transactions in the second step, the hedge-relevant target values that were previously updated by the dissolution are reset and the derived postings reversed. The reversal postings are made on the original posting date of the dissolution. The system updates all the business transactions affected by the reversal as of this date.

You can also reverse a dissolution business transaction that was triggered due to the results of the effectiveness test. For more inofrmation see *Reverse Editing of Effectiveness Test Results* in the *Processes* section.

Reversal of the Dissolution by Operational Event

When it reverses the operational business transaction due to a position outflow, the system also reverses all the documents previously generated during the derivation of this business transaction. The system resets the hedge-relevant calculation base data. Information is sent from Accounting to the Hedge Accounting Workplace by means of a reversal event and the hedge relationship is automatically set to the status *Designated*. If the posting date of the reversal business transaction is after the posting date of the original business transaction, the status of the hedging relationship remains *Dissolved.*. You can then redesignate the hedging relationship in the Hedge Accounting Workplace on the posting date of the reversal business transaction.

Processes

Process: Key Date Valuation

The key date valuation of financial positions in hedging relationshsips results in an adjustment of the hedge-relevant book value components and involves the following steps:

- The calculation procedure is selected in accordance with the characteristic value (Holding category)
- The valuation-relevant key figures are provided for financial positions (for example, *Hedge Fair Value*, *Hedge Amortized Cost*)
- The book value components are updated (for example, *Hedge Adjustment* and *Hedge Adjustment to Be Deferred* (Hedge Adjustment Deferral).

As part of the key date valuation, a *key date valuation* business transaction is generated to update the book value components of a position with the following possible items:

- Hedge adjustment write ups/downs
- Hedge adjustment write ups/downs

The following valuation systematic can be used to value the financial positions of a fair value hedging relationship:

- Hedging Instruments
  Designating a fair value hedging relationship does not result in a change in the *Full Fair Value* valuation approach used in a key date valuation.
- Hedged items
  Designating a fair value hedging relationship results in a change in the valuation approach used in a key date valuation. To take the change in value caused by the hedged risk into consideration, the additional value component, hedge adjustment, is managed.
  The simplified procedure can be used for hedged items in perfect one-to-one hedging relationships. As a controlling measure, a fixed descriptive characteristic in the financial positions of the hedged items can be

used as a variable to determine the calculation procedure. When the hedge adjustment is calculated for the hedged item following the simplified procedure, the change in the value of the hedged item (caused by the hedged risk) is assumed to be equivalent to the change in the value of the hedging instrument.

Process: Process Results of Fair Value Effectiveness Test

The Fair Value Effectiveness Test [page 1039] service function provides hedge accounting with the results accepted in the results database for effectiveness measurement. As part of the process for determining the effectiveness for Accounting, the results are flagged to show if they have already been used within another key date valuation. For more information about the effectiveness test, see Fair Value Effectiveness Test for Hedging Relationships [page 1036].

The process is carried out in the following stages:

1. An inquiry is made to the results database about ineffective hedging relationships from the test result accepted for the current key date.
2. Internal primary dissolution business transactions are generated for all financial positions for the hedging relationships that have become ineffective.
3. Dissolution business transactions are processed for accounting purposes (see above).
4. The financial positions concerned are selected for update as of the date of the dissolution.

"Reverse Editing of Effectiveness Test Results" Process

The process is carried out in the following stages:

1. The processed effectiveness test result for the key date selected are queried
2. Reversal business transactions are created for the dissolution business transactions that were generated in the *Process Results of Fair Value Effectiveness Test* process.
3. The financial positions concerned are updated on the reversal date in the *Update Secondary Business Transactions* process.

Activities

You assign one item type for credit and one for debit to a business transaction type in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Hedge Accounting* > *Assign Item Types to Hedge Business Transaction Types*.

You can find the settings listed below in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Hedge Accounting* > *Processing of Position Changes*:

- You can define your own strategies for dissolving hedge positions or use the FIFO and LIFO dissolution strategies we provide. You assign these strategies to an accounting system. To do this, choose *Define Dissolution Strategy for Hedge* and *Assign Dissolution Strategy to Accounting System*.
  You need to do this if the quantity of the unhedged financial position does not suffice for a financial position outflow, which means that any hedged positions must be dissolved. If several hedging relationships exist, you must use a strategy to determine which hedged positions have to be dissolved. We provide the following default dissolution strategies that are based on the designation day:
  - LIFO: Last In First Out (dissolves those hedges most recently designated for a position)

- FIFO: First In First Out (dissolves the longest standing hedges for a position)

- In the Customizing activity *Assign Dissolution Method to Accounting System*, you assign a calculation procedure and a derived business transaction type for dissolving hedging relationships to an accounting system.
- In the Customizing activity *Derive Consequences of Position Changes*, you define how financial position changes affect hedge positions.

If you want to define and use your own dissolution strategy, you must create your own BAdI implementation (transaction SE19) for the Business Add-in (BAdI) `/BA1/B1_DISS_ORD` (BAdI definition). Use the dissolution strategy you defined in Customizing as a filter value.

For more information, see the documentation for the relevant Customizing activity.

You start processing the analytical event *Hedge is Ineffective* on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Automatic Processing* > *Process Results of Effectiveness Test*.

To reverse the process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Automatic Processing* > *Reverse Editing of Effectiveness Test Results*.

4.4.5.2.8.2 Portfolio Fair Value Hedge (Subledger Scenario)

Use

This function supports the mapping of portfolio fair value hedges. Portfolios of hedged items to be hedged are defined by the hedge processes [page 975] and the Hedge Accounting Workplace [page 976] (HAW).

The portfolio is used for determining the amount for designation (in other words, the amount to be hedged and its percentage) instead of individual hedged items from the portfolio. The portfolio represents the hedged item of the hedging relationship. A portfolio can be held in one currency only. It cannot simultaneously contain contracts from the holding categories *Available for Sale (AfS)*and *Loans and Receivables (LaR)*. Always hold AfS contracts in separate portfolios.

For an overview of the chronological sequence of activities involved in processing portfolio fair value hedges, see the documentation about the Hedge Accounting Workplace (HAW) under Portfolio Fair Value Hedges [page 1005].

Processing Analytical Events in Financial Position Management

Both operational events, such as position outflows, and analytical events [page 689], such as a hedge management decision, result in the dedesignation, redesignation or dissolution of a portfolio fair value hedge:

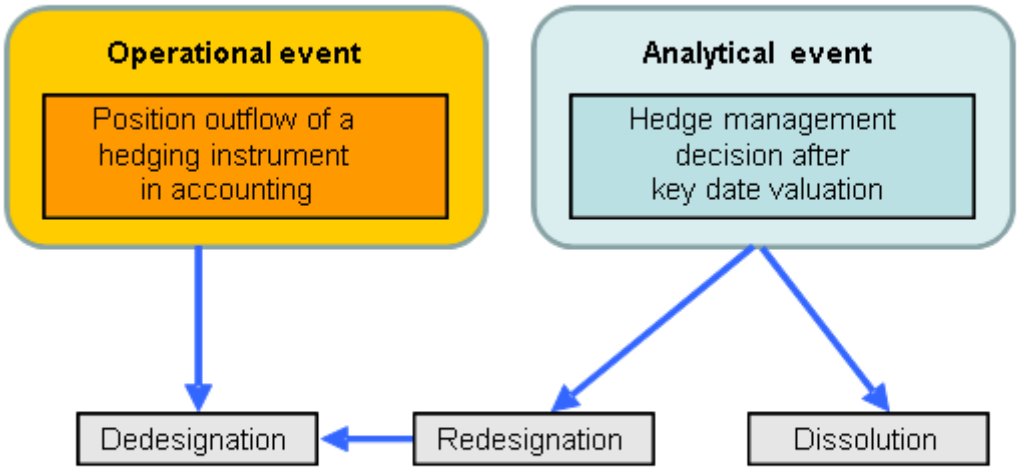


Figure: Events that trigger the dedesignation, redesignation, and dissolution of a portfolio fair fair value hedge

Initial Designation

Designation of hedging instruments and of portfolio positions takes place in the Hedge Accounting Workplace, in accordance with the amounts to be hedged, and is valid until the end of the valuation period. The HAW generates an internal primary business transaction *Designation* and automatically triggers the Update of Secondary Business Transactions [page 786]. When the secondary business transactions are updated, the system derives an internal business transaction of the type *Designation*. The business transaction items relate to the hedging instruments and the time bucket objects.

- Hedging Instruments
  The hedging instruments are designated individually, as in the macro cash flow hedge or the micro fair value hedge. However, the designation of the portfolio fair value hedges is not rejected, as is the case with the designation of micro fair value hedges. An event is used to inform the HAW of the amount available at the time of designation. The HAW updates the designated amount accordingly. You use the calculation procedure template 2131 - *Hedge: Designation* to derive the designation of hedging instruments..
- Time bucket objects
  When the initial primary designation business transaction is posted, time bucket objects are created according to the portfolio positions to be hedged.

Editing of Effectiveness Test Results

You must perform the effectiveness test [page 1036] before the key date valuation. During the key date valuation, the system requires the results of the effectiveness test in order to determine whether it needs to post a new hedge adjustment [page 895] for each time bucket object. If no effectiveness test results are available, the system automatically assumes that the portfolio fair value hedging relationship is effective. The results of the effectiveness test are saved in the RDB (Result Database).

The results of the effectiveness test and micro fair value hedges are processed in different ways. This is because a portfolio fair value hedge may not be dissolved if it is determined to be ineffective. If the system determines that

the hedge is invalid, all time bucket objects remain unchanged. However, the relevant hedging instruments are dedesignated. When this is done, they are removed from the portfolio fair value hedging relationship.

Key Date Valuation

Each hedging period is closed by defining a period end key date in the HAW. The hedging period for the portfolio fair value hedge does not necessarily have to correspond with the valuation period for accounting. After the period end key date has been defined, you can trigger the period end run for portfolio items directly from the HAW (see Portfolio Item Runs [page 1172]). The results of this process are used as the basis for the calculations during key date valuation for time bucket objects and hedging instruments. These are described below:

- Time bucket objects
  The key date valuation procedure is derived from the characteristics for time bucket objects. A separate process is available for the key date valuation for time bucket objects of a portfolio fair value hedge. You can use the calculation procedure template 2202 - *Valuation - Time Bucket Objects* with calculation steps [page 867] for this. The key date valuation results in an adjustment of the hedge-relevant book value components of the individual time bucket objects and involves the following steps:
  - Accrual/deferral of hedge adjustment to be deferred
  - Reduction of hedge adjustment to be deferred through unplanned position outflows
  - Adjustment of hedge adjustment for current period
  - Adjustment of the fair value adjustment for portfolio fair value hedges that hedge hedged items in the holding category *Available for Sale* (AfS correction posting)

  For more information, see Measurement of Time Bucket Objects [page 942].
- Hedging Instruments
  You can use the calculation procedure template 2201 - *Valuation* for the key date valuation for hedging instruments of a portfolio fair value hedge. The key date valuation results in an adjustment of the hedge-relevant book value components. For more information, see Key Date Valuation [page 934].

During the key date valuation process for time bucket objects, an event is sent to the Hedge Accounting Workplace informing it that the period end run is complete. The system automatically creates a new version of the hedging relationship with the status *to be designated* in the HAW. At the start of the new valuation period, the user has to perform a redesignation of the hedges manually in the HAW.

Redesignation

At the start of the new hedging period, the portfolio fair value hedge is redesignated due to a decision by the user in the HAW. The net positions and thus the relevant amount to be hedged have to be determined again. The positions and the hedged items from the portfolio can have changed as a result of early payments and sales. The redesignation takes place in two steps: dedesignation and a renewed designation. These are processed in Hedge Accounting by the corresponding analytical business transactions. These internal business transactions are automatically generated as primary business transactions from the HAW and derived as secondary business transactions. The business transaction items relate to the hedging instruments and to the time bucket objects. The processing differs for the hedging instruments and the time bucket objects.

- Hedging Instruments
  All hedging instruments are first dedesignated and then redesignated if the user assigns them to the volume to be hedged in the HAW. The same calculation procedures are used for hedging instruments as those used for dissolution and designation.
- Time bucket objects
  When the *Dedesignation* secondary business transaction is updated, the system calls a calculation procedure that transfers the hedge adjustment balances to the *Hedge Adjustment to Be Deferred* component. You use the calculation procedure template 2132 - *Hedge: Dissolution* for this.. When the *Designation* secondary business transaction is updated, the system calls a calculation procedure that redistributes the hedge adjustment

balance to be deferred from previous periods. This redistribution can occur because the distribution of the hedged volumes across the time buckets has changed due to a shift in the maturity band. You use the calculation procedure template 2133 - *Hedge: Redesignation Time Bucket Objects* for this.

For more information, see Redesignating and Dissolving Portfolio Fair Value Hedges [page 1009].

Dissolution

Dissolution occurs due to a decision by the user. The Hedge Accounting Workplace generates the primary analytical business transaction *Dissolution*, and automatically triggers the update of secondary business transactions. As a result an internal *Dissolution* business transaction is derived that applies to the hedging instruments and time bucket objects of a portfolio fair value hedge. The HAW supplies Accounting with the information required for the dissolution of a hedging relationship. You use the calculation procedure template 2132 - *Hedge: Dissolution* for this.

- Hedging Instruments
  All hedging instruments are dissolved using the same calculation procedure, as in the case of micro fair value hedges [page 905].
- Time bucket objects
  After dissolution, the portfolio remains in the form of time bucket objects. The balance for the *Hedge Adjustment* key figure is transferred to the *Hedge Adjustment to Be Deferred* key figure.

for the following key date valuations, no new hedge adjustment is calculated. A linear amortization of the hedge adjustment takes place in accordance with the maturity of the time bucket and the contract end date of the hedging instrument.

Closeout

The position outflow from the operational business transaction Close-Out does not dissolve the entire portfolio fair value hedging relationship (as done in the case of micro fair value hedges). A close-out brings about the dedesignation of individual hedged instruments. This is then communicated to the Hedge Accounting Workplace. in the HAW, the hedging instruments affected are then removed from the hedging relationship.

> **Note**
>
> After you have set the SDL time stamp for the relevant legal entity, you update the secondary business transactions **before** period-end processing. This ensures that the portfolio item calculation runs with up-to-date data.

Reversal

A reversal is possible for all analytical process steps. The system does not perform the reversal or the reposting automatically. You have to trigger this process manually. Starting with the last process step that was performed, you can reverse the steps performed prior to it one after the other and individually in reverse chronological order. The following restrictions for reversals apply, for example:

- You can only reverse a key date valuation for time bucket objects if no redesignation has taken place in the subsequent hedging period or if this has already been reversed.
- You can only reverse a redesignation if no key date valuation for the time bucket objects has taken place in the current hedging period or if this has already been reversed.
- You can only reverse a dissolution if all subsequent key date valuations for the time bucket objects have been reversed.

The reversal function described under Micro Fair Value Hedge (Subledger Scenario) [page 905] is used for the reversal of a designation or a dissolution.

When the *Reverse Editing of Effectiveness Test Results* process is reversed, the dedesignation is also reversed.

More Information

Procedure Templates (Not Value-Date-Based) [page 839]

4.4.5.2.8.3 Cross-Legal-Entity Hedging Relationships (XLE Hedges)

Use

This function supports the mapping of hedging relationships between legal entities (commonly referred to as cross-legal-entity hedges or XLE hedges), which contain hedged items and hedging instruments from multiple legal entities [page 696]. An XLE hedging relationship is created by the Hedge Accounting Workplace (HAW) [page 976].

> **Note**
>
> - The system does not provide any consolidation functions. Consolidation is the task of downstream systems, such as SEM-BCS.
> - Parallel value maintenance is not carried out, which means there is no parallel closing for groups and legal entities with varying values.
> - The AFI solution maps only fair value hedging relationships between legal entities.

Figure 1 provides an example of XLE hedges:

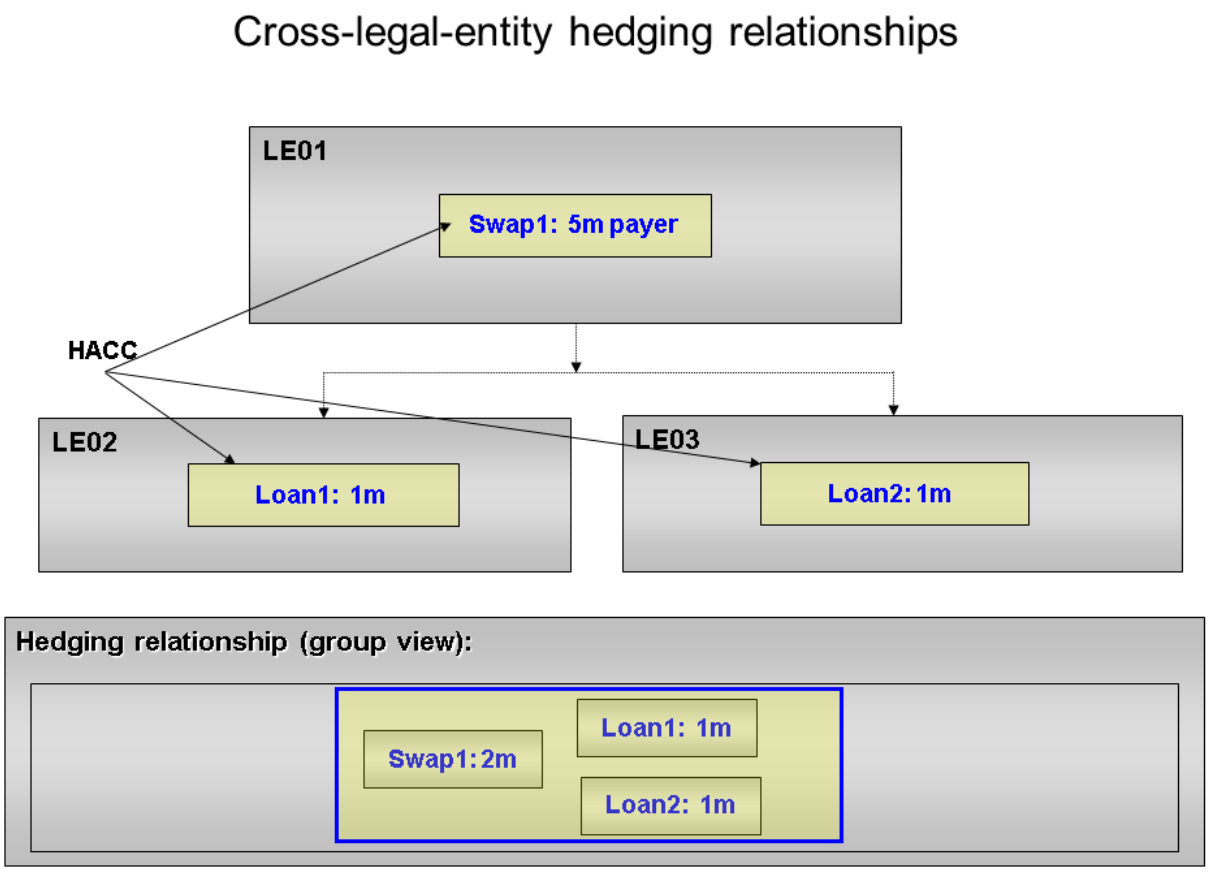


Figure 1: Cross-Legal-Entity Hedging Relationships

Legal entities `LE02` and `LE03` hedge against interest rate risk for *Loan1* and *Loan2*. This is done using *Swap1* in legal entity `LE01`. *Loan1* and *Loan2* are valued according to amortized cost adjusted by hedge (ACHA). This leads to fluctuations in the P/L statement at legal entity level.

The posting results of the legal entities involved have to be shown at group level in accordance with hedge accounting (HACC).

Features

Mirror Transactions

The AFI solution does not require you to include mirror transactions in cross-legal-entity hedging relationships. Reporting at group level is correct even without mirror transactions. However, this may result in unwanted P/L fluctuations at legal entity level.

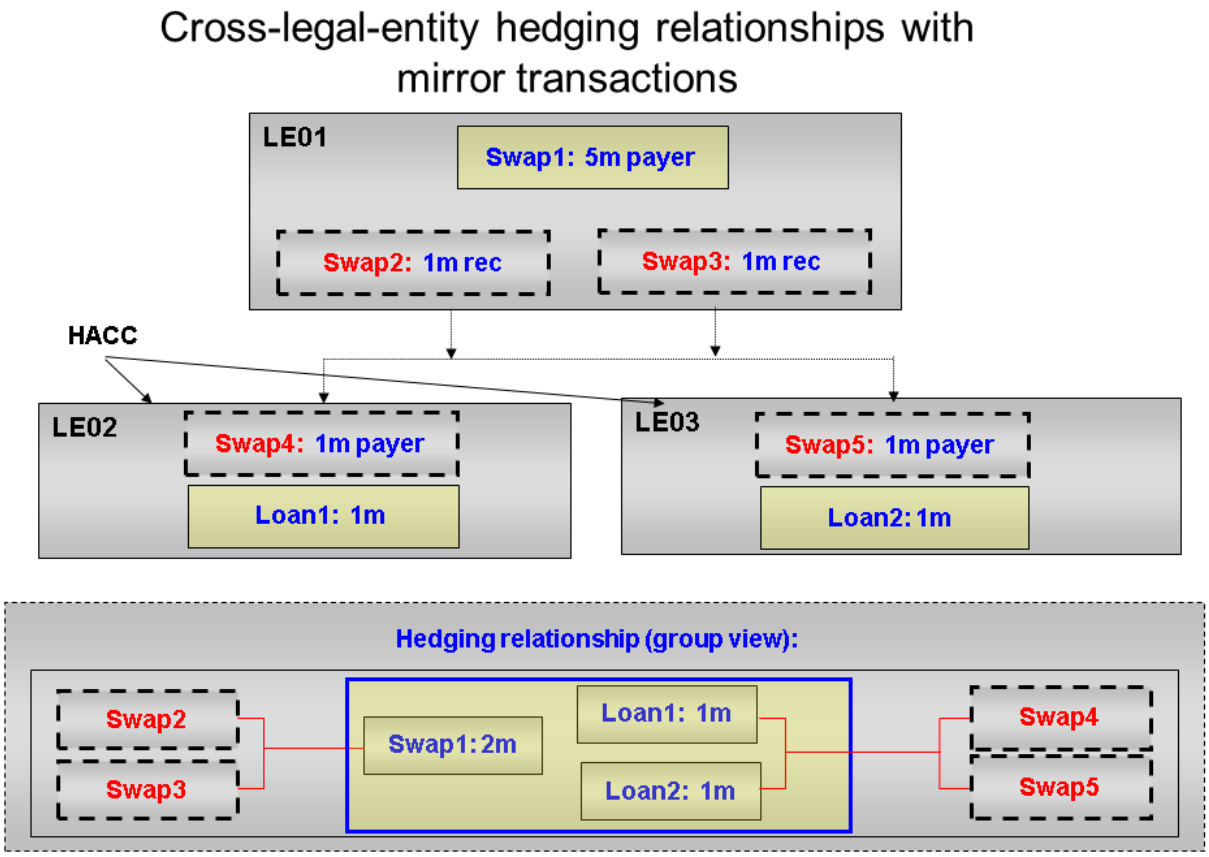


Figure 2: Cross-Legal-Entity Hedging Relationships with Mirror Transactions

Explanations for Figure 2

Swap2, *Swap3*, *Swap4*, and *Swap5* are the mirror transactions. *Loan1* and *Loan2* are no longer valued according to amortized cost, but rather according to amortized cost adjusted by hedge (ACHA). Amortized cost adjusted by hedge is offset by mirror transactions to avoid P/L fluctuations at legal entity level.

Legal entity `LE02` closes payer *Swap4* by means of `LE01` in order to hedge *Loan1* against interest rate changes and avoid fluctuations in the P/L of `LE01`. In `LE02`, the P/L effects from *Swap4* are set against the hedge adjustment from *Loan1*. This enables correct reporting at `LE02` level. Legal entity `LE01` closes receiver *Swap2* by means of `LE02`.

Legal entity `LE03` closes payer *Swap5* by means of `LE01` in order to hedge *Loan2* against interest rate changes and avoid fluctuations in the P/L of `LE03`. In `LE03`, the P/L effects of *Swap5* are set against *Loan2*. In `LE03`, the P/L effects from *Swap5* are set against the hedge adjustment from *Loan1*. This enables correct reporting at `LE03` level. Legal entity `LE01` closes receiver *Swap3* by means of `LE03`.

Even when internal transactions are taken into consideration, no valid hedging relationship exists in `LE01`. The P/L effects of *Swap1* are partially offset by *Swap2* and *Swap3*. The pooling of swaps in `LE01` occurs without special treatment: the external *Swap1* and the mirror transactions *Swap2* and *Swap3* are valued according to full fair value. The pooling of swaps and loans in `LE02` and `LE03` results from the fact that, in accordance with XLE hedging, hedge adjustment is taken into account in the individual financial statement for the loans.

> **Note**
>
> The mirror transactions in figure 2 are, seen from the perspective of Accounting, regular external transactions at legal entity level. This is because no information about the consolidation group exists on this level. Modeling such mirror transactions helps to prevent volatility resulting from XLE hedging from appearing in the P&L of the individual financial statements for legal entities.

Prerequisites for Mirror Transactions

When posting mirror transactions, the following must be taken into consideration: An offset hedge result will occur at legal entity level only when mirror transactions are posted in accordance with hedge accounting.

This is achieved by enabling the mirror transactions to be included in the XLE hedging relationship. When it is designated, the XLE hedging relationship must be entered completely, that is, existing mirror transactions have to be included. The mirror transactions are to be identified using a characteristic (see the *effectiveness test* section). They are not considered in the effectiveness test because effectiveness is checked at group level only. Hedge-relevant key figures for the mirror transactions are determined and posted to as usual.

In addition, to ensure that the results are shown correctly at the individual company level for legal entities that contain only derivatives for the XLE hedging relationship (`LE01` in figure 2), the following procedure should be chosen:

In individual financial statements, the hedge results from derivatives belonging to XLE hedges should be assigned to report items that are shown as the normal trading result rather than the hedge result. Separate functions are not provided for this purpose, however, since this can be achieved using the standard financial statement determination [page 1451] procedures if the appropriate Customizing settings (characteristics and assignment rule sets) have been made.

Note that this concept does not stipulate that mirror transactions have to exist. Reporting at group level is correct without mirror transactions. However, for the individual LEs, P&L effects arise that are not economically feasible.

Business Transactions

XLE hedges are created, managed, and dissolved in the Hedge Accounting Workplace. Primary business transactions are created here. For more information, see Cross-Legal-Entity Hedging Relationships (Subledger Scenario) [page 1016].

Effectiveness Test

Effectiveness is measured at group level. If a hedging relationship contains mirror transactions, these are not taken into consideration during the effectiveness test [page 1036]. For this to happen, it is necessary to identify mirror transactions as mirror transactions. This takes place in the secondary data source of the worklist, and is achieved using the value `M` for the characteristic *Mirror Transaction Indicator* `/BA1/CH0FTMIRR`. The results of the effectiveness test are passed to Accounting.

The effectiveness test for XLE hedges is called during the effectiveness test for legal entities entered on the *Basic Data* tab page in the Hedge Accounting Workplace.

- Prospective Effectiveness Test
  If the effectiveness test is called **before** designation, it queries the XLE hedging relationships with the business status *Pending* in the HAW.
- Retrospective or Prospective Effectiveness Test
  If the effectiveness test is called **after** designation, it queries the XLE hedging relationships with the business status *Designated* in the HAW.

The results of the effectiveness test are saved for further processing in Accounting for the legal entity entered on the *Basic Data* tab page. The system queries the results of the fair value effectiveness test once each period. It does so using the following criteria:

- Accounting system
- Legal entity
- Key date

Key Date Valuation

Key date valuation and posting of relevant financial positions takes places only if an XLE hedge has the status *Designated*.

For financial positions in hedging relationships, you can assign appropriate posting rules and valuation procedures to hedge accounting in Customizing. These can also be used for financial positions in XLE hedges.

You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Accounting Processes* > *Valuation*.

> **Note**
>
> You **cannot** use the *simplified procedure*, which is controlled using characteristic `/BA1/C55HSIMPR` *Procedure for Determination of Hedge Adjustment*, for XLE hedges (see Characteristics in Calculation Management (Subledger Scenario) [page 878]).

4.4.5.2.8.4 Macro Cash Flow Hedge

Use

This function represents the *cash flow hedge* hedging relationship type in accordance with IAS 39. In this type of hedging relationship, cash flows are hedged against fluctuations caused by interest rate risks. This has no effect on the way hedged items are processed from an accounting perspective. Only the hedging instruments are processed differently. The change in the value of the hedging instrument (caused by the hedged risk) is determined and recorded in the Hedge Adjustment book component (see Hedge Adjustment (Subledger Scenario) [page 895]).

For more information about the macro cash flow hedge supported and its role, see Cash Flow Hedge Analysis [page 1107].

Features

Processing Business Transactions in Financial Position Management

Designation

Accounting is informed about the designation of a position as a hedging instrument for a macro cash flow hedge by means of the internal business transaction *Designation*. This business transaction triggers the transfer posting

of position-related key figures from a position that is not used for hedging purposes to a position that is managed for each hedging relationship individually

Process Overview: "Generation of Hedging Relationships/Designations"

In the first step, Hedge Management [page 976] generates an initial hedging relationship in the SDL. The selection conditions for the hedge items are defined as part of this process. The initial hedging relationship does not contain any hedging instruments at this point.

On the basis of this hedging relationship, the user can use the Cash Flow Hedge Analysis [page 1107] component to generate a proposal for a macro cash flow hedging relationship (a hedge proposal). Potential hedging instruments from the SDL are used to do this.

Hedge Management then uses this proposal to generate the *Designation* business transaction. The system only includes those positions that are not currently used in Accounting in a hedging relationship. The business transactions are then transferred to Accounting for posting, and the hedging relationship in the SDL is updated.

The updated hedging relationship is based on the cash flow hedge analysis in the following analyses.

First Designation/Redesignation

Accounting does not need to distinguish between a designation and a redesignation. From an accounting perspective, there is no difference between a position that is designated for the first time, and a position that is redesignated after being removed from a hedging relationship.

Concurrent/Subsequent Designations

A position can be designated concurrently or subsequently when the position that is to be used as the hedging instrument is added. Depending on the time reference for the designation at the time of the inflow, you can choose a calculation procedure [page 821] in Customizing that is then used to valuate the relevant position before it is reposted. The type of business transaction item can be used as a variable to do this. The different types of business transaction items must have already been configured for concurrent and subsequent designations.

Generation of the Designation Business Transaction

Accounting provides interfaces for the generation and posting of the internal business transaction *Designation*. The following information is required to generate a *designation* business transaction:

- Header information
  - Legal entity
  - Accounting system
  - Sender ID
  - SDL hedge ID
  - Type of hedge (macro cash flow hedge)
  - Designation date (as the date from which the designation is to be processed as affecting positions)
- Item information
  - Financial position that you want to use as a hedging instrument
  - Quantity or nominal amount that is to be hedged
  - Type of designation (concurrent or subsequent)

A business transaction is generated that contains two items for every financial position that is to be designated: An item for the outflow of the financial position not used for hedging, and an item for the inflow to the financial position used in the hedge that manages the hedging relationship ID as an additional defining characteristic.

Posting the Designation

The designation is posted in two stages. In the first step, the business transaction is posted to clearing key figures. The financial position concerned is selected for updating.

By updating the business transactions in the second step, at least all of the business transactions from the financial positions concerned are recalculated for the designation date. The business transactions are sorted so that designation business transactions are calculated *after* operational business transactions but *before* key date valuation business transactions.

100% of the available position is used as a hedging instrument in a macro cash flow hedge. The relevant quantity/ nominal value is delivered in the business transaction.

When the position is reposted, all the position components, and any existing revaluation reserves, are reposted. If you update the designation business transaction again, the generated transfer postings are reversed and new postings are generated in line with the result of the check.

Reversing a Designation

A designation business transaction can be reversed by a reversal business transaction.

Accounting provides interfaces for the generation and processing of reversal business transactions. The following information is required to generate a reversal business transaction:

- Sender ID of the business transaction to be reversed
- Source document of the business transaction to be reversed

The reversal business transaction refers to the source document of the original designation business transaction. The designation is reversed in two stages.

In the first stage, the posting of the designation business transaction to the clearing key figures is reversed. The position object concerned is preselected for updating. The derived postings are reversed in the second step when the business transactions are updated. The reversal postings are posted for the original designation date as affecting positions and with the earliest permitted (at the time of the update) posting date. All the business transactions affected by the reversal are updated as of this value date.

Dedesignation

Accounting is informed about the designation of a position used as a hedging instrument for a macro cash flow hedge by means of the internal business transaction "Dedesignation". The business transaction triggers a valuation and transfer posting of the position used up until this point for hedging purposes.

Process Overview: Cash Flow Hedge Analysis/Designation

The user can use cash flow hedge analysis to check the effectiveness of the hedging relationship created in the SDL. Where the results returned show that the relationship is not effective, the user generates a proposal about how to change it. This proposal can include both hedging instruments that are included for the first time, as well as those that are to be excluded. Using this proposal, Hedge Management generates designation and dedesignation business transactions. The business transactions are then transferred to Accounting for posting, and the hedging relationship in the SDL is updated.

The updated hedging relationship is based on the cash flow hedge analysis in the following analyses.

Generation of the Dedesignation Business Transaction

Accounting provides interfaces for the generation and posting of the internal dedesignation business transaction. The following information is required to generate a designation business transaction:

- Header information
  - Legal entity

    - Accounting system
    - Sender ID
    - Hedge ID
    - Dedesignation date (as the date from which the dedesignation affects positions)
- Item information
    - Financial positions that are no longer used as hedging instruments

A business transaction is generated that contains two items for every financial position from the hedging relationship that is to be excluded: A position for the outflow of the financial position used in hedging, and an item for the inflow to the financial position that is not used for hedging purposes.

Posting the Dedesignation

The dedesignation is posted in two stages. In the first step, the business transaction is posted to clearing key figures. The financial position concerned is selected for updating.

By updating the business transactions in the second step, at least all of the business transactions from the financial positions concerned are recalculated for the designation date. The business transactions are sorted so that dedesignation business transactions are calculated *after* operational business transactions but *before* key date valuation business transactions.

Two postings are made in the update:

1. First of all, the position used for hedging purposes is valuated at the time of the dedesignation in accordance with the hedge accounting rules. The effects of hedge accounting up until this time are taken into consideration during this process.
2. All the position components are then reposted from a position that is managed for each hedging relationship to a position that is not used explicitly for hedging purposes. See the next section for information on the way the revaluation reserve accumulated up until this point is handled by the system.

If the dedesignation business transaction is updated for a second time the generated valuations and transfer postings are reversed and new postings are generated.

Treatment of Revaluation Reserve After Dedesignation

Two different procedures can be selected to process revaluation reserves that accumulate in hedge accounting during the dedesignation.

1. The revaluation reserve can be written off the books directly as affecting net income in the profit and loss statement.
2. The revaluation reserve is cleared using the residual maturity of the hedging instrument linearly over time. The position managed for each hedging relationship is transferred with its position components and the revaluation reserve to a position with a revaluation reserve in the reserve to be accrued. In this way the system ensures that the accrued revaluation reserve is cleared on a pro rata basis when position outflows occur in the future as the reserve to be accrued.

The accrual period can be defined in Customizing as the residual maturity of the hedging transaction or fixed time periods specified in months. It is also possible to specify the minimum amounts as of which the distribution is to take place. If the minimum amounts are not met, the revaluation reserve is immediately written off the books as affecting net income.

Reversing a Dedesignation

A dedesignation is reversed in the same way as a designation.

Dissolution

An internal accounting dissolution business transaction is used to inform Accounting that a hedging relationship has been dissolved. The business transaction triggers the valuation and transfer posting of all the positions used for hedging purposes for this hedging relationship.

Process Overview: Dissolution

Dissolving a macro cash flow hedging relationship is a management decision that is entered in Hedge Management as a transaction. Hedge Management generates a dissolution business transaction, and transfers this to Accounting for posting. In addition to this, the hedging relationship is also updated in the SDL.

Generation of the Dissolution Business Transaction

Accounting provides interfaces for the generation and posting of the internal dissolution business transaction. The following information is required to generate a dissolution business transaction:

- Legal entity
- Accounting system
- Sender ID
- Hedge ID
- Dissolution date (as the date from which the dissolution affects positions)

A business transaction is generated that contains two items for every financial position assigned to the hedging relationship so far: A position for the outflow of the financial position used in hedging, and an item for the inflow to the financial position that is not used for hedging purposes.

Posting the Dissolution

The dissolution is posted in two stages. In the first step, the business transaction is posted to clearing key figures. The financial position concerned is selected for updating. By updating the business transactions in the second step, at least all of the business transactions from the financial positions concerned are recalculated for the dissolution date. The business transactions are sorted so that dissolution business transactions are calculated *after* operational business transactions but *before* key date valuation business transactions.

Two postings are made in the update:

1. First of all, the position used for hedging purposes is valuated at the time of the dissolution in accordance with the hedge accounting rules. The effects of hedge accounting up until this time are taken into consideration during this process.
2. All the position components are then reposted from a position that is managed for each hedging relationship to a position that is not used explicitly for hedging purposes. See the next section for information on the way the revaluation reserve accumulated up until this point is handled by the system.

If the dissolution business transaction is updated for a second time the generated valuations and transfer postings are reversed and new postings are generated.

Reversing the Dissolution

A dissolution is reversed in the same way as a designation.

Treatment of Revaluation Reserve After Dissolution

The two procedures described in the dedesignation can be selected to process revaluation reserves that accumulate in hedge accounting.

Position Inflow Caused by Operational Business Transaction

A position inflow caused by an operational business transaction increases the position identified by the characteristics in the business transaction. If the SDL hedging relationship is not specified in the business

transaction, then the system uses the transaction's financial position that is not used for hedging purposes. The business transaction, therefore, does not have any effect on the positions that are used for hedging purposes.

Process Overview: Position Inflow

Accounting selects the operational business transaction from the SDL and, using the *Post External Business Transactions* process, documents it as an accounting document. The system determines the effects on the position and result using the Update Secondary Business Transactions [page 786] process. These effects are then documented in the form of a derived accounting document.

Posting a Position Inflow

The system posts the position inflow for the position not used for hedging purposes in accordance with the general posting logic.

Reversing a Position Inflow

An operational reversal business transaction is used to trigger the reversal of a position outflow. The reversal is processed in accordance with the general posting logic.

Position Outflow Caused by Operational Business Transaction

A position outflow caused by an operational business transaction reduces the position identified by the characteristics in the business transaction. If the SDL hedging relationship is not specified in the business transaction, then the system uses the transaction's financial position that is not used for hedging purposes. In contrast to the position inflow, positions that are used for hedging purposes can also be affected by the operational business transaction. The system supports two different procedures for operational business transactions with position outflows:

- Planned position outflow
- Unplanned position outflow

You assign these procedures to the operational business transaction in Customizing.

Process Overview: Position Outflow

IAS Accounting selects the operational business transaction from the SDL and, using the *Post External Business Transactions* process, documents it as an accounting document. The system determines the effects on the position and result using the Update Secondary Business Transactions [page 786] process. These effects are then documented in the form of a derived accounting document.

Processing a Planned Position Outflow

A business transaction with a planned position outflow reduces the positions that exist for the identified transaction that are both used and not used for hedging purposes in proportion to the share of the total position volume.

Posting a Planned Position Outflow

The planned position outflow is posted in two stages. In the first step, the business transaction is posted to the clearing key figures for the position not used for hedging purposes. The financial position concerned is selected for updating.

Two types of postings are made in the update:

1. A pro rata transfer posting takes place for the clearing key figures (from the position not used for hedging purposes to the position managed for each hedging relationship).
2. The effects on the position and result (profit and loss) are then determined and posted for both types of position.

If the planned position outflow is being updated for a second time, the postings generated in the update are reversed and new postings are generated.

Reversing a Planned Position Outflow

An operational reversal business transaction is used to trigger the reversal of a position outflow. The reversal is processed in accordance with the general posting logic.

Processing an Unplanned Position Outflow

The operational business transaction is selected from the SDL by Accounting, and then posted. When Accounting is updating the secondary business transactions, the system first uses the positions not used for hedging purposes to constitute the position outflow. If the quantity required for the outflow cannot be met by these positions, the system transfers position components from a position managed for each hedging relationship to a position not used for hedging purposes until the volume required for the outflow exists in the latter position.

You can define the sequence in which the positions are to be transferred by selecting a FIFO or LIFO procedure in Customizing (using the "Valid From" date of the hedging relationship as the decisive criterion) or by presetting a different extraction strategy using a BAdI implementation. The position inflow is then processed after the position not used for hedging purposes has been transferred. The procedure concerned here is the same as the procedure used for an internally triggered dedesignation of a position, and is, therefore, published as an event using a BAdI interface. The event can be entered by Hedge Management using a BAdI implementation. Hedge Management is also responsible for the correction of the hedging relationship in the SDL.

Posting an Unplanned Position Outflow

The unplanned position outflow is posted in two stages. In the first step, the business transaction is posted to the clearing key figures for the position not used for hedging purposes. The financial position concerned is selected for updating.

Three types of postings are made in the update:

- If positions are to be transferred, the position used for hedging purposes is firstly valuated at the time of the position outflow in accordance with the hedge accounting rules. The effects of hedge accounting up until this time are taken into consideration during this process.
- If positions are to be transferred, the system then reposts all the position components from a position that is managed for each hedging relationship to a position that is not used explicitly for hedging purposes. For more information about how the accumulated revaluation reserves are processed, see *Treatment of the Revaluation Reserve with an Unplanned Position Outflow*.
- The effects on the position and income caused by the position outflow are determined and posted for the positions not used for hedging purposes.

If the unplanned position outflow is updated for a second time the generated valuations and transfer postings are reversed and new postings are generated.

Reversing an Unplanned Position Outflow

A reversal of a position outflow is triggered using an operational reversal business transaction. The internally triggered dedesignation cannot be reversed. The system can only reverse the position outflow from the position not used for hedging purposes.

Treatment of Revaluation Reserve with Unplanned Position Outflow

The two procedures described in the dedesignation can be selected to process revaluation reserves that accumulate in hedge accounting.

Key Date Valuation

In a key date valuation, the hedge adjustment is calculated and is written to/off the *Hedge Adjustment* book value component. This hedge adjustment is posted against the revaluation reserve (assuming the hedging relationship concerned is a priori 100% effective). For those hedging relationships that are not a priori 100% effective, the Hypothetical Derivative Approach [page 927] offers an alternative procedure to measure effectiveness and to valuate it.

Activities

You can find all of the following Customizing settings for Hedge Accounting in Customizing for *Bank Analyzer*:

Subledger Scenario

Choose *Processes and Methods* > *Accounting for Financial Processes* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Hedge Accounting* > *Assign Item Types to Hedge Business Transaction Types* and assign one item type for credit and one for debit to a hedge business transaction type.

Choose *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Hedge Accounting* > *Processing of Position Changes*:

- Choose *Define Dissolution Strategy for Hedge* or *Assign Dissolution Strategy to Accounting System* to assign procedures for an immediate dissolution or a linear distribution of revaluation reserves accumulated in hedge accounting over time for the business transactions *Dedesignation*, *Dissolution* and *Unplanned Position Outflow*.
- To define the sequence in which the positions are transferred internally for an unplanned outflow from a position used for hedging to a position that is not, choose *Derive Effects of Position Changes*.

For more information, see the documentation for the relevant Customizing activity.

4.4.5.2.8.4.1 Hypothetical Derivative Approach

Use

Revaluation reserves can be created for value changes (caused by changes in the hedged risk) in the hedging instruments used in macro cash flow hedging relationships. This is only possible, however, if corresponding value changes in the opposite direction have occurred for the hedged items (qualified positions). Where these corresponding opposite changes in value do not exist (unqualified positions), the system immediately records this as affecting net income.

The system calculates the qualified and unqualified positions for each individual hedging instrument. Since the hedging instrument concerned cannot be assigned a specific hedged instrument within the macro cash flow hedging relationship, the system calculates the corresponding opposite change in value using a hypothetical derivative.

The *hypothetical derivative* approach is used in cash flow hedge analysis and during the fair value effectiveness test to valuate the effectiveness of the hedging relationships. The effectiveness test is always carried out for the individual hedging instruments of a hedging relationship. The system generates a hypothetical derivative for each

hedging instrument in the hedging relationship. For more information, see Generation of Hypothetical Derivatives [page 515].

The system does not generate a financial position for the hypothetical derivative. The hypothetical derivative approach is simply a way to valuate the effectiveness of a macro cash flow hedging relationship that is not a priori 100% effective. As part of this approach, the hedging instrument is valuated in two steps:

1. In the first step, all the value changes for a hedging instrument are posted as revaluation reserves.
2. In the second step, the ineffective parts of the value change are determined and reposted from the revaluation reserve to the P&L statement. This transfer posting is generally carried out for the entire hedging instrument on the fixed leg.
   This fixed leg must have been found in a secondary data source in Hedge Management using logic, and its *Procedure for Calculation of Hedge Adjustments* (/BA1/C55HSIMPR) characteristic must have been assigned the value *H* (for "hypothetical derivative").

Features

Processing Business Transactions in Financial Position Management

Designation

The designation of an actual swap as a hedging instrument in a macro cash flow hedging relationship is transferred from

the Hedge Accounting Workplace [page 976] to accounting in the form of a business transaction. This internal or secondary business transaction *designation* is posted and then updated. This procedure triggers the transfer posting of position-related key figures from an unhedged financial position to a hedged financial position.

Generation of the "Designation" Internal Business Transaction

The internal business transaction *designation* has already been described in detail in Macro Cash Flow Hedge [page 920].

The hedging instruments in a macro cash flow hedging relationship can also be designated on a pro rata basis (<100% of the nominal amount of an unhedged financial position). The relevant quantity, or rather the nominal value share, is provided using the internal business transaction *designation*. The subsequent transfer posting contains all the book value components of the financial position, including a revaluation reserve (where this exists).

Update of the "Designation" Secondary Business Transaction

When the *designation* secondary business transaction is updated [page 786], the system checks whether the unhedged position that is still available is large enough for the designation. If this is the case, the position is reposted. If not, it is rejected for this particular hedging instrument.

During the update, the unhedged positions and their key figures are reposted on a pro rata basis to the financial positions that belong to the hedging relationship. You must have created the required calculation procedures [page 821] for the designation. These calculation procedures define the designation and the calculation of the business transaction.

Rejection of Designation

The relevant quantity, or rather the nominal value share, to be designated is delivered using the internal business transaction *designation*. If the quantity or nominal amount for the unhedged financial position of the derivative is not sufficient, the system rejects the designation. An event is used to notify the Hedge Accounting Workplace of the designation's rejection.

If the quantity is not sufficient, designations in the Micro Fair Value Hedge (Subledger Scenario) [page 905] and the Macro Cash Flow Hedge [page 920] are always completely rejected using the internal business transaction *Rejection of Designation*. With the *Hypothetical Derivative* valuation approach, however, the designations are **always** rejected **at the item level** of the business transactions (that is, for each hedging instrument). As a result, it is also sometimes possible to reject individual business transactions. The rejected items are not updated (they are not transferred to derivation management), but rather rejected, and following the update they are mixed with the updated business transactions once again.

Business Transaction: "Reversal of Designation"

In the majority of cases, a reversal of a designation is triggered by Hedge Management. For more information, see Macro Cash Flow Hedge [page 920].

Dedesignation

The dedesignation is an internal business transaction that triggers the transfer posting of the position key figures of the hedging instrument from a hedged financial position to an unhedged financial position.

It can be triggered by several events:

1. Position outflow (for example, a partial close-out swap)
2. The hedging relationship is ineffective
3. Proposal from cash flow hedge analysis [page 1107]

In the first two cases (*position outflow* and *ineffectiveness*), the dedesignation takes place in accounting. An event is used to inform the Hedge Accounting Workplace. Where the hedge is found to be ineffective, the system dedesignates the hedging instrument that is no longer effective. In the last case (*proposal from cash flow hedge analysis*), the dedesignation takes place in the HAW, and is then released to accounting.

Reversal of a Dedesignation

In the majority of cases, a reversal of a designation is triggered by Hedge Management. For more information, see Macro Cash Flow Hedge [page 920].

Change to the Designated Quantity

If the designated quantity or nominal amount of the actual swap changes as a result of a dedesignation, the system executes the internal business transaction *dedesignation* for the “old” quantity or the “old” nominal amount, and the internal business transaction *designation* for the “new” quantity or the “new” nominal amount. The designation date of the actual swap is changed to the change date.

Dissolution/Reversal of a Dissolution

For more information, see Macro Cash Flow Hedge [page 920].

Figure 1 illustrates the events that can trigger the dedesignation/dissolution of a macro cash flow hedging relationship with the *hypothetical derivative* approach.

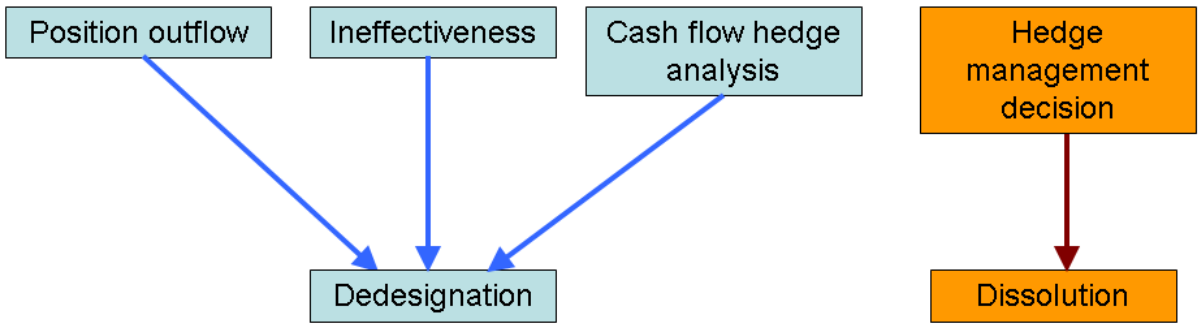


Figure 1: Events that trigger a dedesignation/dissolution

Processes

Process Results of Effectiveness Test

In the *hypothetical derivative* approach, the yield curve that is valid at the time of the designation is fixed for the designation date using the hypothetical derivative so that the fair value calculated in this process is identical to the fair value with a fixed interest rate (spot rate). The system tests the effectiveness against this spot rate

For more information about the fair value effectiveness test for macro cash flow hedging relationships, see Hypothetical Derivative Approach [page 1037] under Fair Value Effectiveness Test [page 1036].

The results of the effectiveness test are selected by accounting based on the following criteria:

- Accounting system
- Legal entity [page 696]
- Key date

The effectiveness indicator is provided for each derivative of the hedging relationship, along with the date as of which the ineffectiveness occurred. The aim is to designate all the ineffective hedging instruments from the hedging relationship The positions of the dissolved hedges are no longer valued using the special rules from hedge accounting as a result.

Key Date Valuation

The system does not generate a financial position for the hypothetical derivative. The financial positions that are processed in accounting using the hypothetical derivative approach are identified and valued using the *Procedure for Calculation of Hedge Instruments* (/BA1/C55HSIMPR) characteristic mentioned above. The book value components of the financial positions are updated and accounting documents are generated as part of the key date valuation. The key date valuation can be run periodically or independently of period.

Determining the Accounting Results

Once the relationship has been designated, the hypothetical derivative undergoes fair value changes. The financial positions that are corrected using the *hypothetical derivative* approach require their own calculation procedure. This calculation procedure is different to the traditional calculation procedure for the leg that is not selected in that it contains the *correction posting* calculation step. This calculation step transfers the unqualified position of the fair value change of the swap in the hedging relationship from the revaluation reserve (equity capital) to the P&L account.

Step-by-Step Procedure

An interest rate swap in a hedging relationship consists of two financial positions: One for the fixed leg and one for the variable leg. The book value of the swap is calculated as the sum of both legs. When the swap in a macro cash flow hedging relationship is designated, the cumulated fair value changes that have occurred since the designation was made are recorded in the position component *hedge adjustment* and posted to the revaluation reserve (equity capital). For more information, see Hedge Adjustment (Subledger Scenario) [page 895]. In order for this procedure to be successful, the hedging relationship must be 100% effective. If this is not the case, the unqualified position of the cumulated fair value change must be recorded as affecting net income. This is done by posting the unqualified position of the entire swap using the fixed leg only. It is assumed that it is the fixed leg that makes the main contribution to the change in fair value. The variable leg is therefore not changed in accounting. The fair value changes related to this leg are always posted to the revaluation reserve.

The fixed leg is valued using two calculation step types: The first calculation step type only affects the fair value changes to the fixed leg and is applied in the same way as with the micro fair value hedge. In the second calculation step type, the unqualified position of the cumulated fair value changes to the entire swap is calculated and posted using the fixed leg as a result of the comparison with the hypothetical derivative. In this way, the effects on the fixed leg remain separate from the effects on the whole swap.

First calculation step type

The cumulated value changes to the hedging instrument, which can be traced back to the hedged risk, are recorded from the time of the designation in the *hedge adjustment* position component. It is important to note here that at the time of the designation, the position components *acquisition value*, *amortization*, and *instrument valuation* are reposted from the unhedged position to the hedged position. The change to the hedge adjustment is calculated as follows:

Delta (HAt) = HFVt - HACtt - HAt-1

t = current key date

t-1 = last key date

HA = hedge adjustment

HFV = hedge fair value (equal to full fair value)

HAC = hedge amortized cost

Second calculation step type: correction of unqualified position of hedge adjustment with hypothetical derivative approach

The algorithm below is used to calculate the correction posting. Although the unqualified position of the cumulated change in value for the whole hedging instrument is calculated, it is posted using only the fixed leg.

The system carries out the following substeps:

1. Calculation of the fair value change to the hypothetical swap (H) between the date t and designation date d
2. Calculation of the cumulated fair value change to the entire swap (A) between the date t and the designation date d: A = FVt - FVd
3. Calculation of the cumulated unqualified position N of the cumulated fair value change A
    1. Case 1: Completely ineffective
       N = A if plus/minus sign (A) ≠ plus/minus sign (H)
    2. Case 2: Completely effective
       N = 0 if plus/minus sign (A) ≠ plus/minus sign (H) and H A
    3. Case 3: Partly ineffective

N = A- H if plus/minus sign (A) ≠ plus/minus sign (H) and H < A

Activities

Create a valuation rule for the calculation step type `T090` with a write-up rule, write-down rule, and book value rule.

You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Accounting Processes* > *Valuation* > *Edit Valuation Rules for Key Date Valuations*.

Example

Line No.	Event	Designation	t1	t2	t3	t4
1	Fair Value Swap Fixed Leg	100	105	106	105	107
2	Fair Value Swap Variable Leg	-100	-101	-103	-104	-101
3	Fair Value Swap Total A	0	4	3	1	6
4	Hypothetical Derivative Fair Value H	0	-2	-1	2	4
5	Rule for Calculation of Unqualified Position N		Plus/Minus Sign (A) ≠ Plus/Minus Sign (H): Completely Ineffective	Plus/Minus Sign (A) ≠ Plus/Minus Sign (H): Completely Ineffective	Plus/Minus Sign (A) = Plus/Minus Sign (H) and H A: Completely Effective	Plus/Minus Sign (A) = Plus/Minus Sign (H) and H < A: Partly Ineffective
6	Unqualified Position		N = A 4	N = A 3	N = 0 0	N = A-H 2
7	Cumulated Hedge Adjustment: Fixed Leg	0	105 -100 = 5	106 -100 = 6	105 -100 = 5	107 - 100 = 7

Line No.	Event	Designation	t1	t2	t3	t4
8	Posting to Revaluation Reserve (Fixed Leg)	0	5 - 0 = 5	6 - 5 = 1	5 - 6 = -1	7 - 5 = 2
9	Cumulated Revaluation Reserve: Fixed Leg Before Correction	0	0 + 5 = 5	1 + 1 = 2	3 - 1 = 2	5 + 2 = 7
10	Correction of Revaluation Reserve: Fixed Leg	0	4 - (5 - 5) = 4	3 - (6 - 2) = -1	0 - (5 - 2)= -3	2- (7 - 7) = 2
11	Cumulated Revaluation Reserve: Fixed Leg After Correction	0	5 - 4 = 1	2- (-1) = 3	2 - (-3) = 5	7 - 2 = 5
12	Cumulated Revaluation Reserve: Variable Leg	0	-1	-3	-4	-1
13	Cumulated Revaluation Reserve: Variable + Fixed Leg	0	0	0	1	4

Table 1: Posting example

Initial values:

- **Line 1:** Fair value of fixed swap leg
- **Line 2:** Fair value of variable swap leg
- **Line 3**: Fair value total of swap: line 1 + line 2
- **Line 4:** Fair value of hypothetical derivative

Posting results:

- **Lines 5 and 6:** Rule for calculation of unqualified position N: Calculated using the algorithm described above for the correction posting with the results from the lines 3 and 4.
- **Line 7:** The difference between two consecutive values in line 1
- **Line 8:** Posting of HA value changes from line 7
- **Line 9:** Cumulated revaluation reserve for fixed leg after correction t-1 (line 11) + posting to revaluation reserve for fixed leg t (line 8)

- **Line 10:** The difference between the recalculated unqualified position N (line 6) and the unqualified position posted until now (determined as the difference from the cumulated HA (line 7) and cumulated revaluation reserve before correction (line 9))
- **Line 11:** line 9 - line 10
- **Line 12:** The cumulated difference between two values in chronological order in line 2.
- **Line 13:** Revaluation reserve for entire swap. Line 11 + line 12.

4.4.5.2.9 Key Date Valuation

Use

Financial positions are valued on a key date. This process comprises multiple calculation steps. These steps include the calculation of amortization, fair value, and deferred tax, and the distribution of revaluation reserves across time. You can define your own key date valuation types. For example, you can do this to reflect a change in tax rates.

The table below contains the categories of financial assets and liabilities, and shows their subsequent measurement:

	Category	Holding Categories That Can Be Assigned	Classification	Accrual/Deferral	Measurement
1	Banking book (tactical items)	IFRS: LAR, HTM, OLI German HGB: banking book, financial liabilities	Financial investments that are held to maturity	Amortized costs incl. display of accruals (affecting net income)	None
2	Banking book (strategic items)	IFRS: AFS German HGB: liquidity reserve	Financial assets that are available for sale	Amortized costs incl. display of accruals (affecting net income)	Measurement that does not affect net income
3	Trading book	IFRS: HFT German HGB: trading book	Financial assets held for trading	Display of accruals (affecting net income); special case: amortized cost is also calculated for derivatives (OTC)	Measurement that affects net income

You must assign derivative financial instruments according IFRS specifications to the “Held-for-Trading” category and measure them at fair value (affecting net income).

So that the fair value is calculated correctly, the following sources are needed:

- Publicly listed price on an active stock exchange
- Measurement by an independent rating agency
- Recognized valuation models, such as a net present value model, or an option price calculator
- Market for a similar financial instrument

> **Note**
>
> When you are reinitializing the accounts, note the following:
>
> You cannot start the process under the following conditions:
>
> - If an accounting system is "frozen" in terms of re-initialization
> - If the client does not have a re-initialization status

Prerequisites

All secondary business transactions up to the key date have been updated.

Features

During the key date valuation process, the system calls different functions (manager). Each of these managers generates its own business transaction for each financial position. You can define the following managers in the Customizing settings for key date valuation:

- Accrual/Deferral
- Valuation
- Transfer of "unremitted earnings" (expenses, income and results)
- Foreign currency valuation

The valuation manager can support a further subdivision of the valuation process (depending on the Customizing settings). This may be necessary if you want to execute balance offsetting, for example. In this case, the valuation manager generates several documents with different business transaction categories.

You can reset the postings for the transfer of unremitted earnings triggered by the key date valuation. The reset is posted on the day after the key date valuation, and is a separate business transaction.

The key date valuation type controls which process steps are executed during a key date valuation run. This means that you can design a key date valuation process to meet your own requirements since individual process steps can be carried out independently of each other. The key date valuation type also influences which of the postings it has triggered are reversed (key date valuation postings, tax rate changes). If the key date valuation type is initial, the system reverses the documents with the following business transaction categories:

- 1003 Accrual/Deferral Business Transaction
- 1004 Valuation Business Transaction
- 1020 Write Down and Release Valuation Remnants
- 1021 Valuate

- 1040 P/L Transfer of Unremitted Earnings
- 1050 Valuation of Foreign Currency Position
- 1051 Foreign Currency Valuation of Balance Sheet Position (MAR)
- 1052 Foreign Currency Restatement

The business transaction categories 1020 and 1021 are provided as an alternative to business transaction category 1004 if the valuation is subdivided into two subprocess steps.

Asset/liability postings or reclassification postings for hedge positions can also occur during a key date valuation. The system does not reverse these using the key date valuation process. Instead, it uses the subsequent run *Update Secondary Business Transactions*.

For more information, see Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Hedge Accounting* > *Reclassification of Hedging Instruments*.

You can display the accounting documents that are generated by the key date valuation in the Results Data Layer. For more information, see Fair Value Measurement [page 896] and Amortization [page 890].

> **Note**
>
> You can run several key date valuations for the same key date.

Integration in the Calculation and Valuation Process Manager (CVPM)

The key date valuation is integrated in the CVPM. More specifically, the CVPM controls the generation of the worklist using a primary data source. This means that you can use the restart function of the CVPM process for incorrectly processed objects. The key date valuation is configured as a system process. This means that you cannot change most of the settings. You have to adjust the following parameters in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Fixed Step Sequences for Analytical Processes* in the *Create Worklist* step:

1. Primary Data Source
2. Package Size

Flexible key date process

The key date valuation is structured in modules and you can subdivide these modules into smaller process steps that you have to execute separately. The key date process can then interact with other processes, such as balance offsetting.

Separate step sequences

The process comprises the following step sequences:

1. Step sequence for *FPO Financial Positions*
2. Step sequence for *GFPO General Financial Positions*

You must execute these in this sequence so that the system correctly analyzes the general financial positions because the documents for the first step sequence (FPO) are used in the valuation of the second step sequence.

Separate valuation procedures

You can split the valuation procedure into separate parts according to the key date valuation type. For example, you can separate the accruals and deferrals and the release of valuation remnants from each other.

Example

In this case, you have to execute the step sequences for the *Financial Positions (FPO)* three times in succession with three different key date valuation types:

1. *[A] Accrual/Deferral*
2. *[R] Release Valuation Remnants*
3. *[V] Valuation*

After you have executed all the processes for the FPO step sequence, you have only to execute the key date valuation type *[V] Valuation* for the *General Financial Positions (GFPO)* for the key date valuation type.

Zero documents

The system saves zero documents both in accrual procedures and in valuation procedures. You must enter a technical calculation step (based on the calculation step category `1Z01` or `2Z01`) for the processing contexts *Operational Financial Accounting* and *Analytical Financial Accounting*. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Assign Calculation Steps for Zero Documents*. When you assign the calculation steps to the accrual or valuation procedure, you can then set the *No 0 Docs* checkbox for each step to indicate whether you want the system to suppress the generation of zero documents. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Accounting Processes* in the activity *Edit Accrual Procedures* or *Edit Valuation Procedures for Key Date Valuation*.

Calculation of deferred tax

The key date valuation also calculates deferred tax [page 817].

Transform and move unremitted earnings

In Customizing, you can add the characteristic `/BA1/C55FXFLOW` to the step in which posting key figures are derived to define that profit in foreign currency is initially posted only to unremitted earnings. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Derive Posting Key Figures*.

You can use the key date valuation type to define whether the system moves and transforms unremitted earnings in addition to the valuation. You must select the *Transform and Move* checkbox for the key date valuation type. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Accounting Processes* > *Define Key Date Valuation Type*. This enables you to define when the system moves and transforms unremitted earnings.

Note

To use the function for posting values to unremitted earnings, you have to define a separate calculation procedure for the moving and transformation of unremitted earnings for the calculation procedure template *Transfer: Unremitted Earnings in FX* (2602) and define a business transaction type for the business transaction category *P/L Transfer of Unremitted Earnings* (1040). You also have to assign the calculation procedure and the business transaction type to the transfer of unremitted earnings. You enter the settings required in Customizing

for *Bank Analyzer* under ▶ *Processes and Methods* ▶ *Accounting for Financial Products* ▶ *After Generation* ▶ *Financial Position Processes* ▶ *Processing of Internal Business Transactions* ▶ *Transfer Postings for Foreign Currency Profit* ▶.

Reset the key date valuation in the subledger scenario

You can reset the key date valuation for all financial positions. To do this, in the *Key Date Valuation* process (for the path, see **Activities** under *Key Date Valuation*) select the entry **`With Reset`** in the *Reset* field.

The following applies to the subledger scenario: If the system transfers unremitted earnings in addition to the valuation in the key date valuation, it no longer reverses all valuations when it carries out a reset, but only the **transform and move steps**. This reset is carried out on the day after the key date.

The prerequisite for this is that you have set the *Transform and Move* checkbox for the relevant key date valuation type in the Customizing activity *Define Key Date Valuation Type* (for the path, see **Activities**, step 1). You must have started the key date valuation run with the entry **`With Reset`**.

Errors in the key date valuation

The key date valuation process is integrated in the framework of the Calculation and Valuation Process Manager (CVPM). This means that all the options provided in the CVPM for restarting incorrect runs are also available here.

For more information, see Calculation and Valuation Process Manager (CVPM) [page 1622].

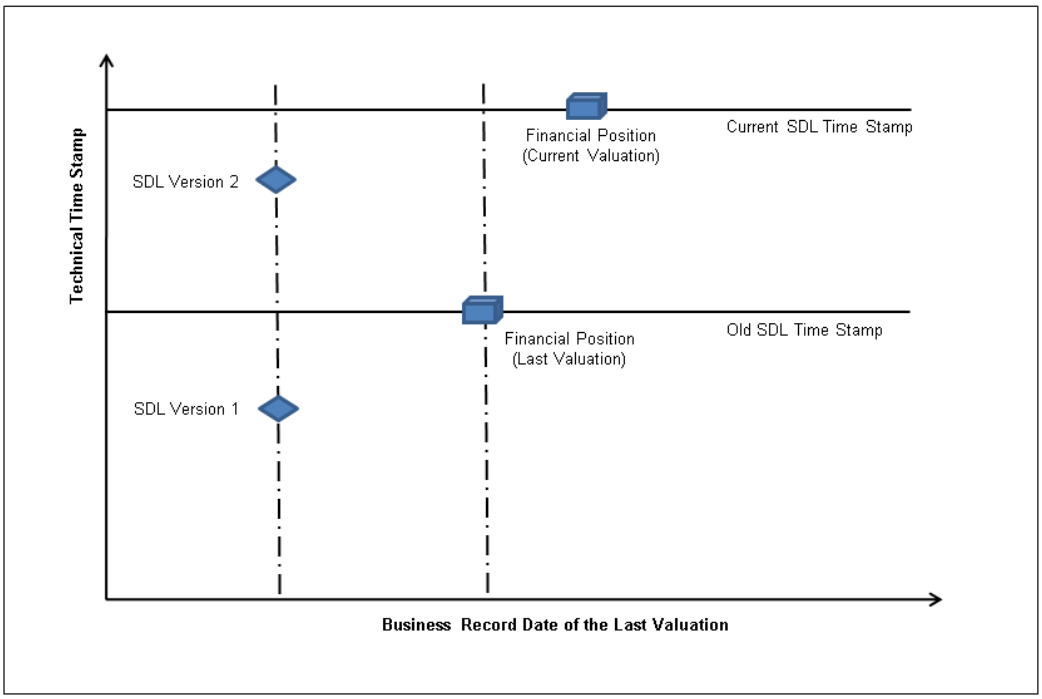
SDL version and time stamp problems (also relevant for the update of secondary business transactions)

The system might raise the following error messages:

1. Invalid versions of the Source Data Layer (SDL) were found
2. Error when getting the primary object

Cause

- Error message 1:
  The last valuation can be incorrect when the second version (SDL version 2) in the SDL is taken into consideration. This is because the financial position was measured on the basis of the first version (SDL version 1) in the SDL. The current time stamp can be used to find a second version in the SDL (SDL version 2), and hence also a different value for this financial position.



The last valuation can be incorrect when the second version (SDL version 2) in the SDL is taken into consideration.

- Error message 2:
  No primary object can be found for the technical time stamp and the business record date specified, since you have not set the current SDL time stamp.

Solution

- Error message 1:
  - Either reverse the old valuation or
  - Create a new version that is between the business valuation time points, and enter a new SDL time stamp [page 367]
- Error message 2:
  Enter a new SDL time stamp

Activities

1. Define key date valuation types that you want to use to derive the calculation procedures for the key date valuation.
   You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Define Key Date Valuation Type*.
2. You can create the following procedures for the key date valuation:
   - Accruals/deferrals
   - Valuation

- Foreign currency valuation

You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Accounting Processes*.

- Create **accrual or deferral procedures for the key date valuation** using calculation steps.
  Choose Customizing for the key date valuation and *Accounting Processes* > *Accruals/Deferrals* > *Accrual Methods* > *Edit Accrual Procedures*.
- Create **valuation procedures for the key date valuation** using calculation steps.
  Choose Customizing for the key date valuation and *Accounting Processes* > *Valuation* > *Edit Rating Model for Key Date Valuation*.
- Create **valuation procedures for foreign currencies for the key date valuation** using calculation steps.
  Choose Customizing for the key date valuation and *Accounting Processes* > *Foreign Currency Valuation* > *Edit Valuation Procedures for Foreign Currencies*.

3. Use the financial position characteristics to define the derivation of a calculation procedure or a business transaction type.

   You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Accounting Processes*.

   - To derive an accrual procedure, choose *Accounting Processes* > *Accruals/Deferrals* > *Accrual Methods* > *Derive Accrual Procedures*.
   - To derive a calculation procedure, choose *Accounting Processes* > *Valuation* > *Derive Calculation Procedures for Key Date Valuation*.
   - For foreign currency valuation, choose *Accounting Processes* > *Foreign Currency Valuation* > *Assign Foreign Currency Valuation Procedures and Business Transaction Types*.
4. To reset the key date valuation, you have to assign reset business transactions to the business transaction types used in the key date valuation.

   You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Reset* > *Reset Internal Business Transactions*.

Key date valuation

1. In Customizing under *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Manager (CVPM)* > *Edit Fixed Step Sequences for Analytical Processes*, you have defined the primary data source and the package size for the step sequences *Financial Positions* `FPO` and *General Financial Positions* `GFPO` in the worklist creation step.
2. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Automatic Processing* > *Run Key Date Valuation*.
3. Enter appropriate selection criteria. The following entries are mandatory:
   - Legal entity
   - Accounting system
   - Key date
   - Key date valuation type
   - Undo

- Step sequence
  You have to execute the key date valuation for the step sequences of the financial positions objects for all required key date valuation types one after another (for example, from accrual/deferral to valuation). After this you execute the key date valuation for the step sequence of the general financial positions objects for the key date valuation type *Valuation*.

If you want to use a log variant, you need to have created the log variant in Customizing.

You do this in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Edit Application Log* > *Define Application Log Variant*.

4. Choose to run the key date valuation.
   The system displays a log.

> **Note**
>
> When you restart the key date valuation (with the same run group), the system updates the time stamp. The system can then include any data that is delivered or corrected in the meantime when it restarts the key date valuation.

Reverse key date valuation

1. You reverse a key date valuation on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Automatic Processing* > *Reverse Key Date Valuation*.
2. Enter appropriate selection criteria. The following entries are mandatory:
   - Legal entity
   - Accounting system
   - Key date
   - Key date valuation type
   - Reason for reversal
   - Step Sequence
     You do not have to enter the step sequence. If you have not selected a step sequence, the system reverses all documents that belong to this key date. If you select a step sequence, the system reverses the documents for this step sequence only.
3. If you want to use a log variant, you have to have created one in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Edit Application Log* > *Define Application Log Variant*.
4. Choose .
5. Choose to reverse the key date valuation.
   The system displays a log.

Key date valuation (start-of-day)

The start-of-day key date valuation is based on the start-of-day balances, unlike the conventional key date valuation which is based on end-of-day balances.

You use this key date valuation in the following cases:

- To transfer the balance of subledger accounts.
  See also: Transfer of Subledger Accounts [page 950]
- To determine the effect of a tax rate change on the subledger accounts for deferred tax.

See also: Deferred Taxes (Subledger Scenario) [page 817]

In Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Define Key Date Valuation Type*, you create your own key date valuation type for the key date valuation (start-of-day).

You use the key date valuation type to specify that the system valuates the start-of-day position and you access the corresponding procedure templates using the relevant valuation procedures. For example, `2214` *Transfer Product Financial Position* and `2215` *Transfer General Financial Position*.

Run key date valuation in special period

If you run the key date valuation in a special period you can perform specific activities at the end of the year and prepare the new fiscal year. For example, you can initialize the impairment process [page 973].

The following conditions must be met:

- You have entered a key date valuation type for special periods in Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Accounting Processes* > *Key Date Valuation for Special Periods* > *Define Key Date Valuation Types for Special Periods*.
- The current fiscal year has been closed.
- The special period is open.

You run a key date valuation in a special period on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Automatic Processing* > *Run Key Date Valuation in Special Period*.

Reverse key date valuation in special period

You reverse the results of the key date valuation in a special period on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Automatic Processing* > *Reverse Key Date Valuation in Special Period*.

4.4.5.2.10 Measurement of Time Bucket Objects (Subledger Scenario)

Use

You use this function to run a key date valuation of time bucket objects in a portfolio fair value hedging relationship [page 1005] in the legal entities involved.

You can reverse the valuations in an additional process.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Automatic Processing* > *Run Key Date Valuation for Time Bucket Objects*.
2. Enter the following data:
   - Accounting system
   - Hedging relationship identifier
   - Key date
     By entering this data, you can, for example, save variants for dissolved portfolio fair value hedges, whose key date you can choose. When the system analyzes time bucket objects for designated hedges, it always takes as the key date the period-end date defined in Hedge Accounting Workplace (HAW), regardless of the key date specified.
   - Key date valuation type
   - Reset
     You can use this field to define whether the system is to reverse the valuation, and whether it should read the Customizing settings as well.

   You can specify technical data, such as filter variant, test run, and layout.
3. Choose  .
4. The system determines all the corresponding time bucket objects for the portfolio fair value hedging relationship and runs the key date valuation. A general portfolio fair value hedging relationship can affect multiple legal entities.

To reverse the key date valuation for time bucket objects, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Automatic Processing* > *Reverse Key Date Valuation for Time Bucket Objects*.

1. Enter the following data:
   - Accounting system
   - Hedging relationship identifier
   - Key date
   - Reason for reversal

   > **Note**
   >
   > When you reverse the analysis if time bucket objects for pending hedging relationships, the system uses as the key date the period-start date defined in the Hedge Accounting Workplace, regardless of the key date specified. A key date must be entered for dissolved hedging relationships.
   >
   > You can specify technical data, such as filter variant, test run, and layout.
2. Choose  .
3. The system determines the corresponding time bucket objects and runs the key date valuation.

4.4.5.2.11 Manual Postings (Subledger Scenario)

Use

The transaction Enter Manual Posting [page 945] enables you to manually post to the individual key figures of a financial position. This posting is then reset by the system the day after it is posted.

Prerequisites

Make the following settings in Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Business Transaction Types and Item Types*:

- In the Customizing activity *Edit Business Transaction Types*, create at least one business transaction type for manual postings and one for resetting the manual posting.
- In the Customizing activity *Assign Business Transaction Categories to Business Transaction Types*, assign the business transaction category `1060` to the business transaction type for **manual postings**.
- In the Customizing activity *Assign Business Transaction Categories to Business Transaction Types*, assign the business transaction category `1061` to the business transaction type for **resetting manual postings**.
- In the Customizing activity *Create Item Types*, create the required item types for the manual postings and assign these to the relevant item templates in the Customizing activity *Assign Item Templates to Item Types*. The following item templates are permitted:
  - `1JC1` Manual Posting: Operational Key Figures Off-Balance-Sheet
  - `1JC2` Manual Posting: Operational Key Figures
  - `2JC1` Manual Posting: Analytical Key Figures
- Edit the Customizing activity *Assign Item Types to Business Transaction Types* accordingly.
- Under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Derive Posting Key Figures*, create a derivation rule in which you assign a pair of credit and debit posting key figures to each item type for manual postings.
  Enter the following condition for this derivation rule:
  Manual Posting `FLG_MANU_CORR = X`.

Under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Reset* > *Reset Internal Business Transactions*, assign the relevant reset business transaction type to each business transaction type for manual postings.

Features

Make this posting once the processes for day processing have run. Once you have made a manual posting for a financial position for a specific posting date, you cannot make any further postings with this posting date or with an earlier date for this financial position using automatic processes.

When you make a manual posting, the system automatically posts a reset on the next posting date.

At the end of the year, you can post to a special period if you are using a fiscal year variant with special periods.

Manual postings can be made to product-related financial positions only and not to general financial positions.

> **Note**
>
> When you enter a manual posting in foreign currency to profit or reserve, the system automatically transforms and moves the posting.

4.4.5.2.11.1 Entering/Reversing Manual Posting

Use

Procedure

Enter Manual Posting

1. To enter a manual posting, on the *SAP Easy Access* screen choose:
   Bank Analyzer > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Manual Processing* > *Enter Manual Posting*.
2. Select an accounting system and a legal entity, and enter the posting date on which the posting is to be made.
3. You can also make the selection process for the financial position you want to manually post to more specific by using the selection criteria.
4. Choose *Execute* (F8).
5. Select a financial position.
6. Choose *Execute* (F8).
   The posting screen appears. You can make a manual posting for the selected financial position. You can enter a description or reason for the manual posting and then select a business transaction type.
7. In the table of posting items at the bottom of the screen, select debit and credit key figures for each row. The item type is automatically filled. Enter a positive amount in transaction currency and specify the transaction currency:
   If necessary, the system translates the currency into the functional currency and into the other currencies assigned to the accounting system and legal entity.
8. Save your entries.
   This completes the entries for the manual posting and initiates the posting in the system.

Reverse Manual Posting

1. To reverse a manual posting, on the *SAP Easy Access* screen choose:
   Bank Analyzer > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Manual Processing* > *Reverse Manual Posting*.
2. Select an accounting system, a legal entity, and a posting date.
3. Enter the reason for reversal.

4. Choose *Execute* (F8).
   A selection list is displayed that contains all the items of documents that have not been reversed for the manual postings on the selected posting date.
5. Select the documents that you want to reverse from the list.
6. Choose *Execute* (F8).
   This triggers the reversal of the selected documents.
   If you select an item in a document, this always triggers the reversal of the entire document and all its items.

> **Note**
>
> Only manual postings that have not been reversed are displayed.
>
> For more information, see Reversal [page 795].

Result

You can display the posting in the results data on the *SAP Easy Access* screen under *Results Data Layer* > *Display Results Data*.

4.4.5.2.11.2 Postprocessing of Manual Postings

Use

In postprocessing, the system carries out the following activities:

- Foreign currency valuation
- Asset/liability determination and asset-liability transfer if the asset/liability status has changed as a result of the manual postings.
- Reclassification of hedging instruments
  For more information, see Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Hedge Accounting* > *Reclassification of Hedging Instruments*.

The system carries out postprocessing for all positions that were posted to manually on the selected posting date and that meet the selection criteria.

> **Note**
>
> For this selection, you need to enter the key date valuation type to specify which valuation procedures the system uses for foreign currency valuation.

> **Note**
>
> At the end of the year, you can post to a special period if you are using a fiscal year variant with special periods.

Prerequisites

You have made the relevant settings in the Customizing activities for foreign currency valuation, asset-liability transfer, and reclassification of hedging instruments in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Postprocessing of Manual Postings*.

Features

When you carry out postprocessing, the system automatically posts a reset on the next posting date.

Reversal: If the manual postings contain errors and postprocessing has already run, you have to reverse the manual postings and postprocessing.

Activities

You carry out manual postprocessing after the manual postings [page 944].

4.4.5.2.12 Year-End Processing

Use

Year-end processing comprises the balance carryforward [page 947] function and, optionally, the transfer of subledger accounts [page 950].

More Information

Processes in the AFI Scenario [page 678]

4.4.5.2.12.1 Balance Carryforward

Use

You can use the balance carryforward function to determine the opening balance of the subledger for financial instruments in Bank Analyzer for the current fiscal year. As a basis for this, the system determines the balance of

on-balance-sheet and off-balance-sheet key figures for product-related and general financial positions, and the balance of the reserve key figures for product-related financial positions at the end of the previous year, including any special periods that might exist. On the first day of the current fiscal year, the system posts the calculated balance as a balance carryforward against retained earnings.

Prerequisites

- In Customizing for Bank Analyzer under *Basic Settings* > *Accounting Settings* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Define Accounting Systems*, you have set a result type in the result category `HFBCF`*Balance Carryforward* for the purpose *Balance Carryforward*.
- In Customizing for Bank Analyzer, choose *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Business Transaction Types and Item Types*.
  - In the Customizing activity *Edit Business Transaction Types*, you define the required business transaction types and under *Assign Business Transaction Categories to Business Transaction Types*, you assign them the business transaction categories `1090` *Balance Carryforward for Product Financial Position* and `1091` *Balance Carryforward for General Financial Position*.
  - In the Customizing activity *Create Item Types*, you define the required item types and assign them to the corresponding item templates under *Assign Item Templates to Item Types*.
    - For a balance carryforward for an operational, on-balance-sheet key figure, you use the item templates `1JI0` and `1JI1`.
    - For a balance carryforward for an operational, off-balance-sheet key figure, you use the item templates `1JI2` and `1JI3`.
    - For a balance carryforward for an analytical, on-balance-sheet key figure, you use the item templates `2JI0` and `2JI1`.
- In Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Year-End Processing* > *Edit Calculation Procedures for Year-End Processing*, you edit the calculation procedure that the system uses to determine the balance carryforward .
- In Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Year-End Processing* > *Accounting Processes* > *Assign Procedure and Business Transaction Type for Balance Carryforward*, you define the calculation procedure and business transaction type for each accounting system.
- In Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Hedge Accounting* > *Reclassification of Hedging Instruments* > *Assign Reclassification Calculation Procedures to Accounting System*, you define the business transaction type for reclassifying hedging instruments as part of the balance carryforward process.
  Note that the business transaction type must be assigned to the business transaction category `1090` *Balance Carryforward for Product Financial Position*.

Features

The system stores the balance carryforward as a business transaction (of the result type defined in Customizing) in the form of one accounting document for each product-related financial position object and one for each general financial position.

If you use hedge accounting, the system also divides the balance carryforwards for hedging instruments up according to the hedged and unhedged part.

For more information about reclassifying hedging instruments, see the Customizing documentation for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Hedge Accounting* > *Reclassification of Hedging Instruments* > *Assign Reclassification Calculation Procedures to Accounting System*.

> **Note**
>
> You cannot use the *Balance Carryforward* function for *SAP Profitability Analysis* scenarios.

> **Note**
>
> Note that you cannot use balance carryforwards as input data for balance processing and for aggregated transactions.

Activities

Determine Balance Carryforward

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Automatic Processing* > *Determine Balance Carryforward*.
2. Enter the data. In particular, enter the previous fiscal year and choose *Execute*.
3. The system first reverses any existing balance carryforwards posted in the new fiscal year, it recalculates the balance carryforwards and posts the documents in the new fiscal year.

Reverse Balance Carryforward

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Automatic Processing* > *Reverse Balance Carryforward*.
2. Enter the data. In particular, enter the previous fiscal year and choose *Execute*.
3. The system reverses the balance carryforwards posted in the new fiscal year.

4.4.5.2.12.2 Transfer of Subledger Accounts

Use

You usually use this function as part of year-end processing to transfer subledger account balances, for example, the balances of subledger accounts for the valuation components risk provision, hedge adjustment and fair value adjustment, and the subledger account balances for financial positions of the currency position.

You use the corresponding procedure templates `2214` *Transfer Product Financial Position* and `2215` *Transfer General Financial Position*.

The system transfers the balances of subledger accounts as part of the start-of-day key date valuation.

Prerequisites

In Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Define Key Date Valuation Type* , you create your own key date valuation type for the key date valuation (start-of-day).

Activities

1. You have executed the activities for the end of the day and for the end of the period.
2. You have carried out the required manual postings.
3. At the start of the new fiscal year, you run a key date valuation (start-of-day).

More Information

For more information, seeKey Date Valuation [page 934] (section **Key Date Valuation (Start-of-Day)** and Processes in the AFI Scenario [page 678].

4.4.5.2.13 Initialization of Profit and Loss Accounts (Subledger Scenario)

Use

> **Note**
>
> We recommend that new customers use only the year-end processing [page 947] functions.

> **Note**
>
> You can continue to use this function for your year-end activities and for year-end activities for imported subledger documents without a financial position object.
>
> For more information, see Initialization of Profit and Loss Accounts for Imported Subledger Documents Without Financial Position Object [page 960].

> **Note**
>
> If you want to continue to use the initialization of profit and loss accounts for year-end activities, you **cannot** use the following functions:
>
> - Balance Carryforward [page 947]
> - Transfer of foreign currency positions (in the context of Transfer of Subledger Accounts [page 950])

You can use this function to reset the balances of key figures affecting net income to zero for the next fiscal year. Resetting the balances is a business transaction. This requires a business transaction type and item types for the document posting. When you initialize profit and loss accounts, the system reverses the postings from any previous initialization of profit and loss accounts.

Prerequisites

- You have made the following settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Year-End Processing* > *Accounting Processes*:
  - *Select Business Transaction Type for P/L Initialization*
  - *Assign Posting Key Figures for P/L Initialization*
- You have defined the reversal reasons for reversing the postings for a prior profit and loss initialization in Customizing under *Bank Analyzer* > *Results Data Layer* > *Settings for Flow Results* > *Edit Reversal Reasons*.

Features

When it resets the balances from the end of the fiscal year, the system creates a document. This document is assigned to period zero of the following fiscal year and the posting date is the first day of the following fiscal year. The processing categories *1IX_PL* or *2IX_PL* are used for key figures affecting net income. During the initialization of profit and loss accounts, the balances of these key figures are transferred to a key figure called *Cumulated Profit/Loss*. The *Cumulated Profit/Loss* key figure is assigned to the processing category *1BG_EQUITY* or *2BG_EQUITY*.

Activities

1. Ensure that the first posting period of the new fiscal year is open. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Results Data Layer* > *Periodic Tasks* > *Open and Close Periods* .
2. To initialize profit and loss accounts, choose transaction code `/BA1/B1_YEC_SL`.
   The system displays a selection screen.
3. Enter the following data:
   - Legal entity
   - Accounting system
   - Fiscal year
   - Reason for reversal

   You can specify technical data such as whether the transaction is to run sequentially or as a test run.
4. Choose .

4.4.5.2.14 Analysis (Subledger Scenario)

The line item and balance display are the central functions available to users for analyzing, controlling, and reporting on the data processed in the accounting processes. This includes data provided from external systems as well as the balances based on them for financial positions.

For more information about the displays from a financial reporting perspective rather than an accounting processes perspective, see SAP Library for Business Information Warehouse.

For more information about the individual reports, see Application Log [page 723] and Displaying Financial Position Balances (Subledger Scenario) [page 954].

You can also use the Display Results Data [page 1407] function.

The layouts of all the displays in accounting can be adjusted easily to display data in list or graphical form. You can replace any alphanumeric term or part of a term with an asterisk.

Example: selection criterion *Primary Key*

Query	ISIN	Securities Account Number
1	123456	4711
2	123456	*
3	*	*
4	123*	47*

When you print, spool requests are generated in the SAP spool system for the lists. You can use output control to:

- Output spool requests (send to the host spool, for printing or output on other devices).
- Display details about a spool request and change certain attributes, such as destination printer, number of copies, and priority.
- Display the data to be printed and download the data to a pure ASCII file.
- Display output requests and track the status of a job that is being printed.
- Delete spool requests and the generated lists.

4.4.5.2.14.1 Display of Financial Positions

Use

You use this function to display financial positions.

You can choose from a range of selection criteria on the selection screen, for example financial position type, legal entity, and key date. The system then displays a results list containing all of the financial positions that match your selection. You can select individual financial positions and then, for example, choose from the following functions:

- Display posting results
- Display details for a financial position
- Display history of descriptive characteristics
- Analysis of financial position
- Save the list as a local file

The system highlights any financial positions in the list that match the following criteria:

- *Accounting update required*
  For example, you are using the accounting processes without cost accounting processes and are yet to update secondary business transactions.
- *Update of calculation of internal costs required*
  For example, you are using the profitability analysis and are yet to update internal costs.

For more information about financial positions, see Financial Positions (Subledger Scenario) [page 744].

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Analysis* > *Display Financial Position*.
2. Specify the legal entity and the accounting system.
3. Choose a date and time for calculating the financial positions.
4. Choose a financial position type.
   The input fields of the *Position* screen area change according to your selection.
5. Enter further data to select a financial position.
6. Configure the settings for administrative data and output options.
7. Choose  .
   The system displays the financial positions that you selected (using the selection criteria) in a table, taking into account the specified output options. The options available in this view include:
   - Choose  to display all financial position characteristics for the selected financial position.
   - Choose  to display all versions of a previously selected financial position.
   - Choose  to display to which color coding the various statuses of a financial position are assigned.

4.4.5.2.14.2 Displaying Financial Position Balances (Subledger Scenario)

Context

Financial position balances are the balances of the key figures and processing categories of financial positions. Internally, displaying financial position balances uses exactly the same functions as Calculating Financial Position Balances [page 752]. In addition to the financial position balances, the system also displays data from calculation bases and characteristics of the financial position.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Analysis* > *Display Financial Position Balances*.
2. Specify the legal entity and the accounting system.
3. Specify the relevant dates.

> **Note**
>
> Note that the system records balances by posting date and by posting date and value date. You can specify which balances you want to display by entering values in the *Posting Date* and *Value Date* fields:
>
> - If you leave the *Value Date* field empty, the system displays balances by posting date.
> - If you specify a value date, the system displays balances by posting date and value date.
>   Note that the system may contain these balances for specific processing categories only based on the settings configured.
> - If you leave the *Posting Date* field empty, the system uses the current date to form the balances.

4. Choose a layout for displaying the financial position balances.
5. Choose  .

   For each financial position, the system displays a list of all the characteristics concerned, the totaled key figures or processing categories, and further information.
6. To display the details, choose *Details*.

   In this view, the characteristics that have values and the key figure values that are not zero are displayed below one another.

4.4.5.2.15 Reclassification

Use

You use a reclassification to change holding categories and, consequently, to also change the valuation approach for a financial transaction or a securities position account that is linked to a holding category. In doing so, you can also switch between the following valuation approaches:

- Amortized Cost
- Fair Value (Other Comprehensive Income)
- Fair Value (Profit and Loss)

Prerequisites

- You can only reclassify financial transactions and securities position accounts that are not processed by source data aggregation.
- The system makes sure that the relevant financial position objects contain the descriptive SAP characteristic `/BA1/C55HOLDCT`. The holding category is stored in this SAP characteristic.

  You define the holding categories in Customizing for *Bank Analyzer* under *Processes and Methods* > > *Accounting for Financial Products* > *After Generation* > *Basic Settings* > *Define Holding Categories*.

Features

Calculation Procedure

We provide the calculation procedure templates `1111` (for the value-date-based, two-dimensional part) and `2111` (for the not value-date-based, one-dimensional part). You can use these calculation procedure templates to implement all the relevant reclassification methods in the various directions.

The calculation procedure templates are structured as follows:

1. Update of position
2. Change of approach
3. Change of reporting due to transfer posting

For more information, see Value-Date-Based Procedure Templates [page 826] and Procedure Templates (Not Value-Date-Based) [page 839].

You can view the procedure template `1111` for the value-date-based part and the procedure template `2111` for the not value-date-based part in Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Procedures* > *Display General Overview of All Templates*.

Reclassification business transaction

When you create a financial position object, depending on the Customizing settings the system derives the contents of the holding category characteristic from the master data. The characteristic can then only be changed by an analytical event (a reclassification business transaction).

A reclassification business transaction must be structured as follows:

- It contains all the information needed to identify the financial position object, that is, all the defining characteristics of the financial position object.
- It does **not** have to provide information about values or quantities. If the business transaction does provide this information it is ignored by the system.
- The posting date is the date of the reclassification decision.
- The value date in the header data contains information about when the change of holding category takes effect. As a rule, this means that the value date is before the posting date.
  The system does not support a business transaction with a value date after the posting date.
- It contains one item that provides the target holding category in characteristic `/BA1/C55HOLDCT`.
- You can use the business transaction type and the item type to determine which calculation procedure matches the combination of source and target holding category.

> **Note**
>
> Reclassification is not possible by changing the master data and processing the application event.

There are two reclassification variants:

1. In the first variant, you assign the business transaction category `1011` *Reclassification* to a business transaction type. If there is a reclassification business transaction of this kind, business transactions with the following conditions are not permitted:
   - Value date of the business transaction < value date of the reclassification business transaction
   - Posting date of the business transaction > value date of the reclassification business transaction

2. In the second variant, you assign the business transaction category `1016` *Reclassification (Valuation Approach Extended)* to a business transaction type. The restriction mentioned under 1. does not apply to this case. However, you can use this business transaction type only for the following two cases:
    - Change in the name of the holding category without a change in the valuation approach
    - Change of amortized cost valuation approach to amortized cost and fair value approach

Activities

Calculation Procedures

In Customizing create a value-date-based and a not value-date-based calculation procedure for each reclassification direction required. Each calculation procedure must contain the relevant subset of steps based on the step categories of the calculation templates `1111` or `2111` (see "Features").

Execute the following activities in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting Processes* > *Elementary Calculation Procedures*:

- for *Elementary Calculation Procedures (Value-Date-Based)*
    - *Edit Elementary Calculation Procedures (Value-Date-Based)*
    - *Derive Elementary Calculation Procedures (Value-Date-Based)*
- for *Elementary Calculation Procedures (Not Value-Date-Based)*
    - *Edit Elementary Calculation Procedures (Not Value-Date-Based)*
    - *Derive Elementary Calculation Procedures (Not Value-Date-Based)*

Triggering a Reclassification

To trigger a reclassification you have to provide an analytical event.

Example

When you post a new financial transaction for the first time, the system derives the holding category `HFT` from the master data. You now want to switch to the holding category `HTM`.

To do this, you have to provide an analytical event in the form of a business transaction. This reclassification business transaction must have the following properties:

- It contains one item that provides the target holding category `HTM` in characteristic `/BA1/C55HOLDCT`.
- The system derives a calculation procedure from the item type of the business transaction. This procedure converts the valuation approach to *Amortized Cost*.

4.4.5.3 Use of Imported Subledger Documents in the Subledger Scenario

Use

You can import subledger documents from external systems directly into the Results Data Layer (RDL). You can enrich the imported subledger documents (for example, by deriving a general ledger account). You can also define whether the system creates a financial position object for every financial transaction or for every financial instrument and securities account, even if the corresponding master data does not exist in the Source Data Layer (SDL).

The system can continue to process the imported documents for which it has created a financial position object, and you can use the following additional functions:

- Implicit currency exchange
- Immediate transform and move
- Transform and move at period-end
- Foreign currency valuation
- Asset/liability transfer
- Reclassification when changes are made to characteristics
- Balance carryforward

You implement these functions in the system if you reuse the *Update Secondary Business Transactions*, *Key Date Valuation*, and *Balance Carryforward* processes from the AFI scenario. The system saves the documents in three different result types in the RDL:

- Imported Subledger Documents
  This result type contains the imported subledger documents.
- Accounting Documents
  This result type contains the documents from all additional functions except balance carryforward.
- Balance Carryforwards
  This result type contains the balance carryforwards.

You define the result types in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities* > *Define Accounting Systems*.

> **Note**
>
> Use the enterprise service `FinancialInstrumentsAnalyticalResultsManagementLedgerNotificationIn` to import the documents. It provides the following operations:
>
> - `CreateDocumentsBasedOnBulkNotification_V1` for the creation of subledger documents
> - `CancDocumentsBasedOnCancellationBulkNotification_V1` for the reversal of subledger documents

> **Note**
>
> A reversal document must always have the same posting date as the reversed document.

> **Note**
>
> You **cannot** use the additional functions for imported subledger documents for *SAP Profitability Analysis* scenarios.

> **Note**
>
> SAP does not support the following functions for financial positions that belong to the imported subledger documents:
>
> - Hedge accounting
> - Maturity Grouping
> - Fair Value Determination
> - Calculation of Percentage Expected Loss

Prerequisites

To be able to use the functions mentioned above, when the subledger documents are imported, you need to enrich them with the reference to a financial position object and with the relevant business transaction types and item types.

You can use only financial positions of the product category `10` *Position without Source Data* for this. In Customizing for *Bank Analyzer* under *Processes and Methods* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Business Transaction Types and Item Types* > *Assign Business Transaction Categories to Business Transaction Types*, you need to assign the business transaction type to one of the following business transaction categories:

- `1006` *Direct Posting (posting dependent on the value date)*
- `1081` *ISD Not Dependent on Value Date*
- `1082` *ISD Characteristic Change*

In Customizing for *Bank Analyzer* under *Processes and Methods* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Business Transaction Types and Item Types* > *Assign Item Templates to Item Types*, you need to assign one of the following item templates to the item type:

- `1I91` *ISD Operational*
- `2I91` *ISD Analytical*
- `1I92` *ISD Characteristic Change*

You can make settings for the enrichment of documents in Customizing for *Bank Analyzer* under *Results Data Layer* > *Settings for PI Interfaces* > *BAdI: RDL Inbound After the Mapping of the XI Structure to the RDL Data Model*.

More Information

- Reverse Subledger Documents [page 1406]
- Foreign Currency Concept (Subledger Scenario) [page 725]
- Posting Process Steps for Foreign Currency Valuation (Subledger) [page 730]
- For more information about enterprise services, see http://help.sap.com/bankingservices. Choose *Integration & Analytics Information*, open the documentation and choose *Process Components in banking services from SAP* > *Financial Instruments Analytical Results Management* > *Analytical View of Financial Instrument Ledger Document*.

4.4.5.3.1 Initialization of Profit and Loss Accounts for Imported Subledger Documents Without Financial Position Object

Use

You use this function to initialize the profit and loss statement for subledger documents that are imported from feeder systems and for which the system has not created a financial position object.

You use this function for subledger documents that you do not want to process with the Bank Analyzer accounting processes.

Integration

In Customizing for the RDL, you can specify the granularity with which the imported subledger documents are to be aggregated in the RDL. For example, documents can be aggregated at the G/L account level.

To accelerate processing, you can use preaggregation [page 1408] within the RDL. For example, you can carry out preaggregation on a weekly or monthly basis.

Prerequisites

- In Customizing for *Bank Analyzer*, you have entered the following data under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*:
  - Result type
  - Result view
  - Filter
    This must contain the year but no key figure.
  - Aggregation levels

The results data area [page 1366] that you use for imported subledger documents is the same as that for other accounting processes.

- In `SM30` in the view `/BA1/VB1_YEC_ISD`, you have assigned the filter and the aggregation levels to the accounting-relevant data.
- In Customizing for *Bank Analyzer*, you have specified the derivation rule for G/L accounts under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Derivation of General Ledger Accounts*.
  Make sure that there are no financial positions for the imported subledger documents.
- In addition, the prerequisites for initializing profit and loss accounts also apply. For more information, see Initialization of Profit and Loss Accounts (Subledger Scenario) [page 951].

Activities

1. Choose transaction code `/BA1/B1_YEC_SL`.
   The system displays a selection screen.
2. Enter the following data:
   - Legal entity
   - Accounting system
   - Fiscal year
   - Reason for reversal

   You can specify technical data such as whether the transaction is to run sequentially or as a test run.
3. Select the *Imported Subledger Documents* indicator.
4. Choose  .

4.4.6 Implementation of Financial Statement Entities

Use

You use this process to initialize the use of financial statement entities in accounting.

You can create one balance sheet for each different financial statement entity. These entities are defined by one or more dimensions, for example by legal unit and profit center.

You can make additional settings for the dimensions of financial statement entities in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Derivations for Dimensions of Financial Statement Entities*.

Depending on the initialization reason, you perform accounting processes, that is to say initialization tasks, in order to lay the foundation for the use of new or changed functions. These initialization tasks include the generation of new technical objects and the transfer of existing documents.

SAP supports the following initialization reason. You perform the corresponding initialization tasks (see the *Process* section).

- Implementation of financial statement entities

Prerequisites

You have already defined the following settings in Customizing for *Bank Analyzer*:

- Under *Results Data Layer* > *Basic Settings* > *Edit Results Data Area*, you have assigned the financial statement entity dimension to an appropriate results data area.
- Under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Implementation of Financial Statement Entities*, you have made the required settings based on your initialization reason.

Process

1. To start the initialization tasks, on the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Tools* > *Implementation of Financial Statement Entities*.
2. Start the following transaction:

Initialization Reason	Initialization Task		
	Assign financial positions to financial statement entities	Generate required additional general financial positions	Initialize required additional general financial positions
Implementation of dimensions of financial statement entities	x	x	x

4.4.7 Use of Source Data Aggregation

Using source data aggregation [page 286] in the Source Data Layer (SDL) has an impact on the accounting processes and cost accounting processes.

> **Note**
>
> Source data aggregation is only relevant for the subledger scenario.

- Accounting Processes
  - By aggregating data, you reduce the data volume of inbound documents in the accounting processes. The system analyzes a smaller number of documents and financial positions. This allows you to reduce the amount of data and improve system performance in the accounting processes. The worklist in the accounting processes becomes smaller. The system supports the aggregation of transactions for specific

account types (such as checking accounts or savings accounts). Cash-flow transactions (such as loans) can also be aggregated if these do not have to be analyzed at cash flow level in the accounting processes.

- Valuations and calculations of financial positions that refer to aggregation objects are only performed for current postings and not for future ones.
- Since the financial position type cannot be transferred from the SDL financial transaction to aggregation objects, you can derive the financial position type in Customizing from the characteristics of the aggregation object. You can find these settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Characteristics of Financial Positions* > *Derive Financial Position Type for Aggregated Financial Products*.
  Alternatively, you can also derive this using a secondary data source. You find this setting in Customizing for Bank Analyzer under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Characteristics of Financial Positions* > *Edit Secondary Data Source for Descriptive Characteristics*.
- If you are working in an accounting system in which you use source data aggregation, all aggregation processes must be complete before you trigger the *Post External Business Transactions* (`/BA1/B1_PBTP_FDB_SL`) process. It does not matter whether you have aggregated accounts or not. As with the aggregation processes, you have to schedule this process separately for all legal entities and source systems.
- All dimensions of financial statement entities and partner organizational units must be granularity characteristics in the aggregation object in the SDL.
- Every financial position class that is an *aggregated financial product* has fixed, defining characteristics that cannot be altered (aggregation object ID, legal entity, accounting system, financial position currency). The descriptive characteristics are also fixed and cannot be altered.
- All characteristics of the aggregation object that are relevant for accounting processes have to be financial position characteristics.

- Cost Accounting Processes
  The system processes aggregation-specific cost accounting events for the calculation of internal costs for periods.
- Financial Statement Preparation
  Both an aggregation-specific balance object category and a balance object class are provided. You can use assigned balance object types and balance subobject types for aggregation.

4.5 Impairment Processes (FS-BA-PM-IP)

Use

This component provides functions for the impairment of financial transactions and financial instruments that are measured at amortized cost under IFRS. It comprises the following processes:

- Impairment Event Processing [page 966]
- Determination of Impairment Attributes [page 968]
- Calculation of the Percentage Expected Loss [page 970]

- Initialization of Impairment [page 973]

You can use the impairment processes in the *Accounting for Financial Instruments* (AFI) scenario.

Integration

The Impairment Server of the *Impairment Methods (FS-BA-PM-GM-IM)* component uses the percentage expected loss to calculate the risk provision. The Impairment Server provides the impairment processes and other applications (for example, accounting processes) with results from the calculation of the risk provision, the write-down and write-off, and the unwinding. For more information, see Impairment Server [page 661].

Enterprise services are available for communication between an impairment workplace (UI) and the impairment components in Bank Analyzer.

More Information

For more information, see the enterprise services documentation for *banking services from SAP* on SAP Help Portal at http://help.sap.com/bankingservices. Under *Integration*, open the enterprise services documentation and choose:

- *Enterprise Services in SAP for Banking* > *Financial Instruments Calculation and Valuation* > *Financial Instrument Impairment Calculation*
- *Enterprise Services in SAP for Banking* > *Financial Instruments Analytical Results Management* > *Financial Instrument Impairment Attribute Values Analytical Result*
- *Enterprise Services in SAP for Banking* > *Financial Instruments Analytical Results Management* > *Financial Instrument Impairment Expected Cash Flow Analytical Result*
- *Enterprise Services in SAP for Banking* > *Financial Instruments Analytical Results Management* > *Financial Instrument Impairment Expected Loss Analytical Result*

4.5.1 Impairment in Accounting for Financial Products

```
SAP_BA1_SHOW_IMPRMT, SAP_BA1_USER_IMPRMT, SAP_BA1_EXPERT_IMPRMT,
SAP_BA1_TECH_IMPRMT_WEBSERVICE, SAP_BA1_USER_IMPRMT_WEBSERVICE
```

Use

Users with this role in the Financial Accounting for Banks scenario determine impairment key figures, such as risk provision and write-down for financial products. This includes executing impairment processes used to determine expected cash flows, expected loss, or the free line for financial products, for example.

The following typical activities are associated with this role:

- Impairment Display Role (`SAP_BA1_SHOW_IMPRMT`)
  This example role contains authorization for various display functions in the context of impairment as part of *Financial Accounting for Banks*.
- Impairment User Role (`SAP_BA1_USER_IMPRMT`)
  This example role contains application authorization for transactions in the context of impairment as part of *Financial Accounting for Banks*. This authorization allows the user to execute specific processes in Calculation and Valuation Process Management (CVPM). A business expert usually works with the user role in a production system at a bank.
- Impairment Expert Role (`SAP_BA1_EXPERT_IMPRMT`)
  This example role contains Customizing and application authorization in the context of impairment as part of *Financial Accounting for Banks*. This authorization allows users to make settings in Customizing and execute transactions, for specific CVPM processes, for example. A development /IT expert usually works with the expert role in the development system at a bank.
- Enterprise Services for Impairment
  In the context of impairment for the *Financial Accounting for Banks* scenario, some enterprise services are available, which write data to and read data from the Results Data Layer (RDL). The two example roles listed below contain the name of these enterprise services, which are assigned to the authorization object `S_SERVICE`:
  - Asynchronous (`SAP_BA1_TECH_IMPRMT_WEBSERVICE`)
    This example role contains authorization for the Results Data Layer and for contracts in the Source Data Layer. Only asynchronous enterprise services are assigned to this role.
  - Synchronous (`SAP_BA1_USER_IMPRMT_WEBSERVICE`)
    This example role is assigned to the authorization object `F_BAMI_IMP`, which is required to simulate an impairment calculation. This example role also contains authorization for the Results Data Layer and for contracts in the Source Data Layer. Only synchronous enterprise services are assigned to this role.

Integration

You also have to assign the user an accounting role, for example `SAP_BA1_EXPERT_PAIA_PML` - *Expert in Accounting and Profitability Analysis*.

Configuration

You have to copy the selected role before you can enter values for the authorizations:

- Display role
  Ensure that the field values for the results data area and result types of the Results Data Layer are completed in the authorization objects.
- User role/expert role
  Ensure that the field values for the step sequences in the CVPM processes, as well as for the results data area and result types of the Results Data Layer are completed in the authorization objects.

You have the following options for assigning a user an individual role:

- You can directly assign the individual role to the user.
- You can group the individual role in a composite role with other individual roles, and then assign the composite role to the user.

4.5.2 Impairment Event Processing

Use

The system reads new application events that are relevant for impairment and, if required, determines which financial transactions and financial instruments are affected by the relevant event. This process determines a minimum date that is used as the basis for creating the worklist in the subsequent Determination of Impairment Attributes [page 968] process.

The processing of impairment events is configured in the Calculation and Valuation Process Manager (CVPM).

Prerequisites

- You have set an SDL time stamp. For more information, see Set the SDL Time Stamp [page 367].
- You have configured the `/BA1/RRI_IEP` – *Process Impairment Events* process in the Calculation and Valuation Process Manager (CVPM). For more information, see Calculation and Valuation Process Manager (CVPM) [page 1622].
- You have made the following settings in Customizing for *Bank Analyzer*:
  - *Source Data Layer* > *Primary Objects* > *Flow Data* > *Operational Events*
  - *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Break Down Business Partner into Financial Transactions & Fin. Instruments*
    You have to make settings in this Customizing activity to be able to process operational events for changes to business partner data.
  - *Processes and Methods* > *Application Event Management*:
    - *Define Application Event Types*
    - *Derive Application Events from Operational Events*
    - *Derive Application Events from Analytical Events*
    - *Define Derivation Relevance of Application Events*
  - **Optional:** *Processes and Methods* > *Impairment Processes* > *General Settings* > *Define Impairment-Relevant Products*
    In this Customizing activity, you can restrict the products for which you want the system to calculate the impairment.

Process

To run this process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *Impairment Processes* > *Process Impairment Events*:

- You can restrict the run according to **legal entity** and **accounting system**, and optionally according to the **source system** of the operational event.
- You also specify the **process partition** that was used when the time stamp was set.

When you execute the process, the following activities run in the system:

1. **Read application events**
   The process reads all the application events for impairment that have not yet been imported, and that match the process parameters, up to the SDL time stamp of the selected process partition. The process also selects application events for which no legal entity is specified.
2. **Break down business partner event**
   If an application event is derived from an operational event that relates to the business partner (for example, if there is a deterioration in the rating of a business partner), the system determines the financial transactions and financial instruments in which the business partner adopts the selected role (for example, account holder):
   - If a legal entity is defined on the *Bank Analyzer Attributes* tab page in the business partner and thereby also in the application event, the system uses this as an additional selection criterion. It only selects contracts that are assigned to the legal entity from the process parameters.
   - If there is **no** legal entity defined for an application event, for each legal entity to be processed the system searches for the contracts of the business partner that are assigned to the legal entity from the process parameters. If an application event for a financial instrument does not have a legal entity, it is processed for all the legal entities that are to be processed.
3. **Determine date**
   For each contract, the system chooses the earliest business record date on which there is an application event that has not been imported.
4. **Update status**
   The system sets the status of the application events with legal entity to *Supplied*. For application events without a legal entity, you have to manually set the status to *Supplied* in the monitor for application event management.

Result

The process saves the date determined for each contract as the result. The system uses this information in the subsequent process to update the impairment attributes from this date onwards and generate analytical events.

More Information

Impairment Server [page 661]

Processing of Events in the AFI Scenario [page 684]

4.5.3 Determination of Impairment Attributes

Use

The system creates a worklist of financial transactions and financial instruments for which it determines the impairment attributes, and uses this to generate analytical events. The system later derives accounting events from these analytical events.

The determination of impairment attributes is configured in the Calculation and Valuation Process Manager (CVPM).

Prerequisites

- You have made settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Impairment Processes* > *Determination of Impairment Attributes*.
  In Customizing activity *Edit Modules*, you have created a module that defines how the system is to determine the impairment attributes. For example, if you want to derive the impairment attributes for financial instruments from rating information, you have made settings for the master rating in the Customizing activity *Edit Master Rating Grade*.
- You have configured the `/BA1/RRI_IAD` *Determine Impairment Attributes* process in the Calculation and Valuation Process Manager (CVPM). For more information, see Calculation and Valuation Process Manager (CVPM) [page 1622].
- In the Impairment Server environment, you have defined a result type for storing impairment attributes. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Impairment Methods* > *Define Impairment Server Environments*.
- You have made settings for the storage of analytical events in the Results Data Layer (RDL). You do this in Customizing for *Bank Analyzer* under *Results Data Layer* > *Settings for Flow Results* > *Make Settings for Storage of Analytical Events in RDL*.
- You have successfully processed impairment events. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Impairment Processes* > *Process Impairment Events*. For more information, see Impairment Event Processing [page 966].

Process

To run this process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *Impairment Processes* > *Determine Impairment Attributes*:

- You restrict the run according to **legal entity** and **accounting system**.
- You select an **environment** for the determination of impairment attributes.

When you execute the process, the following activities run in the system:

1. **Create worklist**

The system creates a worklist of financial transactions and financial instruments by selecting all application events for a contract, irrespective of its status, as of the date transferred by the *Process Impairment Events* process. The system compares the contract ID defined for the application event or the business partner with the corresponding information in the contract.

> **Note**
>
> This process creates the worklist irrespective of the source system.

2. **Map event data**
   The system maps the event structure to characteristics and key figures so that the module editor can further process the application event data.
3. **Derive impairment attributes**
   The process calls the module editor to update the values of the impairment attributes of the contract:
   - Impairment status
   - Accruing status
   - Impairment calculation approach
   - Processing mode
   - Significance
   - Write-down adjustment status (WDA)
   - Market conformity status of the contract

   You can adjust the module that controls this process to suit your own requirements.

Reprocessing application events

This process also takes previously imported results into account, which can result in application events being processed again. For example, if you import a correction event, it causes a resetting of the dependent future impairment events that were processed in an earlier run. The results of the corresponding events are adjusted or deleted.

Result

If the processing mode of the last valid version is **not** *manual* and the data record has changed since the last version, the system saves one data record for each application event in the Results Data Layer (RDL). When a contract is created, the process always creates results in the RDL. The system determines the results data area and the result type from the Impairment Server environment that is assigned to the accounting system.

The process generates a corresponding analytical event for an impairment attribute from the RDL data record that has changed since the previous version. This applies only to attributes that are based on characteristics delivered by SAP. The system does not create events for attributes that are based on custom characteristics.

The system uses the analytical events to later derive application events for processing in accounting.

Reversal

You can reverse the processed events. The system derives a reversal application event from an operational reversal event. When this event is processed by the impairment processes, the impairment results of the original event are deleted and the relevant analytical events in RDL are reversed. Impairment attributes are not derived for reversal events.

Example

An example for determining impairment attributes is available in Business Content. For more information, see the Business Content Guide under .

More Information

Impairment Server [page 661]

Processing of Events in the AFI Scenario [page 684]

4.5.4 Calculation of the Percentage Expected Loss

Use

The system determines the percentage expected loss and the percentage write-down for each contract that is relevant for the impairment processes and that is assigned to the *Percentage Expected Loss* or *Lifetime Expected Loss* calculation approaches.

- The percentage expected loss is read from the generic market data based on the imported custom risk provision rates or calculated on the basis of the risk parameters. In the *Lifetime Expected Loss* calculation approach, the process determines the percentage expected loss for each time bucket.
- The percentage write-down is read from the generic market data.

The calculation of the percentage expected loss is configured in the Calculation and Valuation Process Manager (CVPM) [page 1622]. You can schedule the process as a periodic background job.

> **Recommendation**
>
> Since scheduling the calculation as a background job enables better system performance than with an on-the-fly call, we recommend that you schedule the process before the key date valuation.

> **Note**
>
> If **Source Data Aggregation** is activated in your system, the following applies in the *Percentage Expected Loss* calculation approach: For each financial transaction that is assigned to an aggregation object, the process determines the corresponding absolute EL (amount) in addition to the percentage expected loss. If the percentage write-down is not equal to zero, the process also calculates a write-down amount. The source data aggregation processes use the amounts mentioned above to determine the EL amount and the write-down amount for each aggregation object.
>
> For aggregation objects, you cannot call the process on the fly.
>
> The *Lifetime Expected Loss* calculation approach does not support the processing of aggregation objects. Therefore, the relevant financial transaction has to be separated if a contract changes its status to stage 2 (IFRS 9).

Prerequisites

Values of the impairment attributes of all the impairment-relevant contracts are available in the Results Data Layer (RDL). You can determine these values using the Determine Impairment Attributes [page 968] process or import them using enterprise services.

In the generic market data, either the risk provision rates of the selected contracts are available or risk parameters from which the risk provision rate can be calculated (for example, probability of default (PD) und loss given default (LGD)). You can store the PD and LGD for the *Lifetime Expected Loss* calculation approach in the RDL according to the structure of the maturity band.

You need to import the credit conversion factor (CCF) for the calculation of the expected loss for the off-balance-sheet part of a contract.

You have made settings in Customizing for *Bank Analyzer* in the following places:

- *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Custom Step Sequences for Analytical Processes*
  You adjust the step sequences of the process `/BA1/RRI_EL_PROCESS` to meet your requirements.
- *Processes and Methods* > *General Calculation and Valuation Methods* > *Impairment Methods* > *Create Maturity Band*
  If you use the *Lifetime Expected Loss* calculation approach, you first need to define the structure of the maturity band.
- *Processes and Methods* > *General Calculation and Valuation Methods* > *Impairment Methods* > *Define Impairment Server Environments*
  You define a result type each for the storage of impairment attributes and expected losses in the RDL. For the *Lifetime Expected Loss* calculation approach, you define a result type each for the contractual effective interest rate and for the expected loss in time buckets.
- *Processes and Methods* > *Impairment Processes* > *Calculation Approach: Percentage Expected Loss* > *Module Editor*
  You define the required flow logic for the process in a module. You store this module for the application `RIEL` in the required step sequence of the CVPM process.
- **Optional:** *Processes and Methods* > *Impairment Processes* > *Calculation Approach: Percentage Expected Loss* > *Derivation*
  You can use a derivation that you create here in the module for the EL process.

If **source data aggregation** is activated in your system, the following conditions also apply:

- Position generation was executed successfully. For more information, see Generating Positions [page 304].
- If you are calculating an EL amount for the off-balance-sheet part of a financial transaction that is assigned to an aggregation object, you first need to determine the free line for financial products. For more information, see Determination of the Free Line for Financial Products [page 388].

Process

To run this process, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *Impairment Processes* > *Calculate Percentage Expected Loss* >:

- Restrict the run according to **legal entity** and **accounting system**.
- Select one of the provided **CVPM step sequences**:
  - `EL`: Calculation of 1-year EL rates for financial transactions or financial instruments
  - `EL_SDA`: Calculation of 1-year EL amounts for aggregation objects
  - `EL_LT`: Calculation of EL rates in time buckets for financial transactions or financial instruments

According to how you have configured the step sequence for the CVPM process, corresponding activities run in the system. The following is an example of the process flow:

1. **Create worklist**
   Using the CVPM step sequence, select which option you want to use to create a worklist.
2. **Read impairment attributes**
   For each contract in the worklist, the system reads the impairment status, the calculation approach, and the processing mode from the RDL. For financial transactions, the system also reads the delinquency bands. If there is no delinquency band, the system uses the band "0" (no delinquency). This data is determined using the time stamp of the selection screen.
3. **Derive subportfolio**
   The system can derive the subportfolio to which a contract belongs from its attributes or from the rating of the business partner, for example, as well as from custom characteristics.
4. **Determine risk provision rate**
   The system derives a risk provision rate for the contract from the generic market data. The system bases this on the subportfolio assigned to the contract and its delinquency band (for financial transactions) or its rating (for financial instruments/securities). If a risk provision rate is not imported for a contract, it is possible to calculate this from risk parameters, in particular from the probability of default (PD) and loss given default (LGD).
5. Optional: **Determine percentage write-down**
   Once it has read the risk provision rate or the risk parameters, the system tries to read a percentage write-down amount from the generic market data depending on the subportfolio and the delinquency band. If there is no amount, the system uses a percentage write-down of zero percent.
6. Optional: **Derive percentage expected loss from risk provision rate**
   In the standard system, one-year risk provision rates are used. However, in the module, you can scale risk provision rates to a different time period.
   Depending on how the process is defined, the percentage expected loss for the contract corresponds to the risk provision rate scaled to the maturity. You can make customer-specific adjustments here, for example a CCF weighting for the free line.
7. Source data aggregation: **Determine absolute EL and absolute write-down**
   When you use source data aggregation, the process determines the EL amount for a financial transaction by adding the write-down amounts to the results of the position generation and multiplying this amount with the percentage EL. To determine the write-down amount, the process multiplies the results of position generation with the percentage write-down.

Result

If the process is run periodically as a background job, the percentage expected loss after write-down and the percentage write-down, if required, are stored in the RDL. The system determines the results data area and the result type from the Impairment Server environment that is assigned to the accounting system.

If the Impairment Server calls the process on the fly during risk provision calculation, the system does not save any results.

Example

An example for calculating the percentage expected loss is available in Business Content. For more information, see the Business Content Guide under .

4.5.5 Initialization of Impairment

Use

You can use this initialization process to put an impairment solution into operation for the first time, or to replace your old solution.

Prerequisites

- The results data from the calculation approaches *Expected Cash Flow* (ECF) or *Percentage Expected Loss* in the Results Data Layer have already been initialized or imported.
- You have executed the Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Impairment Methods*.
- You have transferred impairment key figures (such as write-down) from the legacy system using initialization business transactions.

Impairment attributes

- You have created a module for the *Determine Impairment Attributes* process or have customized a module from the Business Content to suit your requirements.

Delinquency data

- You have created a type of operational event for initialization events that process delinquency in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Flow Data* > *Operational Events* > *Edit Operational Event Type*.

- You have made the following settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Application Event Management*:
  - *Define Application Event Types*
    Here you create an application event type for the processing of impairment events (IEP), which is assigned to the application event category `IMP_INITIALIZE` (Initialization of Impairment).
  - *Derive Application Events from Operational Events*
    Here you assign the operational event type for initialization events to the generated application event type and the application *Processing of Impairment Events* (IEP).
  - You have imported information about the delinquency status of financial transactions as operational events (days past due, for example). The impairment processes require this information to determine the impairment attributes.

Impairment-specific characteristics and accounting key figures

- You have assigned the impairment-specific descriptive characteristics to the financial position class in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *Before Generation* > *Financial Positions* > *Define Financial Position Classes*.
- You have completed the Customizing settings for key date valuation for special periods under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Key Date Valuation* > *Accounting Processes* > *Key Date Valuation for Special Periods*.
- You have created the impairment-specific accounting key figures. You have derived these key figures in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic Settings* > *Posting Rules* > *Derive Posting Key Figures*.

Process

1. **Initialization of impairment attributes**
   You initialize the relevant impairment attributes on the *SAP Easy Access* screen, under *Bank Analyzer* > *Processes and Methods* > *Impairment Processes* > *Initialization of Impairment* > *Initialize Impairment Attributes*.
   The system creates a worklist of financial positions to be initialized. To do this, the process collects all positions from financial transactions and financial instruments relevant for impairment. In the data enrichment step of the Calculation and Valuation Process Manager (CVPM) the process calls the module editor to derive the characteristics from the financial positions. The results are saved in the results data area for impairment attributes in the Results Data Layer (RDL).
2. **Initialization of delinquency bands**
   If the source data aggregation is **not** activated, the system derives application events for the impairment processes synchronously from the imported operational delinquency events. The same applies if the source data aggregation and the enhancement of the source data aggregation are activated in the aggregation view. If you are using source data aggregation, the system writes the imported granularity characteristics (such as the delinquency band) for the financial transaction in the aggregation index for operational events.
   The derived impairment events are processed sequentially by the *Process Impairment Events* and *Determine Impairment Attributes* processes. The system updates the values for the impairment attributes in a contract based on the delinquency band. It saves both the impairment attributes and the delinquency band in the Results Data Layer (RDL).

3. **Initialization of impairment-specific characteristics and accounting key figures.**
   You initialize the impairment-specific characteristics in the descriptive characteristics of the financial position on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *Financial Accounting* > *Periodic Processing* > *Automatic Processing* > *Run Key Date Valuation in Special Period*.
   You initialize the values of the impairment-specific accounting key figures by generating corresponding initialization postings. The initialization postings relate to the following processing categories:
   - For **Risk Provision** and **Unwinding** use the processing category *Credit Risk Adjustment*.
   - For **Write-Down** use the processing category *Write-Down*.

   If you have not been using an impairment solution, the data is generated for the first time when you run the initialization. Otherwise the initialization makes delta postings so that the key figures in the new impairment solution are posted with current values at the time of initialization.

> **Note**
>
> You can only initialize the impairment solution at the end of the year. We recommend that you use special periods. The key date for the characteristic change and the posting date of the initialization posting is the last day of the year. The initialization posting takes place after year-end closing and the manual postings.
>
> The initialization must take place after year-end closing. Once the initialization is complete you cannot make any retroactive postings in the previous fiscal year.

More Information

Impairment Event Processing [page 966]

Determination of Impairment Attributes [page 968]

Deriving Application Events [page 310](Source Data Aggregation)

Regenerating Aggregation Indices [page 303]

4.6 Hedge Processes (FS-BA-PM-HP)

Use

You use this component as part of your hedge accounting processes under the International Financial Reporting Standards (IFRS). It is a tool that you can use to manage and analyze hedging relationships. The data generated by the Hedge Processes can be processed further in Accounting Processes.

Features

- Hedge Accounting Workplace [page 976] (FS-BA-PM-HP-HAW)
  You use this component to create and manage hedging relationships. The Hedge Accounting Workplace uses events to communicate with Accounting Processes. An event is the dissolution of a hedging relationship, for instance.
- Fair Value Effectiveness Test for Hedges [page 1036] (FS-BA-PM-HP-FVH)
  You use this component to test the effectiveness of hedging relationships. This is necessary before a key date valuation in preparation for hedge accounting, for example.
- Cash Flow Hedge Analysis [page 1107] (FS-BA-PM-HP-CFH)
  You use Cash Flow Hedge Analysis to analyze cash flow hedges and to generate hedge proposals that you can then use in the Hedge Accounting Workplace to define new hedging relationships.
- Portfolio Fair Value Hedge [page 1156] (FS-BA-PM-HP-PFV)
  This component contains auxiliary functions used to process portfolio fair value hedges. These include the calculation of portfolio items, the basis for the determination of the hedged item in a portfolio fair value hedge, and the calculation of hedge adjustments for hedged items.

More Information

Accounting Processes (Subledger Scenario) [page 693]

4.6.1 Hedge Accounting Workplace (FS-BA-PM-HP-HAW)

Use

You can use this component to access the functions for the management and processing of hedging relationships. Hedge Accounting Workplace is a supplement to Hedge Accounting. It is part of the Accounting Processes component, which covers accounting processes for IAS hedging relationships. Hedge Accounting Workplace provides the functions required for creating and dissolving hedging relationships. It manages and documents existing, dissolved, or expired hedges, plus hedging relationships that have not yet been designated.

In Hedge Accounting Workplace, you can monitor the hedging relationships from Accounting Processes. This gives you an overview of the risks and the transactions hedged (either partly or completely) under IAS. This enables you to react to changes in the market.

You can use Hedge Accounting Workplace for all operational activities, such as searching for suitable transactions for establishing hedging relationships by means of candidate determination, or filling the header data of the hedge using information from the transactions concerned. The system documents all steps in the operational processing and publication of hedging relationships.

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Call Hedge Accounting Workplace*.

Integration

Hedge Accounting Workplace uses financial positions from Accounting Processes and imports hedge information, such as financial transaction and financial instrument data, from the Source Data Layer (SDL). Hedge accounting takes place in the *Accounting Processes* component. For more information, see Accounting Processes (Subledger Scenario) [page 693].

The system calls the Fair Value Effectiveness Test component to test the prospective effectiveness of hedges. For more information, see Fair Value Effectiveness Test [page 1036].

To manage macro cash flow hedging relationships, the system access the hedge proposals that were generated in the Cash Flow Hedge Analysis component. For more information, see Cash Flow Hedge Analysis [page 1107].

Hedge Accounting Workplace provides the data for all hedging relationships. This ensures that hedging relationships are mapped consistently in *Bank Analyzer*. The system uses internal business transactions to ensure that the hedging relationships in the Hedge Accounting Workplace and their financial positions from Accounting Processes are consistent. In the Hedge Accounting Workplace you use the *Dispatch* function (during the designation process) and the *Dissolve* function (during the dissolution process) to generate the business transactions for hedges. These business transactions are then posted in Accounting Processes. Accounting Processes uses events to inform the Hedge Accounting Workplace that the hedges have been processed successfully. The Hedge Accounting Workplace reacts in the same way to events in Hedge Accounting. For example, it reacts in the same way to the dissolution of hedging relationships due to position outflows or because the hedge is ineffective.

> **Note**
>
> The Hedge Accounting Workplace is integrated into the generic concept of *Bank Analyzer*. For this reason, you must make sure that the settings you use in the Hedge Accounting Workplace are compatible with those used in other Bank Analyzer components. For example, you can use only those fields you have defined in the Source Data Layer (SDL) as custom fields.

Features

The following functions are available:

- Designation of hedging relationships [page 987]
- Management of micro fair value hedging relationships [page 999] and portfolio fair value hedging relationships [page 1005]
- Management of macro cash flow hedging relationships [page 1019] and processing of hedge proposals [page 1029]
- Prospective effectiveness test [page 1020] to measure the future effectiveness of hedging relationships
- Information enrichment and validation [page 993] for hedging relationships
- Automatic processing of events in financial position management, such as the dissolution of hedging relationships
- Display of hedging relationships, the results of effectiveness tests, and logs

4.6.1.1 Worklists

Definition

Worklists contain financial transactions and financial instruments that are used to create hedging relationships (see Hedging Relationships [page 987]) in the Hedge Accounting Workplace.

Use

Worklists contain the data that is used in the Hedge Accounting Workplace. They contain financial transactions and instruments that you can use to create hedging relationships between hedged items and hedging instruments.

The Hedge Accounting Workplace contains the following types of worklist:

- **HM** : Set of potential hedged items and hedging instruments for fair value hedges.
- **HP** : Buffer for the hedged items and hedging instruments from the hedge proposals (see Hedge Proposals [page 1029]) that you import from Cash Flow Hedge Analysis to change macro cash flow hedges.
- **HM_VP** : Set of hedging instruments for portfolio fair value hedges

The worklist is processed in the following steps:

1. Define worklists (in Customizing)
   You define the data sources and selection criteria that are used to fill the worklists with data. For more information, see Defining Worklists [page 981] .
2. Fill worklists (in the application)
   The system fills the worklist with the data from the financial positions containing the hedged items and hedging instruments using the selection criteria you specified. For more information, see Filling Worklists with Data [page 985] .
3. Import data to the pools
   For performance reasons, you import data from worklists into pools that exist at runtime only. Once you have done this, you can use the financial transactions and instruments in the worklists to create hedging relationships. Pools contain a selection of the positions that are in a worklist.
   You use separate pools for hedged items and hedging instruments. You use selection criteria to define which data is imported into which pool.

Structure

Data Sources

The set of data records in the worklist is described by a primary data source (*worklist* type, multiple selection) and any number of secondary data sources (SDS).

When you choose the secondary data sources, make sure that the input parameters of the secondary data source match the output parameters of the primary data source, or those of previous the secondary data source.

Selection Criteria

You can enter your own selection criteria for the generation of the worklist. By using selection criteria, you influence the performance of the Hedge Accounting Workplace, since the selection characteristics define which data is added to the worklist.

For example, you can use selection criteria to define the properties of candidates for hedging relationships. For instance, you can specify that the candidates for hedged items must have the product type *loan* .

Note that the system does not use the selection criteria for the secondary data source, and that the quality of the selection characteristics varies in terms of the extent to which they restrict the volume of data that is selected.

> **Example**
>
> We recommend that you use the characteristics /BA1/C40FTRAN (for financial transactions) and /BA1/C41FINST (for financial instruments), instead of /BA1/C11EXTCON (external transaction ID for financial instruments and financial transactions), to select unhedged financial positions.

Data Basis

The system generates a table that contains the content of the worklist.

Primary and Secondary Worklists

You can define a secondary worklist for an existing worklist. The secondary worklist has its own definition, which is independent of the definition of the primary worklist. When the system compiles the fields of the secondary worklist, the fields of the primary worklist are used first, and any secondary worklist fields that are different are then added. If a field is included in both the definition of the primary worklist and the definition of the secondary worklist, the selection criteria of the primary worklist apply for this field.

See also:

Management of Worklists [page 979]

4.6.1.1.1 Management of Worklists

Use

You use this function to create, fill, and display worklists.

You can display the definitions of existing worklists and change their selection criteria. You can also start the process that fills worklists with data. By filling the worklists with data, you ensure that the data you use in the Hedge Accounting Workplace is up to date. For example, you need to do this if you have changed the selection criteria for a worklist.

Other functions are available that you can use to manage worklists. For example, you can create folders to group related worklists. It is useful to create separate folders for the worklists for hedging relationships and for the worklists for hedge proposals. You can use Drag&Drop to move the worklists from one folder to another. You can use filters to select which categories of worklists are displayed, and for which time periods.

Activities

Using the *Edit Worklists* Transaction

Subledger Scenario

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Edit Worklists*.

Merge Scenario

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Accounting: Merge Scenario* > *Service Functions for Risk Basis* > *Hedge Accounting Workplace* > *Edit Worklists*.

The system displays an overview that contains all the worklists, their statuses, the number of data records they contain, and their time stamps.

Using the Worklist Repository in the Hedge Accounting Workplace

You can display the worklists in the Hedge Accounting Workplace as follows:

1. Call the Hedge Accounting Workplace [page 976].
   The system displays a selection screen.
2. Enter the selection criteria that you want to use to choose the worklist, and choose  .
   The system displays the Hedge Accounting Workplace.
3. In the navigation structure, choose *Worklist Repository*.
   The system displays the worklists that were selected.

> **Note**
>
> Note that in the worklist repository in the Hedge Accounting Workplace, you can only display worklists. You can create and assign worklists, and change the worklist definition, in the transaction *Edit Worklists* only.
>
> You can import worklists into pools only in the worklist repository of the Hedge Accounting Workplace.

Functions in the Context Menu of the Worklist and Directories

Action	Function
Generate Worklist	The system generates a worklist. For more information, see Defining Worklists [page 981].
Directory > *Generate New Directory*	The system creates a new directory on the next level.
Worklist Definition > *Display*	The system displays the definition of the worklist.
Worklist Definition > *Change*	The system displays the definition of the worklist in change mode.

Action	Function
Fill > *Direct Processing*	The system adds data tot he worklist in direct processing mode. For more information, see Filling Worklists with Data [page 985].
Fill > *In Job*	The system schedules the filling of the worklist to run as a job. For more information, see Filling Worklists with Data [page 985].
Reporting > *Content of Worklist*	The system displays the worklist data.
Reporting > *Export to Local File*	The system saves the content of the worklist to a local file. The system displays a dialog box in which you can specify the format in which the data is to be saved.
Reporting > *Application Log*	The system displays the application for the process in which data was placed in the worklist.
Import in Pools	The system imports the content of the worklist into the pool for hedged items and hedging instruments. The system displays a dialog box in which you specify which data is to be selected, and to which pool the data is to be imported (hedged items or hedging instruments).
Goto > *Worklist: Collective Selection*	The system generates multiple worklists at the same time. The system displays a dialog box in which you enter the list of worklists.

Other Functions in Worklist Management

Action	Function
→ *On Main Level*→ *Generate New Directory*	The system creates a new directory for worklists on the main level.
	The system restricts the number of worklists that are displayed.

4.6.1.1.2 Defining Worklists

Use

You can edit hedging relationships in the Hedge Accounting Workplace only if you have made the necessary settings for providing the required data. You do this by defining worklists. In Customizing for *Bank Analyzer*, you

need to have created primary and secondary data sources, and, if required, additional characteristics and key figures, which you use in the definition of worklists.

Activities

The following activities describe how you can define worklists systematically. The worklist category *HM*, which is used for fair value hedging relationships, is used here as an example.

Primary Data Sources

You use primary data sources (PDS) to select financial position IDs and to read the defining and descriptive characteristics of financial positions and the positions themselves. In the worklist definition, you specify only the primary data sources used to select the data for the worklist.

- Primary data sources for the worklist containing financial position IDs

  In Customizing for *Bank Analyzer*, choose *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.
  Create a new primary data source of type 2 *Selection of Worklist* and category *General Primary Data Source*. Select *Financial Position Object IDs* as the server. On the *Characteristics* tab page, select all the characteristics, and use the blue arrow to add them to window to the left. Then activate the primary data source. For example, you can choose *SFP-IDS* as the name.
  The *SFP-IDS* primary data source, which is added to the worklist, only supplies the financial position object IDs that are valid on the date for which data is selected. Note that two different dates are provided for the query that is sent to the primary data source:
  - The *key date* is the *posting date* from the external business transaction (see Posting Date and Value Date [page 764]).
  - *Date* and *Time* correspond to the system time when the external business transaction data was transferred. This is the point in time at which IAS Accounting is informed about the conclusion of a swap or the purchase of a loan, for example.
    You can restrict the volume of data the system selects by using selection criteria for all the defining and descriptive characteristics of the financial position. When you define primary data sources, you have to make the appropriate settings for the *Financial Position Object IDs* server in Customizing. You can find the characteristics for this selection process in Customizing for the primary data source (PDS) on the *Characteristics* tab page. The quality of the characteristics used to select data (their ability to restrict the amount of data processed) can vary considerably.

    Example

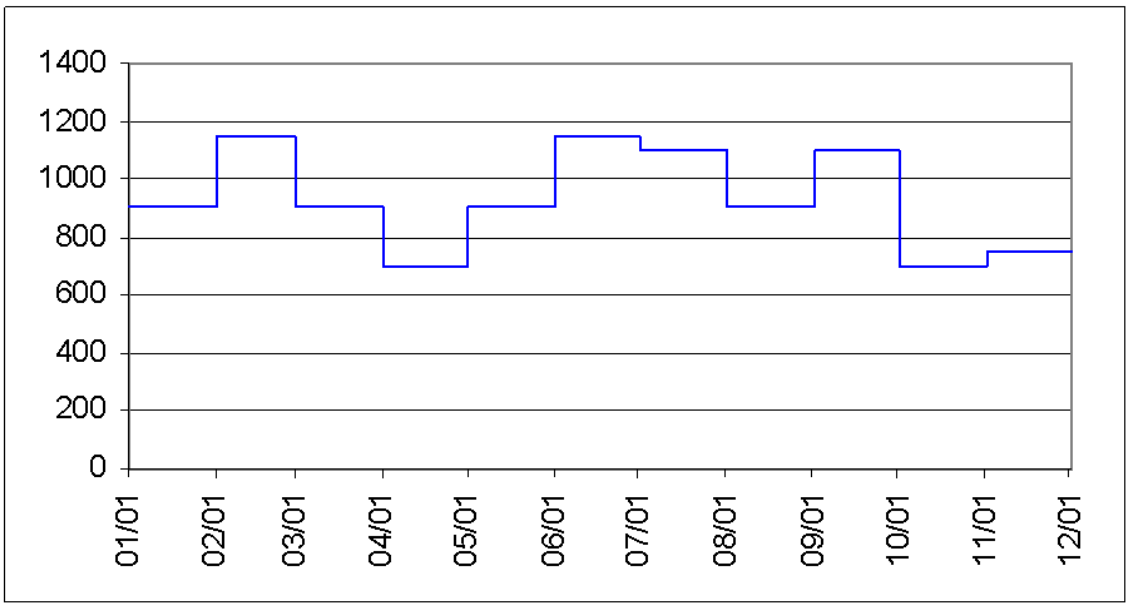
    We recommend that you use the characteristics /BA1/C40FTRAN (for financial transactions) and /BA1/C41FINST (for financial instruments), instead of /BA1/C11EXTCON (external transaction ID for financial instruments and financial transactions), to select unhedged financial positions.

- Primary data sources for the selection of characteristics

  In Customizing for *Bank Analyzer*, choose *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.
  Create a new primary data source of the type 1 *Selection of Single Records* and category *General Primary Data Source SR*. Choose *Financial Position Object Charac.* as the server. On the *Characteristics* tab page, select all the characteristics, and use the blue arrow to add them to window to the left. Then activate the primary data source. For example, you can choose *SFP-CHARACTERISTICS* as the name.

You define the primary data source *SFP_CHARACTERISTICS* using the *Financial Position Object Charac* server. It provides all the defining and descriptive characteristics of the financial position for the financial position ID in question. This means that the system returns a single record. This primary data source can only be included indirectly in the definition of the worklist by means of a secondary data source. However, you still have to define the primary data source first.

- Primary data sources for the selection of positions
  You define the primary data source *SFP_KEYFIGURES* using the *Financial Position Totals* server. The primary data source returns the available balance for the specified financial position object ID, and for the period between the *From Key Date* (/BA1/C55STARKD) and the *To Key Date* (/BA1/C55ENDKD). In the example in the figure below, the primary data source returns the value 700 as the available balance for the entire period January 1 - December 1.



Define Secondary Data Sources

Since you can use any number of steps in the definition of a secondary data source, you only need to define one secondary data source for the Hedge Accounting Workplace.

In Customizing for *Bank Analyzer*, choose *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Data Retrieval* > *Worklist Generation* > *Define Data Sources* > *Edit SDS Definitions for Data Enrichment*.

> **Note**
>
> If the worklist belongs to the *HP* (hedge proposal) category for the hypothetical derivative approach [page 1037], make sure that a secondary data source provides the *Indicator for Calculation of Hedge Adjustment* field

with a value for the fixed-rate leg of the swap. This is required for the *hypothetical derivative approach* effectiveness test method and to measure the transactions in Accounting Processes.

The secondary data source must have the following structure:

Input fields

Characteristic	Description	Use
/BA1/C55ENDKD	To Key Date	Input for PDS FP_TOTALS_KEY
/BA1/C55STARKD	From Key Date	Input for PDS FP_TOTALS_KEY
/BA1/C55FP_ID	Position object ID	Input for PDS FP_TOTALS_KEY and FPO_CHAR

Output fields:

This list is dependent on the defining and descriptive characteristics. Add all the characteristics and key figures here that return the primary data sources *SFP_CHARACTERISTICS* and *FP SFP_KEYFIGURES*. You can find this list in the definition of the primary data sources in Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*. The *Characteristics* and *Key Figs* tab pages indicate which fields are returned.

The Hedge Accounting Workplace also needs the start of the term (/BA1/C11STARTT) and the end of the term (/BA1/C11ENDT) of the financial transactions and instruments from the Source Data Layer (SDL). You must therefore define these two characteristics as additional output fields. These dates are used to check the designation date and the expiry date of the hedging relationship. It may be necessary to supply these dates by means of a different logic if, for example, the listing date of the security does not match the purchase date.

If you integrate function module /BA1HM/DA_SDS_HM_API_GET_INTRA, you can also determine the following values (for each child node) from the SDL:

- Nominal interest
- Reference interest rate
- Purchase/sale indicator
- Name of the condition
- Date from which the interest rate is valid
- Date from which the period is valid
- Repeat period
- Repeat unit

So that the secondary data source can return these values, the output fields must include the relevant characteristics and key figures.

Step sequence:

1. Primary data source *SFP_CHARACTERISTICS*
2. Primary data source *SFP_KEYFIGURES*
3. Condition - IF-branch based on the template category (/BA1/C11PTYPE) Call of Primary Data Source (OTS for financial transaction or financial instrument) to determine start/end of term.
4. Function module /BA1HM/DA_SDS_HM_API_GET_INTRA

Define the Worklist

In Customizing for *Bank Analyzer*, choose *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Data Retrieval* > *Worklist Generation* > *Define Worklists*.

4.6.1.1.3 Filling Worklists with Data

Prerequisites

You have defined and successfully activated a worklist. For more information, see Defining Worklists [page 981].

Context

To be able to process the content of the worklists, you have to fill certain tables with data first. This process fills the worklists with data, and you ca do this in the worklist overview and in the Hedge Accounting Workplace. You can trigger this in the worklist overview and in the Hedge Accounting Workplace, either online, or as a background batch job.

Procedure

1. **Subledger Scenario**

   On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Edit Worklists*.

   Merge Scenario

   On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Accounting: Merge Scenario* > *Hedge Accounting Workplace* > *Edit Worklists*.

   The system displays an overview of your worklists.
2. Select a worklist and choose one of the following functions:
   - *Fill* > *Direct Processing*
   - *Fill* > *In Job*

   Note that if you schedule the filling of worklists with data as a batch job, you must not change any of the characteristics used as the selection criteria for the worklist.
3. Enter the following data:
   - In the *Key Date* field specify the posting date for which the data is to be selected.

- In the *System Date* and *Time* fields enter the system time stamp for which the data is to be selected.

Note that the system time stamp for filling worklists must be after the system time point at which the business transactions for the financial transactions and instruments were posted in Accounting Processes.

4. Set the *With Application Log* checkbox if you want the system to record error messages in an application log.
5. Select one of the following functions to process the run:
   - Normal Run
     The system processes all the data in one step. You use the normal run if you want to process just a small amount of data (fewer than 1000 data records).
   - Select by Package
     The system processes the data package by package, and selects the packages sequentially. This improves system performance during worklist selection, as compared to the normal run.
   - Parallel Selection
     The system processes the data package by package, and as a parallel job. The *Parallel Selection* option is more performance-intensive than the *Select by Package* option. Therefore, we recommend that if you have a large number of data records (more than 30,000) that you use Parallel Selection.

   If you choose *Selection by Package* or *Parallel Selection*, specify the package size.

   Set the *Online Mode* indicator if you want to analyze the selection process.
6. Choose  .

Results

The system has filled the selected worklist with data. You can then import all, or just some, of the data in the worklist into pools, which you can then use in the Hedge Accounting Workplace to create hedging relationships or process hedge proposals.

> **Note**
>
> To check how the worklist was filled, you can display the content of the worklist and the application log. For more information, see Management of Worklists [page 979].

> **Note**
>
> You can generate time bucket data for the hedging instruments of portfolio fair value hedging relationships when the system fills the worklist, and add this data to the hedging relationship.
>
> You do this by defining an HM_VP worklist and assigning it to the hedging relationship. You then fill the worklist from the hedging relationship. When you transfer the worklist, the system calls the mapping data for the hedging instruments automatically.
>
> For more information, see Mapping of Hedging Instruments [page 1171].

4.6.1.2 Hedging Relationship

Definition

Hedging relationships are combinations of one or more hedged items or parts of hedged items and one or more hedging instruments or parts of hedging instruments. These combinations are used to hedge risks and for hedge accounting under the International Financial Reporting Standards (IFRS).

Structure

You use the following parameters in the Hedge Accounting Workplace to map hedging relationships:

- Hedge category
  Specifies whether the hedging relationship hedges fair value or cash flow risks.
- Aggregation level
  Specifies whether the hedged item contains only a few hedged items or a portfolio of hedged items.

You can combine the parameters as follows:

Hedging Relationship	Hedge Category	Aggregation Level
Micro Fair Value Hedging Relationship [page 999]	Fair Value	Micro
Portfolio Fair Value Hedging Relationship [page 1005]	Fair Value	Portfolio
Macro Cash Flow Hedging Relationship [page 1019]	Cash Flow	Portfolio
Cross-Legal-Entity Hedging Relationships [page 1016] (XLE Hedges)	Fair Value	Micro [page 1004] or Portfolio [page 1014]

In addition to the hedge category and aggregation level, the header data of a hedging relationship contains the following:

- Accounting system
- Legal entity
- Currency
- Designation date and end date

For cross-legal-entity hedging relationships, you enter a main legal entity and a legal entity group in the header data.

Each hedging relationship has two statuses: a technical status and a business status. For more information about the statuses of hedging relationships, see Status and Status Changes of Hedging Relationships [page 1021].

The item data contains information about the hedged items and hedging instruments in the hedging relationship. The hedging relationship itself contains only the IDs of the financial positions of the hedged items and hedging instruments. The actual instrument and transaction data is stored in the Source Data Layer (SDL).

4.6.1.2.1 Management of Hedging Relationships

Use

You use the Hedge Accounting Workplace for all operational activities regarding the management of hedging relationships. For more information, see Hedging Relationships [page 987].

Activities

1. Call the Hedge Accounting Workplace [page 976].
   The system displays a selection screen.
2. Enter selection criteria to restrict the data that is selected to the data that you want to process in the Hedge Accounting Workplace. Choose .
   The system displays the Hedge Accounting Workplace.
3. In the navigation structure, choose .
   In the navigation structure, the system displays the list of hedging relationships and the hedging relationship groups you created.
   For each hedging relationship, the system displays the current version and below it the previous versions.
4. Double-click a hedging relationship to choose this relationship.
   On the right side of the screen the system displays the data for the hedging relationship. This includes the header data as well as item data, such as information about the hedged items and hedging instruments and about the effectiveness tests, and the logs for processing the hedging relationship.
   The tab pages contain the following information:
   - Basic data
   - Portfolio or hedged items
     To display financial position balances, choose .
   - Hedging instruments
     To display financial position balances, choose .
   - Portfolio items and a time bucket item overview (for portfolio fair value hedging relationships)
   - Results of the fair value effectiveness test
   - Time series for the retrospective fair value effectiveness test (for cash flow hedging relationships)
   - Events
   - Logs
     To display detailed information, choose a tab page by double-clicking it.

   For previous versions of hedging relationships, the system displays only the basic data and the tab pages for the portfolio or for the hedged items and hedging instruments.

You can use the following functions:

General Functions

Action	Function
Goto > *Online Run for FV Effectiveness Test*	The system starts the online run for the fair value effectiveness test. For more information, see Online Runs [page 1054] and Fair Value Effectiveness Test [page 1036]. The effectiveness of cash flow hedging relationships can be tested only in the hypothetical derivative approach (see Hypothetical Derivative Approach [page 1037]).
Goto > *Online Run for Cash Flow Hedge Analysis*	The system executes an online run of the cash flow analysis (see Cash Flow Hedge Analysis [page 1107]).
Goto > *Hedge Proposal* > *Hedge Proposal: Overview*	The system displays the hedge proposals (see Hedge Proposal [page 1029]). You can enter selection criteria to specify which hedge proposals are displayed.
Goto > *Hedge Proposal* > *Reverse Hedge Proposal*	The system reverses the selected hedge proposal.
Goto > *Processes Hedges*	The system updates the business transactions for a hedging relationship. This function can be used only for hedging relationships that have an interim status. For more information about this status, see Status and Status Changes of Hedging Relationships [page 1021].
	The system creates a new hedging relationship.
	The system updates the display in the navigation structure.
	The system displays a list of all hedging relationships that were selected when the Hedge Accounting Workplace was called.

Action	Function
	The system displays a list of all events sent from Accounting Processes to the Hedge Accounting Workplace. Note that the system displays only events for hedging relationships that were selected when the Hedge Accounting Workplace was called. To process events in the Hedge Accounting Workplace, choose  . In the *Processed in Hedge Accounting Workplace* column, the system displays information as to whether an event has already been processed. If this column contains an `X`, the event has already been processed. The dissolution (`DISSO`) event type and reason code `INEFF`, for example, mean that the hedging relationship has already been dissolved in the Hedge Accounting Workplace. To display the financial position data for an event (if this data is available), double-click the event in question.
Double-Clicking a Hedging Relationship	The system displays the header and item data for the hedging relationship.

Functions in the Context Menu of Hedging Relationships and Hedging Relationship Versions

Action	Function
Change	The system displays the header and item data for the hedging relationship. You can change the attributes of the hedging relationship.
Display	The system displays the header and item data for the hedging relationship.
Use as Copy Template	The system creates a new hedging relationship that has the same header and item data as the hedging relationship that was selected.
Check	The system executes validation for the hedging relationship (see Enrichment and Validation of Information [page 993]).
Delete	The system deletes the hedging relationship. You can delete only those hedging relationships that are not in the hedging relationship worklists.

Action	Function
Additional Functions > *Long Text* > *Display*	The system displays the long texts that you assigned to the hedging relationship. In the dialog box, you define whether the system displays texts relating to the designation or dissolution of a hedging relationship, or the general texts.
Additional Functions > *Long Text* > *Change*	The system opens the long texts that you assigned to the hedging relationship in change mode.
Update Business Transactions (for cash flow hedging relationships only)	The system updates the business transactions for a hedging relationship.

Functions available after you have called a hedging relationship

Action	Function
	You can switch to change mode or display mode.
	The system deletes the hedging relationship.
	The system simulates the dispatch of the hedging relationship.
	The system executes validation for the hedging relationship.
	The system tests the effectiveness of the fair value hedging relationship.
	The system updates the data displayed for the hedging relationship.
	The system executes information enrichment for the header and item data for the hedging relationship (see Standard Information Enrichment [page 994]).
	The system dispatches the hedging relationship, without generating secondary business transactions.
	The system dispatches the hedging relationship.
	The system reverses the hedging relationship.
	The system dissolves the hedging relationship.

Action	Function
	The system displays the posting results for the hedging relationship. You can display the financial position for each line item.
	The system displays the long texts that you assigned to the selected hedging relationship.
	The system displays or hides the navigation structure.
Goto > *Graphical Version Overview* 	The system displays the versions of the hedging relationships saved in the Hedge Accounting Workplace in a graph. The business and technical time stamps are used as the coordinates. For more information about these versions, see the Tabular Version Overview.
Goto > *Tabular Version Overview* 	The system displays the versions of the hedging relationships saved in the Hedge Accounting Workplace in chronological order in a table. The tables lists the status, the technical and business status, and the validity date of the versions.
	The system branches to the *Initial Generation Run* or *Period-Start Run for Portfolio Items*. The test mode and the hedging relationship ID are entered by default; in the case of the initial generation run, the portfolio item environment and the name of the run are also entered by default. You can access this screen only in the status *Initial (20)*; in the subledger scenario, it can also be accessed in the status *Initial Adjustment (25)*.
	The system branches to the *Portfolio Item Run (Merge Scenario)* or *Period-End Run for Portfolio Items*. The hedging relationship ID, process category, and test mode are entered by default. You can access this screen only in the status *Designated (52)*.
 (*Hedging Instruments* tab page in portfolio fair value hedging relationships)	The system displays a dialog box in which you use the *Candidate Determination* or *Use Existing Worklist and Replace Hedging Instruments* functions to select the required worklist. The system finds suitable hedging instruments for the portfolio fair value hedging relationship, and displays them in a list. For more information, see Automatic Proposal for Hedging Instruments [page 1011].

Action	Function
 (*Hedging Instruments* tab page) (*Hedged Items* tab page in micro fair value hedging relationships)	The system updates the currently designated nominal value and the current total volume of the selected financial transactions. If you want to display the current total volume or the currently designated nominal value, select at least one financial transaction in the list and choose *Update Volumes*. If you want to call the current total volumes or the currently designated nominal amounts of all the listed financial transactions, choose the arrow on the *Update Volumes* button and select *Update All Volumes*.
 (*Hedging Instruments* tab page in portfolio fair value hedging relationships)	The system displays a dialog box in which you can trigger the mapping of hedging instruments to time buckets (see Mapping of Hedging Instruments [page 1171]). You can execute the run sequentially or in parallel.

> **Note**
>
> Note that not all of these functions are available for all hedging relationships. For example, you can call only the *Use as Copy Template* and *Display* functions for hedging relationship versions.

See also:

Designating Micro Fair Value Hedging Relationships [page 1000]

Creating Micro Fair Value Hedging Relationships Automatically [page 1001]

Dissolving and Reversing Micro Fair Value Hedging Relationships [page 1003]

Initial Designation of Portfolio Fair Value Hedging Relationships [page 1007]

Redesignating and Dissolving PFV Hedging Relationships [page 1009]

Calling the Prospective Effectiveness Test [page 1020]

4.6.1.2.2 Enrichment and Validation of Information

Use

The Hedge Accounting Workplace has many functions that you can use to enrich and validate the data from hedging relationships.

You can use information enrichment, for example, to ensure that the header and item data for hedging relationships is consistent with the data for the hedged items and hedging instruments, or to display in Reporting information for hedging relationships that are not relevant for valuation. The functions for validation and for checking consistency help to ensure that the Processes and Methods component of Bank Analyzer can further process the hedging relationships.

Features

Information Enrichment

Standard Information Enrichment [page 994]

Custom Information Enrichment [page 995]

Validation and Consistency Check

Custom Validation of Hedging Relationships [page 996]

Consistency Check for Hedging Relationships [page 998]

4.6.1.2.2.1 Standard Information Enrichment

Use

You use this function to fill certain fields for basic data, hedging instruments and hedged items in a hedging relationship.

Prerequisites

You have created at least one hedging relationship. The system contains at least one hedging instrument or hedged item.

In Customizing for *Bank Analyzer*, under *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Hedging Relationship Management* > *Creation of Hedging Relationships* > *Define Default Field Value Variants for Information Enrichment* you have entered fixed values for the fields whose content cannot be derived.

In Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Hedging Relationship Management* > *Creation of Hedging Relationships* > *Edit Tolerance for Determination of Subsequent/Simultaneous*, you have defined under which conditions the start date and designation date of a hedging relationship are considered to be the same.

Features

- Reference currency: When the system determines the reference currency, it assumes that the reference currency is the currency that does not occur in multiples of two when viewed across all financial position objects.
- Designation date: The latest start date of all the transactions is used as the designation date.

- Concurrent/subsequent characteristic: The designation date must be filled for this enrichment process. The start date is compared with the designation date for every transaction. The tolerance value stored in Customizing is read, and the *Concurrent/Subsequent Designation* indicator is set accordingly.
- End date of hedging relationship: The earliest end date of a transaction is taken to be the end date of the hedging relationship.
- Legal entity: The system uses the legal entity of the transaction. If there is more than one legal entity at transaction level (the hedging relationship is a cross-legal-entity hedging relationship) the system checks whether the legal entity group contains the legal entities. An error message is raised if the hedge is not a cross-legal-entity hedge.
- Variant for default field value: You can use one of the variants for default values that you stored in Customizing. The fields defined there are filled with the relevant values

Activities

In the hedge repository of the Hedge Accounting Workplace, select a hedging relationship and choose *Edit* > *Info Enrichment*.

See also:

Custom Information Enrichment [page 995]

4.6.1.2.2.2 Custom Information Enrichment

Use

You can use this function to fill custom fields in the header or item data of a hedging relationship automatically. You call this function directly in the Hedge Accounting Workplace (see Addition of Information to Standard Fields [page 994]).

Prerequisites

- If you want to use your own function module to transfer values, you have to program and register the function module beforehand. You register the function module in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Hedging Relationship Management* > *Custom Enrichment of Information and Validation* > *Function Modules for Modules* > *Assign Function Modules of the Function Library*.
- If you want to derive the data to fill the fields you have to define a derivation in Customizing You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Hedging Relationship Management* > *Custom Enrichment of Information and Validation* > *Derivations*.

- In all other cases, you can edit a module directly. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Hedging Relationship Management* > *Custom Enrichment of Information and Validation* > *Modules* > *Edit Modules* > *Register Modules*.
- In Customizing for *Bank Analyzer* you have assigned your module to a hedging relationship group under *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Hedging Relationship Management* > *Custom Enrichment of Information and Validation* > *Modules* > *Assign Modules*.

Features

You can fill fields automatically with the following data:

- Fixed values
- Fixed values linked to conditions
- Values derived using the derivation tool
- Custom function modules
- Values from the item data of the hedging relationship for the header data, and vice versa

Activities

1. Call the Hedge Accounting Workplace [page 976].
   The system displays a selection screen.
2. In the hedge repository of the Hedge Accounting Workplace, select a hedge relationship and choose *Edit* > *Info Enrichment*.
   The system displays a dialog box.
3. Set the *Add Custom Information* checkbox and choose ✔.
   The system displays a dialog box.
4. In Customizing, select another defined information enrichment and choose ✔.
   The system enriches the information and changes the fields for the hedging relationship. The system records these activities in a log.

4.6.1.2.2.3 Custom Validation of Hedging Relationships

Use

You can use this function to check the fields for a hedging relationship automatically. This affects the header data and the item data. You use this function to ensure that the data for a hedging relationship is consistent before you designate the hedge.

Custom validation enhances the visual check and the consistency check (see consistency check [page 998]) when hedging relationships are saved in the Hedge Accounting Workplace.

Prerequisites

- If you want to use your own function module to check the data, you have to program and register this beforehand. You register the function module in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Hedging Relationship Management* > *Custom Enrichment of Information and Validation* > *Function Modules for Modules* > *Assign Function Modules of the Function Library*.
- If you want to use a derivation to check the information, you have to define a derivation in Customizing. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Hedging Relationship Management* > *Custom Enrichment of Information and Validation* > *Derivations*.
- In all other cases, you can edit a module directly. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Hedging Relationship Management* > *Custom Enrichment of Information and Validation* > *Modules* > *Edit Modules*.
- In Customizing for *Bank Analyzer* you have assigned your module to a combination of hedging relationship category and hedging relationship group under *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Hedging Relationship Management* > *Custom Enrichment of Information and Validation* > *Modules* > *Assign Modules*.

Features

The system uses your custom validation automatically when you call one of the following functions in the Hedge Accounting Workplace:

- Check
- Simulation of Dispatch
- Dispatch

You have can check the field values by comparing them to:

- Fixed values
- Fixed values linked to conditions
- Values derived using the derivation tool
- Custom function modules
- Values from the item data of the hedging relationship for the header data, and vice versa

Incorrect values can be identified by an error message in the log.

If the validation finds errors, the hedging relationship cannot be designated.

> **Note**
>
> Although it is possible to overwrite the field values of hedging relationships using a validation, the system does not accept such changes for the hedging relationships in the Hedge Accounting Workplace.

Activities

In the hedge repository of the Hedge Accounting Workplace, select a pending hedging relationship and choose one of the following options:

- 
- 
- 

The system displays a log of the custom validation. The hedging relationship fields remain unchanged.

4.6.1.2.2.4 Consistency Check for Hedging Relationships

Use

You can use this function to check whether modified hedging relationships are consistent, and whether it is still possible to use processes such as the fair value effectiveness test even though the hedging relationships were changed.

The consistency check is applied automatically when you change and save hedging relationships in the Hedge Accounting Workplace.

The system checks whether the data in the fields relevant for measuring the hedging relationship have changed. If so, the hedging relationship is treated as inconsistent, and cannot be saved.

Integration

When the system checks whether the data is consistent, it checks all the fields that you assigned in Customizing to a combination of consistency level and consistency group.

- The **consistency level** defines how the characteristics are used. SAP provides consistency levels for the calculation of qualified positions and for the fair value effectiveness test. There is also a basic consistency level, which contains the fields forboth the calculation of qualified positions and the fair value effectiveness test.
- The **consistency group** defines the hedging relationship category. SAP provides consistency groups for micro fair value hedges, portfolio fair value hedges, and macro cash flow hedges.

You can add your own custom fields for the consistency check. You enter this setting Customizing for Bank Analyzer under *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Basic Settings* > *Assign Customer-Specific Fields to Consistency Groups* .

Example

For portfolio fair value hedging relationships, the system checks all the fields that are used to determine portfolio items. If the hedging relationships are inconsistent, the portfolio items that were already calculated are no longer appropriate for the modified hedging relationship, This means that you have to start the calculation of the portfolio items again.

Consistency Check for Portfolio Fair Value Hedging Relationships

Field Name	Description
/BA1/CH0HEDSTB	Business status of the hedging relationship
/BA1/C11REFINT	Reference interest rate
/BA1/C15BCURR	Reference currency
/BA1/C55ACCSY	Accounting system
/BA1/C55LGENT	Legal entity
/BA1/C47VALFROM	Valid-from date of the hedge data
/BA1/CH0SYNTMS	Time stamp for synchronization with Accounting Processes
/BA1/KXH0HRDSS (complex key figure &H0HRDSS)	Portfolio of hedged items (data sources and selection criteria)
/BA1/CH0XLEGRP	Main legal entity for cross-legal-entity hedges

4.6.1.2.3 Micro Fair Value Hedging Relationship

Hedging relationships [page 987] that comprise only a small number of similar hedged items and hedging instruments. They are used to hedge risks related to the fair value of a financial transaction or a position of financial instruments, and can have an effect on the reported profit.

In micro fair value hedging relationships, the hedged items and hedging instruments have a n:m relationship. All or some of the hedging instruments are used to hedge the hedged items, or some of the hedged items.

See also:

Designating Micro Fair Value Hedging Relationships [page 1000]

Creating Micro Fair Value Hedging Relationships Automatically [page 1001]

Dissolving and Reversing Micro Fair Value Hedging Relationships [page 1003]

Micro Fair Value Hedging Relationship Between Legal Entities [page 1004]

4.6.1.2.3.1 Designating Micro Fair Value Hedging Relationships

Prerequisites

You have defined one or more suitable worklists, and filled them with data (see Worklists [page 978]).

In Customizing for *Bank Analyzer* you have created a hedging relationship group under *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Basic Settings* > *Hedging Relationship Management* > *Edit Hedging Relationship Groups*. You use hedging relationship groups to sort the hedging relationships in the hedge repository of the Hedge Accounting Workplace.

If required, you have created a default field value variant in Customizing. In Customizing for *Bank Analyzer* choose *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Creation of Hedging Relationships* > *Define Default Value Variants for Information Enrichment*.

Context

You create and designate hedging relationships in multiple steps in the Hedge Accounting Workplace.

Procedure

1. **Call the Hedge Accounting Workplace**

   Call the Hedge Accounting Workplace [page 976].

   The system displays a selection screen in which you can enter selection criteria to define which data you use in the Hedge Accounting Workplace. Choose  .
2. **Create a hedging relationship**

   Choose *Hedging Relationship* > *Create*.

   The system displays a dialog box.

   Chose the hedge category *FVH* and the aggregation level *MICRO*. Enter the hedging relationship group.

   Choose  .

   In the hedge repository, the system displays the new hedging relationship under your hedging relationship group.
3. **Import the worklists to the pools**

   In the navigation structure, choose *Worklist Repository*.

   The system displays the worklists that are available.

In the context menu of the required worklist, choose *Import in Pools*.

The system displays a dialog box in which you can enter selection criteria. Choose ✔.

The system displays a dialog box in which you define whether the system is to import the data to the pool for hedged items or to the pool for hedging instruments. Select the pool you require, and then choose ✔.

In the lower half of the screen, the system displays the pool and the financial positions that were selected.

Repeat this process so that you have one pool for the hedged items and one pool for the hedging instruments.

4. **Assign the transactions**

   Drag and drop the required transactions from the pools to the relevant tab pages. You can make multiple selections here. The system detects which financial positions belong together, and transports these to the hedging relationship. You can therefore simply drag just one leg of the swap to the *Hedging Instrument* tab page. The system then automatically finds the other swap leg, and adds it during Drag&Drop. Enter data in the required entry fields. You can do this manually or using information enrichment.

   Choose *Hedging Relationship* > *Save*.

5. **Transfer hedging relationship to hedge accounting**

   Choose *Hedging Relationship* > *Check* to check your entries and, if required, simulate the transfer to hedge accounting by choosing *Hedging Relationship* > *Simulate Designation*. If there are no errors, you can transfer the hedging relationship. To do so, choose *Hedging Relationship* > *Dispatch*.

   > **Note**
   >
   > When you simulate the hedge, the system simulates the generation of the business transaction and the transfer to hedge accounting. However, for technical reasons it is not possible to simulate the derivation of the business transaction (see Updating of Secondary Business Transactions [page 786]).

4.6.1.2.3.2 Creating Micro Fair Value Hedging Relationships Automatically

Prerequisites

- You have created and filled a worklist for both the hedged items and the hedging instruments (see Worklists [page 978]).
- In Customizing for *Bank Analyzer*, you have created a hedging relationship group under *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Hedging Relationship Management* > *Basic Settings* > *Edit Hedging Relationship Groups*. You use hedging relationship groups to group the hedging relationships in the hedge repository of the Hedge Accounting Workplace.
- In Customizing for *Bank Analyzer*, you have created a field mapping definition under *Processes and Methods* > *Hedge Accounting Workplace* > *Hedging Relationships Management* > *Creation of Hedging Relationships*

Define External Hedge ID for Transaction Hedges. The system uses the field mapping definition to identify the transactions that belong to a hedging relationship. The field names for hedged items and hedging instruments must therefore have the same value.

- In Customizing for *Bank Analyzer*, you have created a variant for information enrichment under *Processes and Methods* > *Hedge Accounting Workplace* > *Hedging Relationships Management* > *Creation of Hedging Relationships* > *Define Default Value Variants for Information Enrichment*.
- You can also enter settings for automatic information enrichment and validation. For more information, see Custom Information Enrichment [page 995] and Custom Validation of Hedging Relationships [page 996].

> **Note**
>
> Note that the hedging relationship group specified in the hedge determination (or overwritten using custom information enrichment) must be the same as the hedging relationship group you assigned.

Context

You can let the system create hedging relationships automatically. You use this function if you know exactly which financial transactions and instruments you want to include in the hedging relationships. Identical characteristic values, for example, indicate that transactions belong to the same hedging relationship. The hedging relationships that are created automatically can be designated immediately.

Using a worklist that you defined in Customizing, the system identifies the hedged items and hedging instruments for each hedging relationship. It uses a default value variant to fill the fields, and then creates the hedging relationships. If the *Automatic Release* checkbox has been selected, then the hedging relationship is validated (see also Custom Validation of Hedging Relationships [page 996]). If the validation is completed successfully, then the hedging relationship is designated.

Procedure

1. Call the Hedge Accounting Workplace [page 976].

   The system displays a selection screen.
2. Choose *Goto* > *Hedge Determination*.
3. Use the input help (F4) to select the worklist for hedged items and the worklist for hedging instruments.
4. Choose a legal entity.
5. Select a source worklist. The system searches for the values of the external hedge ID. Using these values, the system searches for the related transactions in the other worklist. For performance reasons, we recommend that you use the smaller worklist as the source worklist.
6. Use the input help (F4) to choose a field mapping variant that you created in the Customizing activity *Define External Hedge ID for Transaction Hedges*.
7. Choose the hedging relationship group. If you use a custom validation, you can define that the validation checks the hedging relationship group.
8. If you want to create a cross-legal-entity hedge, enter the legal entity group.

9. If you want to fill the fields of the hedging relationship whose values are not derived, you must also specify a default field value variant. This is also necessary if you choose *Automatic Dispatch*.
10. To ensure that the hedging relationship is designated, set the *Automatic Dispatch* checkbox.
11. Choose  .

Results

The system displays an application log. The hedging relationships you have created can be found in the hedge repository of the Hedge Accounting Workplace under the hedging relationship group you specified.

4.6.1.2.3.3 Dissolving and Reversing Micro Fair Value Hedging Relationships

Prerequisites

Reversal of the Designation or Dissolution

You can reverse the designation or dissolution of micro fair value hedging relationships only if you have already designated or dissolved them or triggered this designation or dissolution. If you tried to designate or dissolve a micro fair value hedging relationship but this was not successful (see the error messages), the system assigns this relationship an intermediate status. You can remove the hedging relationship from this status by reversing the designation or dissolution. This means that it can then be used for other hedging relationships (see Status and Status Changes of Hedging Relationships [page 1021]).

Context

Dissolution

You use the dissolution function to dissolve micro fair value hedging relationships if the decision was made to dissolve a hedging relationship, for instance. In this function, you can document the reason why the hedging relationship was dissolved. The Hedge Accounting component can also trigger the dissolution of hedging relationships.

Reversal of the Designation or Dissolution

You can reverse the designation or dissolution of micro fair value hedging relationships if these processes are not required, or if they were not successful due to incorrect settings in Customizing, for example. The hedging relationship is reversed on the posting date of the original business transaction. Pending micro fair value hedging relationships can be deleted only; they cannot be reversed.

Procedure

1. Call the Hedge Accounting Workplace.

   The system displays a selection screen.
2. In the Hedge Repository in the Hedge Accounting Workplace, double-click a micro fair value hedging relationship (see Management of Hedging Relationships [page 988]).
3. Choose *Display <–> Change (CTRL + F1)*.
4. Choose one of the following options:
   - Reverse (CTRL + SHIFT + F1)
     Enter a reversal reason.
   - Dissolve (SHIFT + F8)
     Enter a dissolution date.

Results

Depending the status of the hedging relationship, you have either reversed the designation or dissolution, or dissolved the hedging relationship.

Next Steps

Micro Fair Value Hedge (Subledger Scenario) [page 905]

Hedge Accounting Workplace [page 976]

4.6.1.2.3.4 Micro Fair Value Hedging Relationship Between Legal Entities

Definition

Micro fair value hedging relationship that groups hedged items and hedging instruments from multiple legal entities.

Structure

Hedged Items

You can collateralize a hedged item within a hedging relationship partially or fully.

Hedging Instruments

- You can assign hedging instruments to only one specific part of the hedging relationship.
- If you use mirror transactions, these must also be assigned to the hedging relationship.

Legal Entities

- All legal entities involved must be relevant for hedge accounting.
  Relevant legal entities are ones that you have assigned to the hedging relationship using legal entity groups.
- The main legal entity specifies the time stamp for the calculations in all legal entities.
- The main legal entity must contain at least one hedging instrument or one hedged item.

Rejection of the Designation

If the designation in one of the legal entities involved in a micro cross-legal-entity hedging relationship is rejected when internal business transactions are updated, all designations that have already been generated in the other legal entities are reversed. A new designation is then no longer possible.

First, the status of the legal entity that sends the event is set to *Designation Rejected*. The business status of the hedging relationship is set to *To Be Rejected*. Choose  (F7) to reverse the designation of the hedging relationship in the other legal entities. This sets the business and technical statuses of the hedging relationship to *Designation rejected*.

Integration

Key Date Valuation

You run the key date valuation for hedged items and hedging instruments for each legal entity. To do so, you must have posted all mirror transactions and all hedged items and hedging instruments.

More Information

Cross-Legal-Entity Hedging Relationships [page 1016]

Micro Fair Value Hedging Relationship [page 999]

4.6.1.2.4 Portfolio Fair Value Hedging Relationship

Definition

Hedging relationships [page 987] contain a portfolio of hedged items, are used for hedging risks related to the fair value of the financial transaction or position in the financial instruments, and can affect the reported net income.

Portfolio fair value hedging relationships are synthetic hedging relationships and consist of IAS financial positions.

> **Note**
>
> You can create portfolio fair value hedging relationships within a legal entity and between legal entities. For more information, see Portfolio Fair Value Hedging Relationship Between Legal Entities [page 1014].

Use

You can manually link and manage portfolio fair value hedging relationships in the Hedge Accounting Workplace. For example, it contains functions for the designation, dissolution, and reversal of hedging relationships. It also contains functions for enriching information and assigning hedging instruments to a defined portfolio of hedged items. Validation functions are provided or you can create your own. These are used to check hedging relationships before they are released to make sure they are complete and plausible.

Integration

To process a portfolio fair value hedging relationship, carry out the following steps in the specified components.

Start of Initial Valuation Period

1. Set SDL Time Stamp [page 367] (Processes and Methods)
   For hedging relationships across multiple legal entities, the time stamp of the main legal entity indicates the start of the valuation period with the designation date of the hedging relationship as the business key date. After setting the SDL time stamp, you post external business transactions, and update internal business transactions.
2. Create Portfolio Fair Value Hedging Relationship [page 1007] (Hedge Accounting Workplace)
3. Start Initial Generation Run [page 1169] (Portfolio Fair Value Hedge)
4. Assign Financial Instruments [page 1007] (Hedge Accounting Workplace)
5. Carry Out Prospective Effectiveness Tests [page 1036] (Fair Value Effectiveness Test)
6. Publish Portfolio Fair Value Hedging Relationship [page 1007] (Hedge Accounting Workplace)

End of Initial Valuation Period

1. Set SDL Time Stamp [page 367] (Processes and Methods)
2. Set End of Valuation Period [page 1009] (Hedge Accounting Workplace)
   Defines the key date and time stamp for all dependent processes such as the portfolio item run, effectiveness test, and key date valuation.
3. Carry Out Portfolio Item Run [page 1172] (Portfolio Fair Value Hedge)
4. Carry Out Effectiveness Tests [page 1036] (Fair Value Effectiveness Test)
5. Key Date Valuation [page 934]
   The key date valuation for hedging instruments takes place as part of the regular key date valuation.
6. Run Key Date Valuation for Time Bucket Objects [page 942] (Accounting Processes)
   Measures the time bucket objects of a published portfolio fair value hedging relationship. This step is not dependent on the key date valuation for hedging transactions.
   For hedging relationships between multiple legal entities, the system distributes the calculated hedge adjustments to the legal entities. This is carried out based on the Customizing settings, for example, according to the distribution of hedging instruments.

7. Adjust Portfolio Fair Value Hedging Relationships [page 1009] (Hedge Accounting Workplace)
8. Redesignation or Dissolution of Portfolio Fair Value Hedging Relationships [page 1009] (Hedge Accounting Workplace)

> **Note**
>
> The Hedge Accounting Workplace uses financial positions of the *Accounting Processes* component to create the worklist for portfolio fair value hedging relationships. To define the granularity of financial positions for hedging, you assign a selection of defining characteristics for the corresponding financial positions to the financial position types in Customizing for *Bank Analyzer* under *Processes and Methods* > *Accounting for Financial Products* > *Before Generation* > *Financial Positions* > *Define Granularity of Financial Positions for Hedges*.

More Information

- Initial Designation of Portfolio Fair Value Hedging Relationships [page 1007]
- Redesignating and Dissolving Portfolio Fair Value Hedging Relationships [page 1009]
- Adjustment of Initial Portfolio Fair Value Hedging Relationships [page 1011]
- Portfolio Fair Value Hedge (Subledger Scenario) [page 912]
- Portfolio Fair Value Hedges [page 1156] (component documentation)
- Portfolio Fair Value Hedging Relationship Between Legal Entities [page 1014]

4.6.1.2.4.1 Initial Designation of Portfolio Fair Value Hedging Relationships

Prerequisites

- You have generated primary data sources for the definition of the portfolio of hedged items.
- You have set the SDL time stamp [page 367] for the legal entity in which the portfolio fair value hedging relationship is to be created. For cross-legal entity hedging relationships, you have set the SDL time stamp for all relevant legal entities.
- You have posted external business transactions [page 774] and updated internal business transactions [page 786].
- You have defined one or more suitable worklists, and filled them with data.

Context

Portfolio fair value hedges are designated just for the duration of the valuation period. After the key date valuation for time bucket objects [page 942], the hedging relationship has to be either redesignated or dissolved. For more information, see Redesignating and Dissolving Portfolio Fair Value Hedging Relationships [page 1009].

Procedure

1. Call the Hedge Accounting Workplace [page 976].

   The system displays a selection screen.

2. Choose *Hedging Relationship* > *Create*. Choose the hedging relationship category *FVH* and the aggregation level *PORTFOLIO*, and enter your hedging relationship group in the relevant field. Once you have saved this information, the new hedging relationship is displayed in the hedge repository under your hedging relationship group.
3. Make the following entries, which are required for calculating portfolio items:
   1. Enter data in all of the fields on the *Basic Data* tab page.
   2. On the *Portfolio* tab page, define a portfolio of hedged items using primary data sources and appropriate selection criteria.
4. Save your entries.
5. On the *Basic Data* tab page, choose the *Start Initial Generation Run* pushbutton or call the transaction *Start Initial Generation Run*:

   On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Run Administration* > *Create and Schedule* > *Start Initial Generation Run*.

   The system calculates the qualified positions for the initial portfolio fair value hedging relationship.
6. Call the Hedge Accounting Workplace again and open the initial portfolio fair value hedging relationship.

   On the *Portfolio Positions* tab page, the system displays the qualified positions for the following valuation period.
7. Choose the *Portfolio Positions* tab page and make the required settings for the qualified positions for each time bucket. The following options are available for assigning hedging instruments to the hedging relationship:
   - Manual assignment
   - Automatic assignment
   - Scheduling assignment as part of background processing

   For more information about the options for assigning hedging instruments, see Adjustment of Initial Portfolio Fair Value Hedging Relationships [page 1011].

   The *Time Bucket Item Overview* tab page contains an overview the hedging items that are assigned in each time bucket.
8. Validate your entries by choosing *Hedging Relationship* > *Check* and simulate the designation process by choosing *Hedging Relationship* > *Simulation of Des*.

   If there are no errors, you can dispatch the hedging relationship. To do so, choose *Hedging Relationship* > *Dispatch*. You can also schedule this step in a process scheduler for background processing.

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Background Processing* > *Create and Schedule* > *(Re)Designate Initial Portfolio Fair Value Hedges*.

> **Note**
>
> When you simulate the hedge, the system simulates the generation of the business transaction and the dispatch process. However, for technical reasons it is not possible to simulate the derivation of the business transaction (see Updating of Secondary Business Transactions [page 786]).

4.6.1.2.4.2 Redesignating and Dissolving PFV Hedging Relationships

Use

The designation of a portfolio fair value hedging relationship applies only for the duration of the valuation period. After the key date valuation, the hedging relationship has to be either redesignated or dissolved. A portfolio fair value hedging relationship is redesignated or dissolved based on a hedge management decision.

Prerequisites

Redesignation and Dissolution

1. You have set the SDL time stamp [page 367] (subledger scenario) or created a financial reporting process (merge scenario).
2. You have closed the valuation period for the portfolio fair value hedging relationship.
   In the merge scenario, the end date for this period is also the start date (designation date) of the new valuation period. In the subledger scenario, the new valuation period starts one day after the end date of the previous valuation period. This ensures that there are no gaps between the periods.
   You carry out this step in the Hedge Accounting Workplace on the *Basic Data* tab page. You can also trigger this step via background processing, which means that you can schedule the process in a process scheduler.
   Subledger Scenario
   On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Background Processing* > *Define End of Valuation Period*.
   Merge Scenario
   On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Accounting: Merge Scenario* > *Service Functions for Risk Basis* > *Portfolio Fair Value Hedge* > *Background Processing* > *Define End of Valuation Period*.
3. You have carried out the portfolio item run [page 1172].
4. You have executed an effectiveness test [page 1036] and processed the results of this test in Accounting.
5. You have completed the key date valuation [page 934].
6. You have completed the key date valuation for time bucket objects [page 942].

Redesignation

You have defined one ore more suitable worklists [page 978], and filled them with data.

Procedure

1. Call the Hedge Accounting Workplace [page 976].
   The system displays a selection screen.
2. Then choose *Hedge Repository* in the navigation structure and open the *Hedging Relationships Worklists* folder.
   The system displays the hedging relationships that have to be redesignated or dissolved because their valuation period is closed. If you do not redesignate these hedging relationships, they cannot be used in other key date valuations.

Redesignation

In the **merge scenario** the time stamp is taken from the period end date and is used with the period-end items to calculate the period-start items. In the **subledger scenario** proceed as follows:

1. On the *Basic Data* tab page, choose the *Define Start of Valuation Period* pushbutton.
2. Save your entries.
3. On the *Basic Data* tab page choose the *Start Initial Generation Run* pushbutton or on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Run Administration* > *Create and Schedule* > *Period-Start Run for Portfolio Items*.
   The system calculates the qualified positions for the initial portfolio fair value hedging relationship.

Regardless of the accounting scenario, proceed as follows:

1. Call the Hedge Accounting Workplace again and open the initial portfolio fair value hedging relationship.

   > **Note**
   >
   > Note that you cannot change the entries on the *Basic Data* and *Portfolio* tab pages after the initial designation.

2. Choose the *Portfolio Positions* tab page.
   The system displays the qualified positions for the subsequent valuation period.
3. For each time bucket, enter the data required for the qualified positions. The following options are available for assigning hedging instruments to the hedging relationship:
   - Manual assignment
   - Automatic assignment
   - Scheduling assignment as part of background processing

   For more information about the options for assigning hedging instruments, see Adjustment of Initial Portfolio Fair Value Hedging Relationships [page 1011].
   The *Time Bucket Item Overview* tab page contains an overview the hedging items that are assigned in each time bucket.
4. Validate your entries by choosing *Hedging Relationship* > *Check* and simulate the designation process by choosing *Hedging Relationship* > *Simulation of Des*.
   If there are no errors, you can dispatch the hedging relationship. To do so, choose *Hedging Relationship* > *Dispatch*.

You can also schedule this step in a process scheduler for background processing. On the *SAP Easy Access* screen, choose *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Background Processing* > *(Re)Designate Initial Portfolio Fair Value Hedges*.

> **Note**
>
> When you simulate the hedge, the system simulates the generation of the business transaction and the dispatch process. However, for technical reasons it is not possible to simulate the derivation of the business transaction (see Updating of Secondary Business Transactions [page 786]).

Dissolution

1. In the Hedge Accounting Workplace, choose *Hedging Relationship* > *Dissolve*.
2. Enter a dissolution date.

4.6.1.2.4.3 Adjustment of Initial Portfolio Fair Value Hedging Relationships

Use

You can use this function to adjust portfolio fair value hedging relationships with the status *Initial* to the next valuation period by assigning hedging instruments to them. You call this function after you have defined the portfolio of hedged items for the hedging relationship and calculated the portfolio positions.

You can assign hedging instruments manually. However, this is not practical for a large number of hedging instruments, since you have to take into account the portfolio items (see Portfolio Items [page 1158]) of each hedging instrument.

You can use the automatic proposal function for hedging instruments to create hedging relationships so that the portfolio items of the hedging instruments match the portfolio positions of the hedged items as closely as possible.

You can assign all of the hedging instruments proposed by the system, or just some of them, and change the assignment manually afterwards.

Features

You can assign suitable hedging instruments to a hedging relationship as follows:

Assigning Hedging Instruments to the Hedging Relationship Manually

Choose hedging instruments from a worklist in the Hedge Accounting Workplace and assign these to a hedging instrument pool.

The system detects which financial positions belong together, and transports these to the hedging relationship. You can, therefore, simply drag just one leg of the swap to the *Hedging Instruments* tab page. The system then automatically finds the other leg, and adds it.

Assigning Hedging Instruments Automatically

Candidate Determination

In *Find Candidates* function, the system searches for suitable hedging instruments by looking at all the time buckets in the maturity band of the hedging relationship. It starts with the most recent time bucket.

For each time bucket, the system finds hedging items whose portfolio items fit in the current time bucket, and that do not have any portfolio items in the next time buckets. The system starts with the hedging instruments that have the largest nominal, and then searches for the hedging instruments with the next largest nominal.

The system chooses a hedging instrument when the portfolio items for the selected hedging instruments do not exceed the limit required when their nominal value is taken into account. The system considers the portfolio items, the assets, and the liabilities. The total nominal value of the hedging instrument is selected.

The system terminates the search in a given time bucket when the following criteria have been met:

The portfolio items of the hedging instruments already selected cover the portfolio items of the hedged item up to the required lower limit.

If another hedging instrument is selected, this would cause the portfolio items of the hedging instruments to exceed the required upper limit.

When the system processes the next time bucket, it takes into account the hedging instruments that were already selected.

You define the lower and upper limits for the portfolio items for the hedging instruments when you call the *Fill Automatically* function for the automatic proposal of hedging instruments. You define the limits for the portfolio items for the hedged item. For example, you can define that the values in the portfolio items for the hedging instruments can be up to 10% above or 10% below the values of the portfolio items for the hedged instrument.

Use Existing Worklist and Replace Hedging Instruments

If you have already assigned hedging instruments in a different place (such as in a legacy system) and you want to use these assignments, we recommend that you use the *Use Existing Worklist and Replace Hedging Instruments* function. The system uses all the hedging instruments in the worklist that have free nominals. If a hedging instrument no longer has any free nominal, but it is still designated for the same hedging relationship in the prior period, it is taken over into the next period with the same nominal. When you use the *Use Existing Worklist and Replace Hedging Instruments* function, the system removes all the hedging instruments that have already been assigned and replaces them with a new selection.

Background Processing

You can use the transaction /BA1HM/AH_M_PFVH_ADJ *Adjust Initial Portfolio Fair Value Hedges* to schedule the adjustment of initial portfolio fair value hedging relationships in a process scheduler, such as in the Schedule Manager.

This transaction calls the Business Add-In *BAdI: Adjust Initial Portfolio Fair Value Hedges for Next Period* . You can use this Business Add-In (BAdI) to define your own methods for adjusting initial portfolio fair value hedges automatically. The SAP standard implementation is the same as the *Use Existing Worklist and Replace Hedging Instruments* function: The system fills the worklist assigned to the hedging relationship. It then uses all the available nominal amounts of the transactions as hedging instruments.

For more information, see Customizing for *Bank Analyzer* under *Processes and Methods* → *Hedge Processes* → *Portfolio Fair Value Hedge* → *Process* → *BAdI: Adjust Initial Portfolio Fair Hedges for Next Period.*

Activities

Assigning Hedging Instruments to the Hedging Relationship Manually

On the *SAP Easy Access screen* , choose *Bank Analyzer → Processes and Methods → Hedge Processes → Hedge Accounting Workplace → Call Hedge Accounting Workplace.*

Select a worklist in the worklist r *epository in the* Hedge Accounting Workplace (see Management of Hedging Relationships [page 988]).

In the context menu of the worklist, choose *Import in Pools*

Choose the required selection criteria and then specify that the worklist is selected for the hedging instrument pool.

The system displays the hedging instrument pool.

Drag and drop the transactions required from the pool to the *Hedging Instruments* tab page. You can make multiple selections here.

Enter data in the required entry fields. You can do this manually or using information enrichment (see Enrichment and Validation of Information [page 993]).

Assigning Hedging Instruments Automatically

On the *SAP Easy Access screen* , choose *Bank Analyzer → Processes and Methods → Hedge Processes → Hedge Accounting Workplace → Call Hedge Accounting Workplace.*

In the Hedge Repository in the Hedge Accounting Workplace (see Management of Hedging Relationships [page 988]), choose a portfolio fair value hedging relationship.

Choose the *Hedging Instruments* tab page .

Choose  (*Fill Automatically*).

The system displays a dialog box. Double-click the worklist required. The system displays a dialog box in which you can choose between the following options for finding hedging instruments:

Candidate determination

Select the *Find Candidates* checkbox and then choose *Continue* .

In the next dialog box, enter the upper and lower tolerance levels for the portfolio items. Choose *Continue* .

The system finds suitable hedging instruments for the portfolio fair value hedging relationship, and displays them in a list.

Use Existing Worklist and Replace Hedging Instruments function

Select the *Use Existing Worklist and Replace Hedging Instruments* checkbox.

Choose *Continue* .

The system displays all the hedging instruments it took over in one list.

If required, remove the hedging instruments that are proposed from the hedging relationships, and assign other hedging instruments.

Background Processing

If you want to use your own implementation of *BAdI: Adjust Initial Portfolio Fair Hedges for Next Period* , you create this in Customizing for *Bank Analyzer* under *Processes and Methods* → *Hedge Processes* → *Portfolio Fair Value Hedge* → *Process* → *BAdI: Adjust Initial Portfolio Fair Value Hedges for Next Period*

On the *SAP Easy Access* screen, choose *Bank Analyzer* → *Processes and Methods* → *Hedge Processes* → *Portfolio Fair Value Hedge* → *Background Processing* → *Adjust Initial Portfolio Fair Value Hedges*

Use the hedging relationship ID to select one or more portfolio fair value hedge relationships.

Choose *Execute*

More Information

Redesignating and Dissolving PFV Hedging Relationships [page 1009]

Portfolio Fair Value Hedging Relationships [page 1005]

Initial Designation of PFV Hedging Relationships [page 1007]

4.6.1.2.4.4 Portfolio Fair Value Hedging Relationship Between Legal Entities

Definition

Portfolio fair value hedging relationship that groups hedged items and hedging instruments from multiple legal entities.

A portfolio fair value hedging relationship between legal entities can represent, for example, the entire bank portfolio and the corresponding hedging instruments.

Structure

Hedging Instruments

- You can assign hedging instruments to only one specific part of the hedging relationship.
- If you use mirror transactions, these must also be assigned to the hedging relationship.

Time Buckets

- The time buckets must be identical and equidistant for all legal entities.
  You make the time bucket settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Create Maturity Band* .
- The time bucket objects are created for all legal entities of the hedging relationship when it is designated.

Legal Entities

- All legal entities involved must be relevant for hedge accounting.
  Relevant legal entities are ones that you have assigned to the hedging relationship using legal entity groups.
- The main legal entity specifies the time stamp for the calculations in all legal entities.
- The main legal entity must contain at least one hedging instrument.

Integration

Portfolio Items

You have to define a specific portfolio of hedged items for each legal entity. To do so, choose the *Portfolio* tab page for the hedging relationship in the Hedge Accounting Workplace and define separate primary data sources with relevant selection parameters for each legal entity.

For the aggregated view of portfolio items, the system adds the valid and invalid items across all legal entities for each time bucket.

Key Date Valuation

You run the key date evaluation for hedging instruments for each legal entity. To do so, you must have posted all mirror transactions and all hedging instruments.

You run the key date valuation for time bucket objects in an independent process that includes multiple legal entities. To do so, the valuation periods of the legal entities involved must be closed.

Distribution of the Hedge Adjustment

The system determines a hedge adjustment for each time bucket and distributes it to the legal entities involved. The proportion of hedge adjustment HA_{LE} for a legal entity *LE* is calculated as follows:

$$HA_{LE} = HA_{LE}^{HIT} + HA_{LE}^{HIN}$$

The proportion of hedge adjustment of a legal entity for hedged items HA^{HIT}_{LE} and hedging instruments HA^{HIN}_{LE} is calculated as follows:

$$HA_{LE}^{HIT} = \frac{Vol_{LE}^{HIN}}{Vol^{HIN}} \cdot HA^{HIT}$$

$$HA_{LE}^{HIN} = \frac{Vol_{LE}^{HIN}}{Vol^{HIN}} \cdot HA^{HIN}$$

This calculation is based on the proportion of the total hedge adjustment for the hedged items HA^{HIT} and hedging instruments HA^{HIN}, the total volume of designated hedging instruments Vol^{HIN}, and the total volume of hedging instruments designated in a legal entity VOL^{HIN}_{LE}.

More Information

Cross-Legal-Entity Hedging Relationships (Subledger Scenario) [page 1016]

Portfolio Fair Value Hedging Relationship [page 1005]

Editing Maturity Bands [page 1152]

4.6.1.2.5 Cross-Legal-Entity Hedging Relationships

Use

In addition to hedging relationships whose hedged items and hedging instruments are in one legal entity, you can also create hedging relationships between legal entities (XLE hedges). This makes hedge accounting possible at group level, for example. You can use the following hedging relationships as cross-legal-entity hedges:

- Micro Fair Value Hedging Relationship [page 999]
  For more information, see Micro Fair Value Hedging Relationship Between Legal Entities [page 1004].
- Portfolio Fair Value Hedging Relationship [page 1005]
  For more information, see Portfolio Fair Value Hedging Relationship Between Legal Entities [page 1014].

Prerequisites

- You have created at least one group of legal entities. To create a group of legal entities, in Customizing for *Bank Analyzer* choose *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Hedging Relationship Management* > *Creation of Hedging Relationships* > *Cross-Legal-Entity Hedging Relationships* > *Edit Groups of Legal Entities*.
- If you want to use mirror transactions, you must assign these the characteristic `/BA1/CH0FTMIRR`, which is predefined in the system. Also, when you define the worklist for cross-legal-entity hedges, you must add this characteristic as an output field to the secondary data source. You must also provide the value of the characteristic. For more information, see Defining Worklists [page 981].

Features

Designation

When you create a cross-legal-entity hedge, the system first creates a pending version. If you choose *Release*, the system creates a new version in which the designation date is the posting date.

Unlike hedging relationships that exist within one legal entity, when you create a cross-legal-entity hedging relationship you specify the group of legal entities and the main legal entity (in the *Legal Entity* field) as part of the basic data for the hedging relationship. Before the relationship is designated, the Hedge Accounting Workplace

checks whether all the legal entities in the hedging relationship belong to the same group of legal entities specified in the basic data for the hedging relationship.

The hedge is designated in the Hedge Accounting Workplace for all the legal entities in question. For each legal entity in the XLE hedge, the system generates a primary business transaction of the type *Designation* containing all the financial positions that belong to the legal entity. The Hedge Accounting Workplace saves the source document of the designation of the hedging relationships (the key for the business transaction "designation") for each legal entity as an event.

For each legal entity, the internal business transactions are updated automatically when the hedging relationship is designated (see also Update Internal Business Transactions [page 786]). This means that a secondary *Designation* business transaction is generated for each primary business transaction of the category *Designation*. An event informs the Hedge Accounting Workplace whether the hedging relationship was designated successfully.

Once the cross-legal-entity hedging relationship has been designated, it has the business status and technical status *Designated*. All the legal entities involved are also assigned the status *Designated*.

In the case of a micro XLE hedge, the designation can be rejected because the amounts are not available (see Micro Fair Value Hedging Relationship Between Legal Entities [page 1004]).

If an error caused the updating of the internal business transactions to terminate, the legal entity involved is assigned the status *Designated To Be Derived*, and the XLE hedge is given the business status *To Be Derived*. You then have the following options:

- The hedging relationships that are to be derived are displayed in the Hedge Repository [page 988] of the Hedge Accounting Workplace under *Hedging Relationship Proposals*. Select the hedging relationship that is to be derived and choose  to trigger the update of internal business transactions for all legal entities with the status *Designated To Be Derived*.
- The internal business transactions are updated in the legal entities involved, and the designation business transactions are derived. Accounting sends an event to the Hedge-Accounting-Workplace, and the legal entities are assigned the status *Designated* or *Designation Rejected*.

Dissolution or Dedesignation

A **micro XLE hedging relationship** is dissolved at group level due to a hedge management decision, position outflow, or ineffectiveness.

A **portfolio XLE hedging relationship** is dissolved at group level due to a hedge management decision. In the case of position outflows, *individual* hedging instruments are dedesignated. In the case of ineffectiveness, *all* hedging instruments are dedesignated.

Hedge Management Decision

If the decision is made to dissolve the hedging relationship, the system generates a *dissolution* primary business transaction for each legal entity. The Hedge Accounting Workplace saves the source document of the dissolution for each legal entity. When the internal business transactions are updated, a secondary *Dissolution* business transaction is generated for each primary business transaction of the category *Dissolution*. Once the cross-legal-entity hedging relationship has been dissolved, it has the business status and technical status *Dissolved*. All the legal entities involved are also assigned the status *Dissolved*.

Position Outflow and Ineffectiveness

- **Micro XLE hedging relationships**

Position outflows are imported from feeder systems for each legal entity. A possible consequence of a position outflow is that the associated hedging relationship is dissolved as follows:

1. During the *Update Business Transactions* process, the position outflow generates a secondary dissolution business transaction in accounting that dissolves the hedge financial positions for the affected legal entity. The reversal of a position outflow results in the reversal of the dissolution.
2. Information about the derivation of the dissolution is sent from the *Update Business Transactions* process to the Hedge Accounting Workplace as an event.
3. The Hedge Accounting Workplace dissolves the hedging relationship.
   The Hedge Accounting Workplace generates a new version of the XLE hedge and changes the status of the legal entity for which the event was sent from *Designated* to *Dissolved by HA* (Hedge Accounting). The business status of the XLE hedge is changed from *Designated* to *Dissolved*. The technical status is changed from *Designated* to *Dissolved by HA* (Hedge Accounting).
   The hedging relationships that are to be dissolved are displayed in the Hedge Repository [page 988] of the Hedge Accounting Workplace under *Hedging Relationship Proposals*.
4. So that the hedging relationships are dissolved in other legal entities, you must do the following:
   1. In the Hedge Accounting Workplace, choose  .
   2. Choose  .
      The system generates primary dissolution business transactions for the other legal entities and triggers the *update of internal business transactions*. The cross-legal-entity hedges are assigned the business status *Dissolved by HA*, and the other legal entities are also given the status *Dissolved* (except those for which the event was sent).

This procedure for dissolving micro XLE hedges also applies if the hedging relationships are **ineffective**.
The starting point for the hedge dissolution is the main legal entity to which the effectiveness test results are assigned.

- **Portfolio XLE hedging relationships**
  In the case of a **position outflow**, the system dedesignates only the relevant hedging instrument. This means that only a legal entity of the XLE hedge is affected. The hedged positions are dedesignated as follows:
  1. The position outflow generates a dedesignation business transaction in Accounting for the affected hedging instrument.
  2. Information about the dedesignation is sent from accounting to the Hedge Accounting Workplace as an event.
  3. The Hedge Accounting Workplace generates a new version of the hedging relationship and removes the hedging instrument. The status of the portfolio XLE hedging relationship remains *Designated*.

  In the case of **ineffectiveness**, the system dedesignates the hedging instruments of all legal entities as follows:
  1. If the accounting process *Process Results of Effectiveness Test* finds that a portfolio XLE hedge is ineffective, accounting generates a dedesignation business transaction for the hedging instruments for the main legal entity.
  2. Information about the derivation of the dedesignation is sent from accounting to the Hedge Accounting Workplace as an event.
  3. The Hedge Accounting Workplace dedesignates all the hedging instruments for the legal entities. The status of the hedging relationship and the legal entities remains *Designated*. The calculation step for the hedge adjustment determination checks the effectiveness test results during the key date valuation for the time bucket.
  4. The dissolution of the PFVH relationship by the *Process Results of Effectiveness Test* process step can be reversed in the main legal entity on the period-end key date.

Reversal of the Designation and the Dissolution

The reversal of an XLE hedge is either triggered from the Hedge Accounting Workplace, by reversing the position outflows in the legal entities involved, or using the *Reverse Editing of Effectiveness Test Results* function. The system reverses the source documents saved when the relationships are designated or dissolved in each legal entity involved when the designation or dissolution of the hedging relationships is reversed. If a hedging relationship is reversed, the current version is set to *Deactivated*. The previous version then becomes the *active* version.

Activities

1. To create an XLE hedge, call the Hedge Accounting Workplace [page 976].
   The system displays a selection screen.
2. In the Hedge Repository of the Hedge Accounting Workplace, choose  .
3. The system displays the *Create Hedging Relationship* dialog box. In the *Group Leg. Entities* field, specify the group of legal entities for which the cross-legal-entity hedging relationship is to be created. By specifying the group of legal entities, you define that the hedging relationship is a cross-legal-entity hedging relationship.

   > **Caution**
   >
   > After you have saved the data, you cannot change the group of legal entities assigned to the hedging relationship. If you want to do so, you have to create a new hedging relationship.

4. On the *Basic Data* tab page, in the *Legal Entity* field, enter the main legal entity. This legal entity must be one of those included in the group of legal entities that you specified. You can also use a default value variant to provide the value for this field To do this, choose *Info. Enrichment* > *Default Field Value Variant*. You need to test the effectiveness of the hedging relationship in the main legal entity at a later point in time.
5. Assign hedged items and hedging instruments to the hedging relationship.
   For more information, see Designation of Micro Fair Value Hedging Relationships [page 1000] and Initial Designation of PFV Hedging Relationships [page 1007].
6. Save your entries.

4.6.1.2.6 Macro Cash Flow Hedging Relationship

Definition

Hedging relationship [page 987] that contains a portfolio of hedged items and that serves as a kind of protection against risks associated with cash flows.

Use

Together with the Cash Flow Hedge Analysis [page 1107], the Hedge Accounting Workplace supports the processing of macro cash flow hedging relationships in accordance with IAS 39.

You create and release macro cash flow hedging relationships in the Hedge Accounting Workplace. The initial hedging relationships do not contain any hedging instruments. Instead they contain only a primary data source and the selection criteria for the portfolio of hedged items. The system requires these entries in order to execute the Cash Flow Hedge Analysis.

In the Cash Flow Hedge Analysis, you can present in a maturity band the cash flows of a portfolio of hedged items. You can use an existing worklist of hedging instruments to partially or completely assign individual hedging instruments to the portfolio of hedged items. You can then save this assignment as a hedge proposal [page 1029].

In the Hedge Accounting Workplace, you can further process the saved hedge proposals and establish macro cash flow hedging relationships.

See also:

Management of Hedge Proposals [page 1031]

Processing of Hedge Proposals [page 1033]

4.6.1.2.7 Calling the Prospective Effectiveness Test

Use

You can call a prospective fair value effectiveness test for pending hedging relationships from the Hedge Accounting Workplace. The volume of the hedging instruments and hedged items involved can therefore deviate from the volume of the corresponding financial position objects. You can display the results of the effectiveness test and the detail logs of the effectiveness test in the Hedge Accounting Workplace.

Prerequisites

- You have configured the prospective effectiveness test in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness Test*.
- You have entered the parameters for calling the prospective effectiveness test in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Hedging Relationship Management* > *Creation of Hedging Relationships* > *Assign Effectiveness Test Variants to Run Type Data*.
  To start the prospective effectiveness test for the pending hedging relationship you must assign the effectiveness test category *Prospective for Unsettled Hedging Relationships* as a control parameter. You can also use defined run types in the fair value hedge effectiveness test for the prospective effectiveness test by assigning these to an effectiveness test variant. All necessary parameters must be defined in the Hedge Accounting Workplace without reference to the fair value effectiveness test.

Procedure

1. Call the Hedge Accounting Workplace [page 976].
   The system displays a selection screen.
2. Double-click a pending fair value hedging relationship in the hedge repository of the Hedge Accounting Workplace and choose  .
3. Choose  and enter the key date for the fair value effectiveness test.
4. Choose an effectiveness variant from the *Prospective for Unsettled, Undesignated Hedging Relationships* category using input help (F4).
   The system displays the result of the prospective fair value effectiveness test.
5. To display the detailed results of the fair value effectiveness test, choose  .

See also:

Fair Value Effectiveness Test [page 1036]

Prospective Regression Analysis [page 1088]

Management of Hedging Relationships [page 988]

4.6.1.2.8 Status and Status Changes of Hedging Relationships

Use

A hedging relationship always has a technical status and a business status that is derived from the technical status. The following events can cause a status change:

- User actions in the Hedge Accounting Workplace (HAW) [page 976]
  For example, the creation or dispatch of hedging relationships
- Actions in Hedge Accounting (HA) [page 904]
  For example, the dissolution of hedging relationships due to ineffectiveness. In this case, the Accounting Processes component sends events that are processed further in the Hedge Accounting Workplace.

The system only permits changes in status that are predefined. New hedging relationship versions can be generated when a status is changed. The versions are used for version management of the hedging relationship lifecycle and, if necessary, to undo executed actions.

An action triggered by the user, such as the designation of a hedging relationship, may fail as a result of incorrect system settings, for example. In this case, although the system is still able to run the substeps, the ultimate aim (the designation) cannot be realized. The system then assigns the hedging relationship the temporary technical status *Designation to Be Derived* . The business status (*pending*) remains the same. The user can then decide whether the partly designated hedging relationship is to be reversed, or whether the designation is to be completed (by correcting the incorrect settings in Customizing).

To summarize, Hedge Accounting Workplace procedures involved in the designation, dissolution and reversal of a hedging relationship are always carried out in two steps.

Note that the data must be supplied to the interfaces in a particular order to ensure the process runs smoothly. This order and the possible statuses are therefore predefined by the system and cannot be changed. These steps

include the generation and forwarding of an internal business transaction to Hedge Accounting (HA) and the triggering of the derivation of the internal business transaction (update of internal business transactions) in Hedge Accounting (HA).

Status Changes and Version Generation for Hedging Relationships

The table shows the changes in status and contains information about the generation of new versions and the relevant hedging relationship categories. The statuses and status transitions are predefined in the system and cannot be changed by the user. Hedging relationships are assigned the correct status synchronously.

Original Technical Status	New Technical Status	Versions Generation	Function	Relevant Hedge Category	Relevant Aggregation Level
0	New (10)	New version is generated (N)	Create a hedging relationship	All (ALL)	All (ALL)
New (10)	Initial (20)	Status change for same version (O)	Save hedging relationship in the Hedge Accounting Workplace	All (ALL)	All (ALL)
Initial (20)	Initial (20)	Version is deactivated (D)	Delete hedging relationship	All (ALL)	All (ALL)
Initial (20)	Designated to be derived (50)	New version is generated (N)	Generation and transfer of the business transaction	Fair value hedge (FVH)	All (ALL)
Initial (20)	Released (56)	New version is generated (N)	Release of the macro cash flow hedge in the Hedge Accounting Workplace	Cash flow hedge (CFH)	Portfolio hedge (PORTFOLIO)
Initial (20)	Initial (20)	Status change for same version (O)	Save hedging relationship in the Hedge Accounting Workplace	All (ALL)	All (ALL)
Initial adjustment (25)	Initial adjustment (25)	Version is deactivated (D)	Delete hedging relationship	Cash flow hedge (CFH)	Portfolio hedge (PORTFOLIO)
Initial adjustment (25)	Dissolved to be derived (80)	New version is generated (N)	Generation and transfer of the business transaction	Fair value hedge (FVH)	Portfolio hedge (PORTFOLIO)

Original Technical Status	New Technical Status	Versions Generation	Function	Relevant Hedge Category	Relevant Aggregation Level
Initial adjustment (25)	Initial adjustment (25)	Version is deactivated (D)	Reversal of key date valuation event by Hedge Accounting	Fair value hedge (FVH)	Portfolio hedge (PORTFOLIO)
Initial adjustment (25)	Designated to be derived (50)	New version is generated (N)	Generation and transfer of the business transaction	All (ALL)	Portfolio hedge (PORTFOLIO)
Initial adjustment (25)	Initial adjustment (25)	Status change for same version (O)	Save hedging relationship in the Hedge Accounting Workplace	All (ALL)	All (ALL)
Designated to be derived (50)	Designated (52)	Status change for same version (O)	Designation event generated by Hedge Accounting	All (ALL)	All (ALL)
Designated to be derived (50)	Designated to be derived (50)	Version is deactivated (D)	Dissolution event generated by Hedge Accounting	Fair value hedge (FVH)	Micro hedge (MICRO)
Designated to be derived (50)	Designated to be derived (50)	Version is deactivated (D)	Designation attempt reversed by user	All (ALL)	All (ALL)
Designated to be derived (50)	Designated to be derived (50)	Status change for same version (O)	Update all hedged positions after a transfer posting	All (ALL)	All (ALL)
Designated to be derived (50)	Designated (52)	Status change for same version (O)	Designation event generated by Hedge Accounting	All (ALL)	All (ALL)
Designated (52)	Dissolved to be derived (80)	New version is generated (N)	Generation and transfer of the business transaction	All (ALL)	All (ALL)
Designated (52)	Initial adjustment (25)	New version is generated (N)	Hedge proposal imported to the Hedge Accounting Workplace	Cash flow hedge (CFH)	Portfolio hedge (PORTFOLIO)

Original Technical Status	New Technical Status	Versions Generation	Function	Relevant Hedge Category	Relevant Aggregation Level
Designated (52)	Designated (52)	New version is generated (N)	Dedesignation event generated by Hedge Accounting	All (ALL)	Portfolio hedge (PORTFOLIO)
Designated (52)	Designated (52)	New version is generated (N)	Designation event generated by Hedge Accounting	Fair value hedge (FVH)	All (ALL)
Designated (52)	Designated (52)	New version is generated (N)	Designation event generated by Hedge Accounting	All (ALL)	All (ALL)
Designated (52)	Dissolved by Hedge Accounting (85)	New version is generated (N)	Dissolution event generated by Hedge Accounting	Fair value hedge (FVH)	All (ALL)
Designated (52)	Designated (52)	Status change for same version (O)	Update all hedged positions after a transfer posting	All (ALL)	All (ALL)
Designated (52)	Initial adjustment (25)	New version is generated (N)	Key date valuation generated by Hedge Accounting	Fair value hedge (FVH)	Portfolio hedge (PORTFOLIO)
Designated (52)	Designation reversed to be derived (60)	Status change for same version (O)	Designation reversed by user	All (ALL)	All (ALL)
Designated (52)	Designated (52)	Status change for same version (O)	Save hedging relationship in the Hedge Accounting Workplace	All (ALL)	All (ALL)
Designated (52)	Designated (52)	Status change for same version (O)	Define End of Valuation Period	Fair value hedge (FVH)	Portfolio hedge (PORTFOLIO)
Released (56)	Released (56)	Version is deactivated (D)	Delete hedging relationship	All (ALL)	All (ALL)
Released (56)	Initial adjustment (25)	New version is generated (N)	Hedge proposal imported to the Hedge Accounting Workplace	Cash flow hedge (CFH)	Portfolio hedge (PORTFOLIO)

Original Technical Status	New Technical Status	Versions Generation	Function	Relevant Hedge Category	Relevant Aggregation Level
Released (56)	Released (56)	Status change for same version (O)	Save hedging relationship in the Hedge Accounting Workplace	All (ALL)	All (ALL)
Designation reversed to be derived (60)	Designated (52)	Status change for same version (O)	Internal business transactions updated by the Hedge Accounting Workplace	All (ALL)	All (ALL)
Dissolved to be derived (80)	Dissolved by Hedge Accounting (85)	Status change for same version (O)	Dissolution event generated by Hedge Accounting	Fair value hedge (FVH)	Micro hedge (MICRO)
Dissolved to be derived (80)	Dissolved (81)	Status change for same version (O)	Internal business transactions updated by the Hedge Accounting Workplace	All (ALL)	All (ALL)
Dissolved to be derived (80)	Dissolved to be derived (80)	Version is deactivated (D)	Dissolution attempt reversed by user	All (ALL)	All (ALL)
Dissolved (81)	Dissolution reversed/to be derived (90)	Status change for same version (O)	Dissolution reversed by user	All (ALL)	All (ALL)
Dissolved (81)	Dissolved (81)	Status change for same version (O)	Save hedging relationship in the Hedge Accounting Workplace	All (ALL)	All (ALL)
Dissolved by Hedge Accounting (85)	Dissolved by Hedge Accounting (85)	Status change for same version (O)	Dissolution event generated by Hedge Accounting	Fair value hedge (FVH)	Micro hedge (MICRO)
Dissolved by Hedge Accounting (85)	Dissolved by Hedge Accounting (85)	Status change for same version (O)	Internal business transactions updated by the Hedge Accounting Workplace	Fair value hedge (FVH)	Micro hedge (MICRO)

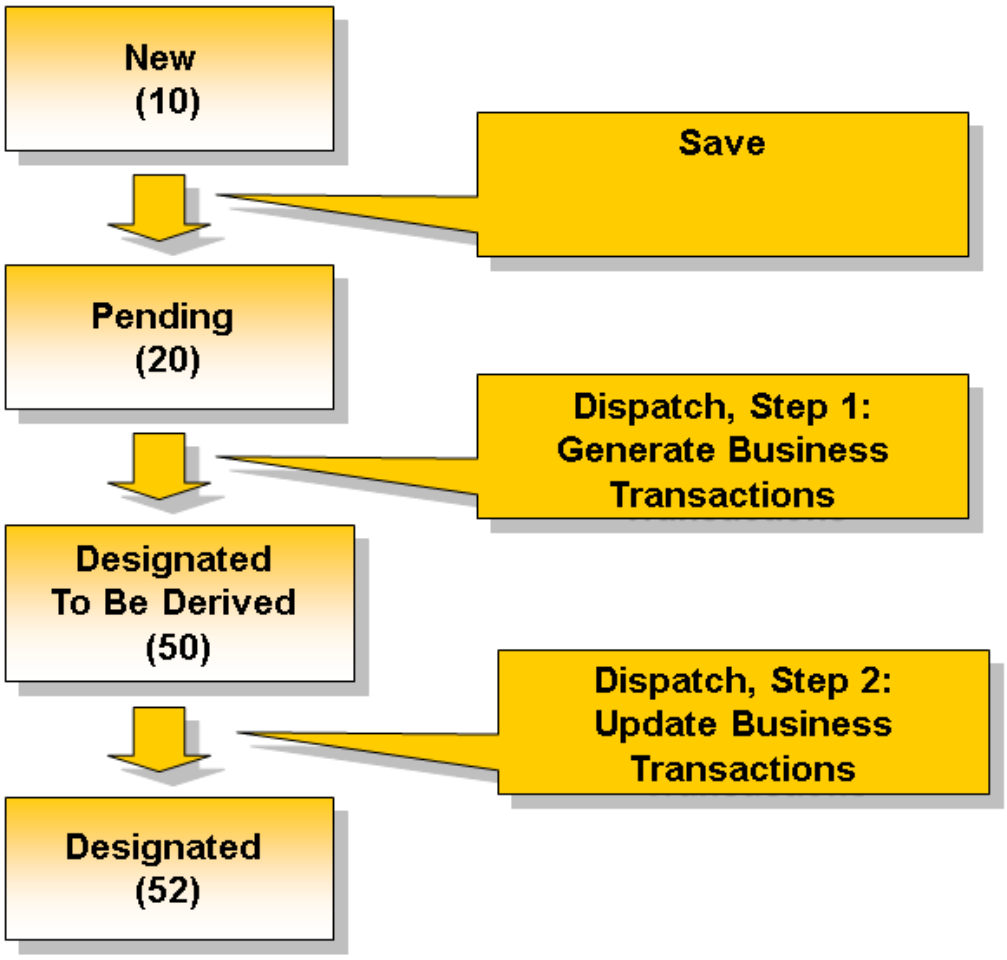
Original Technical Status	New Technical Status	Versions Generation	Function	Relevant Hedge Category	Relevant Aggregation Level
Dissolution reversed/to be derived (90)	Dissolved (81)	Status change for same version (0)	Internal business transactions updated by the Hedge Accounting Workplace	All (ALL)	All (ALL)

Hedging relationships whose designation was reversed successfully by the user, or whose designation in Hedge Accounting failed, have the technical status *Initial* (20). In the Hedge Accounting Workplace, the system uses the technical status *designation reversed* (21) and *designation rejected* (21) to log that the designation of a hedging relationship was reversed or rejected. These statuses are displayed only.

Example

The following figure gives an overview of the changes in status for a *designation* . The vertical sequence shows the statuses of a micro fair value hedge. It includes the description of the status and, in brackets, the numerical status. The arrows between the statuses represent the changes in status. On the right, beside each arrow, there is an explanation of the action that triggers or describes the status change.

Changes in the Status of Micro Fair Value Hedges During Designation



When you create and dispatch a hedging relationships, the status of the hedge changes as follows:

1. In the Hedge Accounting Workplace, choose *Hedging Relationship* > *Create* .
   Once the hedging relationship category has been selected, the system generates an initial hedging relationship with the status *New* (10).
2. Fill in the required entry fields and choose *Save* .
   The system changes the status of the hedging relationship to *Pending* (20).
3. Add the positions to the hedging relationship, enter any additional information, and validate them successfully.
   The status does not change.
4. Choose *Dispatch* to complete the designation of the hedging relationship.
   The system creates a new version of the hedging relationship. The system then starts to generate an internal business transaction in the background and acquires the status *Designated to be Derived* (50).
   The system then triggers the update of internal business transactions [page 786] for this hedging relationship.
   During the update of internal business transactions, Hedge Accounting sends an event to the Hedge Accounting Workplace reporting on the success of this process. If it is successful, the hedging relationship acquires the status *Designated (52)* .

4.6.1.2.9 Reading of Data from the Hedge Accounting Workplace

Use

You can use this function to read hedging relationship data from the Hedge Accounting Workplace This process is used for the communication with other Bank Analyzer components. For example, after the prospective effectiveness test has been called, the fair value effectiveness test gets the data it needs for pending hedging relationships from the Hedge Accounting Workplace.

It reads this data by using general primary data sources that use the HAW - Hedging Relationship (HEDGE_REL) server. The server can be used for primary data sources both for single records and for worklists. You can get the data for fixed characteristics, key figures, and for custom fields. The primary data sources return the current version of a hedging relationship.

> **Note**
>
> Specifying a primary data source for the selection of hedging relationships is optional. If you want to use this feature, you enter the primary data source in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Valuation* > *Edit Portfolio Item Environment.If you do not enter a primary data source, the system automatically uses an API to read the details for the hedging relationship from the Hedge Accounting Workplace.*

Prerequisites

If you want to read custom fields from the header data of a hedging relationship, you must first store these field in the HEADGE_HEADER_GDS field catalog. You find the field catalog in Customizing for *Bank Analyzer* under

Processes and Methods > *Hedge Processes* > *Hedge Accounting Workplace* > *Basic Settings* > *Edit Field Catalog for Mapping Accounting and Hedge Accounting Workplace.*

You create primary data sources in Customizing for Bank Analyzer under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources.* Use the HAW - Hedging Relationship (HEDGE_REL) server.

On the *Characteristics* tab page, enter the characteristics that you want the system to read. The hedging relationship ID (/BA1/C47HEDGE) is a required field.

The relevant complex key figures for item data is stored in the *Key Figures* tab page. The complex key figures are:

- &H0HRDSS (Definition of the portfolio for hedged items)
  Data element /BA1/F1_TAB_CONTR_DSS_DATA
- &H0HRITM (Item data)
  Data element /BA1HM/ID_GDS_TAB_HR_ITEM
- &H0HRLE (Hedging relationship data for a given legal entity)
  Data element /BA1HM/ID_GDS_TAB_HR_LE
- &H0HRVP (Qualified positions)
  Data element /BA1HM/ID_GDS_TAB_HR_VP

On the *Attributes* tab page, you specify whether the primary data source supplies mirror transactions: When you call up the fair value effectiveness test, for example, the system does not consider mirror transactions:

4.6.1.3 Hedge Proposal

Definition

Proposal for inflows and outflows of hedging instruments in a macro cash flow hedging relationship [page 1019].

Use

The hedge proposals support the management of macro cash flow hedging relationships and contain information about hedging instrument inflows and outflows, as well as changed hedging instruments and, where applicable, the results of the effectiveness tests.

You create hedge proposals in the cash flow hedge analysis [page 1107]. In this analysis, you can simulate the influence of new hedging instruments on macro cash flow hedging relationships and change the assigned hedging instruments. When you save changed hedging relationships, the system creates hedge proposals that you can process further in the Hedge Accounting Workplace (in accordance with IAS 39, Implementation Guidance Committee 121-2).

Structure

The header data contains the following information about creating hedge proposals in the cash flow hedge analysis:

- Technical time stamp and key date
- Valuation run and valuation run version
- Aggregation run, aggregation run category and version
- Name of the hedge proposal in the cash flow hedge analysis
- Whether a fair value effectiveness test is planned based on the *hypothetical derivative* approach
- Version of the hedge proposal
- Status of the hedge proposal

The hedge proposal also contains a table that displays the changes to the cash flow hedging relationships. The table has the following structure:

- Hedging relationship ID
- External ID of the hedging instrument
- Inflowing or outflowing nominal of the hedging instrument
- Final effectiveness test (if this was run based on the *hypothetical derivative* approach)

Integration

The hedge proposals created in the cash flow hedge analyses are automatically registered in the Hedge Accounting Workplace and can be managed here.

In the Hedge Accounting Workplace, you enrich the data of the existing financial positions, transfer the volume of the hedge proposals, and validate the legal entity of the hedging instruments concerned for each hedging relationship.

Once these steps have been run, you can import the hedge proposals in the Hedge Accounting Workplace and add the information from the hedge proposal to the cash flow hedging relationships. The hedging relationships can then be dispatched for the accounting processes.

The disclosure triggers a transfer posting in accounting. In the case of a designation, the transfer posting is made to a hedged financial position. If a hedging instrument is dedesignated, the transfer posting is made to an unhedged financial position.

See also:

Generation of Hedge Proposals [page 1146]

Management of Hedge Proposals [page 1031]

Processing of Hedge Proposals [page 1033]

4.6.1.3.1 Management of Hedge Proposals

Use

You use this function to display and to reverse hedge proposals [page 1029].

You can display hedge proposals to obtain an overview of their properties. You can also display the results of the fair value effectiveness test (under the hypothetical derivative approach) for the hedging instruments used in the hedge proposals.

Integration

You generate hedge proposals in reporting in Cash Flow Hedge Analysis [page 1140]. In the Hedge Accounting Workplace, you can use display the hedge proposals saved by this function. Once the hedge proposal has been released, you can reverse it here.

Activities

1. **Subledger Scenario**
   On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Manage Hedge Proposals*.
   Merge Scenario
   On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Accounting: Merge Scenario* > *Hedge Accounting Workplace* > *Manage Hedge Proposals*.
2. Enter the selection criteria, for example, the valuation run type, time period for the time stamp, or the key date.
   The system displays an overview of all the hedge proposals that meet the selection criteria. The proposals are sorted by valuation run and aggregation run, and the status of each hedge proposal is displayed.

   > **Note**
   >
   > You can call the report for displaying hedge proposals from the Hedge Accounting Workplace [page 987] . To do this, choose *Hedge Proposal Repository* in the navigation structure. Note that you can only see the hedge proposals for which you can release, designate or redesignate cash flow hedging relationships.
3. Choose the hedge proposals that you want to display.
   To display the definition of the worklist that belongs to the hedge proposal, choose the pushbutton with the name of the worklist. The system displays the definition of the worklist for the selected hedge proposal.
   The following functions are available in the context menu for the hedge proposal:

Action	Function
Set Status to "Obsolete"	The system sets the status of the hedge proposal to “Obsolete”.
Start: Worklist Selection	The system fills the hedge proposal with the financial positions. Select a worklist for this.
Validate Worklist	The system checks the worklist. If the data is consistent, it sets the status of the worklist to *Valid*. The hedge proposal can then also be viewed in the Hedge Proposal Repository of the Hedge Accounting Workplace.
Reverse Hedge Proposal	The system reverses the hedge proposal.
Reporting > *Application Log*	The system displays the application log for hedge proposal processing.
Reporting > *HP Reporting*	The system displays the details of the hedge proposal.
Reporting > *HP Results Data of Aggregation Run*	The system branches to the reporting functions in the cash flow hedge analysis [page 1140] and displays the results for the hedge proposal.
Reporting > *WL Reporting*	The system displays the content of the worklist for the hedge proposal.

Note

In the hedge proposal repository in the Hedge Accounting Workplace you can use the context menu of valuation runs, aggregation runs, and aggregation run versions to display the relevant runs in run administration.

To display the results from Cash Flow Hedge Analysis, in the context menu of the aggregation run version choose *Display Version Results for Aggregation Run*. The system branches to the reporting functions in Cash Flow Hedge Analysis and displays the results for the selected aggregation run version.

See also:

Managing Macro Cash Flow Hedging Relationships [page 1033]

4.6.1.3.2 Processing of Hedge Proposals

Prerequisites

- You have created and released a macro cash flow hedging relationship [page 1019].
  The initial hedging relationship does not contain any hedging instruments. It contains only a data source and selection criteria for hedge items to run the cash flow hedge analysis.
- You have carried out cash flow hedge analysis [page 1107] and generated a hedge proposal.
  Note that you can modify hedging instruments only by starting the cash flow hedge analysis again, and generating a new hedge proposal.
- You have created an *HP* worklist [page 978].
  This worklist is used to add financial position data to hedge proposals.

> **Note**
>
> The system can generate the worklist for hedge proposals automatically. You define this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Accounting Workplace* > *Data Retrieval* > *Hedge Proposal* > *Store Target Folder and Worklist Definition for Hedge Proposal Processing*.

Context

To process hedge proposals [page 1029] you need to carry out various steps in the Hedge Accounting Workplace.

The hedge proposal information is registered automatically in the Hedge Accounting Workplace when these proposals are generated, and can be maintained there. For example, the financial position data has to be added to the hedge proposals, and imported into the Hedge Accounting Workplace.

Procedure

1. **Subledger Scenario**

   On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Manage Hedge Proposals*.

   Merge Scenario

   On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Accounting: Merge Scenario* > *Hedge Accounting Workplace* > *Manage Hedge Proposals*.

   The system displays a selection screen in which you can enter selection criteria to define which hedge proposals are to be displayed. Choose ✔.

The system displays an overview of all the hedge proposals registered in the Hedge Accounting Workplace. The status of each hedge proposal shows the processing status.

2. Select a hedge proposal.

   To do so, in the navigation structure of a valuation run choose a valuation run version, an aggregation run, and an aggregation run version. Then select the hedge proposal you require.

3. In the context menu choose *Start: Worklist Selection*.

   The system displays a dialog box in which you can select the worklist. Choose ✔.

   The system fills the hedge proposal with the financial positions.

4. In the context menu choose *Validate Worklist*.

   For each hedging relationship the system checks that the legal entities of the hedging instruments in question are consistent, and sets the status of the hedge proposal to *Valid*. The hedge proposal is then displayed in the *Hedge Proposal Repository* in the Hedge Accounting Workplace.

5. Call the Hedge Accounting Workplace [page 976].

   The system displays a selection screen in which you can enter selection criteria to define which data you use in the Hedge Accounting Workplace. Choose .

6. In the navigation structure, choose Hedge Proposal Repository.
7. Select the hedge proposal. In the context menu choose *Import into Hedge Repository*.
8. In the hedge repository, choose the *Hedging Relationship Worklists* repository and the *Cash Flow* subdirectory.

   When the hedge proposals are transferred, an import folder that has the ID of the valuation run (run ID) is automatically generated for each hedge proposal. By opening the import folder you can display all the corresponding hedging relationships. The key of the hedging relationship proposals has the same name as the released hedging relationship. The imported hedging relationships for a proposal all have the same *Initial Adjustment* status.

   When the hedge proposal is imported, new pending versions of the hedging relationships are generated with the hedging relationship ID of the original designated version. The designated versions are displayed in the overview of the cash flow hedging relationships.

9. In the context menu of the import folder choose *Publish*.

   The system dispatches the hedging relationships to the accounting processes. A new pending version of the hedging relationship is designated as a result of the release of the imported hedging proposal. This generates a new designated version from the pending version. This new version contains the unchanged, the changed, and the new hedging instruments with the corresponding financial positions. Where hedging instruments are changed or new, the *Designation Date* field contains the changed or new designation date.

   The new designated version, and not the originally designated version, is displayed in the cash flow hedging relationship overview. The successfully designated hedge proposal can no longer be seen in the *Hedging Relationship Proposals* directory.

> **Note**
>
> To reverse a hedge proposal, in the Hedge Proposal Repository select the hedge proposal and in the context menu choose *Reverse Hedge Proposal*.

Next Steps

Management of Hedge Proposals [page 1031]

4.6.1.4 Archiving

Use

You can use this function to archive expired, dissolved, and cancelled hedging relationships. You do this if you want to remove data from the database without deleting the data permanently.

Features

Hedging relationships are archived in three stages: First, the system makes preparations for the archiving process. Then, the system selects the hedging relationships and writes them to the archive. After saving them successfully, you can remove the hedging relationships from the Hedge Accounting Workplace.

The system has functions with which you can read and reload archived hedging relationships.

Transactions for Archiving Hedging Relationships

Function	Transaction
Prepare for archiving hedging relationships	/BA1HM/AH_M_HR_APRP
Write hedging relationships to the archive	/BA1HM/AH_M_HR_AWRT
Delete from Hedge Accounting Workplace the archived hedging relationships	/BA1HM/AH_M_HR_ADEL
Read archived hedging relationships	/BA1HM/AH_M_HR_AREAD
Reload archived hedging relationships	/BA1HM/AH_M_HR_ARLD

> **Note**
>
> You can call up the archiving function from the SAP Easy Access Screen under *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Hedge Accounting Workplace* > *Archiving* .

4.6.2 Fair Value Effectiveness Test (FS-BA-PM-HP-FVH)

Use

This component provides functions to measure the effectiveness of hedging relationships and to manage the test results. You use this component for hedge accounting under the International Accounting Standards (IAS).

Hedge accounting can be used only if the effectiveness of the hedging relationships posted can be proven. Ongoing checks must be run to prove the effectiveness, at least for every key date. You can use the fair value effectiveness test to do the following:

- **Prepare for key date valuations**
  The results of the effectiveness test can be used in the Hedge Accounting component. If appropriate, ineffective hedging relationships are dissolved in the Hedge Accounting component. You usually schedule this process for the end of the accounting period.
- **Monitor the effectiveness of existing hedging relationships**
  If you do not want the results of the effectiveness test to be processed in Hedge Accounting, you can test the effectiveness during the period to monitor the effectiveness of the hedging relationships.

While the International Accounting Standards do not specify a particular method for measuring the effectiveness, they do stipulate that a standard method be used. We provide suitable test methods for this purpose.

Integration

The following data is the basis for the fair value effectiveness test:

- Hedging relationships from the Hedge Accounting Workplace [page 976]
- Saved financial positions of hedging relationships from the Accounting [page 693] component
- Results from external measurements of effectiveness from the Source Data Layer (SDL) [page 100]
- History of effectiveness measurements from the Result Database (RDB)
  This history contains runs for a given legal entity that have already been processed in Hedge Accounting. The system uses the history to access earlier results of the effectiveness test and compare them with current test results. When you run a tolerance value check, for example, it is useful to set up a history that records the sequence of tolerance value checks that have been applied.

The fair value effectiveness test calls the following functions:

- Fair Value Server [page 628]
  This is used to calculate the following key figures: fair value, amortized cost, hedge fair value, hedge amortized cost for micro fair value hedging relationships and cash flow hedging relationships.
- Portfolio Fair Value Calculator [page 1175]
  This is used to calculate the hedge adjustment for portfolio fair value hedging relationships.

You can also call the effectiveness test from Hedge Accounting Workplace [page 976] and

Cash Flow Hedge Analysis [page 1107] (for the hypothetical derivative approach only).

Features

The fair value effectiveness test measures the effectiveness of the following hedging relationships:

- Micro fair value hedging relationships
- Portfolio fair value hedging relationships
- Macro cash flow hedging relationships in the hypothetical derivative approach [page 1037]
- Cross-legal-entity hedging relationships [page 1038]

It does so by using the following methods:

- Internal test methods for measuring effectiveness [page 1071]
  In addition to using the effectiveness test methods predefined in the system, you can create and use your own custom effectiveness test methods [page 1099].
- Transfer of external effectiveness measurements [page 1100]
- Reporting [page 1055] (display of the results of the effectiveness measurement)
  You can recalculate the results for particular hedging relationships if required, or change the final effectiveness indicator manually.
- Run administration [page 1050]
  Displays the runs for the effectiveness test. You can release individual runs for Hedge Accounting [page 904].

> **Note**
>
> In Hedge Accounting, the system generates business transactions to dissolve ineffective hedging relationships. Effective hedging relationships are posted in line with IAS in the key date valuation, and not in the fair value effectiveness test.

For more information, see the document Processes of the Fair Value Effectiveness Test [page 1039].

4.6.2.1 Hypothetical Derivative Approach

Use

You use this function to measure the effectiveness of macro cash flow hedging relationships.

Prerequisites

You have stored a characteristic with the field instance to identify the hedging instruments in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Configuration* > *Field Instance* > *Set Up Field Instance*. If you do not store a characteristic, you cannot distinguish between the hedging instruments in reporting for the effectiveness test.

You have analyzed the hedging relationships in Cash Flow Hedge Analysis [page 1107] whose effectiveness you want to measure. You have analyzed the cash flow hedges using the hypothetical derivative approach.

Features

Although the hypothetical derivative approach essentially uses the same infrastructure of the fair value effectiveness test, it differs in that:

- At the beginning of the effectiveness test, the system generates a hypothetical derivative [page 515] for each hedging instrument in the cash flow hedging relationship.
- The system does not determine the setting of the final effectiveness indicator for the whole hedging relationship. Instead, it does this for each hedging instrument. As a consequence, you can see how each hedging instrument contributes to the overall effectiveness of the hedging relationship.
- The system uses the existing effectiveness methods but generates fictitious hedging relationships for them. These fictitious hedging relationships consist of a real hedging instrument and the corresponding hypothetical derivative. In the fictitious hedging relationship, the hypothetical derivative represents the hedged item, and the real derivative the hedging instrument.
  To be able to use the existing effectiveness test methods, the system compares the value changes in the hypothetical derivative with the real derivative. As these changes in value are always consistent, the system reverses the +/- sign before the final effectiveness indicator is derived. If this was not the case, the determined key figures would not be consistent with the results in the micro fair value hedging relationships.

Activities

You can either schedule the effectiveness test using Schedule Manager, or you can execute an online run. Set the *Use Hypothetical Derivative Approach* checkbox.

For more information, see Run Management [page 430] and Online Run [page 1054].

4.6.2.2 Testing of Cross-Legal-Entity Hedging Relationships

You can test the fair value effectiveness of cross-legal-entity hedging relationships. All processes are the same as in the effectiveness test for hedging relationships that belong to the same legal entity. The only differences are in the measurement of the transactions in the hedging relationship.

Measurement Without Hedging Instruments

Cross-entity hedging relationships can contain mirror transactions. Since the effectiveness of these hedging relationships is tested across multiple legal entities, rather than for just one, mirror transactions are ignored by the measurements of the effectiveness test.

Synchronization of Legal Entities

The fair value effectiveness test takes place as part of the key date valuation.

In the subledger scenario, start the Set SDL Time Stamp [page 367] function in the main legal entity and close the valuation period for hedging relationships in the Hedge Accounting Workplace. The system uses the SDL time stamp from the main legal entity to determine the SDL time stamp for accessing other legal entities.

If the time stamps of the legal entities do not match, in the effectiveness test the system might select incorrect versions of transactions, or errors might occur because some transactions no longer exist.

4.6.2.3 Processes Used in the Fair Value Effectiveness Test

Purpose

The effectiveness test measures the effectiveness of hedging relationships. The result of the test is the final effectiveness indicator. For micro fair value hedging relationships and portfolio fair value hedging relationships, the final effectiveness indicator refers to the entire hedging relationship. For cash flow hedging relationships, the system uses the hypothetical derivative approach, and derives a separate final effectiveness indicator for each hedging instrument. The final effectiveness indicator shows whether a hedging relationships is effective or ineffective, and to what extent the portion of a hedging instrument used in the hedge contributes to its effectiveness (for cash flow hedging relationships).

Prerequisites

The effectiveness test is a self-contained process that is scheduled and called up independently of the Accounting Processes such as the key date valuation. You schedule the effectiveness test using the Schedule Manager [page 437]. When you schedule the effectiveness test, you can set an indicator to define that the program is to use the hypothetical derivative approach. In other words, you define whether the effectiveness of cash flow hedging relationships is to be measured.

Before you call up the test, note the following:

Ensure that all the external effectiveness measurements have been completed and the transferred results are available in the Source Data Layer (SDL).

Ensure that all the external business transactions for your legal entity have been updated in the Accounting Processes.

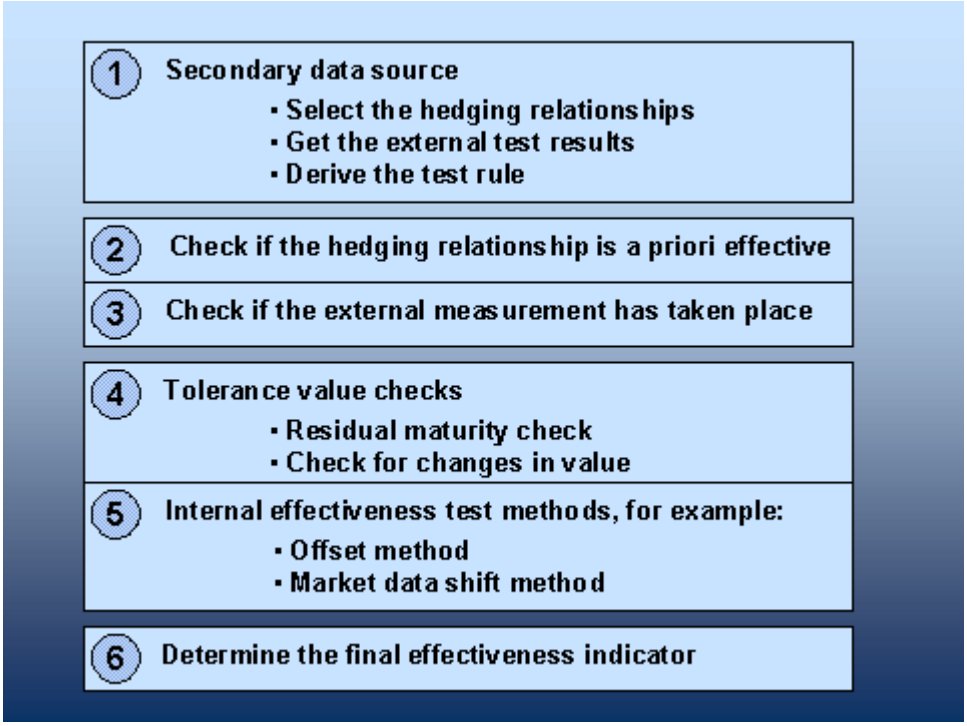
You usually call the effectiveness test before the key date valuation [page 934].

If you want to test the effectiveness of cash flow hedging relationships, you must have already carried out a cash flow hedge analysis [page 1107] using the hypothetical derivative approach.

> **Note**
>
> Instead of scheduling the runs in the Schedule Manager, you can start the effectiveness test as an online analysis [page 1054]. The results of the online run are stored in the main memory; they are not saved in the Result Database (RDB).
>
> You can start the effectiveness test as a batch job from within the area menu. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Run Administration* > *Create and Schedule* > *Start Effectiveness Test*.

Note that the runs that you start from the area menu use work processes. In certain circumstances, this can lead to a shortage of system resources, and delay the start of the runs that you have scheduled in Schedule Manager.

Process Flow



The system selects the hedging relationships that it is to analyze:

The system creates the worklist from the Hedge Accounting Workplace by using a primary data source belonging to the *InfoSet* category.

The system then selects the data relevant for the test from the SDL. This data includes the designation date, the end date of the hedging relationship, the list of the hedged risks, and the reference currency of the hedging relationship. The system selects the data using a secondary data source.

If the effectiveness was tested externally, the system reads the results from the SDL.

Within the secondary data source, the system derives the effectiveness test rule [page 1042], which enables the hedging relationships to be analyzed based on characteristics.

When selecting the hedging relationships, the system uses a current time stamp for the worklist and for the single records. This ensures that changes to the hedging relationship are considered, including any hedging relationships that have been dissolved. Versioning [page 103] in the Hedge Accounting Workplace involves storing the data in two dimensions, using the key date and the time stamp.

The following steps are carried out separately for each hedging relationship and for each time bucket (for portfolio fair value hedging relationships).

The system checks whether the hedging relationship that is to be analyzed is to be set to *a priori effective*.

If this is the case, the system sets the final effectiveness indicator to *effective* and ends the test.

The system checks whether an external effectiveness measurement [page 1100] has been carried out.

If this is the case, the system copies the final effectiveness indicator that was transferred.

The system carries out a tolerance value check [page 1074], which can include the following steps:

- Residual maturity check
- Check for changes in value

If the tolerance value check is successful, the system sets the final effectiveness indicator to *effective* and ends the test.

You can configure the tolerance value check in Customizing for the fair value effectiveness test.

The system carries out an internal effectiveness measurement, which includes the following steps:

Prospective effectiveness test (for example, prospective regression analysis [page 1088])

The system sets the prospective effectiveness indicator as per the results.

Retrospective effectiveness test (for example, the offset method [page 1076] or retrospective regression analysis [page 1083])

The system sets the retrospective effectiveness indicator in line with the results.

> **Note**
>
> You can configure each effectiveness test method in Customizing for the fair value effectiveness test. In Customizing, you can also define which tolerance value checks or test methods the system uses, and in what order. It uses the test schema for this purpose. SAP provides test schemas, and you can also create your own test schemas.

Using characteristics and the effectiveness test rule [page 1042] you can define how certain hedging relationships are tested. You use the effectiveness test rule to specify that you do not want to test the effectiveness internally for hedging relationships that are *a priori effective*, for example.

The internal effectiveness test methods require the *hedge fair value* and *hedge amortized cost* key figures. Before the system tests the effectiveness, it generates the cash flows for the transactions in the hedging relationship. When selecting data in a batch run, the system uses the SDL time stamp for the current financial reporting process (subledger scenario) or the time stamp of the current preliminary process for financial reporting (merge scenario). If the system cannot determine the time stamp, it uses the current time stamp. This is recorded in the application log.

In an online run, you can choose the time stamp. If you do not specify a time stamp, the system uses the current time stamp.

The system derives the final effectiveness indicator from the prospective and retrospective effectiveness indicators. It uses the rule set [page 1103] for this purpose. It also sets the start date of ineffectiveness.

Result

The system has measured the effectiveness of the hedging relationships. You can display the final results in reporting [page 1055]. In run administration [page 1050], you release the results of the effectiveness test runs for Hedge Accounting [page 904].

> **Note**
>
> It may be the case that runs finish but they are not complete (run status RINC). You can restart these runs by adding the data that is missing. By setting the *New Version* indicator on the selection screen for runs you define that the system creates a new version of the run when it restarts the run.

See also:

Effectiveness Test Methods [page 1071]

4.6.2.3.1 Effectiveness Test Rules

Definition

Rules that uses characteristics to define how fair value effectiveness is tested for hedging relationships.

Use

You use this object to configure the effectiveness test methods separately for particular groups of hedging relationships. For example, in Customizing you can store different settings for the offset method. By means of the test rule, you can then use these settings for different types of transaction in the effectiveness test. In the effectiveness test, the system derives the effectiveness test rule using the characteristics of the hedging relationship or of the hedged items and hedging instruments.

The effectiveness test rule is used together with the effectiveness test environment. While the test environment contains general settings, the effectiveness test rule enables the actual test methods within a test environment to be defined more precisely. You have to assign an effectiveness test rule to each hedging relationship. Only one test environment is possible for each call of the effectiveness test. However, you will usually use multiple effectiveness test rules.

The effectiveness test environment contains the following general settings:

Information for calling up the Fair Value Server [page 628]

Global valuation environment

Cash flow set

Market data area

Rule set for determining the final effectiveness indicator [page 1103]

Inversion of declared and declaring variables (only in retrospective regression analysis and the market data shift method)

The effectiveness test rule includes the following settings:

A Priori Effective indicator

If you set the indicator, the system does test the effectiveness of the hedging relationship.

Test schema

In the test schema, you define which test methods the system is to use to test the effectiveness of hedging relationships, and the order in which the methods are applied.

Indicator that you use to specify when the currency translation is to be carried out (only for hedging relationships with hedged items and hedging instruments containing more than one currency)

You use this indicator to specify the extent to which the currency risk is part of the hedging relationship.

If a currency translation is carried out on the designation date, currency fluctuations during the term of the hedging relationship are not considered.

If a currency translation is carried out on the current key date, the currency fluctuations during the term of the hedging relationship are considered.

If required, you can use specific settings for each effectiveness test method (for example, "noise" in the offset method).

Integration

The effectiveness test rule is a Customizing concept. You cannot change test rules in your operative system.

To define effectiveness test rules, enter the following settings in the Customizing for the fair value effectiveness test for hedging relationships:

Define the effectiveness test rule

In Customizing for *Bank Analyzer* , you define the effectiveness test *rule* by choosing *Processes and Methods* → *Hedge Processes* → *Fair Value Effectiveness Test* → *Valuation* → *Environment and Rules for Effectiveness Test* → *Edit Rule for Effectiveness Test*

Enter the parameters for the test methods

You enter the parameters for the test methods in the Customizing for *Bank Analyzer* by choosing *Processes and Methods* → *Hedge Processes* → *Fair Value Effectiveness Test* → *Valuation* → *Environment and Rules for Effectiveness Test* . These settings are made for a combination of effectiveness test environment and effectiveness test rule.

Define how the effectiveness test rule is derived

You define how test rules are derived in the Customizing for Bank Analyzer under *Processes and Methods* → *Hedge Processes* → *Fair Value Effectiveness Test* → *Valuation* → *Derivation of Rule for Effectiveness Test*

Note that the only characteristics you can use for the derivation are those that are available to the system after the selection of the hedging relationships and their hedged items and hedging instruments. For example, you can

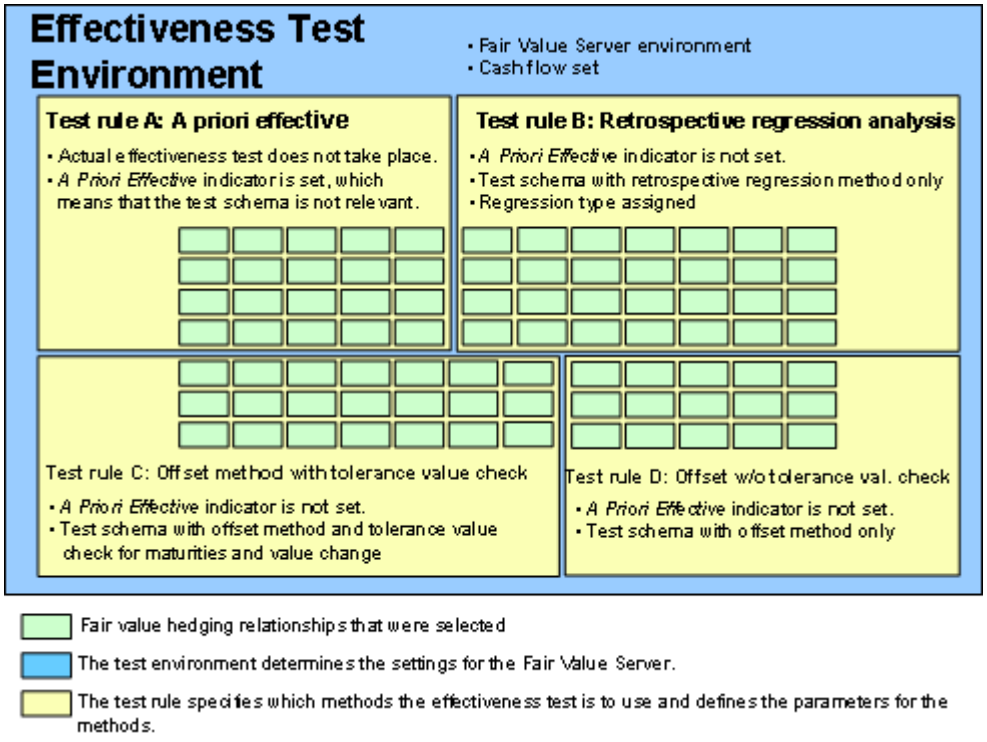
define a characteristic *test method* and include this in the hedging relationships so that the system can recognize which effectiveness test method it is to use for the hedging relationship.

Link the derivation process to the secondary data source

You define secondary data sources in the Customizing for Bank Analyzer under *Processes and Methods* → *Hedge Processes* → *Fair Value Effectiveness Test* → *Processes* → *Secondary Data Source* → *Edit Module for the Selection of Hedging Relationships.*

Example

You can use effectiveness test rules to split the worklist containing the hedging relationships, and to use different settings to test the effectiveness of individual groups.



4.6.2.4 Run Administration

Definition

Run administration includes the following functions:

- Execute or create run
- Display an overview of runs
- Display Application Log

- Edit run
- Manage run
- Replace run
- Select run for archiving
- Delete run
- Log of the deletion function

> **Note**
>
> The above functions are not all available for each application. For more information, see the application-specific documentation.

Use

The following table lists the runs available for each application:

Application	Run
General Calculation and Valuation Methods [page 376]	Account Pooling [page 383] or Account Pooling (CVPM) Facility Distribution [page 399] or Facility Distribution (CVPM) Determination of the Free Line for Facilities [page 395] or Determination of the Free Line (CVPM) Collateral Distribution [page 404] or Collateral Distribution (CVPM) Determination of Default [page 410] Stress Runs [page 429]: • Stress run for account pooling or stress run for account pooling (CVPM) • Stress run for facility distribution or stress run for facility distribution (CVPM) • Stress run for determination of the free line or stress run for determination of the free line (CVPM) • Stress run for collateral distribution or stress run for collateral distribution (CVPM) • Stress run for determination of default

Application	Run
Credit Risk [page 1191]	Credit Exposure Run [page 1238] or Credit Exposure Run (CVPM) Country Risk Run [page 1336] (prototype for country risk) **Stress run:** Credit Exposure Stress Run [page 1261] or Credit Exposure Stress Run (CVPM)
Historical Database [page 1478]	**Version management:** Run for generating historical data for data layers [page 1481] Run for generating historical data for the bank's own models [page 1484] Uploading of Files [page 1486] **Calculation functions:** Determining Default Rates [page 1488] Determining Average Default Rates [page 1491] Determining Default Figures [page 1493] Calculation of Migration Matrices [page 1496] **Data retrieval:** Run for supplying data to models [page 1498] Downloading of Files [page 1500] **Stress tests:** Stress Run for Supplying Models with Data [page 1505] Generating Scenario Data in the Source Data Layer [page 1506]
Generic BI Data Extraction [page 1638]	Testing the BI Extractor [page 1639] BI extraction run **Note** Extraction runs are created and executed in SAP NetWeaver Business Intelligence (BI). The system displays information about extraction runs in run administration of Bank Analyzer.
Regulatory Reporting Interface	Data Extraction Runs
Limit Manager [page 1529]	Limit Utilization Run [page 1551]

Application	Run
Fair Value Effectiveness Test for Hedging Relationships [page 1036]	Effectiveness Test Run
Cash Flow Hedge Analysis [page 1107]	Valuation Run [page 1128] Subvaluation Run Aggregation Runs [page 1129]
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169] Portfolio Item Runs [page 1172] • Period-end run for portfolio items (subledger scenario) • Period-start run for portfolio items (subledger scenario) • Portfolio item run (merge scenario)

> **Note**
>
> For some of the Bank Analyzer components, you can use the Schedule Manager to schedule and control jobs. If you use multiple applications, you can define the sequence in which the runs are to be carried out. For more information, see Schedule Manager [page 437].

More Information

Status Overview for Run Administration [page 1248]

4.6.2.4.1 Schedule Manager

Definition

A tool in Bank Analyzer that you can use to execute and monitor complex business processes, such as period-end closing. When you schedule a run, you always have to specify the program with which the system calls the run. There is a separate program for each run category in an application.

The table below shows for which runs and run categories the Schedule Manager can be used, and their corresponding programs.

Application	Run Category	Program
Account Pooling	Account Pooling [page 383]	/BA1/DE_GM_MAIN_COMP

Application	Run Category	Program
Account Pooling	Stress Run for Account Pooling [page 429]	/BA1/DE_GM_MAIN_COMP_SCEN
Determination of the Free Line for Facilities	Determination of the Free Line for Facilities [page 395]	/BA1/DE_GM_MAIN_FL
Determination of the Free Line for Facilities	Stress Run for Determination of the Free Line for Facilities [page 429]	/BA1/DE_GM_MAIN_FL_SCEN
Distribution of Facilities	Distribution of Facilities [page 399]	/BA1/DE_GM_MAIN_RLV
Distribution of Facilities	Stress Run for Facility Distribution [page 429]	/BA1/DE_GM_MAIN_RLV_SCEN
Collateral Distribution	Collateral Distribution [page 404]	/BA1/DE_GM_MAIN_SV
Collateral Distribution	Stress Run for Collateral Distribution [page 429]	/BA1/DE_GM_MAIN_SV_SCEN
Determination of Default	Determination of Default [page 411]	/BA1/RA_DFLT4280_RUN
Determination of Default	Stress Run for Determination of Default [page 429]	/BA1/RA_DFLT4280_RUN_SCENARIO
Fair Value Effectiveness Test [page 1036]	Effectiveness Test Run [page 1039]	/BA1/R8_FVHE_MAIN_RUN
Fair Value Effectiveness Test	Generation of Time Series for Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MAIN_RUN
Fair Value Effectiveness Test	Modifying a Time Series for the Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MODIFY
Cash Flow Hedge Analysis [page 1107]	Subvaluation Run [page 1128]	/BA1/R5_CFHA_BTC_RUN
Cash Flow Hedge Analysis	Aggregation Run [page 1129]	/BA1/R5_START_BATCH_AGGR
Cash Flow Hedge Analysis	Joint call of Valuation Run [page 1111] and Aggregation Run [page 1123]	/BA1/R5_EXECUTE_CFHA
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169]	/BA1/R9_VPM_BTC_ID_RUN
Portfolio Fair Value Hedge	Portfolio Item Run (Merge Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_RUN
Portfolio Fair Value Hedge	Period-End Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PE_RUN
Portfolio Fair Value Hedge	Period-Start Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PS_RUN

Application	Run Category	Program
Portfolio Fair Value Hedge	Define End of Valuation Period	/BA1HM/RAH_M_PFVH_SET_VP_END
Portfolio Fair Value Hedge	Adjust Initial Portfolio Fair Value Hedges for Next Valuation Period	/BA1HM/RAH_M_PFVH_ADJUST
Portfolio Fair Value Hedge	(Re-) Designation of Initial Portfolio Fair Value Hedges	/BA1HM/RAH_M_PFVH_DESIG
Credit Exposure [page 1192]	Credit Exposure Run [page 1238]	/BA1/R2_CRA_MAIN_CRE
Credit Exposure	Credit Exposure Stress Run [page 1261]	/BA1/R2_CRA_MAIN_CRE_SCEN
Credit Exposure	Credit Exposure Run for Revolving Retail	/BA1/RR2_CRE_REVRET
Credit Exposure	Approximate Stress Run [page 1274]	/BA1/RR2_CRE_SCENARIO
Credit Exposure	Aggregation Run for Ad Hoc Calculation [page 1276]	/BA1/RR2_SCEN_AGGR
Country Risk [page 1326]	Country Risk Run [page 1336]	/BA1/R2_CRE_MAIN
Historical Database [page 1478]	Historization Run for Data Layers [page 1481]	/BA1/R6_HIST_RUN
Historical Database	Historization Run for Bank's Models [page 1484]	/BA1/R6_HIST_RUN_MODEL
Historical Database	Determine Default Rates [page 1488]	/BA1/R6_CF_DEFAULT_4210
Historical Database	Determine Average Default Rates [page 1491]	/BA1/R6_CF_DEFAULT_4220
Historical Database	Determine Default Figures [page 1493]	/BA1/R6_CF_DEFAULT_KEYFIGURES
Historical Database	Exporting Data to In-House Models [page 1498]	/BA1/R6_MOD_RUN
Historical Database	Calculation of Migration Matrices [page 1496]	/BA1/R6_CF_MIGRATIONS
Historical Database	Stress Run for Supplying Models with Data [page 1505]	/BA1/R6_MOD_RUN_SCENARIO
Historical Database	Stress Run for Supplying Data [page 1506]	/BA1/R6_FDB_RUN_SCENARIO
Regulatory Reporting Interface	Processing Framework Run Copying the Start ID	/BA1/S1_COPY_STARTID

Application	Run Category	Program
Regulatory Reporting Interface	Processing Framework Run Creating Runs	/BA1/S1_CREATE_RUN
Regulatory Reporting Interface	Processing Framework Run Release Run	/BA1/S1_RELEASE_RUN
Regulatory Reporting Interface	Processing Framework Run Creating a Worklist	/BA1/S1_CREATE_WRKL
Regulatory Reporting Interface	Processing Framework Run Processing a Run	/BA1/S1_PROCESS
Regulatory Reporting Interface	Data Extraction Runs	/BA1/S3_EXTRACT_RUN
Limit Manager [page 1529]	Limit Utilization Run [page 1551]	/BA1/R3_UTILIZATION_RUN_DPF

More Information

Schedule Manager

4.6.2.4.2 Managing Runs

Use

Run administration provides an overview of the runs for the effectiveness test. The system displays all the main attributes of the runs, including the key date, the creation date, and the current and previous status of the runs.

You can also release the results of individual runs for Accounting, and set deletion flags.

> **Note**
>
> In run administration, the system displays batch runs only; it does not display online runs [page 1054].

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Run Administration* > *Management* > *Manage Runs*.
   The system displays the initial screen for run administration.

2. Enter the external ID, business key date, version, and system status of the runs that you want to manage. You can also select a layout as the display variant.
3. Choose .
   The system displays an overview of the runs.

Overview of the Selected Runs

The system displays an overview of all the runs for the effectiveness test that match the data entered; this includes information about the business key date, creation date, and status of the runs. The runs can have the following status:

Status	Description
Crtd	Run created
Strt	Run started
Canx	Run cancelled
RCM	Run was changed manually
ROFC	*Run Obsolete* flag was cleared
End	Run finished
RRHA	Released for HA (Hedge Accounting)
CRHA	Release for HA (Hedge Accounting) was reset
FRHA	Hedge Accounting run was fetched
Del.	Deletion flag was set
Rvke	Deletion flag revoked
RINC	Run finished but not complete

To display more information about the run for the effectiveness test, release the run for Hedge Accounting, or flag it for deletion, select the run, and choose . The system displays a detail screen.

Displaying Runs Individually

The system displays detailed information about the selected run.

- *Configuration* tab page
  Contains information about the Customizing settings for the effectiveness test environment and run type.
- *Result Database* tab page
  Contains information about the key date, run type, and Result Database (RDB) package type.
- *Client (HA)* tab page (client run information)
  Contains information about the accounting system, the legal entity, and status displays that indicate whether the run has been released and can be called by Hedge Accounting or whether it has already been called in Hedge Accounting.

You can use the following functions:

Action	Function
	The system switches to display mode
	The system switches to change mode
	The system flags the displayed run for deletion **Note** This action can be carried out in change mode only.
	Hedge Accounting cannot call the results of the effectiveness test until you have set this checkbox (see Micro Fair Value Hedge (Subledger Scenario) [page 905]). You can release only one run per business key date for each legal entity and each accounting system. When a run is released for Accounting, all of the hedging relationships analyzed in the run for the effectiveness test are released for Hedge Accounting. **Note** This action can be carried out in change mode only.
	The system cancels the release of runs that have already been released. **Caution** You cannot cancel the release if Hedge Accounting has already called the results.
	The system branches to reporting [page 1055].
	The system displays the field instance for the run for the effectiveness test.
	The system displays the current and previous status of the run

4.6.2.4.3 Deleting Runs

Prerequisites

- Hedge Accounting has not yet processed the runs that you want to delete.
- Under Managing Runs [page 1050] you have set the deletion flag for the runs.

Context

You can delete runs for the effectiveness test.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Run Administration* > *Delete and Archive* > *Delete Runs*.
2. Enter the external ID, business key date, type, and version of the run that you want to delete.
3. Choose to delete the run immediately or to simulate its deletion.
4. You can also specify a layout as the display variant.
5. Choose *Execute*.

Results

The system has deleted the selected run.

4.6.2.4.4 Archiving of Runs

Use

You use this function to archive runs for the fair value effectiveness test for hedging relationships. You need to do this if you want to remove data from the database without deleting the data permanently.

You can archive only those runs that have been completed. You should archive only the runs that you no longer require for your analyses, or whose results were already processed by Hedge Accounting.

Activities

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Run Administration* > *Delete and Archive* > *Archive Runs*.

The system displays the initial screen for SAP Archive Administration (transaction `SARA`). The default value of the *Archiving Object* field is the archiving object for the effectiveness test (`BA1_R4_056`).

4.6.2.4.5 Online Runs

Use

Runs for the effectiveness test that are scheduled as batch runs usually measure the effectiveness of all the hedging relationships for a given legal entity. You can use an online run to measure the effectiveness for a small dataset. Online runs differ from batch runs in the following ways:

- In reporting for online runs, the results are kept in the main memory but are not saved.
- In reporting for online runs, you can display a detail log that contains information about the processes the Fair Value Server used to assess the hedged items and hedging instruments. This is not possible in reporting for batch runs.
- In reporting for batch runs, you can change the results of the effectiveness test; this is not possible in reporting for online runs.
- Online runs do not consider the history of the fair value effectiveness test.
- In online runs, you can analyze hedging relationships across multiple legal entities. For this reason, you can specify a time stamp for selecting the hedging relationships.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Run Administration* > *Create and Schedule* > *Online Run*.
2. Enter the following data:
   - Primary data source for generating the worklist
     Choose a primary data source and specify the data source category. You can choose from the data source categories *InfoSet* and *General Primary Data Source*.
   - Other selection criteria
     In order to define which hedging relationships are to be analyzed, enter the required values for the characteristics contained in your primary data source.
   - Secondary data source for the selection of hedging relationship and the derivation of the effectiveness test rule
   - Key date
   - Effectiveness test environment
   - Field instance

- Decide whether you want you want to specify an effectiveness test rule [page 1042] along with the effectiveness test, and if so specify the rule required
- Time stamp for calculating the *Hedge Fair Value* and *Hedge Amortized Cost* auxiliary key figures
  The system uses this time stamp when it generates the cash flows for the transactions in the hedging relationship. The time stamp is required because the data is stored in the Hedge Accounting Workplace in two dimensions –the key date and the time stamp. If you do not enter a time stamp, the system uses a current time stamp.
- Decide whether you want to use the hypothetical derivative approach [page 1037] to test the effectiveness of the hedging relationships
- Define whether the system is to create a new version of the run if incomplete runs are restarted
- Specify which message types are to be displayed in the application log

3. Choose *Execute*.
   The system starts the online run and displays the results. For more information, see Reporting [page 1055].

4.6.2.5 Reporting

You can use the following functions to display the results of the fair value effectiveness test for hedging relationships:

- Reporting functions for the effectiveness test [page 1055]
- In regression analysis, you have the following options:
  - Using graphs to analyze value changes [page 1061]
  - Using graphs to analyze regression lines [page 1065]
  - Displaying the time series for the retrospective regression analysis [page 1068]

4.6.2.5.1 Display Reports

Use

Reporting provides you with an overview of the results of the fair value effectiveness test [page 1036]. The system displays the key figures and effectiveness indicators calculated for all the hedging relationships in a run for an effectiveness test.

You can use the following functions for test runs that have not yet been processed by Hedge Accounting [page 904]:

- Recalculate the effectiveness of hedging relationships
  You can use a different effectiveness test environment for the recalculation.
- Manually change the final effectiveness indicator for hedging relationships
- Manually change the start date of ineffectiveness for hedging relationships

Prerequisites

The system has successfully completed the runs you want to display.

If you want to change the test results, Hedge Accounting must not have processed the runs in question. The system displays the status of the runs for the effectiveness test in run administration [page 1050].

If you want to use a new effectiveness test environment to recalculate the effectiveness of hedging relationships, you have to have created a pool of test environments in Customizing for the fair value effectiveness test. To do so, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Effectiveness Test Environment and Rules* > *Edit Effectiveness Test Environment*.

Procedure

You can access reporting in the following ways:

- From run administration [page 1050]
- In the online run [page 1054]
- By calling it up directly (for batch runs)

Selection

To call up reporting directly from the area menu, proceed as follows:

1. On the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Reporting* > *Display Reporting*.
   The system displays the initial screen for reporting.
2. Enter the run name, run version, and key date of the effectiveness test run for which you want to display the results of the effectiveness measurement.
3. Specify whether the system is to display only hedging relationships that were assessed as being effective, ineffective, or incorrect.
4. To display only a selection of the hedging relationships from an effectiveness test run, choose *Segment Sel*.
   Enter values for the characteristics you are using for the selection.

   > **Note**
   >
   > To select the segment, you can use all the characteristics in the field instance you are using. For more information, see Display of Field Instances [page 1106].

5. Choose *Execute*.
   The system displays the results of the selected effectiveness test run.

> **Note**
>
> Note that if you used the hypothetical derivative approach [page 1037] to test the effectiveness of cash flow hedging relationships, the system displays fictitious hedging relationships. These fictitious hedging relationships comprise a hypothetical derivate as the hedged item, and one hedging instrument.
>
> If you want the system to display the object ID of the hedging instruments in reporting, then in Customizing for the fair value effectiveness test you need to have stored the relevant characteristic in the field instance.

Action

The system displays tab pages [page 1059] containing detailed information. You can use the following functions.

Action	Function
Display/Change	Changes to display or change mode. **Note** To change the final effectiveness indicator and the start date of ineffectiveness, and to analyze hedging relationships again, you must be in change mode.
Run administration	Branches to run administration [page 1050].
Application Log	The system displays the log for the effectiveness measurement.
Detail log	The system displays the detail log [page 652] for the Fair Value Server. It contains information about the analysis of the hedged items and hedging instruments that are contained in the hedging relationships in the effectiveness test run. **Note** The detail log can be displayed only after an online run [page 1054].

To edit individual hedging relationships, choose the *Effectiveness Indicator* tab page and select one or more hedging relationships. You can use the following functions.

Action	Function
Display Hedging Relationship	The system displays detailed information about the hedging relationship selected.
Recalculate Hedging Relationship	The system recalculates the effectiveness of the selected hedging relationship. Specify which test environment the system is to use to measure the effectiveness.

Action	Function
Change the value of effectiveness indicator	Place the cursor in the *Effectiveness Indicator* field and select the required value from the input help. **Note** On the *Effectiveness Indicator* tab page, you can change only the final effectiveness indicator. You cannot change the effectiveness indicators for the individual test methods.
Change the start date of ineffectiveness	Enter the required date in the *Ineffective as On* field.
Graphical Analysis	Branch to the following graphical representations: • Development of Value Changes [page 1061] • Regression Line [page 1065] You can display a list for this with all the measured values. **Note** This option is only available in the *Retrospective Regression* and *Prospective Regression* tab pages.
Regression Analyses	The system displays the results of the similar asset test for all the hedged items in the portfolio. **Note** This option is available only on the *Similar Asset Test* tab page.
Display the history	In the context menu (right mouse button), choose *History*. The system displays the results of all runs that are before the current key date, and contain the hedging relationship that was selected.

You can change the effectiveness indicator and the start date of ineffectiveness only for runs that have not yet been processed by Hedge Accounting. Run administration [page 1050] provides an overview of the runs that have already been processed.

If you access reporting from an online run [page 1054], you cannot change the results of the effectiveness measurement.

4.6.2.5.1.1 Tab Pages

Definition

In reporting for the fair value effectiveness test function, the system displays the results of the effectiveness test on several tab pages.

The *Effectiveness Indicator tab page contains general information, particularly the value of the* final effectiveness indicator. On this tab page, you can change the final effectiveness indicator and the start date of ineffectiveness manually.

Each test method has a separate tab page, which contains all the important information about the valuation as well as the result in the form of an effectiveness indicator.

If you use your own methods for testing the effectiveness of hedges, then the system displays up to three tab pages for these effectiveness methods (*Custom Method 1, Custom Method 2* , etc). If you use more than three custom effectiveness test methods, the system displays the results of the additional methods in the *Custom Method 3* tab page.

Note that these tab pages contain only those fields that you assigned to your custom effectiveness test methods in Customizing for the field instance for *Fair Value Effectiveness Test* . For more information, see Custom Effectiveness Test Methods [page 1099] .

Structure

General Information

Column Heading	Explanation
Hedge Category	
Hedging Relationship ID	Identifier for the hedging relationship
Test Method Effectiveness Indicator	Each test method sets a separate effectiveness indicator. On the *Effectiveness Indicator* tab page, the system displays the final effectiveness indicator, which it derives using the rule set [page 1103] and from the effectiveness indicators of the individual methods.

Effectiveness Indicator Values

Effectiveness Indicator	Explanation
Effective	The hedging relationship tested is effective.
Ineffective	The hedging relationship tested is ineffective.
Incorrect	Errors occurred during the valuation.
Temporarily Ineffective	The hedging relationship tested is temporarily ineffective. Only the effectiveness indicators for the individual test methods can have this value, not the final effectiveness indicator.
Valuation w/o Result	The hedging relationship could not be assessed. For more information, call up the error log.
Not Assessed	No result is available for the effectiveness test.

Detailed Information

Effectiveness Indicator Tab Page

Final Effectiveness Indicator

Prospective Effectiveness Indicator

Retrospective Effectiveness Indicator

Total Changes in Value of Hedging Instruments

Total Changes in Value of Hedged Items

Total Nominal Value of Hedged Items

Ineffective as on Date

Effectiveness Test Method

Reference Currency for Hedging Relationship

Effectiveness Test Environment

On the *Effectiveness Indicator* tab page, you can change the final effectiveness indicator and the ineffective as on date. For more information, see Displaying a Report [page 1055] .

Retrospective Regression AnalysisandProspective Regression AnalysisTab Pages

Axis section of regression lines

Slope of regression lines

Normed hedge ratio

Coefficient of determination

Variance of axis section

Variance of slope

T-quantile

F-quantile

F-statistic

Sample size (number of data points actually used by the system in the effectiveness test)

Criterion that was used to determine the effectiveness (f-test [page 1090] , t-tes t [page 1093] or simple interval test [page 1095])

Indicator showing whether the y or x axis were switched.

Similar Asset TestTab Page

Number of hedged items in the portfolio

Number of hedged items in the portfolio that are not similar assets

Tolerated percentage of hedged items that are not similar assets

Offset MethodTab Page

Hedge ratio

Noise threshold (*Value Changes* field)

Number of temporarily ineffective tests

Criterion for effectiveness in offset method

Tolerance Value CheckTab Page

Expired term of hedging relationship

Percentage change in value of hedged items

Percentage change in value of hedging instruments

Residual maturity of hedging relationship

Number of consecutive tolerance value checks

4.6.2.5.1.2 Using Graphs to Analyze Value Changes

Use

You use this function to display in a graph the changes in the values of the hedged items and hedging instruments in a hedging relationship.

You use the function to depict the development of the changes in the value of the hedged items and hedging instruments over a longer period of time, and so that you can understand the results of the fair value effectiveness test. In retrospective regression analysis, the system displays the historical values for the value changes; in prospective regression analysis it shows the future changes in value. When market data scenarios are used, it shows estimated changes in value.

> ⚠ Caution
>
> You can use this function only if you used retrospective regression analysis [page 1083] or prospective regression analysis [page 1088] in the fair value effectiveness test.
>
> Note that the system uses the same graph of value changes to display the time series for retrospective regression analysis and, in the reporting function, for the fair value effectiveness test.

Activities

To call up the graph showing the value changes, proceed as follows:

- From the reporting function for the fair value effectiveness test [page 1055]:

  Choose the *Retrospective Regression Analysis* or *Prospective Regression Analysis* tab page, and then  .
  The system displays the value changes that it used for fair value effectiveness test.
- From the function for displaying time series for retrospective regression analysis [page 1068]:
  Choose the *Graphic* tab page.
  The system displays the changes in value for the data points in the time series. Note that the time series can contain more data points than the system actually used for measuring the effectiveness.

You can use the following functions.

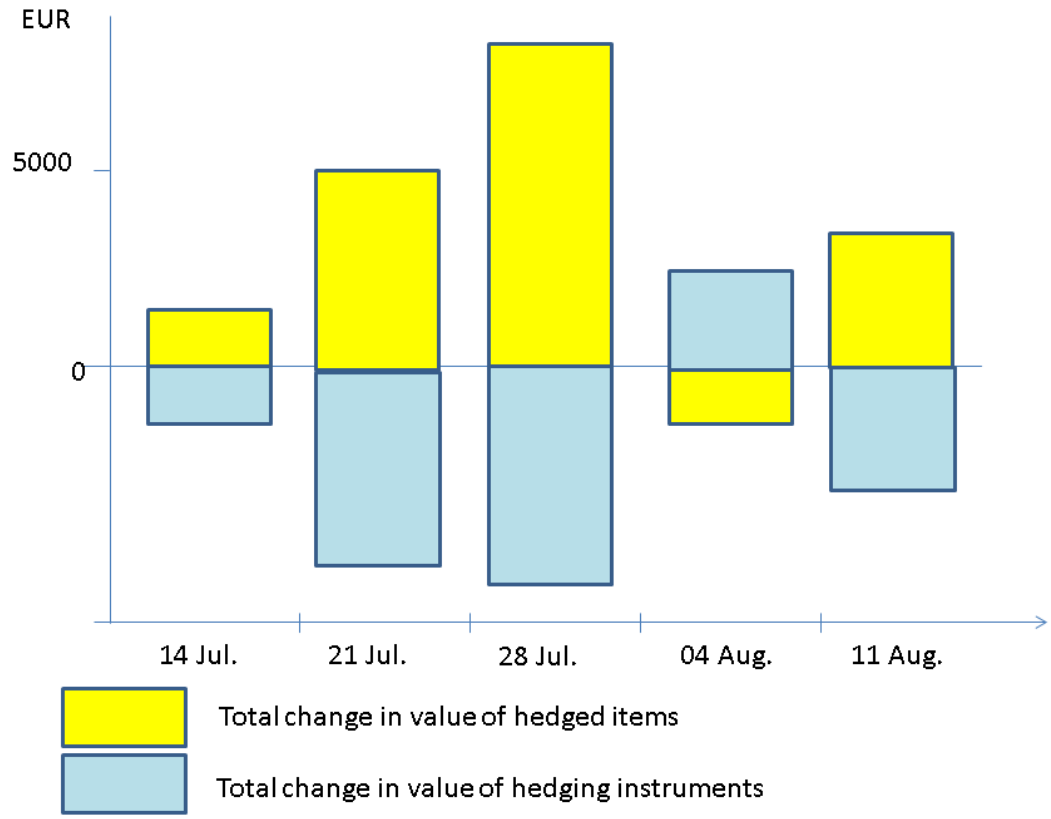
Action	Function
Graphic > *Display* > *Hide/Display Normed Hedge Ratio*	The system hides or displays the normed hedge ratio.
Graphic > *Display* > *Hide/Display Total Nominal Value*	The system hides or displays the total of the nominal values of the hedged items. Note that when you use this function, you change the scaling of the ordinates.
Graphic > *Display* > *Limits of the Range*	The system displays a different section of the time series. You need to enter the start and the end of the section you want the system to display.

Action	Function
Graphic > *Choose Scenario*	The system displays a dialog box that contains all the market data scenarios that you used in the fair value effectiveness test When you have chosen a market data scenario, on the tab page *Development of Value Changes* or *Graphic*, the system displays the changes in value for the selected market data scenario only. **Note** You have this option only for prospective regression analysis [page 1088]. and if you have used a scenario set with multiple scenarios in the fair value effectiveness test.

Example

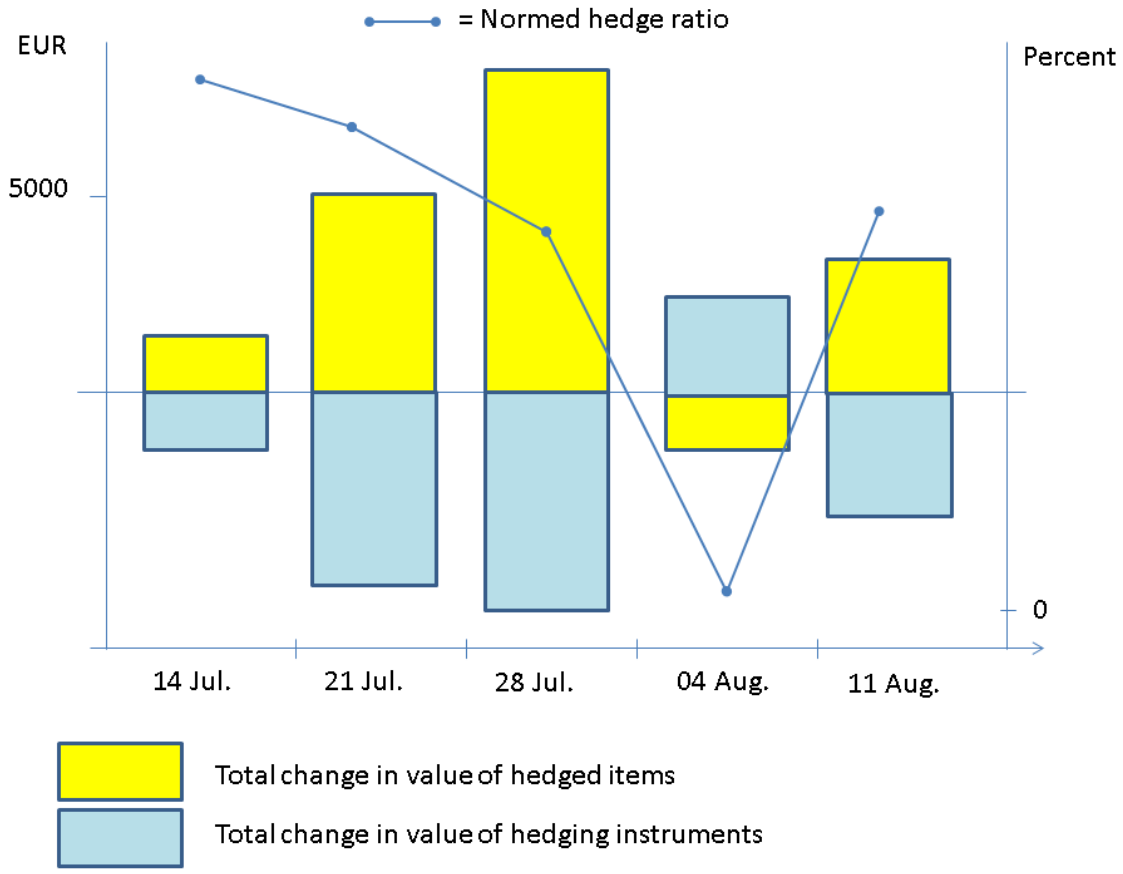
The following diagram shows how a graph for a typical hedging relationship may look.

- **Graph showing only the changes in the values of the hedged items and hedging instruments**



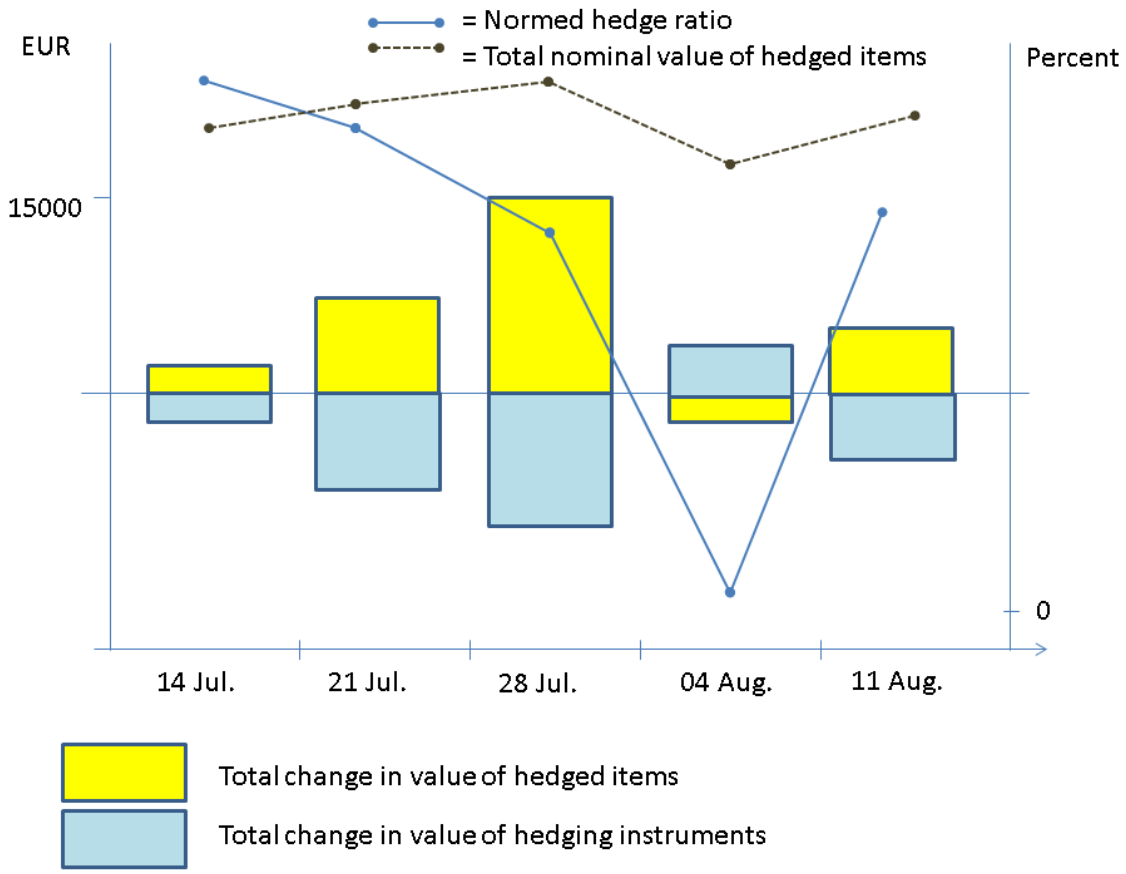
For each date value in the time series, the system displays the value changes that were calculated for the hedged items and hedging instruments. The key figure that is calculated is the hedge adjustment [page 1073]. The left-hand scale shows that the hedge adjustment contains absolute amounts in the currency of the hedging relationship (in our example, the currency is EUR).

- **Graph showing the normed hedge ratio**



If you display the normed hedge ratio, then, for each date value, the system displays the normed hedge ratio and uses a line to connect the points. The scale for the normed hedge ratio is on the right-hand side.

- **Graph showing the total nominal value of the hedged items and the normed hedge ratio**



If the total of the nominal values of the hedged items is displayed, then the system displays the total of the nominal values of the hedged items for each date value, and connects the points with a line. At the same time, the system adjusts the left-hand scale for absolute amounts. This causes the bars for the changes in the value of hedged items and hedging instruments to appear smaller, and gives you an impression of how large the changes are in comparison to the total of the nominal values of the hedged items.

4.6.2.5.1.3 Using Graphs to Analyze Regression Lines

Use

In this function, you can use graphs to analyze regression lines.

This helps you understand the results of the fair value effectiveness test for a hedging relationship. In particular, you can use the graph to check whether the statistical tests that you used are plausible.

Integration

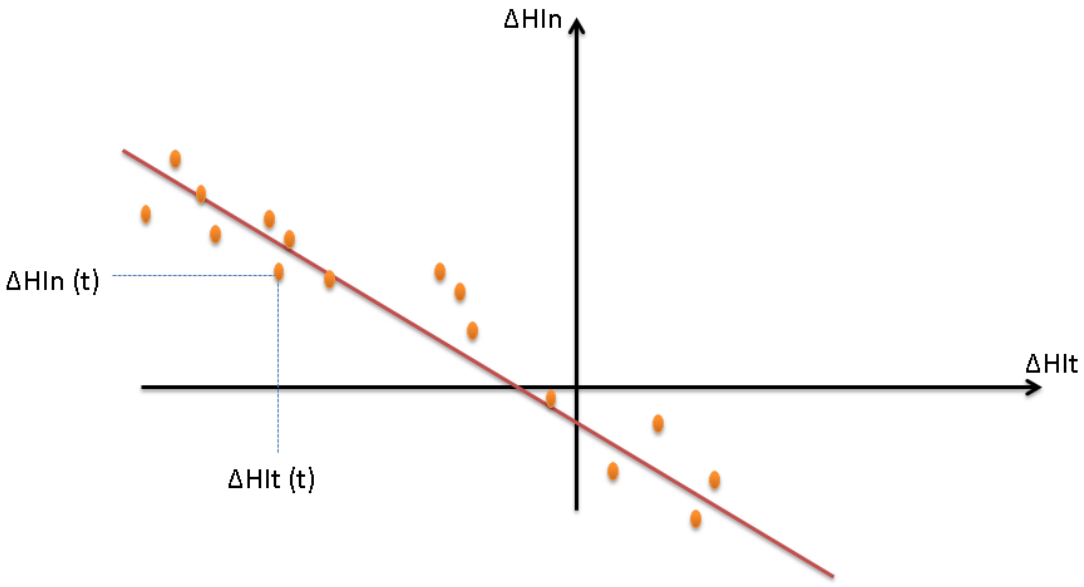
You can use this function only if you used retrospective regression analysis [page 1083] or prospective regression analysis [page 1088] in the fair value effectiveness test.

Activities

1. In the reporting function for the fair value effectiveness test, choose the *Retrospective Regression Analysis* or *Prospective Regression Analysis* tab page, and then *Graphical Analysis*.
   The system displays the graph.
2. Choose the *Regression Line* tab page.
   The system displays the egression line. The system labels each data point in the regression analysis with the date of the measurement. In the prospective regression analysis, it also adds name of the market data scenario that was applied.
   To hide the labels, choose *Graphic* > *Display* > *Hide Description*.

Example

- **Retrospective Regression Analysis and Prospective Regression Analysis Using Market Data Scenarios**
  For each time point in the graph, the system shows the changes in the value of the hedging instruments and the changes in the value of the hedged items, and draws the regression line through the points.



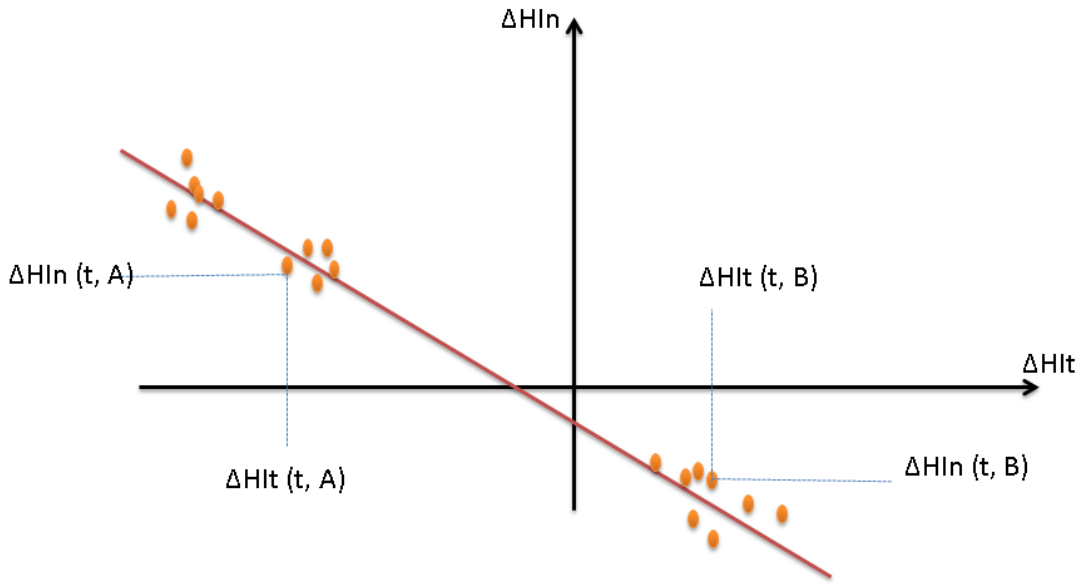
= Data point for time point t

Regression line

ΔHIt = Changes in the value of the hedged items in the hedging relationship

ΔHIn = Changes in the value of the hedging instrument in the hedging relationship

- **Prospective Regression Analysis with Multiple Market Data Scenarios**
  If you use multiple market data scenarios in regression analysis, then the system displays the changes in values for each time point, and for each market data scenario that is valid for this time point.



= Data point for time point t and market data scenario X

Regression line

ΔHIt (t,X) = Changes in the value of the hedged items in the hedging relationship for time point t and market data scenario X

ΔHIn (t,X) = Changes in the value of the hedging instruments in the hedging relationship for time point t and market data scenario X

4.6.2.5.2 Display of Runs for a Legal Entity

Use

You can use this function to display all the runs for the fair value effectiveness test that belong to a particular legal entity. You can also restrict the runs that are displayed to those that were already processed by Hedge Accounting.

By doing so, you obtain an overview of which effectiveness test runs were used in the key date valuation by the Accounting Processes.

You can display details about each run by branching from the screen displaying the runs to run administration.

Activities

1. On the *SAP Easy Access screen* , choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Reporting* > *Display Runs for a Legal Entity.*
   The system displays a selection screen.

2. Choose a legal entity.
3. Decide whether you want to display all the runs for this legal entity, or just the runs that were fetched from Hedge Accounting (default setting).
4. Choose  w ith the quick info *Execute* .
   The system displays a table containing the runs that were selected. For each run, it displays the key date, the version, the accounting system, and the legal entity.
   To display detailed information about the run, select the run and choose *Run Administration.*

4.6.2.5.3 Display of Time Series for Retrospective Regression Analysis

Use

You use this function to display the time series that the system generated for retrospective regression analysis. Doing so enables you to understand how the fair value effectiveness of hedging relationships was tested. Once you have selected a time series and a hedging relationship that has been analyzed, the system displays for each date value the changes in the values of the hedged item and hedging instrument in a hedging relationship, plus the total nominal value of the hedged items. You can also have the system display this information in a graph.

You can use the reporting function for time series in order to see how the changes in the value of the hedged items and hedging instruments in a hedging relationship have developed over time. However, this is not possible in the reporting functions of the fair value effectiveness test, since only the regression line is displayed there, and it is not possible to display the underlying time series in a detail view.

Integration

You use this function only if you use the retrospective regression analysis [page 1083] method in the fair value effectiveness test.

Prerequisites

The system has already generated the time series that you want to display. For more information, see the document Generation of Time Series for Retrospective Regression Analysis [page 1084] .

Activities

1. On the *SAP Easy Access screen* , choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Reporting* > *Display Time Series for Retrospective Regression Analysis.*

The system displays the time series for retrospective regression analysis. In the left part of the screen, the system displays a navigation structure that contains all the time series and the associated hedging relationships. You open the tree by clicking on the arrow to the left of the *Time Series* name, or on the arrow on the level that you want to open.

2. In the navigation structure, choose the hedging relationship for which you want to display the time series.
   In the right-hand screen area, the system displays the name of the time series, its start date, and the distance between the data points.
   On the **Measurements tab page** , the system displays the individual data points in the time series. For each date, it displays the change in the value of the hedged item and hedging transaction, their nominal values, and the time stamp of the valuation.
   On the **Graphic tab page** , the system displays the changes in the value of the hedged items and hedging instrument as a graph. For more information, plus some examples, see the document Using Graphs to Analyze Value Changes [page 1061] .

When you use the graph to analyze the changes in the value of transactions in hedging relationships, the functions in the table below are available. You access all of these functions by using the context menu (right mouse button).

Action	Procedure	Function
Simulate Time Series	In the navigation structure, select the time series required, and then in the context menu choose the option *Simulate Time Series* . Enter a start date and an end date.	The system generates a list of date values, which have the same distance between them as the measurements in the time series you selected (distance and unit are the same).
Delete Obsolete Versions	In the navigation structure, select the time series required, and then in the context menu choose the option *Delete Obsolete Versions* .	The system deletes all older versions of the selected time series. Only the version with the most recent time stamp is retained on the database. You use this function if you generate multiple time series that have the same values for the parameters Start Date, Distance, and Unit. The system creates a new version for each of these time series. Since the fair value effectiveness test uses only the latest version of a time series, the older versions can be deleted.
Delete Data Points from Time Series	In the navigation structure, select the hedging relationship required, and then in the context menu choose the option *Delete Data Points from Time Series* .	The system deletes the data points of the selected hedging relationship from the time series.
Navigate to Run	In the *Measurements* tab page, select a row with measurements, and in the context menu choose *Navigate to Run* .	The system branches to run administration [page 1070] and displays the run in which the time series was generated and to which the selected measurements belong.

4.6.2.5.4 Displaying the Runs Used to Generate Time Series

Use

You use this function to display all the runs that were started for the generation of the time series for retrospective regression analysis [page 1084]. You can branch from the list of runs to the application log and the detail log. You can also delete obsolete runs.

Integration

You use the runs to generate time series only if you want to carry out retrospective regression analysis [page 1083] in the fair value effectiveness test.

Activities

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Reporting* > *Display Runs for Generation of Time Series*.

The system displays an overview of all the runs that you started in order to generate time series.

To process the data further, select the run required, and choose one of the following functions:

Action	Function
Details	The system displays more information about the run you selected.
Detail Log	The system displays the detail log [page 652] for the Fair Value Server.
Application Log	The system branches to the application log.
Delete run	The system deletes the selected run.

The system displays the following information about the run: name of the run, primary data source, secondary data source, effectiveness test environment, field instance, market data area, global valuation environment, set of rules for deriving the effectiveness indicator, cash flow set, run type, date, time stamp, and the name of the user who created the run.

4.6.2.6 Effectiveness Test Methods

Use

The International Accounting Standards (IAS) stipulate that the effectiveness of hedging relationships be checked before hedge accounting is used. IAS does not specify which test methods have to be used. For this reason, the fair value effectiveness test provides you with several accepted methods for testing the effectiveness of hedges.

In the Customizing for the fair value effectiveness test, you can configure the individual test methods and how you want the effectiveness test to be carried out.

Integration

For information about calling up the effectiveness test and about the process flow of the test methods, see Processes of the Fair Value Effectiveness Test [page 1039] .

In the calculation bases [page 1072] document you can find information about using hedge ratios and about hedge adjustment [page 1073] , which is used as a key figure for the value changes to the hedged items and hedging transactions in a hedging relationship.

Features

The fair value effectiveness test contains the following methods for testing the effectiveness of hedges:

- **Tolerance Value Checks**
  - Test of the maturity and residual maturity
    If the hedging relationship was designated only a few days ago, or if it is about to be dissolved, the system sets the final effectiveness indicator to effective, without carrying out any further checks.
  - Test of value changes
    If only minor changes in value have taken place, the system sets the final effectiveness indicator to effective, without carrying out any further checks.
- **Internal Effectiveness Measurement**
  SAP provides the following methods for testing the effectiveness of hedges:
  - Prospective effectiveness test
    - Prospective regression analysis [page 1088]
      The methods for the prospective test set their own indicator, the prospective effectiveness indicator.
  - Retrospective effectiveness test
    - Offset method [page 1076]
    - Retrospective regression analysis [page 1083]
      The methods for the retrospective test set their own indicator, the retrospective effectiveness indicator.
      In addition to using the effectiveness test methods predefined in the system, you can create and use your own custom effectiveness test methods [page 1099] . Usually, custom effectiveness test methods set the prospective and retrospective effectiveness indicators in the same way as the methods predefined in the system.

The system derives the final effectiveness indicator from the results of the prospective and retrospective effectiveness tests. It uses the rule set for effectiveness [page 1103] for this purpose.

- External Effectiveness Measurement [page 1100]
  Transfer of test results determined externally

4.6.2.6.1 Calculation Bases

The following prerequisites and conventions apply for the methods used for testing effectiveness:

Use of the Hedge Ratio

The central key figure for the effectiveness measurement is the hedge ratio *HR*. This key figure represents the relationship between the change in value of the hedged items and the change in value of the hedging instruments. It specifies the effectiveness of the hedging instruments with reference to the changes in the fair value of the hedged items.

The individual test methods calculate the hedge ratio in different ways.

A hedging relationship is usually classified as effective if the hedge ratio is between 80% and 125%. You can choose these barriers in the Customizing for the *offset method*, for example.

Use of the Hedge Adjustment

To describe the change in the values of the hedged items and hedging instruments in a hedging relationship, the system uses the hedge adjustment, which is defined as follows:

Hedge adjustment = hedge fair value – hedge amortized cost

For more information, see Hedge Adjustment [page 1073].

Prorated Hedge

As well as a 1:1 relationship between hedged items and hedging instruments, the system allows a prorated hedge. Any references made below to the hedged items and hedging instruments of a particular hedging relationship are to be understood as the portions of the financial transactions or instruments used as hedged items or hedging instruments.

Norming of the Hedge Ratio to the Nominal Values of Hedged items

For some effectiveness test methods, the system norms the hedge ratio to enable it to be displayed consistently in graphical form. It norms the original value for the hedge ratio to the total of the nominal values of the hedged items in the respective hedging relationship.

> **Note**
>
> You can use programs `/BA1/R8_FVAC_SUBCON_EVAL` and `/BA1/R8_FVAC_HEDGE_EVAL` to test the Fair Value Server [page 628]'s valuations.

4.6.2.6.1.1 Hedge Adjustment

Use

To describe the changes in the value of the hedged items and hedging instruments, the systems uses the hedge adjustment rather than a fair value. This value represents only those value changes that result from the hedged interest rate risks of the hedging relationships. The effects of other interest rate risks and any influencing factors that are not relevant for the effectiveness measurement are ignored.

The hedge adjustment is used by all the effectiveness test methods, including the tolerance value check for value changes [page 1074] and the offset method [page 1076] .

Integration

In order to calculate the hedge adjustment, the fair value effectiveness test calls the following functions:

- Fair Value Server [page 628]
  The Fair Value Server provides the auxiliary key figures *hedge fair value* and *hedge amortized cost,* which are used in the fair value effectiveness test to calculate the hedge adjustment.
- Portfolio Fair Value Calculator [page 1175]
  For portfolio fair value hedging relationships, the Portfolio Fair Value Server itself calculates the hedge adjustment.

Calculation

The hedge adjustment is calculated as follows:

$$HA_i(t) = HFV_i(t) - HAC_i(t)$$

where *HA* is the hedge adjustment at time point *t* , *HFV* is the hedge fair value, and *HAC* is the hedge amortized cost. In contrast to the hedge adjustment, the *hedge fair value* and the *hedge amortized cost* are simply auxiliary figures in Accounting and are not directly used there.

The system first calculates the hedge adjustment for each transaction separately, or for each portion *i* in the hedging relationship (using the hypothetical derivative approach for micro fair value hedging relationships and cash flow hedging relationships) or for time bucket *i* (for portfolio fair value hedging relationships). For the actual effectiveness test, the system totals the hedge adjustments of all hedged items and hedging instruments, or for all time buckets:

$$HA_{HR}(t) = \sum_{i} \left(HFV_i(t) - HAC_i(t) \right)$$

$$HA_{HR}(t) = \sum_{i} \left(HFV_i(t) - HAC_i(t) \right)$$

Properties

The hedge adjustment has the following features:

- **It is not dependent on unhedged risks**
  When the system calculates the hedge fair value [page 633] , it discounts the cash flows using the hedge spread [page 641] . This spread is not maturity dependent, and contains all the risks that are **not** hedged by the hedging relationship. No market data is used to calculate the hedge amortized cost [page 636] . For this reason, the hedge adjustment depends only on those risks that are hedged by the hedging relationship.
- **It is not dependent on the structure of the cash flows**
  Net present value (NPV) key figures that are calculated using the *cash flow discounting* pricing model are usually dependent on the structure of the transaction cash flows. For example, the progression of loans with a discount shows a drift. This drift is described by the hedge amortized cost. So that the drift effects do not affect the effectiveness test, the hedge amortized cost [page 636] is deducted from the hedge fair value [page 633] during the hedge adjustment.
- **The recording of value changes after the designation date**
  By calibrating values to the hedge amortized cost, the hedge adjustment records only value changes that occurred after the designation date of the hedging relationship. In particular, if the target values for the hedge amortized cost and the hedge fair value are identical on the designation date of the hedging relationship, the hedge adjustment on this date is equal to zero.

4.6.2.6.2 Tolerance Value Checks

Use

You have to test the effectiveness of certain hedging relationships. This might be necessary if there were only minor value changes, if a hedging relationship was designated only a few days earlier, or if a hedging relationship is to be dissolved soon.

We provide methods that you can use to define which hedging relationships do not require a separate effectiveness test. The methods are not dependent on one another and can be called individually.

Prerequisites

You configure the tolerance value checks in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness Test* > *Set Up Tolerance Value Check*.

The system applies the methods for the tolerance value check only if they are contained in the current test schema. You assign the test schema you want to use to a combination of effectiveness test environment and effectiveness test rule [page 1042]. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness Test* > *Edit Rules for Effectiveness Test*.

Features

Value Changes

The system sets the Final Effectiveness Indicator to Effective when the following criteria are met:

- The maximum number of successful tolerance value checks was not exceeded.

$$k \leq n$$

  Where *k* = the number of tolerance value checks carried out in direct succession and in which the hedging relationship was classified as *effective*, *n* = the maximum number of such checks specified in Customizing.

  > **Note**
  >
  > To be able to apply these criteria, the system constructs a history which shows how often the final indicator was set as a result of the tolerance value check on value changes. This applies only to runs that were processed by Hedge Accounting.

- The value changes for the hedged items offset those for the hedging instruments.

$$sign(HA_{HIt}(t)) \neq sign(HA_{HIn}(t))$$

  Where *t* = key date, *HIt* = quantity of hedged items and *HIn* = quantity of hedging instruments in the hedging relationship, *HA* = hedge adjustment [page 1073]; *sign* = plus/minus sign function.

- The hedged items and hedging instruments show only minor value changes.

$$\left|\frac{HA_{HIt}(t)}{Nom}\right| \leq tol_{HIt} \text{ and } \left|\frac{HA_{HIn}(t)}{Nom}\right| \leq tol_{HIn}$$

  Where *t* = key date, *HIt* = quantity of hedged items and *HIn* = quantity of hedging instruments in the hedging relationship, and *HA* = hedge adjustment. *Nom* = total of the nominal values in the hedged items, where the current remaining debt on the key date is used as the nominal value.

Maturity and Residual Maturity

The system sets the final effectiveness indicator when at least one of the following criteria is met:

- The designation of the hedging relationship is a maximum of tol_{Lt} days in the past.

$$\left|t - t_d\right| \leq tol_{Lt}$$

  Where *td* = designation date of the hedging relationship, *t* = key date, and tol_{Lt} = tolerance limit defined for the maturity in Customizing.
- The hedging relationship is to be dissolved in a maximum of tol_{Rt} days.

$$\left|t_f - t\right| \leq tol_{Rt}$$

  where t_f = date on which the hedging relationship matures, *t* = key date, and tol_{Rt} = tolerance limit defined for the residual maturity in Customizing.

When calculating the difference between the dates, the system uses a calendar that does not distinguish between workdays and public holidays.

Error Processing

The methods of the tolerance value check call the Fair Value Server and other functions. If an error occurs in one of the required functions, the system records an error message in the application log.

4.6.2.6.3 Offset Method

Use

This function measures the effectiveness of a hedging relationship by comparing the changes in the value of the hedged items and the hedging instruments in a particular hedging relationship. The main key figure is the quotient of the two values, the hedge ratio.

The offset method is a retrospective test method, since it takes into account value changes that occurred in the past only. In contrast to retrospective regression analysis, this method does not require the history to cover a long period, nor does it use complex statistical methods.

In its simplest form, the offset method has the disadvantage that its result can fluctuate significantly when minor value changes occur. This can cause effective hedging relationships to be incorrectly classified as ineffective. For this reason, SAP provides an enhanced offset method that you can use to limit the effect that very minor value changes have on the effectiveness test.

Prerequisites

You configure the offset method in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness Test* > *Set Up Offset Method*.

Features

The method used to calculate the effectiveness key figure depends on the control parameters you have defined in Customizing. For each effectiveness test rule [page 1042], you can choose a noise threshold value (NT) parameter and a transition speed (S_T) parameter.

Standardized method

If you do not define a noise threshold value, the system uses the standard method to calculate the hedge ratio HR:

$$\text{(A)} \qquad HR(t) = \frac{HA_{HIt}(t)}{HA_{HIn}(t)}$$

where *t* is the key date, *HA* is the hedge adjustment [page 1073], *HIt* is the hedged item, and *HIn* is the hedging instrument in a hedging relationship (depending on the hedging relationship, these can be complete transactions or portions of transactions).

Enhanced Method

In contrast to the standard method, you enter a noise threshold value when you use the enhanced method. This parameter defines the point at which fluctuations in value are to be considered relevant for the analysis of the hedging relationships. Besides the noise threshold value, you can use the transition speed parameter to influence selectiveness in the effectiveness measurement:

- Calculation with noise threshold value and without transition speed

$$\text{(B)} \qquad HR(t) = \frac{HA_{HIt}(t) \pm NT_A}{HA_{HIn}(t) \pm NT_A}$$

where *NT A* is the noise threshold value as an absolute amount in the reference currency of the hedging relationship. The noise threshold value *NT A* is added or subtracted in the numerator so that the sign (+/-) of the expression does not change; the respectively opposite sign is used in the denominator. If, for example, the change in the value of the hedged items *ΔHIt* is positive, the system adds the noise threshold value *NT A* in the numerator, and subtracts it from the denominator.

If the value changes are large, the formula is the same as that used in the standard method (*ΔHIt* or *ΔHIn* dominates). In the case of minor value changes, the noise threshold value (*HR≈NT A/NT A=-1*) dominates in the numerator and denominator. When *NT A=0*, formula (B) is the same as the standard method formula (A).

> **Note**
>
> In Customizing, you can enter the noise threshold value as a percentage *NT P* of the total of the nominal values *Nom* of the hedged items, or as an absolute amount *NT A* in the reference currency. If you enter the threshold value as a percentage, the system calculates the corresponding absolute amount on the basis of this percentage before applying the above formula: $NT_A = NT_P * Nom$
>
> where *Nom* = nominal value of a hedging relationship; *Nom* is defined as the total of the nominal values of the hedged items in the hedging relationship, and the current remaining debt on the key date is used as the nominal value.

- Calculation with noise threshold value and transition speed

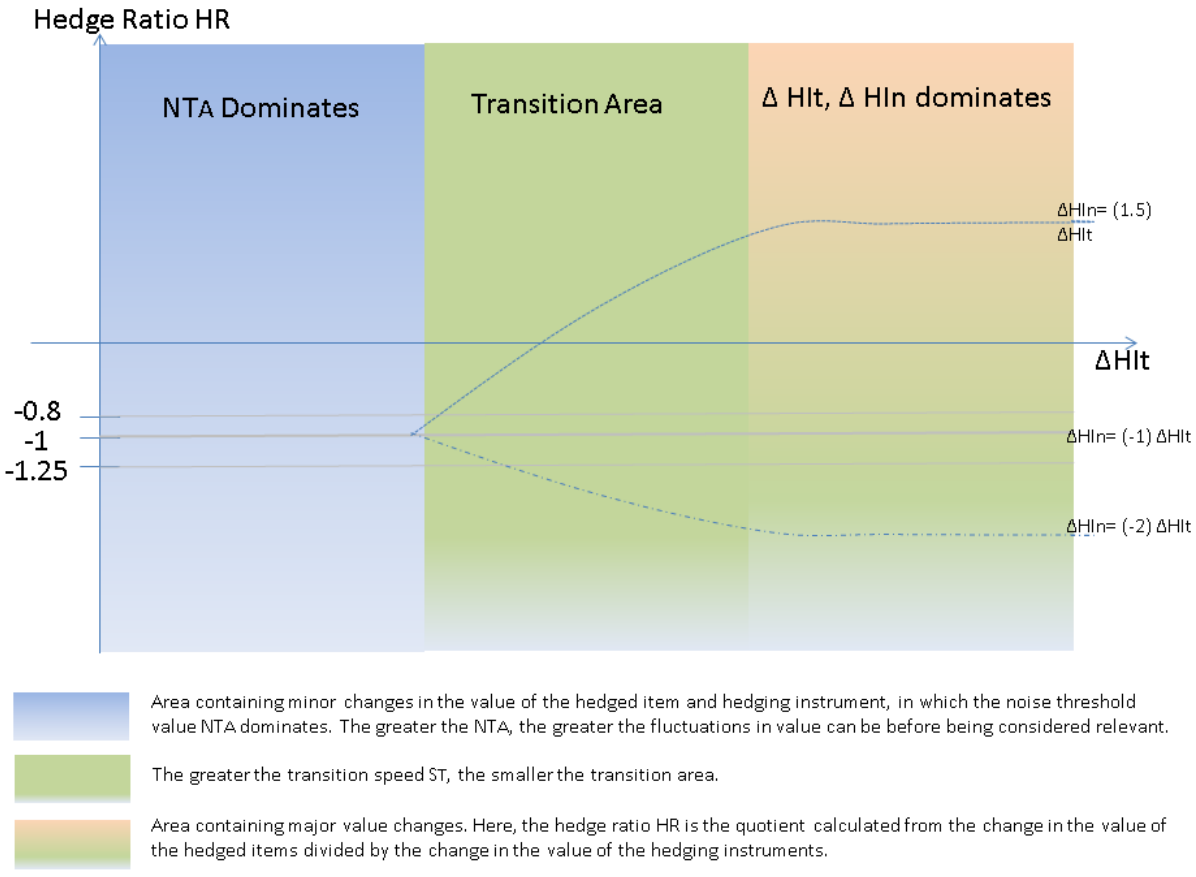
$$\text{(C)} \qquad HR(t) = \frac{HA_{HIt}(t)\left(\frac{\Delta MP(t)}{NT_A}\right)^{S_T} \pm NT_A}{HA_{HIn}(t)\left(\frac{\Delta MP(t)}{NT_A}\right)^{S_T} \pm NT_A} \text{ with } S_T > -1$$

where *S T* is the transition speed. The noise threshold value *NT A* is added and subtracted as above. *ΔMP* is defined as follows:

$$\Delta MP\ (t) = \sqrt{HA_{HIt}\ \ (t) + HA_{_{HIn}(t)}{}^{2}}$$

ΔMP is a measurement of the extent of the value changes in the hedged items and hedging instruments. If *ΔMP* is less than *NT A*, the hedge ratio is 1. If *ΔMP* is greater than *NT A*, the equation (C) is the same as the standard method (A). The calculation rule (C) conforms to the standard method (A) and the enhanced method (B) without transition speed, since, when *ST =0* or *S T =0* and *NT A =0*, it is the same as formulas (B) and (A).

You use the transition speed to influence the transition area between the area in which the noise threshold value dominates and that in which the noise threshold value is irrelevant. The greater the transition speed, the smaller the transition area. The following graphic shows the effect that the noise threshold value and transition speed parameters have on the hedge ratio in calculation rule (C):



Analysis of the Hedging Relationship

1. Setting of the retrospective effectiveness indicator
   The system compares the calculated hedge ratio with the limits specified in Customizing, and sets the retrospective effectiveness indicator to *Effective*, *Temporarily Ineffective* or *Ineffective*.

   > **Note**
   >
   > In reporting [page 1055] for the fair value effectiveness test, the system displays the changes in value of the hedged items *HIt* and of the hedging instrument *HIn*. This enables you to see how the system calculated these values.

2. Check of the number of tests with the result *temporarily ineffective*
   In Customizing, you specify the number of times in succession that a hedging relationship is allowed to be classified as *temporarily ineffective*. If this maximum number is exceeded for a given hedging relationship, the system classifies that hedging relationship as *Ineffective*.
3. Setting of the start date of ineffectiveness

Error Handling

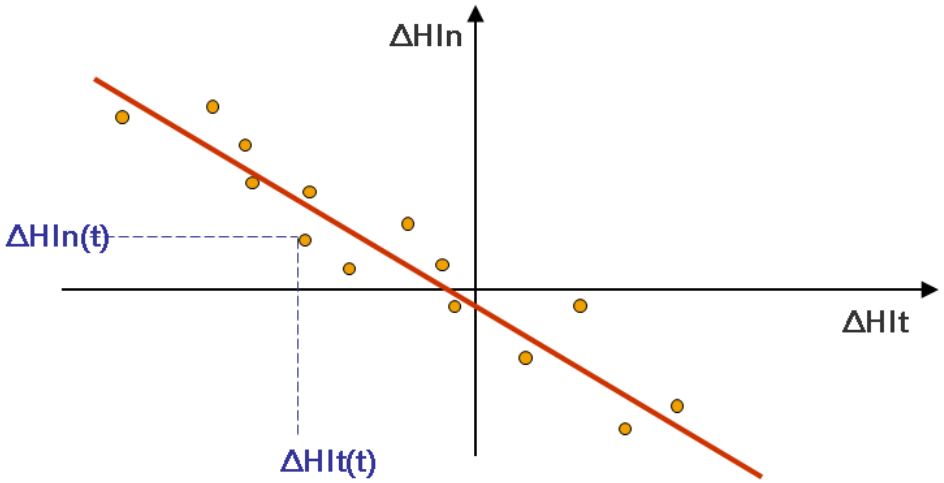
- If the changes in the value *HIt* of the hedging instruments is equal to zero or approximately zero, the system issues a warning.
- When calculating the changes in the value of hedging instruments, the system calls the Fair Value Server. If an error occurs in one of the required functions, the system sets the effectiveness indicator to *Incorrect* and writes an error message in the application log.

4.6.2.6.4 Regression Analyses

Use

When testing the effectiveness of hedging relationships, it can sometimes be useful to analyze the changes in the values of hedged items and hedging instruments over their entire terms, rather than at a particular point in time. To be able to do this, the system contains two types of regression analysis for the effectiveness test: retrospective regression analysis and prospective regression analysis.

The following figure shows an example of regression analysis. The system has measured the changes in the value of the hedged item and hedging instrument in a particular hedging relationship at different points in time, and placed a regression line through these points.



	Data point for time t
	Regression line
ΔHIt	Changes in the value of the hedged items in the hedging relationship
ΔHIn	Changes in the value of the hedging instruments in the hedging relationship

Prerequisites

- You define the general mathematical properties for the regression analysis in the regression type [page 1089]. For example, in the regression type you enter the settings for the statistical tests for the interpolation of the regression lines.
- You also enter separate Customizing settings for retrospective regression analysis and prospective regression analysis.

Features

In both types of regression analysis, the system measures the changes in the values of the hedged items and hedging instruments over their terms. The system contains the following types of regression analysis:

- Retrospective regression analysis [page 1083]
  The system uses historical value changes, and market data scenarios and transaction data for the key date as required.
- Prospective regression analysis [page 1088]

Here the system takes the transaction data on the key date, and measures the change in the value of the hedged item and hedging instrument for future points in time. Market data scenarios are used as the basis for these measurements.

As is the case in other methods for testing the effectiveness of hedges, the system takes the hedge adjustment [page 1073] as the key figure for the change in the value of the hedged item and the hedging instrument. Once it has generated the sample, the system places a regression line through the data points. The system uses the slope and section of the axis of the regression line as a measure of the effectiveness of the hedging relationship.

Properties of the Sample

In Customizing, you define the distribution of the data points or the regression analyses. In retrospective regression analysis, you specify the time series; in prospective regression analysis you specify the maturity band and the scenario set. In addition, in the Customizing for retrospective regression analysis and prospective regression analysis you can enter values for the following parameters:

- Time period after the hedging relationship has been designated in which the system is not to create any data points for regression analysis
- Time period before the hedging relationship expires in which the system is not to create any data points (for prospective regression analysis only)
- Minimum number of data points in a sample
  If this number is not reached, then the effectiveness of the hedging relationship is not tested. You can also specify a minimum number of data points below which the system issues a warning.

In retrospective regression analysis, you can define how far back into the past, starting from the key date, that the system looks when it creates the sample.

To be able to understand the results of regression analysis if you look at them again at a later point in time, you can define in Customizing that the system it to save all the data points that it has actually used in the effectiveness test.

Measurement of the Change in the Value of Hedged Items and Hedging Instruments

The system measures the change in the value of the hedged item and hedging instrument in a hedging relationship by using the hedge adjustment [page 1073] key figure.

In the default setting, the system calculates the hedge adjustment for the hedged item and hedging instrument from the individual data points. These measurements then form the basis of the regression analysis. If this method is used, it can be the case that the analysis is influenced by autocorrelation. In this case, the results of the measurements are distorted because the hedge adjustments for the individual data points are not independent of one another. To avoid the effects of autocorrelation, you can use the delta mode.

In the **delta mode**, at each data point the system calculates the difference between the hedge adjustment at this time point, and the hedge adjustment at the time point of the previous data point. (No value is calculated for the first data point, since no data point precedes it.)

Selection of Statistical Tests

You can use the following statistical tests to interpret the regression lines:

- F-test [page 1090]
- t-test [page 1093]
- Simple interval test [page 1095]

4.6.2.6.4.1 Retrospective Regression Analysis

Use

Retrospective regression analysis is used to test the effectiveness of fair value hedging relationships on the basis of historical data; if appropriate, the function also uses scenario-based data points. As in the offset method, the changes in the value of the hedged item and the hedging instrument are compared. However, the system does not use just one data point. It takes multiple data points, and places a regression line through the historical values. The slope and section of the axis of the regression lines show whether the hedging relationship can be classified as effective.

Prerequisites

You have made the following settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for the Effectiveness Test*:

- You have defined a regression type in the Customizing activity *Edit Regression Type*.
- You have defined at least one time series in the Customizing activity *Edit Time Series for Retrospective Regression Analysis*.
- You have entered the properties for retrospective regression analysis in the Customizing activity *Set Up Retrospective Regression Analysis*.

Features

In retrospective regression analysis, the system proceeds as follows:

1. In a separate process, the system generates the time series that is the basis of retrospective regression analysis.

   > **Note**
   >
   > You schedule the job for generating the time series as a process in Schedule Manager. For more information, see Generation of Time Series for Retrospective Regression Analysis [page 1084]. Note that the generation of time series has to be completed before you can call the function for the effectiveness test.

2. The system reads the time series containing the changes in the values of the hedged item and hedging instrument.
   In the regression analysis, it then takes only those data points that fall after the date on which the hedging relationship that is to be analyzed was designated (or on the designation date). The system also ignores any data points that are after the time period defined in Customizing that is after the designation date.
   In the Customizing for retrospective regression analysis, if you also defined a maximum time period for the data points, then the system considers only those data points that fall in this time bucket up to the current key date. Therefore, the time frame whose data points are taken into account changes with the key date.
3. If the time series contains too few data points to make the effectiveness test meaningful, you can use scenario-based data points [page 1086] in addition.

4. The system carries out the effectiveness test defined in the regression type, which includes the following tests:
   - F-test [page 1090]
   - t-test [page 1093]
   - Simple interval test [page 1095]
5. The system sets the indicator for retrospective effectiveness to *Effective* or *Ineffective*, as per the result of the regression analysis.

4.6.2.6.4.1.1 Generation of Time Series for Retrospective Regression Analysis

Use

In this function, the system generates time series for individual hedging relationships as part of the effectiveness test, and hence as preparation for retrospective regression analysis. You define the period pattern for the time series in Customizing for the fair value effectiveness test. The period pattern is the same for all hedging relationships that have the same combination of effectiveness test environment and effectiveness test rule.

When the time series for retrospective regression analysis are generated, the system adjusts the period pattern for each hedging relationship, and calculates the change in the value of the hedged item and hedging instrument for each time point. In the effectiveness test itself, the system simply selects the time series already generated for the hedging relationships, and adjusts these as per the settings you made in Customizing for retrospective regression analysis.

In addition to the actual generation of the time series, there is a separate function that you can use to modify existing time series. You may need to do this, if, for example, you later realize that for particular time points it was not possible to enter market data correctly. Once you have adjusted the market data, you can correct the time series for these time points without having to regenerate the entire time series.

> **Note**
>
> Note that in addition to the data points of a time series, you can also use scenario-based data points [page 1086] if the time series contains too few data points for the effectiveness test to be meaningful.

Prerequisites

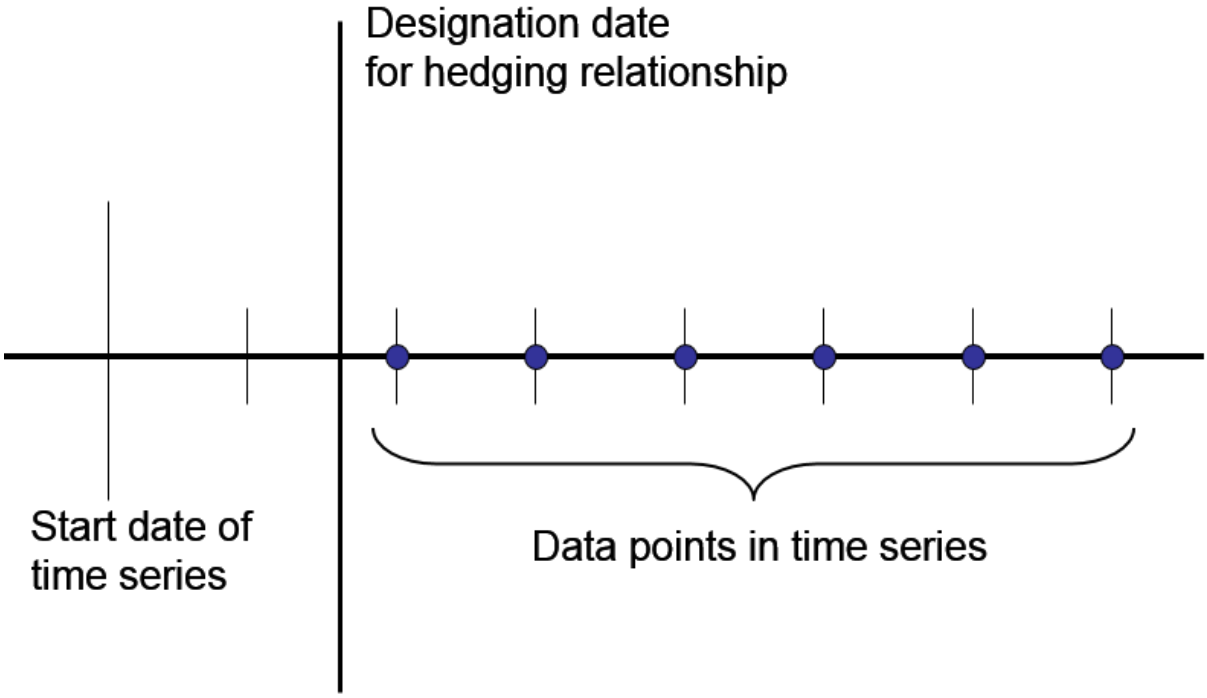
- You need to have already created the time series required, and assigned it to retrospective regression analysis. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness Test* > *Edit Time Series for Retrospective Regression Analysis* or *Set Up Prospective Regression Analysis*.
- You have created a run type for the effectiveness test. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Processes* > *Run Type* > *Edit Run Type*.

Features

Generation of a Time Series

The system generates time series as follows:

1. It finds the first data point.
   In Customizing, you enter a start date for each time series, and specify the distance between the individual time points. The time points are always equidistant. Starting from the start date, the system reads the time points in the series, and finds the first time point, which is either the time point at which the hedging relationship was designated, or after this time point. The start date specified in Customizing should usually be before the date on which the hedging relationships were designated so that the time series of a hedging relationship starts on the designation date or shortly after this.
   The following diagram shows that the data points in the time series start with the designation time point, even if the time series itself starts before the designation time point:



2. The system extends the time series as far as possible.
   It adds more and more time points to the time series for the hedging relationship, and calculates the changes in the value of the hedged item and hedging instrument for all the time points. The system stops generating the time series when it can no longer calculate any changes in value. This is usually because the market data is missing.

Modification of a Time Series

In order to correct a time series, you can repeat the process described above for part of the time series. In order to do so, the system must have already completed the generation of the time series that you want to change. Note the following:

- The system overwrites existing measurements in the time interval specified with the new measurements.
- Unlike in the generation of the time series, the system does not stop the processing when the calculation of the changes in the value of the hedged item and hedging instrument fails. When this happens, the existing measurements in the series are retained.

Activities

You schedule the generation or modification of the time series as separate runs in the Schedule Manager [page 437]. You need to schedule the runs for generating the time series so that they take place before the fair value effectiveness test that carries out retrospective regression analysis. You use the following programs:

Function	Program
Generation of a Time Series	/BA1/R8_FVH_BTS_MAIN_RUN
Modification of a Time Series	/BA1/R8_FVH_BTS_MODIFY

> **Note**
>
> You can call the runs for generating and modifying time series directly in the area menu. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Run Administration* > *Create and Schedule* > *Create Time Series for Retrospective Regression Analysis* or *Modify Time Series for Retrospective Regression Analysis*.
>
> Note that the runs that you start from the area menu use work processes. In certain circumstances, this can lead to a shortage of system resources, and delay the start of the runs that you have scheduled in Schedule Manager.

Once the programs have been called, the system generates the time series, or modifies an existing time series. This happens for all hedging relationships that are in the worklist for the effectiveness test run type. For the date values in the time series, the system creates a complete series of measurements containing the changes in the values of the hedged item and hedging instrument.

> **Caution**
>
> Note that in the actual effectiveness test, the system does not usually use all the data points in the time series. This is because in Customizing for retrospective regression analysis you can specify a time period that is after the time point at which the hedging relationship was designated for which the system is to ignore all measurements in retrospective regression analysis.

Once the runs for generating the time series have been called, you can use the following reporting functions:

- Display of Time Series for Retrospective Regression Analysis [page 1068]
- Displaying the Runs Used to Generate Time Series [page 1070]
  From within run administration, you call the application log and the detail log [page 652], which document how the changes in the values of the hedged items and hedging instruments were calculated.

4.6.2.6.4.1.2 Use of Data Points from Scenarios

Use

In this function, the system adds scenario data points to a time series for retrospective regression analysis.

You use this function if you want to carry out a retrospective effectiveness test just after you have designated a hedging relationship, for example. In such a case, the time series for the retrospective regression analysis would contain too few data points for the effectiveness test to provide meaningful results. By using scenario data points in addition to the time series you can obtain useful results from the retrospective effectiveness test.

Integration

The system generates scenario data points in the effectiveness test method Retrospective Regression Analysis [page 1083]. Unlike for the generation of time series [page 1084] for the retrospective regression analysis, you do not need to schedule a separate job when you use scenario time points.

Prerequisites

In Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness Test* > *Set Up Retrospective Regression Analysis*, you have set the *Add Scenario Data Points* checkbox, and entered the following:

- Scenario set
  Contains the market data scenarios for the scenario data points
- Maturity band (optional field)
  You normally do not need to use a maturity band. However, you have to create as many market data scenarios as the number of scenario data points you require.
- Designation date/evaluation date: Here you specify whether the system is to select the transaction data for the designation date of the hedging relationship or for the evaluation date.
- Minimum number of data points that are to be used

Features

The system uses scenario data points only if the time series for the retrospective regression analysis contains fewer data points than the minimum number specified in Customizing. If the minimum number of data points is not reached, the system adds the relevant number of scenario data points so that this minimum number is reached.

The system does this as follows:

1. It selects the transaction data
   The system selects the data either on the basis of the designation date of the hedging relationship, or for the evaluation date, depending on the setting made in Customizing.
2. It creates a list of scenario data points
   The system then generates a scenario data point for each combination of maturity band date and market data scenario. For relative maturity bands, the maturity starts on the date for which the transactions are selected. If you have not stored a maturity band, when the system generates the scenario data points it uses the date for which the transactions are selected as the only maturity band date.

It sorts the pairs in reverse order first by the scenario date, and then by the maturity band date. The system takes the scenario date from the market data scenarios, and the horizon date from the corresponding maturity band date.

3. It uses the scenario data points
   The system uses only as many scenario data points from the list as are required to meet the minimum number of data points.
   The further back in time the designation date is, the fewer scenario data points are used. When the system uses scenario data points, it first uses those that are at the furthest future point in time.

4.6.2.6.4.2 Prospective Regression Analysis

Use

Prospective regression analysis uses current transaction data and market data scenarios for the future development of the market data to test the effectiveness of hedging relationships. As is the case in the offset method, in prospective regression analysis the changes in the value of the hedged items and hedging instruments are compared, though here the system uses multiple data points and connects the values measured with a regression line. The slope and section of the axis of the regression line show whether the hedging relationship can be classified as effective.

Prerequisites

You need to have already made the following settings in Customizing for *Bank Analyzer* :

- Under *Financial Services* > *Foundation* > *Market Data* > *Market Data Scenarios* , you have created market data scenarios, and grouped them together into at least one scenario set.
- Under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Create Maturity Band* you have created at least one maturity band.
- Under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness Test* > *Edit Regression Type* , you have defined at least one regression type [page 1089] .
- Under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness Test* > *Set Up Progressive Regression Analysis* , you have defined the properties of the prospective regression analysis.

Features

In prospective regression analysis, the system proceeds as follows:

1. The system generates the sample.
   It does this by selecting the hedging relationships that are to be analyzed for the key date, and measuring the changes in the value of the hedged items and hedging instruments on the dates in the maturity band, which

are used as the horizon date or valuation date here. The system uses the hedge adjustment [page 1073] as the key figure for the change in value. It calculates the change in value for each combination of market data scenario and horizon date that you entered in the fair value effectiveness test when you specified the scenario set or maturity band.

2. The system carries out the effectiveness test defined in the regression type:
   - F-test [page 1090]
   - T-test [page 1093]
   - Simple interval test [page 1095]
3. The system sets the indicator for prospective effectiveness to effective or ineffective, as per the result of the regression analysis.

4.6.2.6.4.3 Regression Type

Definition

Customizing term that groups together the general mathematical parameters for a regression analysis.

Use

You use the regression type to specify how the system interprets the regression line in regression analysis. To do this, you can use the statistical tests, such as the f-test or the t-test. You specify the required criterion for the interpretation of the regression type, as well as the necessary parameters for this criterion in the regression type.

You define the regression types for both retrospective and prospective regression analysis. You can, therefore, use the same regression types for retrospective regression analysis and prospective regression analysis.

Structure

Parameters for the Regression Type

Criterion for Determining Effectiveness	Parameters
F-test [page 1090]	• Significance level • Minimum value to be tolerated for the coefficient of determination
T-test for slope of regression line [page 1093]	• Significance level • Minimum value to be tolerated for the coefficient of determination • Intervals for slope of regression line

T-test for slope of regression line and axis distance [page 1093]	• Significance level • Minimum value to be tolerated for the coefficient of determination • Intervals for slope of regression line • Intervals for axis distance of regression line
Simple interval test [page 1095] (ad hoc test)	• Intervals for slope of regression line • Intervals for axis distance of regression line

Example

If you want to use only an f-test in regression analysis, you need only one regression type. You define the regression type (for example, using the name RG1) in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness Test* > *Edit Regression Type*. Choose the f-test as the criterion for determining the effectiveness, and enter a significance level (for example, 0.05), and a (tolerated) minimum for the coefficient of determination (for example, 0.90).

Once you have done this, you can then assign your regression type RG1 to the retrospective and prospective regression analyses. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness Test* > *Set Up Retrospective Regression Analysis* > *Set Up Prospective Regression Analysis*.

4.6.2.6.4.3.1 F-test

Use

In the t-test, the system checks whether the regression line that was calculated can be regarded as a regression line for a perfect hedging relationship. You can use the F-test in retrospective regression analysis and prospective regression analysis in the fair value effectiveness test.

F-tests are usually used to compare two or more normally-distributed random variables. In this comparison, hypotheses about the variance of the random variables are tested. In the F-test in the fair value effectiveness test, the system compares the variance of the data points in terms of the regression line to their variance in terms of the regression line for a perfect hedging relationship.

Prerequisites

In Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness* > *Edit Regression Type*, you have created a

regression type [page 1089] and set it as a criterion for determining effectiveness. You also need to have entered values for the following parameters:

- Significance level *a*
- Minimum tolerance value for coefficient of determination R^2

You need to have entered the regression type for retrospective regression analysis [page 1083] or prospective regression analysis [page 1088].

Features

Mathematical Basis

The system tests the hypothesis that the regression line is in the section of the axis $\beta_1 = 0$ and has the slope $\beta_2 = -1$. In order to test the hypothesis, the system compares the following models:

$$(A) \quad y_t = \beta_1 + \beta_2 \cdot x_t + u_t$$

$$(B) \quad y_t = -x_t + u_t$$

where *t* is the time points at which measurements x_t and y_t were taken, and u_t an error term. Model (A) describes the regression line that was calculated, and model (B) the hypothesis that $\beta_1 = 0$ and $\beta_2 = -1$. If the hypothesis is true, then models (A) and (B) are either identical, or only slightly different.

> **Note**
>
> In the standard setting, *x* represents the changes in the value of the hedged items, and *y* the change in the value of the hedging instruments in the hedging relationship. In Customizing for the effectiveness test environment, you have the option of swapping the two coordinates. For this reason, a more general notation is used here.

The residuals of the models are used to measure the difference between Model (A) and Model (B):

$$e_t = y_t - \beta_1' - \beta_2' \cdot x_t$$

$$e_{0t} = y_t + x_t$$

where β'_1 and β'_2 are the estimates for the axis section and the slope of the calculated regression lines. The estimates are defined as follows:

$$\beta_1' = \bar{y} - \beta_2' \cdot \bar{x}$$

$$\beta_2' = \frac{\sum_{t=1}^{T} (x_t - \bar{x}) \cdot (y_t - \bar{y})}{\sum_{t=1}^{T} (x_t - \bar{x})^2}$$

where T is the number of data points and

$$\bar{x} = \sum_{t=1}^{T} x_t, \quad \bar{y} = \sum_{t=1}^{T} y_t$$

is the arithmetical mean of the values measured.

The main value for the effectiveness test is the difference between the residuals:

$$\sum_{t=1}^{T} e_t^2 - \sum_{t=1}^{T} e_{0t}^2$$

where *T* is the number of data points. If the difference between the sums of the residuals is large, then the hypothesis that $\beta_1 = 0$ and $\beta_2 = -1$ has to be rejected.

The following random variable *X* is used for the actual test:

$$X = \frac{\sum_{t=1}^{T} e_{0t}^2 - \sum_{t=1}^{T} e_t^2}{\sum_{t=1}^{T} e_t^2} \cdot \frac{T-2}{2}$$

In the F-test, the zero hypothesis is tested to find out whether the variances of the numerator and denominator of *X* are the same.

If the variances of the numerator and denominator of *X* are equal, then random variable *X* is F-distributed with *2* and *T-2* degrees of freedom. (The degree of freedom for Model (A) is *T-2*; the degree of freedom of the hypothesis, or Model (B), is *T*.

The Actual F-Test

In the actual effectiveness test for the hedging relationship, the system determines whether random variable X is smaller than the corresponding value of the F-quantile.

$$(I) \quad X < F_{1-\alpha;2,T-2}$$

where *α* is the significance level of the F-test. If condition (I) is met, then the system considers the hypothesis as confirmed, and classifies the hedging relationship as effective. Otherwise, the system classifies the hedging relationship as ineffective.

The F-test for retrospective regression analysis and prospective regression analysis is based on the assumption that the slope and axis section of the regression line is distributed normally, and the sums of the residuals, which as measurements in the F-test take on the role of the standard normal distribution, are distributed independently of each other and x^2 distributed.

Checking the Coefficient of Determination

In the final step, the system checks whether the coefficient of determination of the regression analysis does not exceed the minimum specified in the regression type.

$$(II) \quad R^2 \leq \frac{\beta_2'^2 \cdot \sum_{t=1}^{T} (x_t - \bar{x})^2}{\sum_{t=1}^{T} (y_t - \bar{y})^2}$$

If condition (II) is **not** met, then the system classifies the hedging relationship analyzed as ineffective.

4.6.2.6.4.3.2 T-Tests

Use

In the t-test, the system checks whether the regression line that was calculated can be regarded as a regression line for a perfect hedging relationship. T-tests are normally used to test hypotheses about the arithmetic mean of the measurements when the variance is unknown. In the t-test carried out as part of the fair value hedge effectiveness test, the regression line is used as the mean value.

You can use the following values in the test:

- Test the slope of the regression lines only
- Test the slope and the axis distance of the regression lines

You can use the t-test in retrospective regression analysis and prospective regression analysis in the fair value effectiveness test.

Prerequisites

In Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for EffectivenessTest* > *Edit Regression Type* , you have created a regression type [page 1089] and entered the t-test as a criterion for determining the effectiveness of hedging relationships. You also need to have entered values for the following parameters:

- Significance level *α*
- Minimum tolerance value for coefficient of determination R^{2} ^{2}
- You can specify the additional interval limits $g_2 \leq h_2$ for the slope of the regression line, and, if required, the interval limits $g_1 \leq h_1$ for the axis distance of the regression lines.

You need to have entered the regression type for retrospective regression analysis [page 1083] or prospective regression analysis [page 1088] .

Features

Mathematical Basis

The system tests the hypothesis that the regression line is in the section of the axis $\beta_1 = 0$ and has the slope $\beta_2 = -1$. In order to test the hypothesis, the system compares the following models:

$$(A) \quad y_t = \beta_1 + \beta_2 \cdot x_t + u_t$$

$$(B) \quad y_t = -x_t + u_t$$

where *t* is the time points at which measurements x_t and y_t were taken, and u_t an error term. Model (A) describes the regression line that was calculated, and model (B) the underlying hypothesis. If the regression line has a slope of -1 and an axis distance of 0, then Model (A) matches Model (B).

> **Note**
>
> In the standard setting, *x* represents the changes in the value of the hedged items, and *y* the change in the value of the hedging instruments in the hedging relationship. In Customizing for the effectiveness test environment, you have the option of swapping the two coordinates. For this reason, a more general notation is used here.

The residuals of the models are used to measure the difference between Model (A) and Model (B):

$$e_t = y_t - \beta_1' - \beta_2' \cdot x_t$$

$$e_{0t} = y_t + x_t$$

where β_1 and β_2 are the estimates for the axis section and the slope of the calculated regression lines. The estimates are defined as follows:

$$\beta_1' = \bar{y} - \beta_2' \cdot \bar{x}$$

$$\beta_2' = \frac{\sum_{t=1}^{T} (x_t - \bar{x}) \cdot (y_t - \bar{y})}{\sum_{t=1}^{T} (x_t - \bar{x})^2}$$

where T is the number of data points and

$$\bar{x} = \sum_{t=1}^{T} x_t, \quad \bar{y} = \sum_{t=1}^{T} y_t$$

is the arithmetical mean of the values measured.

In the t-test, the variance of the estimates β'_1 and β'_2 must also be calculated.

$$Var(\beta_1') = \frac{\sigma^2 \cdot \sum_{t=1}^{T} x_t^2}{T \cdot \sum_{t=1}^{T} (x_t - \bar{x})^2}$$

$$Var(\beta_2') = \frac{\sigma^2}{\sum_{t=1}^{T} (x_t - \bar{x})^2}$$

where:

$$\sigma^2 = \frac{1}{T-2} \cdot \sum_{t=1}^{T} e_t^2$$

Actual t-Test

In the actual test for the effectiveness of the hedging relationship, the system compares whether the estimates β' $_1$ and β' $_2$ for the axis section and slope are in the *1-α/2* quantile of a t-distribution with *T-2* degrees of freedom.

$$(I) \quad g_i - t_{1-\alpha/2;T-2} \cdot \sqrt{Var(\beta_i')} \le \beta_i \le h_i + t_{1-\alpha/2;T-2} \cdot \sqrt{Var(\beta_i')}, \quad i \in \{1;2\}$$

where *α* is the significance level and *T* the number of measurements, and g $_i$ and h $_i$ the additional interval limits stored in Customizing for the regression type. In a t-test for the slope of the regression lines, only the interval limits g $_2$ and h $_2$ are used for the slope. In a t-test for the slope and axis distance, the interval limits g $_1$ and h $_1$ are also taken into account for the axis distance.

Checking the Coefficient of Determination

In the final step, the system checks whether the coefficient of determination of the regression analysis does not exceed the minimum specified in the regression type.

$$(II) \quad R^2 \le \frac{\beta_2'^2 \cdot \sum_{t=1}^{T} (x_t - \bar{x})^2}{\sum_{t=1}^{T} (y_t - \bar{y})^2}$$

If condition (II) is **not** met, then the system classifies the hedging relationship analyzed as ineffective.

4.6.2.6.4.3.3 Simple Interval Test

Use

In this test, the system checks whether the slope and the section of the axis of the regression line fall within a defined interval. Unlike the statistical tests, such as the F-test or t-test, distribution of probability is not taken into account here.

You set up the simple interval test in the retrospective regression analysis and prospective regression analysis.

Prerequisites

In Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for EffectivenessTest* > *Edit Regression Type* , you need to have created a regression type [page 1089] and entered it in the simple interval test as a criterion for determining the effectiveness of hedging relationships. You also need to have stored values for the following parameters:

- Intervals g $_2$ ≤h $_2$ for the slope of the regression lines
- Intervals g $_1$ ≤h $_1$ for the axis distance of the regression lines

You need to have entered the regression type for retrospective regression analysis [page 1083] or prospective regression analysis [page 1088] .

Features

The system calculates the axis section β'_1 and slope ß'$_2$ for the regression lines as follows:

$$\beta_1' = \bar{y} - \beta_2' \cdot \bar{x}$$

$$X = \frac{\sum_{t=1}^{T} e_{0t}^2 - \sum_{t=1}^{T} e_t^2}{\sum_{t=1}^{T} e_t^2} \cdot \frac{T-2}{2}$$

where T is the number of data points and

$$\bar{x} = \sum_{t=1}^{T} x_t, \quad \bar{y} = \sum_{t=1}^{T} y_t$$

is the arithmetical mean of the values measured.

The system checks whether the regression line has the properties of the regression type:

1. The slope of the regression lines is within the interval specified:

   $(I) \quad g_2 \le \beta_2' \le h_2$

2. The axis distance of the regression lines is within the interval specified.

   $(I) \quad g_2 \le \beta_2' \le h_2$

If both conditions are met, then the system classifies the hedging relationship as effective. Otherwise, the system classifies the hedging relationship as ineffective.

Example

You have created a regression type, and set up the test for whether the slope and axis section fall within a given interval (simple interval test) as a criterion for determining effectiveness. You entered the values -0.8 and +1.25 as the limits of the interval for the slope of the regression lines, and the values -0.05 and +0.05 as the limits of the interval for the axis section of the regression lines. You assigned the regression type for retrospective regression analysis and prospective regression analysis.

If the regression line calculated has a slope of -0.95, for example, and the axis distance is 0.02, then the system classifies the hedging relationship as effective, since the values for the slope and the axis distance of the regression lines fall within the interval you specified in the regression type.

4.6.2.6.5 Similar Asset Test

Use

You use this function to test whether the hedged items in a micro fair value hedging relationship react to the hedged risks in the same way as the overall portfolio of hedged items. If the assets in the fair value hedging relationship are similar, then the relationship can be tested in the fair value effectiveness test.

Integration

The similar asset test is a separate test method that is part of the fair value effectiveness test. If required, you can add the similar asset test to a custom test schema.

SAP provides the test schemas `SAT` and `SAT_PROSP` that contain the similar asset test.

Prerequisites

You have created at least one market scenario, and stored it in a scenario set in Customizing under *Financial Services* > *Foundation* > *Market Data* > *Market Data Scenarios*.

You have made the following settings in Customizing for *Bank Analyzer*:

You have created at least one maturity band under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Create Maturity Band* (see Edit Maturity Band [page 1152]).

You have defined at least one regression type [page 1089] under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness Test* > *Edit Regression Type*.

> **Note**
>
> Note that the slope of the ideal regression line in the similar asset test is +1. Therefore, in the regression type you must store positive values for the limits of the intervals of the slope. In the prospective and retrospective regression analyses, the ideal regression line has a negative slope. This means that you need two regression types if you want to call the similar asset test and the prospective regression analysis in the fair value effectiveness test.

You have stored all the parameters that are required for the similar asset test under *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness Test* > *Set Up Similar Asset Test*.

> **Note**
>
> In the fair value effectiveness test, if you want to use the similar asset test and the prospective regression analysis, you should use the same maturity band and scenario set for both test methods.

Features

The system carries out a prospective regression analysis and in doing so generates data points for each market data scenario and for each horizon date or key date that you specify.

The system generates a sample.

For the key date, the system selects the hedging relationships that are to be analyzed, and measures the change in the value of each hedged item. It calculates the change in value for each combination of market data scenario and horizon date that you entered in the effectiveness test by specifying the scenario set or maturity band. The system uses the hedge adjustment [page 1073] as the key figure for the change in value.

The system carries out the effectiveness test defined in the regression type:

F-test [page 1090]

t-test [page 1093]

Simple interval test [page 1095]

Unlike the prospective regression analysis and retrospective regression analysis, the similar asset test does not compare the changes in the value of the hedged items with those of the hedging instruments. Instead, it compares the change in the value of a hedge item with the change in the value of the portfolio of hedged items. So that it can compare these values, the system uses the *remaining debt* or *hedge fair value* key figure to scale the values of the single transactions to portfolio level. In the Customizing settings, you define which of the key figures the system is to use to scale the values.

Weighting by the remaining debt

$$x_t = \sum_{g \in GG} HA_g(t), \quad y_t = \left| \frac{\sum_{g \in GG} Nom_g(t)}{Nom_{g'}(t)} \right| \cdot HA_{g'}(t)$$

where *HIt* is the portfolio of hedged items in the hedging relationship, *HA* is the *Hedge Adjustment* key figure, and *Nom g* is the nominal value of the hedging relationship; the current remaining debt is used for the key date.

Weighting by the hedge fair value

$$x_t = \sum_{g \in GG} HA_g(t), \quad y_t = \left| \frac{\sum_{g \in GG} Nom_g(t)}{Nom_{g'}(t)} \right| \cdot HA_{g'}(t)$$

where *HFV* is the *hedge fair value* key figure

Note that the *remaining debt* and *hedge fair value* key figures depend on the horizon and on the market data scenarios. If the remaining debt or the hedge fair value is zero, the hedged item in question is not similar to the portfolio of hedged items.

According to the result of the similar asset test, the system sets the effectiveness indicator of the similar asset test or the final effectiveness indicator.

If the deviations are larger than the maximum permitted deviation, the system sets the final effectiveness indicator to *ineffective*, and locks this setting. The hedging relationships is thus still assessed as ineffective if it is analyzed again by subsequent test methods.

> **Note**
>
> You can display the results of the similar asset test in the reporting functions for the fair value effectiveness test. The *Similar Asset Test* tab page contains the test results for the individual transactions in the portfolio of hedged items. In this tab page, for each transaction you can display the data points that the system used in regression analysis.

If the similar asset test found the hedging relationship to be “ineffective”, the system displays this in reporting [page 1055] on the *Effectiveness Indicator* tab page. The *Prospective Effectiveness Method* column contains the name of the test method *Similar Asset Test* (`SIM_ASSET`), and the *Effectiveness Test Method* column contains the *Final Set of Rules* (`RULEFRMWRK`).

4.6.2.6.6 Custom Effectiveness Test Methods (BAdI)

Definition

Effectiveness test methods that are created by customers as a Business Add-In (BAdI).

Use

We provide methods for measuring the effectiveness of hedging relationships. If you require additional or different effectiveness test methods for your own purposes, then you can create custom effectiveness test methods, and integrate them into the framework of the fair value effectiveness test.

Structure

From a technical point of view, custom effectiveness test methods are implemented in the same way as those predefined in the system. Both types are implementations of the *test method* (/BA1/R8_TEST_METHOD) BAdI. The interface of the BAdI contains two methods. The system first calls a method that supplies the BAdI with metadata for the connection to the framework of the fair value effectiveness test. It then calls a method that applies the effectiveness test method, and returns the results.

You can set up your own custom effectiveness test methods so that they set the prospective effectiveness indicator, or the retrospective effectiveness indicator, or so that they immediately set the final effectiveness indicator, as is the case for the tolerance value check. For more information about the effectiveness indicator, see Determination of the Final Effectiveness Indicators [page 1103].

Integration

Creation of a BAdI Implementation

Effectiveness test methods are implemented as Business Add-Ins (BAdIs) with filter values. This means that you must create a filter value for each of your custom effectiveness test methods. You do this in Customizing for Bank Analyzer under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Configuration* > *Custom Methods for Effectiveness Test* > *Create Methods for Effectiveness Test*.

You define your custom methods for the effectiveness test in Customizing for Bank Analyzer under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Configuration* > *Custom Methods for Effectiveness Test* > *BAdI: Test Method*.

To integrate your custom methods for the effectiveness test, you need to define a transfer structure for the results of your method, and inform the system about the characteristics and key figures that you use as fields in this structure. You do this in Customizing for Bank Analyzer under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Configuration* > *Custom Methods for Effectiveness Test* > *Assign Characteristics and Key Figures to Field Names*.

> **Note**
>
> For more information, see the documentation for the Customizing activity *BAdI: Test Method*.

Connection to Reporting

The system displays the results of custom methods for the effectiveness test in reporting only if you have stored the associated characteristics and key figures in the field instance for the fair value effectiveness test, and if you have assigned your custom method for the effectiveness test. You do this in Customizing for Bank Analyzer under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Configuration* > *Field Instance* > *Set Up Field Instance*.

> **Note**
>
> The system displays the results generated by custom effectiveness test methods in the functions for reporting for the fair value effectiveness test [page 1055]. The results are displayed on a separate tab page for each custom method as follows: *Custom Method 1*, *Custom Method 2*, and so on.
>
> Note that the system displays the value of the effectiveness indicator as an icon only if you include in your output structure a field that refers to data element /BA1/R8_DTE_METH_EFFECT_FLG.

4.6.2.6.7 External Effectiveness Measurement (Data Transfer)

Use

You can also use the results of external effectiveness measurements for the fair value effectiveness test.

Integration

The system reads the imported results from the Source Data Layer (SDL). For this reason, before you call the effectiveness test make sure that all the external effectiveness measurements have been carried out and their results stored as positions in the SDL.

The system requires the following parameters in order to process external results:

Identifier for the hedging relationship

Date of effectiveness measurement

Final effectiveness indicator

Start date of ineffectiveness

You can transfer additional parameters, such as the hedge ratio.

If you want to use the offset method [page 1076] externally, you should also transfer its result in the retrospective indicator. This enables the system to compare the test result with the history and decide whether to classify a temporarily ineffective hedging relationship as *ineffective* .

Prerequisites

Position Classes and Positions

Before you can transfer data, you have to have created a position class in the SDL. To create position classes, in Customizing for *Bank Analyzer* choose *Source Data Layer* → *Primary Objects* → *Transaction Data* → *Positions* → *Edit Position Class* . To store values manually in the SDL positions, on the *SAP Easy Access* screen choose *Bank Analyzer* → *Source Data Layer* → *Primary Objects* → *Position* and then either *Create Position* or *Change Position* , as required.For more information about creating positions, see Primary Object: Position [page 142] .

You do not have to store the values you import in SDL positions. The values can be stored in any storage location, provided the system can use primary data sources to read the data.

Characteristics and key figures

You can transfer any characteristics and key figures. For this reason, you have to assign the characteristics and key figures to the fields used in the fair value effectiveness test. You do so in Customizing for *Bank Analyzer* by choosing *Processes and Methods* → *Hedge Processes* → *Fair Value Effectiveness Test* → *Configuration* → *Transfer* and then either *Assign Characteristics* or *Assign Key Figures* , as required.

Primary data source

To enable the system to read positions from the SDL, you have to create a primary data source of the category *Object Transfer Structure (OTS).* To do so, in Customizing for Bank Analyzer choose *Source Data Layer* → *Source Data Services* → *Access to Source Data* → *Object Transfer Structure (OTS)* → *Edit OTS for Primary Objects* → *Edit OTS for Positions.*

Secondary data sources

The fair value effectiveness test uses a secondary data source to read the SDL positions that contain the transferred test results. The system uses primary data sources to select the transferred values within the module. You create selection modules for the fair value effectiveness test in Customizing for *Bank Analyzer* by choosing *Processes and Methods* → *Hedge Processes* → *Fair Value Effectiveness Test* → *Processes* → *Secondary Data Source* → *Edit Module for the Selection of Hedging Relationships* .

Derivation of the effectiveness test rule [page 1042]

In order to process the data further, the system needs an effectiveness test rule for each hedging relationship. The system calls up the derivation function for the test rule within the secondary data source. You enter the settings that define how test rules are derived in Customizing for *Bank Analyzer* under *Processes and Methods* → *Hedge Processes* → *Fair Value Effectiveness Test* → *Valuation* → *Derivation of Rule for Effectiveness Test* → *Create Derivation Environments* , and in the subsequent IMG activities.

Features

The system reads from the SDL the test results for the hedging relationships whose effectiveness was determined externally. The system then checks the following:

If the SDL does not contain a value for the final effectiveness indicator, the system calls the internal methods of the fair value effectiveness test to measure the effectiveness of the hedging relationship. The test schema defines which test methods are called up and in which order.

If the retrospective effectiveness indicator was transferred when set to *temporarily ineffective* , the system uses the history stored internally to check whether the final effectiveness indicator has to be set to *ineffective* . If this is the case, the system sets the final effectiveness indicator to *incorrect* .

For this check, the system requires the maximum number of times that tests with the result *temporarily ineffective* are permitted one after another. The system reads this information from Customizing for the offset method [page 1076] . For this purpose, the system requires an effectiveness test rule [page 1042] . This rule is derived in the secondary data source for the hedging relationship.

The system sets the Start Date of Ineffectiveness indicator.

If the effectiveness indicator is set to *ineffective* when it is imported, or if it is set to *ineffective* by the offset method, the system uses the history to determine the start date of ineffectiveness. The system sets the start date of ineffectiveness to the date of the most recent fair value effectiveness test that had the result *effective* .

In reporting [page 1055] for the fair value effectiveness test, the system displays the results of the external effectiveness measurement separately. This enables you to see which data was transferred.

In reporting, you can set the final effectiveness indicator to *effective* or *ineffective* manually.

For more information, see the document Processes of the Fair Value Effectiveness Test [page 1039] .

4.6.2.6.8 Determination of the Final Effectiveness Indicator

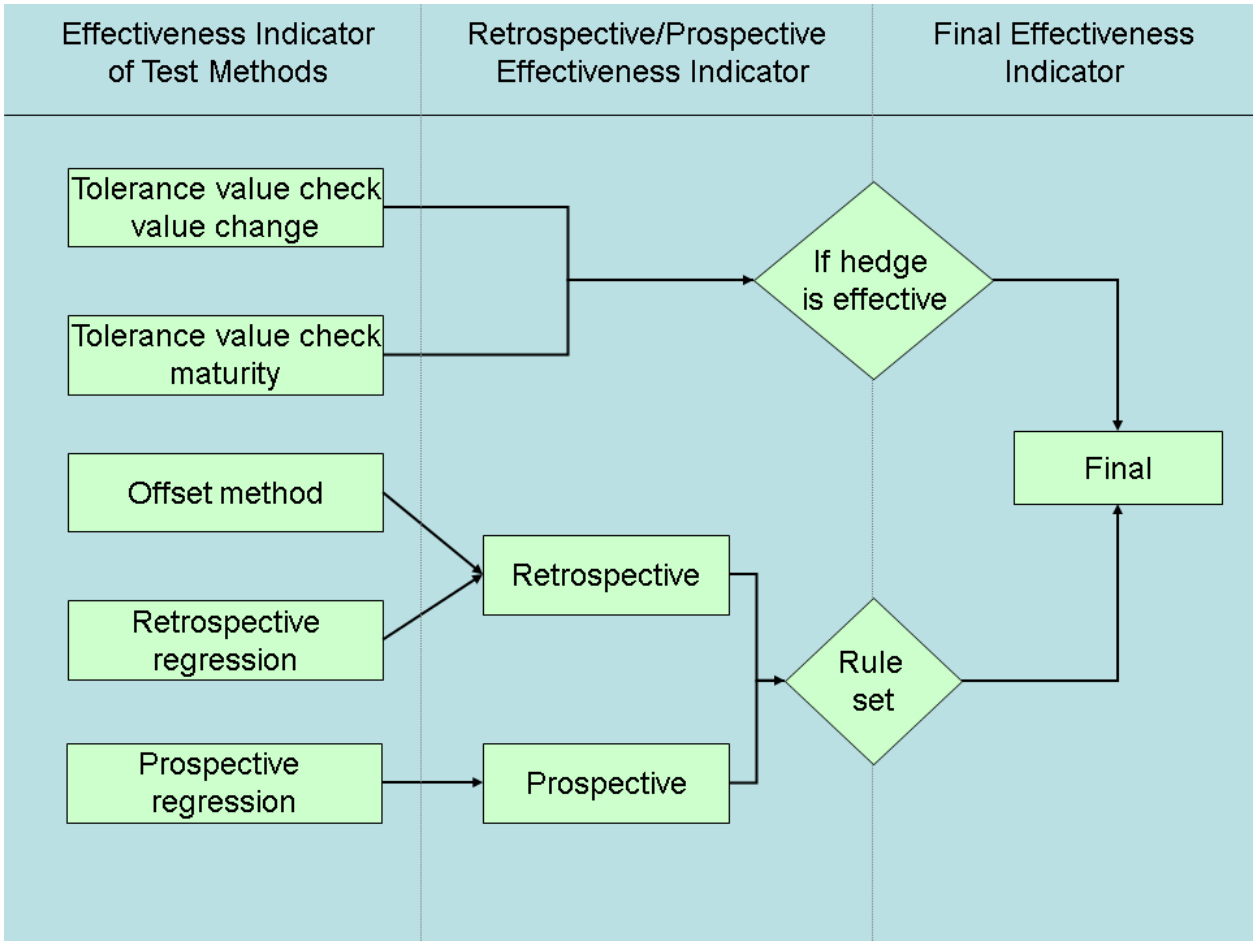
Use

The effectiveness test provides the final effectiveness indicator for each hedging relationship that is tested. This indicator states whether the hedging relationship being analyzed is effective or ineffective. In certain circumstances, the fair value effectiveness test calls up several test methods in order to assess the effectiveness of a hedging relationship, and so the system derives the final effectiveness indicator as follows:

- If a tolerance value check [page 1074] judges the hedging relationship to be effective, the system sets the final effectiveness indicator to *Effective*, without calling up other effectiveness test methods.
- Each effectiveness test method has a separate effectiveness indicator, which, depending on the test method, is incorporated in the indicator for the retrospective or prospective effectiveness test.
- Each test method sets the retrospective and prospective effectiveness indicators in the order defined in the test schema.
- To ensure the test produces a clear result, the system derives the final effectiveness indicator from the retrospective and prospective effectiveness indicators. It uses the rule set for this purpose (see table below).

> **Note**
>
> You can override the effectiveness test methods using the *A Priori Effective* indicator. If the *A Priori Effective* indicator is set, the system sets the final effectiveness indicator, without carrying out any further checks.

The following figure shows the connection between the internal effectiveness test methods and the effectiveness indicator:



Prerequisites

We provide rule sets for deriving the final effectiveness indicator. You can display the rule sets and create your own rule sets in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *System Settings* > *Edit Set of Rules for Determining the Effectiveness Indicator*.

In order to use a rule set, you assign it to an effectiveness test environment and an effectiveness test rule [page 1042]. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Valuation* > *Environment and Rules for Effectiveness Test* > *Edit Rule for Effectiveness Test*.

Rule set provided by SAP:

Retrospective Effectiveness Indicator	Prospective Effectiveness Indicator	Final Effectiveness Indicator

Initial	Initial	Error when determining the effectiveness
Initial	Error when determining the effectiveness	Error when determining the effectiveness
Initial	Effective	Effective
Initial	Ineffective	Ineffective
Error when determining the effectiveness	Initial	Error when determining the effectiveness
Error when determining the effectiveness	Error when determining the effectiveness	Error when determining the effectiveness
Error when determining the effectiveness	Effective	Effective
Error when determining the effectiveness	Ineffective	Ineffective
Effective	Initial	Effective
Effective	Error when determining the effectiveness	Effective
Ineffective	Initial	Ineffective
Ineffective	Error when determining the effectiveness	Ineffective
Ineffective	Effective	Effective
Ineffective	Ineffective	Ineffective
Temporarily ineffective	Initial	Effective
Temporarily ineffective	Effective	Effective
Temporarily ineffective	Ineffective	Ineffective

4.6.2.7 Tools

4.6.2.7.1 Display of Field Instances

Use

Field instances contain all the characteristics used internally by the system, and are used as the basis for valuation structures, which the system generates automatically in line with the settings in the Customizing for the fair value effectiveness test [page 1036] . Therefore it is helpful to be able to check the status of field instances when you are using the system operationally.

Integration

You create field instances in Customizing and assign them to the run type for the fair value effectiveness test. You have to activate field instances before you can use them. You also do this in Customizing. Once field instances have been activated, the system automatically generates the package type required for the evaluations to save results in the Result Database (RDB). When a field instance is activated, the system always uses the Customizing settings that are valid at this particular point in time. Because you can change the Customizing settings for the fair value effectiveness test at any time, the properties of the field instances used must be checked while the system is in operation.

You can find the IMG activities for creating and activating field instances in Customizing for *Bank Analyzer* by choosing *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Configuration* > *Field Instance* .

Activities

On the *SAP Easy Access screen* choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Tools* > *Display Field Instances.*

4.6.2.7.2 Display of Secondary Data Sources

Use

The fair value effectiveness test [page 1036] uses secondary data sourcesto select transaction data and derive the test rule that defines how the fair value effectiveness is tested for the transaction in question. In order to check the settings for the secondary data source while the system is in operation, you can display the secondary data source and with the selection modules. This enables you to track which transactions have been selected, and see how the system prepared the data for the tests.

Prerequisites

You have configured the secondary data source in the Customizing for the fair value effectiveness test. You can find the relevant IMG activities in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Processes* > *Secondary Data Source* .

Activities

On the *SAP Easy Access screen* , choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Tools* > *Edit Secondary Data Sources.*

4.6.3 Cash Flow Hedge Analysis (FS-BA-PM-HP-CFH)

Use

This component contains methods that you can use to analyze and document macro cash flow hedging relationships. The analyses are designed to be used at portfolio level to examine hedging relationships that have a large number of hedged items and hedging instruments. The treatment of the cash flow hedging relationships aims to reflect current practice in banks, where asset/liability management is performed on the basis of aggregated cash flows.

Cash flow hedging relationships and fair value hedging relationships are the two basic models covered by the International Accounting Standards (IAS) for hedging risks. In contrast to fair value hedging relationships, cash flow hedging relationships are based on individual cash flows or the cash flows for a maturity band rather than on the value of financial transactions and financial instruments.

Cash Flow Hedge Analysis helps you do the following:

- Monitor the effectiveness of existing hedging relationships
  The system displays the existing cash flow hedging relationships, and helps you to analyze them using key figures such as the hedge ratio, and by comparing assets and liabilities in terms of interest flows and capital flows.
- Simulate future hedging instruments
  You can define hedging instruments that do not yet exist, and test them by assigning them to hedging relationships. You can use these fictitious transactions to test the effectiveness of hedging instruments in advance.
- Assign non-designated hedging instruments
  You can assign (either partially or completely) hedging instruments that have not yet been designated to a hedging relationship and then carry out further analyses of the cash flow hedging relationships. You can also test the hedging relationship by removing hedging instruments already assigned to it.
- Generate hedge proposals
  If you have assigned hedging instruments to a hedging relationship for test purposes only, you can let the system generate a hedge proposal. You can then use this proposal in the Hedge Accounting Workplace component to change a hedging relationship.

To check hedge proposals, when using the hypothetical derivative approach in Cash Flow Hedge Analysis you can branch to the fair value effectiveness test for hedging relationships.

- Document existing hedging relationships
  You can save all the reports and call them up again in order to meet the IAS requirements at any time regarding documentation of the effectiveness of hedging relationships.

Integration

- Cash Flow Hedge Analysis contains auxiliary functions for the Hedge Accounting Workplace [page 976]. You can call Cash Flow Hedge Analysis directly from the Hedge Accounting Workplace. The hedge proposals that you generate in Cash Flow Hedge Analysis can be used in the Hedge Accounting Workplace to change existing cash flow hedging relationships.
- The system selects the cash flow hedging relationships that are to be analyzed from the worklist in the Hedge Accounting Workplace. It selects the individual financial instruments and financial transactions from the Source Data Layer (SDL) [page 100].
- The system calls the cash flow generation method [page 440] to generate the cash flows for the hedged items and hedging instruments.
- If want to use the hypothetical derivative approach in your cash flow hedge analysis, you can call the fair value effectiveness test for hedging relationships [page 1036] from within the reporting function in Cash Flow Hedge Analysis.

Features

Cash Flow Hedge Analysis contains an infrastructure that you can use in the following application areas of macro cash flow hedging relationships:

- Hedging of Cash Flows Subject to Reinvestment (Approach for Reinvestment Risk)
  This approach focuses on the reinvestment risk that is associated with nominal and interest cash flows. For example, a reinvestment risk arises for future incoming payments because only a lower rate of interest can be achieved if the capital is reinvested at a time when interest rates are falling. A swap can be used to hedge against this interest rate risk.
  Cash Flow Hedge Analysis is based on a method described in Batch 6 of the Implementation Guidance "Questions and Answers" published by the IAS 39 Implementation Guidance Committee (IGC) (IAS 39.IG F6.2; F6.3).
  The cash flow hedging relationships that are to be analyzed can contain both fixed-rate and variable-rate hedged items. The variable-rate hedged items have to have the same reference interest rate as the hedging instruments in the hedging relationships. This is required by IAS.
  In the reporting function, the system displays the nominal values of the hedged items and the hedging instruments. If the nominal values of the hedged items and the hedging instruments are the same, since they have the same reference interest rate the hedge is adequate. In the basic setting, the system displays the nominal values of the hedged items in the section of the maturity band in which the interest rate was fixed. The system displays the hedging instruments in all sections of the maturity band. From the amount of the nominal values you can see whether the nominal cash flows or the interest cash flows are hedged in the respective section of the maturity band.
- Hedging of Future Variable-Rate Cash Flows (Hypothetical Derivative Approach)

This approach focuses on the hedging of cash flows that are expected in the future. These are hedged using swaps, for example, in which fixed rates are agreed for a future point in time.
Since the reference interest rates of the hedged items and hedging instruments are usually not the same in this case, it is not sufficient just to analyze the nominal values of the hedged items and hedging instruments to demonstrate the effectiveness of the hedging relationship. For this reason the system also contains a fair value effectiveness test for use with the hypothetical derivative approach, as described in IAS 39.IG F5.5.
The hedged items in the cash flow relationships have variable interest rates, and should all have the same reference interest rate. For loans, this can be the bank's in-house rate, for example. A different rate is used for the hedging instruments. This reference interest rate is usually taken from the market, such as the 3-month Euribor.
In reporting, the system compares the hedged items in the hedging relationship with the hedging instruments. However, it does not display the actual hedging instruments, but the hypothetical derivatives generated from them. Hypothetical derivatives are fictitious hedging instruments. Their variable sides have the same interest rate period and reference interest rate as the hedged item; the fixed sides are constructed in such a way so that the swap has a value of zero on the designation date.
The comparison of the nominal values, and the fixed rates of the hedged items and hedging instruments provided by the reporting function in Cash Flow Hedge Analysis enable you to obtain an impression of how effective the hedges are. You can also test the fair value effectiveness. In this test, the system measures the contribution of each hedging instruments to the effectiveness of the hedging relationship. It does this by comparing the changes in the fair value of the hypothetical derivatives with the changes in the fair value of the associated actual hedging instruments.

Technical View

Cash Flow Hedge Analysis contains the following functions:

- The valuation run [page 1111] selects and analyzes financial transactions and financial instruments.
- The aggregation run [page 1123] groups the results of the valuation run across a segment hierarchy and along a maturity band.
- Reporting [page 1140] enables you to display the evaluation results, redesignate hedging instruments for test purposes, and to generate hedge proposals.
- You can use an online run [page 1149] for smaller amounts of data. The online run carries out the valuation run and then the aggregation run, and then branches to reporting. The valuation run and the aggregation run should be started as batch runs, though.

By means of the *Approach for Hypothetical Derivatives* indicator you control whether the system is to use the reinvestment risk approach, or the hypothetical derivative approach. The indicator is found in the valuation run type and on the selection screen for the online run.

More Information

Calling Up and Processing a Cash Flow Hedge Analysis [page 1110]

4.6.3.1 Calling and Processing a Cash Flow Hedge Analysis

Use

You can use a cash flow hedge analysis to analyze hedging relationships and aggregate the key figures determined along a freely-definable segment hierarchy. In reporting, you check the effectiveness of the existing hedging relationships, and test alternative assignments of hedge transactions.

The cash flow hedge analysis process comprises the following runs, which you can create and schedule independently of one another:

- Valuation run [page 1111]
- Aggregation run [page 1123]

The aggregation run is based on the valuation run. For this reason, the valuation run and the aggregation run must be executed one after the other. Each aggregation run is assigned uniquely to one valuation run. There can be any number of aggregation runs for one particular valuation run.

Prerequisites

You have defined run types for the valuation run, the subvaluation runs, and the aggregation run in Customizing for Cash Flow Hedge Analysis. You define run types in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Processes* > *Edit Valuation Run Type*, *Edit Subvaluation Run Type* and *Edit Aggregation Run Type*.

Procedure

Schedule the Analyses as Batch Runs in Schedule Manager

1. Create a valuation run [page 1128].
2. Create an aggregation run [page 1129].
3. Schedule the runs in Schedule Manager [page 437].

The system calls the valuation run and the aggregation run. When you schedule runs, you can schedule new subruns and incomplete subruns. If you choose an incomplete run, the run is restarted.

Start the Analysis Directly as a Batch Run

For test purposes, if you want to analyze a small number of hedging relationships in Cash Flow Hedge Analysis, you can start this process directly in the area menu. You do this as follows:

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Run Administration* > *Create and Schedule* > *Start Cash Flow Hedge Analysis*. A selection screen appears.
2. Enter a valuation run type, a key date, and an aggregation run type. Then choose *Execute*.

The system creates a valuation run and an aggregation run, and executes them one after the other. When you start runs, you can choose new subruns or incomplete subruns. If you choose an incomplete run, the run is restarted.

> **Note**
>
> You can start valuation runs and aggregation runs individually as batch runs from within the area menu. You do this on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Run Administration* > *Create and Schedule* > *Start Valuation Run* or *Start Aggregation Run*.

> **Note**
>
> Note that the runs that you start from the area menu use work processes. In certain circumstances, this can lead to a shortage of system resources, and delay the start of the runs that you have scheduled in Schedule Manager.

Online Run

You can start Cash Flow Hedge Analysis as an online run. In this case, the system calls the valuation run and the aggregation run one after the other. It keeps the results in the main memory, and does **not** write them in the Result Database (RDB).

For more information, see Online Analysis [page 1149].

Result

The system has analyzed and aggregated the selected hedging relationships, hedged items and hedging instruments, and stored the results in the RDB.

You can display the results in reporting [page 1140].

4.6.3.1.1 Valuation Run

Use

The valuation run is the first of two processes in Cash Flow Hedge Analysis [page 1107]. In the valuation run, the system analyzes the hedging relationships or hedged items and hedging instruments.

You can divide the valuation run into multiple subvaluation runs. However, you cannot process hedged items and hedging instruments in the same subvaluation run. For this reason, you always have at least two subvaluation runs; one for hedged items, and one for hedging instruments. You can schedule and start the subvaluation runs a different points in time. A valuation run is complete when the system has completed all the individual subvaluation runs.

The second process in cash flow hedge analysis, the aggregation run [page 1123], is based on the valuation run. Each aggregation run is assigned uniquely to one valuation run. There can be any number of aggregation runs for one particular valuation run.

Prerequisites

1. You have defined run types for the valuation run and the subvaluation runs in Customizing for Cash Flow Hedge Analysis.

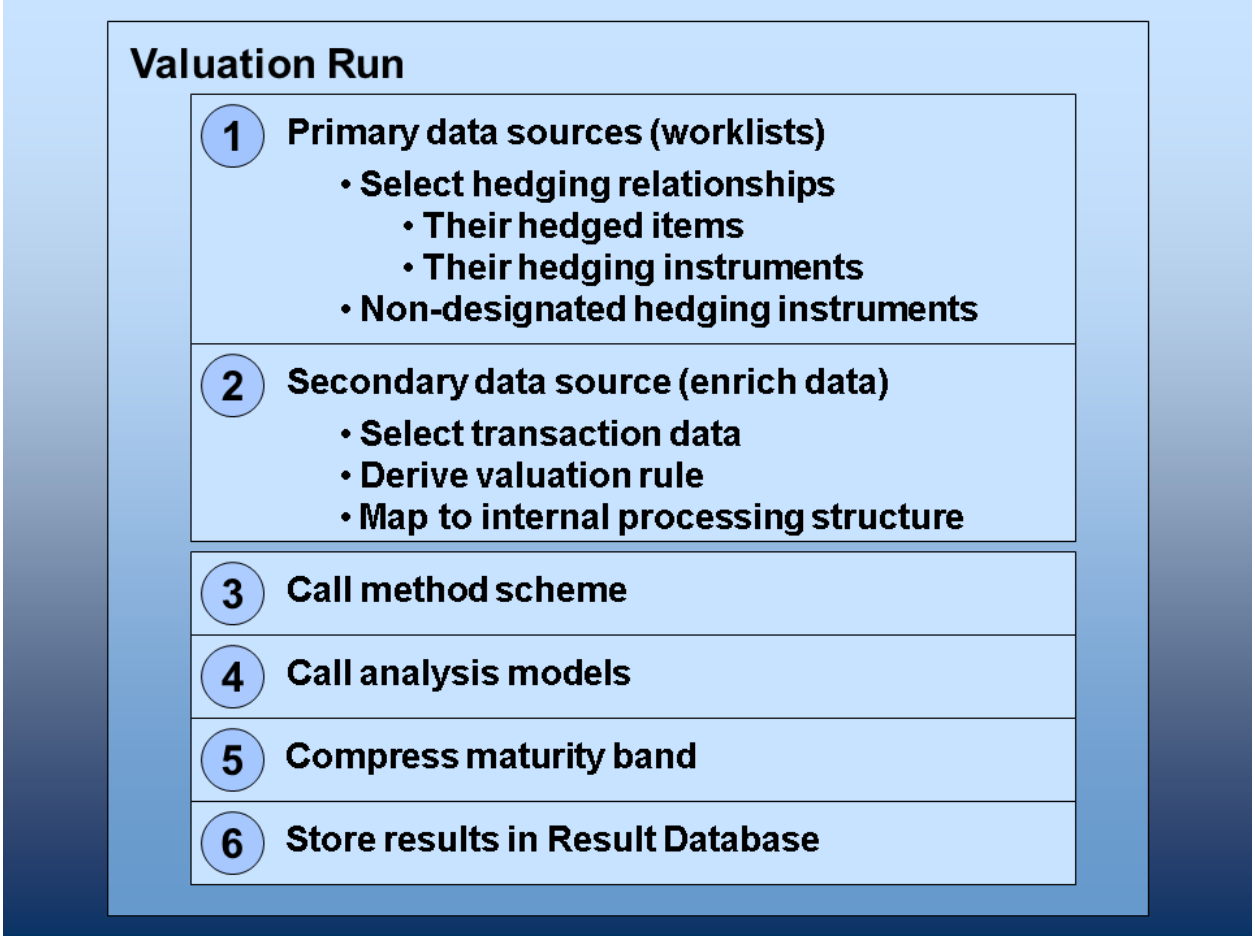
   You define run types in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Processes* > *Edit Valuation Run Type* and *Subvaluation Run Type*.

   > **Note**
   >
   > Note that you select the approach that you want to use to run the cash flow hedge analysis function in the valuation run type. For the hypothetical derivative approach, set the *Use Hypothetical Derivative Approach* checkbox, and choose a hypothetical derivative environment. Do not set this checkbox if you want to use the reinvestment risk approach.
   >
   > For the valuation run, additional settings must also be made for the secondary data source that is used to enrich the selected data, and for the cash flow refinement methods that are used to prepare the data for the analysis.
   >
   > You create secondary data sources in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Configuration* > *Secondary Data Source* > *Edit Selection Modules*.
   >
   > You configure the methods for cash flow refinement in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Basic Settings for Methods and Edit Method Schema*. You can display the method schemas provided by SAP in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Edit Method Schemas*.

2. You have created a valuation run [page 1128] in the application and scheduled this run in Schedule Manager [page 437].

   > **Note**
   >
   > Instead of scheduling the runs in Schedule Manager, you can start the cash flow hedge analysis as an online analysis [page 1149]. In the online analysis, the system calls the valuation run and the aggregation run in direct succession. The results are kept in the main memory; they are not saved in the Result Database (RDB).
   >
   > You can start the valuation run from within the area menu. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Run Administration* > *Create and Schedule* > *Start Valuation Run*.
   >
   > Note that the runs that you start from the area menu use work processes. In certain circumstances, this can lead to a shortage of system resources, and delay the start of the runs that you have scheduled in Schedule Manager.

Process



1. The system uses primary data sources to generate the worklists for the hedged items and hedging instruments of the hedging relationships, as well as for the hedging instruments that have not been designated.
   For more information, see Selection of Hedged Items and Hedging Instruments [page 1115].
2. The system enriches the data in a secondary data source. It does this separately for each primary data source in order to generate the worklist. This process includes the following steps:
   - Reading of the transaction data
     The system reads the transaction data for the hedged items or hedging instruments using the identifiers. It uses primary data sources to read single records.
   - Determination of the valuation rule
     You use the valuation rule [page 490] to define the valuation process on the basis of characteristics. For more information, see Derivation of Global and Local Valuation Rules [page 1116].
   - Mapping of the structure of the transaction to the valuation structure
     To be able to analyze the selected transactions, the system standardizes their structures to match the valuation structure. We provide function modules for this purpose that you can incorporate in the secondary data source. For more information, see Mapping of SDL Templates to the Valuation Structure [page 496].

   Note that the system repeats this step for each worklist. You can use a separate secondary data source for each primary data source used for this purpose; however, you can also use the same secondary data source

for all the primary data sources. Once all the primary and secondary data sources have been processed, the system moves on to the next process step.

3. The system calls the method schema [page 534] to prepare the transaction cash flows for the valuation. A typical method schema contains the following methods for cash flow refinement [page 499]:
   - Determination of market data
     The system uses the derivation tool to derive the market data set and, if necessary, the spread curve type on the basis of characteristics. The market data is required for the analyses later on.
   - Generation of Hypothetical Derivatives [page 515]
     In the hypothetical derivative approach, the system generates a hypothetical derivative for every designated and non-designated hedging instrument.
   - Cash Flow Filter [page 513]
     If required, the system removes cash flows from the selected financial transactions and instruments. In this way, the system can remove hedging instruments and interest rate cash flows from hedged items that are already being used in fair value hedging relationships. For more information, see Removal of Cash Flows that are in Fair Value Hedges [page 1119].
   - Application of market data scenarios
     The system generates a separate valuation structure for each market data scenario and each transaction. This is in preparation for analysis of the transactions on the basis of scenarios, which takes place at a later point in time.
   - Fixing of floating-rate cash flows [page 506]
     The system determines floating-rate cash flows using the forward rates.
   - Cash Flow Determination [page 504]
     The system breaks down optional parts of transactions.
   - Application of interest capitalization
     The system adjusts the transaction cash flows that have with interest capitalization so that the transactions can be analyzed.
   - Use of due date scenarios [page 508]
     The system adjusts account balances in line with the Customizing settings.
   - Updating of cash flows (rollover [page 510])
     If required, the system rolls over those transactions that mature within the time period shown in reporting in order to reflect reinvestment risk. It writes the cash flows from the last maturity band interval of a transaction of this type to the following maturity band intervals. In Customizing, you define which transactions belong to the rollover category.
4. The system analyzes the selected transactions using the current market data and market data scenarios (scenarios and scenario progressions), and assigns them to the relevant sections of the maturity band. For more information, see Analysis of Variable Rate Hedged Items and Hedging Instruments [page 1121].
5. The system aggregates the results of the single transactions.
6. The system stores the interim results in the Result Database (RDB) [page 1823].

Result

The system has analyzed the selected hedging relationships and stored the results in the RDB. Aggregation runs can access the results of the valuation run. For more information, see Creating Payment Orders [page 1129].

4.6.3.1.1.1 Selection of Hedged Items and Hedging Instruments

Purpose

For the evaluations in cash flow hedge analysis, the system needs the hedged items and hedging instruments of the cash flow hedging relationships, and the hedging relationships that have not yet been designated, and which you want to assign in the reporting function to the existing cash flow hedging relationships.

Hedged items and hedging instruments are selected in the valuation run [page 1111] in cash flow hedge analysis. This process is called automatically by the system. Primary data sources from the Hedge Accounting Workplace are used to select the data.

If you want to exclude hedged items and hedging instruments that have already been used in a fair value hedging relationship either completely or partially from cash flow hedge analysis, then you need to use the cash flow filter [page 513] . For more information about using the cash flow filter in cash flow hedge analysis, see Removal of Cash Flows that are in Fair Value Hedges [page 1119] .

Prerequisites

You need to have already defined the primary data sources for selecting hedging relationships, their hedged items, and non-designated hedging instruments. This is done in Customizing for *Bank Analyzer* under *Processes and Methods* → *Hedge Processes* → *Cash Flow Hedge Analysis* → *Processes* → *Edit Valuation Run Type*

Process Flow

The system generates the worklist for the cash flow hedging relationships. In doing so, it uses the primary data source that you have specified in Customizing for the valuation run type.

Note that in the reporting [page 1140] function in Cash Flow Hedge Analysis, you can analyze only those hedging relationships that the primary data source has selected for the hedging relationships. Hedging relationships cannot be read or created in reporting.

The worklist contains the identifier, the reference currency, and the reference interest rate for each hedging relationship together with the following data:

Pointer to the relevant hedged items

List of the relevant hedging instruments

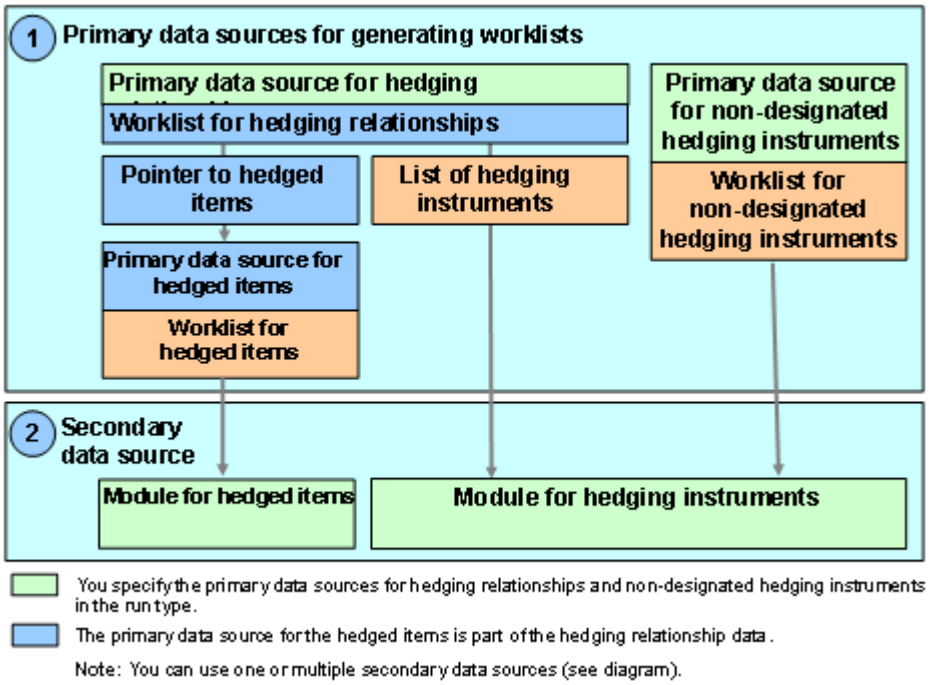
The worklist for the hedging relationships already contains a list for each hedging relationship containing the identifiers of this relationship's hedging instruments. Therefore, no separate primary data sources are required to generate this worklist.

The system generates the worklist for the hedged items. In doing so, it uses the primary data sources that are defined in the hedging relationship.

The system generates the worklist for the non-designated hedging instruments. In doing so, it uses the primary data source that you have specified in Customizing for the valuation run type.

The non-designated hedging instruments only become relevant in reporting, where, for test purposes, you can assign them to the existing hedging relationships as hedging instruments.

The primary data sources for generating the worklists interact as follows:



For performance reasons, a valuation run contains multiple subvaluation runs, which the system processes in parallel. A separate subvaluation run is used for the non-designated hedging instruments. The hedging relationships can be grouped together in one subvaluation run or in multiple subvaluation runs.

4.6.3.1.1.2 Derivation of Global and Local Valuation Rules

Use

During the valuation run in Cash Flow Hedge Analysis and in Strategy Analyzer, the system derives the valuation rule [page 490] so it can use it to define how other methods are used. Unlike in other Bank Analyzer applications, in Cash Flow Hedge Analysis and Strategy Analyzer a distinction is made between global valuation rules and local valuation rules. Global valuation rules apply to financial transactions and financial instruments; local valuation rules contain control information at subcontract level, and can be used to override global valuation rules.

In Cash Flow Hedge Analysis, the valuation rule is used to define how the following methods are applied:

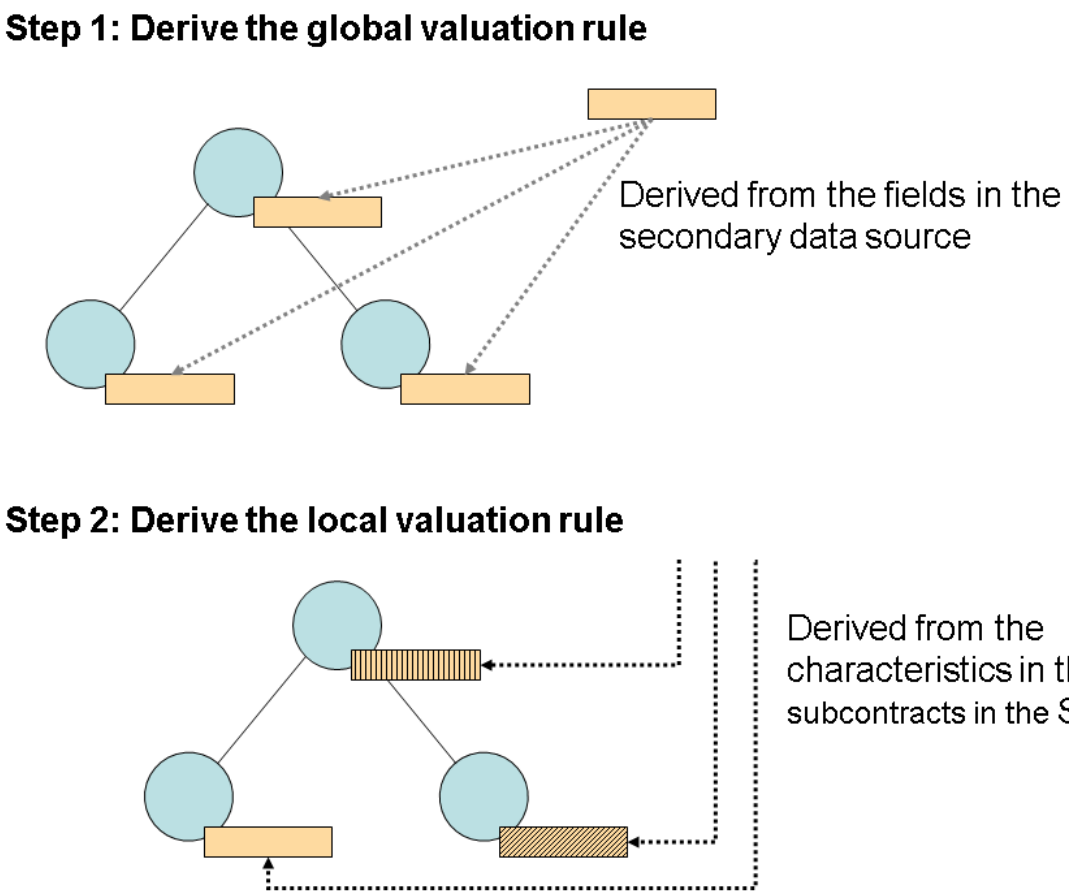
- Price calculator [page 547]
- Cash flow filter [page 513]
- Cash flow fixing [page 506]
- Cash flow determination [page 504]
- Interest capitalization
- Due date scenarios [page 508]
- Rollover [page 510]
- “Short-term” category

Prerequisites

- You have created valuation rules. You define valuation rules in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivations* > *Determination of the Valuation Rule* > *Create Valuation Rules*.
- You have created a derivation strategy for the valuation rule. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivations* > *Determination of the Valuation Rule* > *Create Derivation Environments* and then create derivation environments and derivation modules in the subsequent Customizing activities. You define derivation strategies in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivations* > *Determination of the Valuation Rule* > *Create Derivation Strategies*.
- You need to have already added the derivation strategy of the global valuation rule in the form of a derivation environment and a derivation module to the secondary data source. You edit secondary data sources in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Configuration* > *Secondary Data Source* > *Edit Selection Modules* or *Analytics* > *Strategy Analyzer* > *Configuration* > *Secondary Data Source* > *Edit Selection Modules*.
- You have already assigned the derivation strategy for the local valuation rule to the cash flow hedge environment. You edit cash flow hedge environments in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Valuation* > *Edit Cash Flow Hedge Environment* or *Analytics* > *Strategy Analyzer* > *Valuation* > *Create Valuation Environment and Assign Market Data Set to Derivation Strategy*.

Process

The following figure shows how the valuation rule is derived for a financial transaction that has multiple subtransactions.



The system first derives the global valuation rule, and then distributes it to all the subcontracts. It then derives a local valuation rule for each subcontract. If the system was able to derive a valuation rule for a subcontract, it then overwrites the global valuation rule stored in the subcontract with the local valuation rule. If it could not derive a local valuation rule, then the global valuation rule is retained in the subcontract. Note the following:

- Global valuation rules
  The system derives the global valuation rule for each financial transaction or instrument in the secondary data source for Cash Flow Hedge Analysis. The system does this by means of the combination of derivation environment and derivation module that you entered in the step sequence of the secondary data source. This step has to be carried out before data is mapped to the valuation structure in the Calculation and Valuation Methods.
  You can use any of the characteristics in the secondary data source as the source characteristics for the derivation process. These are the fields in the input structure and output structure, plus the local fields.
- Local valuation rules
  The system also derives the local valuation rule in the secondary data source. This is part of the mapping for the valuation. The system uses the combination of derivation environment and derivation module that you entered in the valuation run type in Customizing for Cash Flow Hedge Analysis to derive the valuation rule.
  You can use any of the characteristics in the subcontracts in the SDL as the source characteristics for the derivation process. These are the characteristics that you find on the *Attributes* tab page when you edit financial transactions and financial instruments.

> **Caution**
>
> If the local valuation rule for a subcontract is initial, then the system does not overwrite the global valuation rule in this subcontract.

> **Note**
>
> The function for deriving local valuation rules can be used to refine processing at subcontract level. Use of this function is optional.

Result

The system assigned valuation rules to individual financial transactions and instruments at contract level, and, if required, at subobject level as well.

4.6.3.1.1.3 Removal of Cash Flows that are in Fair Value Hedges

Use

In cash flow hedge analysis, you can choose whether to include just partially, or ignore completely, financial instruments or transactions that are already used in fair value hedging relationships. Derivatives that are already designated as hedging instruments in fair value hedging relationships must not be designated as hedging instruments in a cash flow hedging relationship. Interest cash flows and other cash flows that have the nature of interest must not be included in cash flow hedge analysis if they relate to financial transactions or financial instruments that are already included as hedged items in a fair value hedging relationship.

For these reasons, the system contains a function that you can use to remove certain cash flows from financial transactions and financial instruments.

Integration

The removal of cash flows is part of the valuation runs [page 1111] in cash flow hedge analysis, and takes place after the worklists have been generated and the valuation rule derived. From a technical point of view, cash flows are removed by means of the cash flow filter [page 513], which is one of the cash flow refinement methods, and is called as part of the method schema for cash flow hedge analysis. You use valuation rules [page 490] and the cash flow indicator to define which cash flows the system removes.

Prerequisites

- You need to have already assigned the predefined method schema [page 534] CFHA1 to the cash flow hedge environment. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Valuation* > *Edit Cash Flow Hedge Environment*.
  If you use your own method schema, then you need to have assigned the cash flow refinement method [page 499] *Cash Flow Filter* to it. You configure method schemas in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Edit Method Schemas*. You have then assigned the method schema that you want to use to the cash flow hedge environment.
- You need to have already defined the valuation rules for the hedged items and hedging instruments that you want to use to control the cash flow filter. You define valuation rules in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Derivations* > *Determination of the Valuation Rule* > *Create Valuation Rules*.
- You have defined the cash flow filter in the method environment that you want to use. You define for which valuation rules the system is to remove cash flows, and which cash flows are to be removed from the financial transactions and financial instruments in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Edit Method Environments*.
- You need to have already assigned the cash flow filter and valuation rules for hedged items and hedging instruments to the cash flow hedge environment. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Valuation* > *Edit Cash Flow Hedge Environment*.

> **Caution**
>
> The system can check only individual hedged items and hedging instruments to determine whether they have already been used in a fair value hedging relationship. When the system checks this, under certain circumstances it accesses tables in which transaction numbers are not part of the primary key. This can increase the runtime. For this reason the use of cash flow filters is optional. If required, you can switch the cash flow filter on or off in the Customizing activity *Edit Cash Flow Hedge Environment* by setting or deselecting the indicator.

Features

1. The system searches the worklist for cash flow hedge analysis, and identifies all the hedged items and hedging instruments that have already been included in fair value hedging relationships.

> **Caution**
>
> Note that the system recognizes only financial transactions and financial instruments that have the object relationship categories HBGFG, HBGFI, HBSFG, or HBSFI in the hedging relationships. The analysis is hard-programmed for these values, so that you cannot enter your own Customizing settings for the object relationship categories.

2. The system removes certain cash flows from the hedged items and hedging instruments that are already in hedging relationships. The International Accounting Standards prescribe the following procedure:
   - Removal of all hedging instruments that are already used in fair value hedging relationships
   - Removal of all cash flows that have the nature of interest cash flows that are already used in fair value hedging relationships
3. The system continues analyzing the cash flow hedging relationships.

Example

For an example of the Customizing settings required for this process, see Cash Flow Filter [page 513].

4.6.3.1.1.4 Analysis of Variable-Rate Hedged Items and Hedging Instruments

Use

This function is used by the system to determine the cash flows of variable-rate hedged items and hedging instruments, and to assign them to the correct sections of the maturity bands for reporting in Cash Flow Hedge Analysis. You can adjust your Customizing settings to determine how the system processes the nominal value and interest flows of the variable-rate transactions, as well as the nominal value of the hedging instruments.

Integration

The analysis of hedged items and hedging instruments transactions is part of the valuation run [page 1111] in Cash Flow Hedge Analysis.

Prerequisites

In *Bank Analyzer* Customizing under *Processes and Methods* → *Hedge Processes* → *Cash Flow Hedge Analysis* → *Valuation* → *Edit Cash Flow Hedge Environment* , you need to have created the cash flow hedge environment and adjusted the analysis model for hedged items and hedging instruments within this environment.

Since you make the settings for a cash flow hedge environment, the settings are always valid for all hedged items and hedging instruments in a valuation run.

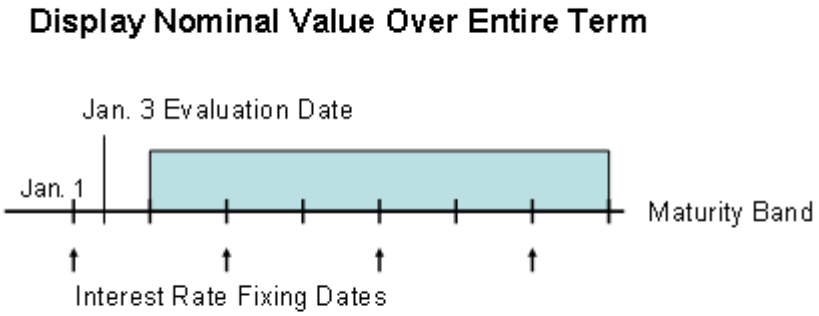
Features

You can define whether the system is to ignore or includes the interest cash flows from hedged items in the analysis.

You can decide if the nominal values of the hedged items and hedging instruments appear over the entire term of the transaction in a maturity band, or whether they appear only in the maturity band sections in which the interest rate is fixed. You can enter the following settings:

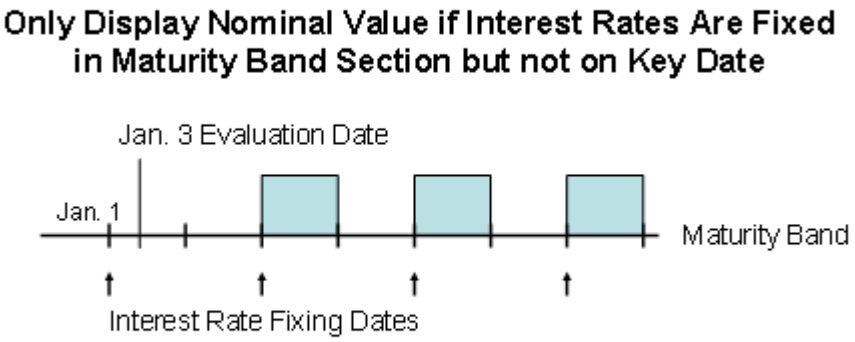
Display the nominal value of the transaction over the entire term

This is the standard setting for hedging instruments.

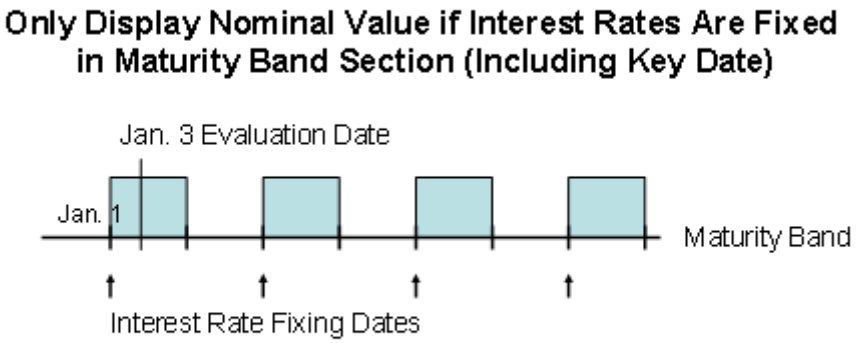


Only display the nominal value of the transactions in those maturity band sections in which the interest rates are fixed. The system displays the nominal value only if the interest rates are fixed after the evaluation date.

This is the standard setting for variable-rate hedged instruments.



Display the nominal value of the transactions only in those maturity band sections in which the interest rates are fixed; the system displays the nominal value in the first maturity band section, even if the interest rates are fixed after the evaluation date.



You use this setting if, for example, you want to analyze swaps with different interest fixing dates and interest periods of varying lengths. In the case of a maturity band with monthly dates, the system displays the nominal value of the individual swaps only in those maturity band sections in which the interest rate fixing for the respective swaps takes place.

4.6.3.1.2 Aggregation Run

Use

The aggregation run is the second process in Cash Flow Hedge Analysis [page 1107]. In the aggregation run, the system aggregates the results of a valuation run [page 1111] along a segment hierarchy. The segment hierarchy is used in reporting [page 1140] as the basis for displaying hedging relationships or hedged items and hedging instruments. Each aggregation run is assigned uniquely to one valuation run. There can be any number of aggregation runs for one particular valuation run.

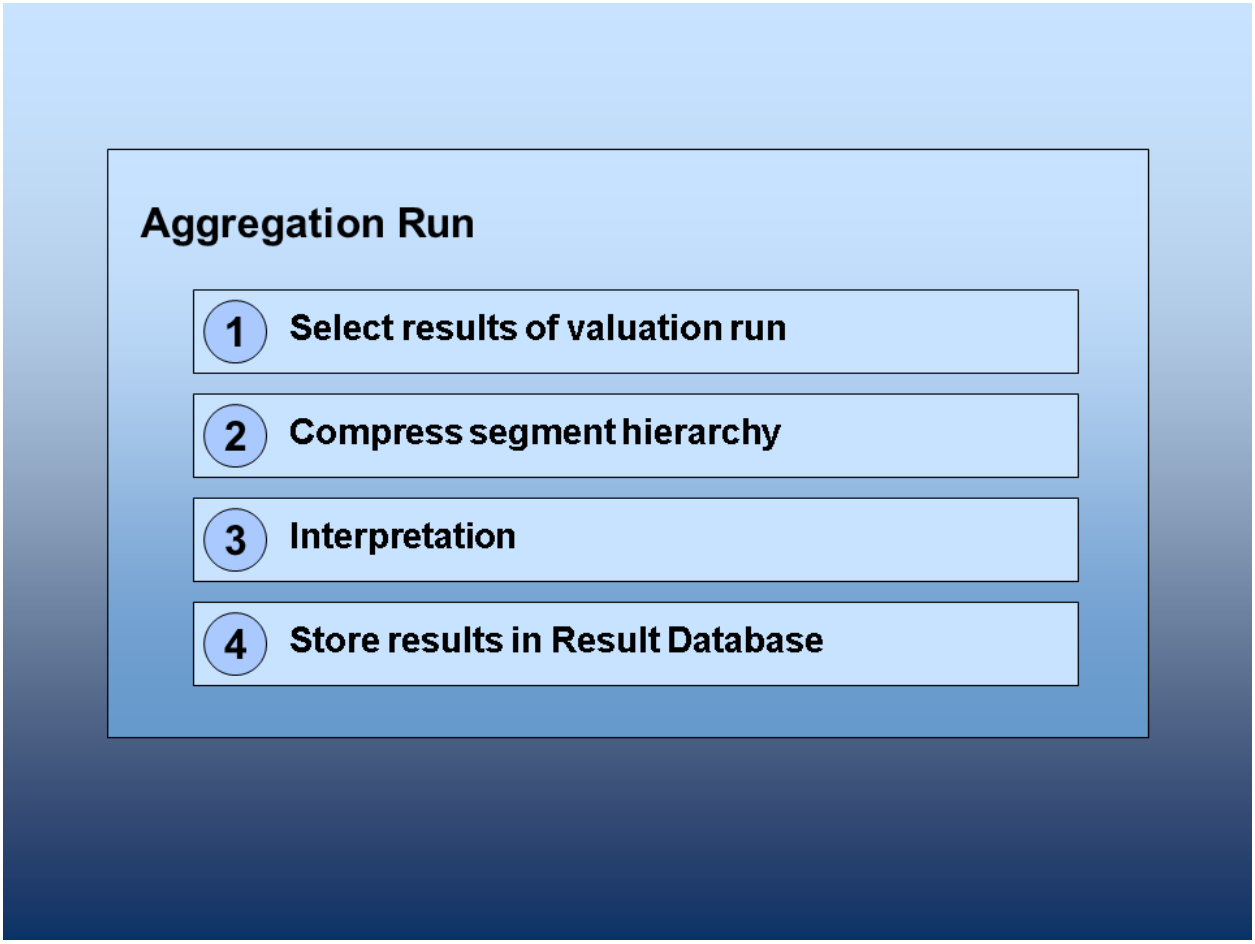
Prerequisites

- You have defined run types for the aggregation run in Customizing for Cash Flow Hedge Analysis.
  You define run types in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Processes* > *Edit Aggregation Run Type*.

- You have created a valuation run [page 1128] and scheduled this run in Schedule Manager [page 1128].
- You have created an aggregation run [page 1129] in the application and scheduled this run in Schedule Manager [page 437].

> **Note**
>
> Instead of scheduling the runs in Schedule Manager, you can start the cash flow hedge analysis as an online analysis [page 1149]. In the online analysis, the system calls the valuation run and the aggregation run in direct succession. The results are kept in the main memory; they are not saved in the Result Database (RDB).
>
> You can start the aggregation run from within the area menu. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Run Administration* > *Create and Schedule* > *Start Aggregation Run*.
>
> Note that the runs that you start from the area menu use work processes. In certain circumstances, this can lead to a shortage of system resources, and delay the start of the runs that you have scheduled in Schedule Manager.

Process



1. The system selects the results of the valuation run.
2. The system aggregates the transactions along the segment hierarchy.
   At the five lowest levels, the transactions are aggregated along the following characteristics, and in the order given below:
   1. Currency
   2. Reference interest rate characteristic
   3. Hedging relationship
   4. Indicator (hedged item, hedging instrument)
   5. Transaction number (for hedging instruments)

   The segment hierarchy is displayed in the dialog structure in reporting. In this dialog structure, you can navigate through the characteristics down to the hedged items and hedging instruments.
3. The system prepares the aggregated results in such a way that a complete result is available for each maturity band date in reporting.
   Before this, for performance reasons the system stored the results in compressed form only, or calculated only interim results for the final key figures.
4. The system stores the final results in the Result Database (RDB).

Result

The system has analyzed the selected hedging relationships and aggregated them along the segment hierarchy. It has also stored the results in the Result Database (RDB).

You can display the results in reporting [page 1140].

4.6.3.2 Run Administration

Definition

Run administration includes the following functions:

- Execute or create run
- Display an overview of runs
- Display Application Log
- Edit run
- Manage run
- Replace run
- Select run for archiving
- Delete run
- Log of the deletion function

> **Note**
>
> The above functions are not all available for each application. For more information, see the application-specific documentation.

Use

The following table lists the runs available for each application:

Application	Run
General Calculation and Valuation Methods [page 376]	Account Pooling [page 383] or Account Pooling (CVPM) Facility Distribution [page 399] or Facility Distribution (CVPM) Determination of the Free Line for Facilities [page 395] or Determination of the Free Line (CVPM) Collateral Distribution [page 404] or Collateral Distribution (CVPM) Determination of Default [page 410] Stress Runs [page 429]: • Stress run for account pooling or stress run for account pooling (CVPM) • Stress run for facility distribution or stress run for facility distribution (CVPM) • Stress run for determination of the free line or stress run for determination of the free line (CVPM) • Stress run for collateral distribution or stress run for collateral distribution (CVPM) • Stress run for determination of default
Credit Risk [page 1191]	Credit Exposure Run [page 1238] or Credit Exposure Run (CVPM) Country Risk Run [page 1336] (prototype for country risk) **Stress run:** Credit Exposure Stress Run [page 1261] or Credit Exposure Stress Run (CVPM)

Application	Run
Historical Database [page 1478]	**Version management:** Run for generating historical data for data layers [page 1481] Run for generating historical data for the bank's own models [page 1484] Uploading of Files [page 1486] **Calculation functions:** Determining Default Rates [page 1488] Determining Average Default Rates [page 1491] Determining Default Figures [page 1493] Calculation of Migration Matrices [page 1496] **Data retrieval:** Run for supplying data to models [page 1498] Downloading of Files [page 1500] **Stress tests:** Stress Run for Supplying Models with Data [page 1505] Generating Scenario Data in the Source Data Layer [page 1506]
Generic BI Data Extraction [page 1638]	Testing the BI Extractor [page 1639] BI extraction run **Note** Extraction runs are created and executed in SAP NetWeaver Business Intelligence (BI). The system displays information about extraction runs in run administration of Bank Analyzer.
Regulatory Reporting Interface	Data Extraction Runs
Limit Manager [page 1529]	Limit Utilization Run [page 1551]
Fair Value Effectiveness Test for Hedging Relationships [page 1036]	Effectiveness Test Run
Cash Flow Hedge Analysis [page 1107]	Valuation Run [page 1128] Subvaluation Run Aggregation Runs [page 1129]

Application	Run
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169] Portfolio Item Runs [page 1172] • Period-end run for portfolio items (subledger scenario) • Period-start run for portfolio items (subledger scenario) • Portfolio item run (merge scenario)

> **Note**
>
> For some of the Bank Analyzer components, you can use the Schedule Manager to schedule and control jobs. If you use multiple applications, you can define the sequence in which the runs are to be carried out. For more information, see Schedule Manager [page 437].

More Information

Status Overview for Run Administration [page 1248]

4.6.3.2.1 Creating Valuation Runs

Prerequisites

You have defined run types for the valuation run and the subvaluation runs in Customizing for Cash Flow Hedge Analysis.

You define run types in Customizing for Bank Analyzer under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Processes* > *Edit Valuation Run Type and Subvaluation Run Type*.

> **Note**
>
> You define in the valuation run itself whether the system is to use the hypothetical derivative approach.

Context

In the valuation run [page 1111], the system analyzes the hedging relationships or the hedged items and hedging instruments of these relationships. The second process in Cash Flow Hedge Analysis [page 1107], the aggregation run [page 1123], is based on the valuation run.

Procedure

1. On the SAP Easy Access screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Run Administration* > *Create and Schedule* > *Start Valuation Run*.

   The *Create Run* selection screen appears.
2. Enter the run type for the valuation run and the required key date.
3. Choose *Execute*.

Results

The system creates a version for the required valuation run. The system always forms the name of the valuation run from the name of the valuation run type that you selected. To distinguish between valuation runs with the same run type and key date, the system also assigns a version number (1, 2, and so on).

You can now create the aggregation run [page 1129] for the valuation run.

You schedule the valuation run in Schedule Manager [page 437], and can display the run along with its status in run administration [page 1133].

> **Note**
>
> As an alternative to scheduling the runs in Schedule Manager, you can start the process as an online analysis [page 1149]. In the online analysis, the system calls the valuation run and the aggregation run in direct succession. The results are kept in the main memory; they are not saved in the Result Database (RDB).

4.6.3.2.2 Creating Aggregation Runs

Prerequisites

- You have defined run types for the aggregation run in Customizing for Cash Flow Hedge Analysis.
  You define run types in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Processes* > *Edit Aggregation Run Type*.
- You have created a valuation run and scheduled this run in Schedule Manager.
  The system cannot start the aggregation run before the related valuation run has finished.

Context

The aggregation run [page 1123] is the second process in Cash Flow Hedge Analysis [page 1107]. In the aggregation run, the system aggregates the results of a valuation run along a segment hierarchy. The segment

hierarchy is used in reporting [page 1140] as the basis for displaying hedging relationships or hedged items and hedging instruments. Each aggregation run is assigned uniquely to one valuation run.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Run Administration* > *Create and Schedule* > *Create Aggregation Run*.

   The selection screen for aggregation runs appears.
2. Enter the name of the valuation run, the key date, and the version of the run, and select an aggregation run type.
3. Choose .

Results

The system creates a version for the required aggregation run. The system forms the name of the aggregation run from a combination of the required valuation run type and the aggregation run type. To distinguish between aggregation runs that belong to the same valuation run and aggregation run type, the system also assigns a version number (1,2, and so on).

You can now schedule the aggregation run in Schedule Manager [page 437] and display it in run administration with its status.

> **Note**
>
> As an alternative to scheduling the runs in Schedule Manager, you can start the process as an online analysis. In the online analysis [page 1149], the system calls the valuation run and the aggregation run in direct succession. The results are kept in the main memory; they are not saved in the Result Database (RDB).

4.6.3.2.3 Schedule Manager

Definition

A tool in Bank Analyzer that you can use to execute and monitor complex business processes, such as period-end closing. When you schedule a run, you always have to specify the program with which the system calls the run. There is a separate program for each run category in an application.

The table below shows for which runs and run categories the Schedule Manager can be used, and their corresponding programs.

Application	Run Category	Program
Account Pooling	Account Pooling [page 383]	/BA1/DE_GM_MAIN_COMP
Account Pooling	Stress Run for Account Pooling [page 429]	/BA1/DE_GM_MAIN_COMP_SCEN
Determination of the Free Line for Facilities	Determination of the Free Line for Facilities [page 395]	/BA1/DE_GM_MAIN_FL
Determination of the Free Line for Facilities	Stress Run for Determination of the Free Line for Facilities [page 429]	/BA1/DE_GM_MAIN_FL_SCEN
Distribution of Facilities	Distribution of Facilities [page 399]	/BA1/DE_GM_MAIN_RLV
Distribution of Facilities	Stress Run for Facility Distribution [page 429]	/BA1/DE_GM_MAIN_RLV_SCEN
Collateral Distribution	Collateral Distribution [page 404]	/BA1/DE_GM_MAIN_SV
Collateral Distribution	Stress Run for Collateral Distribution [page 429]	/BA1/DE_GM_MAIN_SV_SCEN
Determination of Default	Determination of Default [page 411]	/BA1/RA_DFLT4280_RUN
Determination of Default	Stress Run for Determination of Default [page 429]	/BA1/RA_DFLT4280_RUN_SCENARIO
Fair Value Effectiveness Test [page 1036]	Effectiveness Test Run [page 1039]	/BA1/R8_FVHE_MAIN_RUN
Fair Value Effectiveness Test	Generation of Time Series for Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MAIN_RUN
Fair Value Effectiveness Test	Modifying a Time Series for the Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MODIFY
Cash Flow Hedge Analysis [page 1107]	Subvaluation Run [page 1128]	/BA1/R5_CFHA_BTC_RUN
Cash Flow Hedge Analysis	Aggregation Run [page 1129]	/BA1/R5_START_BATCH_AGGR
Cash Flow Hedge Analysis	Joint call of Valuation Run [page 1111] and Aggregation Run [page 1123]	/BA1/R5_EXECUTE_CFHA
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169]	/BA1/R9_VPM_BTC_ID_RUN
Portfolio Fair Value Hedge	Portfolio Item Run (Merge Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_RUN
Portfolio Fair Value Hedge	Period-End Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PE_RUN

Application	Run Category	Program
Portfolio Fair Value Hedge	Period-Start Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PS_RUN
Portfolio Fair Value Hedge	Define End of Valuation Period	/BA1HM/RAH_M_PFVH_SET_VP_END
Portfolio Fair Value Hedge	Adjust Initial Portfolio Fair Value Hedges for Next Valuation Period	/BA1HM/RAH_M_PFVH_ADJUST
Portfolio Fair Value Hedge	(Re-) Designation of Initial Portfolio Fair Value Hedges	/BA1HM/RAH_M_PFVH_DESIG
Credit Exposure [page 1192]	Credit Exposure Run [page 1238]	/BA1/R2_CRA_MAIN_CRE
Credit Exposure	Credit Exposure Stress Run [page 1261]	/BA1/R2_CRA_MAIN_CRE_SCEN
Credit Exposure	Credit Exposure Run for Revolving Retail	/BA1/RR2_CRE_REVRET
Credit Exposure	Approximate Stress Run [page 1274]	/BA1/RR2_CRE_SCENARIO
Credit Exposure	Aggregation Run for Ad Hoc Calculation [page 1276]	/BA1/RR2_SCEN_AGGR
Country Risk [page 1326]	Country Risk Run [page 1336]	/BA1/R2_CRE_MAIN
Historical Database [page 1478]	Historization Run for Data Layers [page 1481]	/BA1/R6_HIST_RUN
Historical Database	Historization Run for Bank's Models [page 1484]	/BA1/R6_HIST_RUN_MODEL
Historical Database	Determine Default Rates [page 1488]	/BA1/R6_CF_DEFAULT_4210
Historical Database	Determine Average Default Rates [page 1491]	/BA1/R6_CF_DEFAULT_4220
Historical Database	Determine Default Figures [page 1493]	/BA1/R6_CF_DEFAULT_KEYFIGURES
Historical Database	Exporting Data to In-House Models [page 1498]	/BA1/R6_MOD_RUN
Historical Database	Calculation of Migration Matrices [page 1496]	/BA1/R6_CF_MIGRATIONS
Historical Database	Stress Run for Supplying Models with Data [page 1505]	/BA1/R6_MOD_RUN_SCENARIO
Historical Database	Stress Run for Supplying Data [page 1506]	/BA1/R6_FDB_RUN_SCENARIO

Application	Run Category	Program
Regulatory Reporting Interface	Processing Framework Run Copying the Start ID	/BA1/S1_COPY_STARTID
Regulatory Reporting Interface	Processing Framework Run Creating Runs	/BA1/S1_CREATE_RUN
Regulatory Reporting Interface	Processing Framework Run Release Run	/BA1/S1_RELEASE_RUN
Regulatory Reporting Interface	Processing Framework Run Creating a Worklist	/BA1/S1_CREATE_WRKL
Regulatory Reporting Interface	Processing Framework Run Processing a Run	/BA1/S1_PROCESS
Regulatory Reporting Interface	Data Extraction Runs	/BA1/S3_EXTRACT_RUN
Limit Manager [page 1529]	Limit Utilization Run [page 1551]	/BA1/R3_UTILIZATION_RUN_DPF

More Information

Schedule Manager

4.6.3.2.4 Managing Valuation Runs

Use

Run administration provides an overview of the valuation runs in Cash Flow Hedge Analysis [page 1107].

The system displays all the important attributes of the runs, including the key date, the creation date, and the current and previous status. You can flag valuation runs for deletion.

Prerequisites

You have created at least one valuation run.

Procedure

Selection

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Run Administration* > *Management* > *Manage Runs*.
   The system displays the initial screen for run administration.
2. Enter selection criteria in order to restrict the display to specific runs.
   You can use the ID of the run, the business key date, the version of the run, and the system status as selection criteria. If you have already displayed runs and have saved a layout as a variant, you can select this as a display variant.
3. Choose *Execute*.
   On an overview screen, the system displays the runs you selected.

Action

- To refresh the list, choose *Refresh List*.
- To display more information about a valuation run or to flag the run for deletion, choose *Display*.
  The system displays detailed information about the selected run.
  - The *Run Parameters* tab page contains information about the run parameters used, such as field instance, maturity band, valuation environment, cash flow set, and scenario set.
  - The *Original RDB* tab page contains information about the run name, the key date, the analyzer of the RDB client, and the version number of the run.

On the detail screen, you can carry out the following actions:

Action	Function
Display Run	The system switches to display mode
Change Run	The system switches to change mode
Set Deletion Flag	The system marks the displayed valuation run for deletion. **Note** This action can be carried out in change mode only. For more information, see Deleting Runs [page 1139].
Display Subruns from CF Hedge Analysis	The system displays an overview of all the subvaluation runs for the valuation run. For more information, see Managing Subvaluation Runs [page 1135].
Field Instance	The system displays an overview of the field instances used. For more information, see Field Instance Display [page 1106].
Status Management	The system displays the current and previous status of the run.

4.6.3.2.5 Managing Subvaluation Runs

Use

Run administration provides an overview of the subvaluation runs in Cash Flow Hedge Analysis [page 1107].

The system displays all the main attributes of the runs, including the key date, the creation date, and the current and previous status of the runs. You can flag subvaluation runs for deletion.

Prerequisites

You have created at least one valuation run [page 1128]. The system has automatically created the relevant subvaluation runs.

Procedure

Selection

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Run Administration* > *Management* > *Manage Subruns*.
   The initial screen for run administration appears.
2. Enter selection criteria in order to restrict the display to specific runs.
   You can use the ID of the run, the business key date, the version of the run, and the system status as selection criteria.
   If you have already displayed runs and have saved a layout as a variant, you can select this as a display variant.
3. Choose *Execute*.
   On an overview screen, the system displays the runs you selected.

Action

- To refresh the list, choose with the quick info *Refresh List*.
- To display more information about a valuation run or to flag the run for deletion, choose with the quick info *Display*.
  The system displays detailed information about the selected run.
  - The *Run Parameters* tab page contains information about the run parameters used, such as field instance, maturity band, valuation environment, cash flow set, and scenario set.
  - The *Original RDB* tab page contains information about the run name, the key date, the analyzer of the RDB client, and the version number of the run.

On the detail screen, you can carry out the following actions:

Action	Function
Display Run	The system switches to display mode.

Change Run	The system switches to change mode.
Set Deletion Flag	The system marks the displayed run for deletion. This action can be carried out in change mode only. For more information, see Deleting Runs [page 1139].
Selection Characteristics	The system displays the selection characteristics that were used to split the valuation run into subvaluation runs.
Error Log	The system displays the error log.
Display CFHA Run	The system displays the valuation run for the subvaluation runs. For more information, see Managing Valuation Runs [page 1133].
Status Admin	The system displays the current and previous status of the subvaluation run.

4.6.3.2.6 Managing Aggregation Runs

Use

Run administration provides an overview of the aggregation runs in Cash Flow Hedge Analysis [page 1107].

The system displays all the main attributes of the runs, including the key date, the creation date, and the current and previous status of the runs. You can flag aggregation runs for deletion.

Prerequisites

You have created at least one aggregation run.

Procedure

Selection

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Run Administration* > *Management* > *Manage Aggregation Runs*.
   The initial screen for run administration appears.
2. Enter selection criteria in order to restrict the display to specific runs.
   You can use the ID of the run, the business key date, the version of the run, and the system status as selection criteria.

If you have already displayed runs and have saved a layout as a variant, you can select this as a display variant.

3. Choose .
   On an overview screen, the system displays the runs you selected.

Action

- To refresh the list, choose .
- To display more information about a valuation run or to flag the run for deletion, choose .
  The system displays detailed information about the selected run.
  - The *Run Parameters* tab page contains information about the run parameters used, such as valuation run type and aggregation run type as well as hierarchy type, segment hierarchy, and maturity band.
  - The *RDB* tab page contains information about the run name, the key date, the analyzer of the RDB client, and the version number.
  - The *Business Settings* tab page is not relevant for cash flow hedge analysis.

On the detail screen, you can carry out the following actions:

Action	Function
	The system switches to display mode.
	The system switches to change mode.
	The system marks the displayed run for deletion. **Note** This action can be carried out in change mode only. For more information, see Deleting Runs [page 1139].
	The system displays the valuation run for the aggregation run.
	The system displays the cash flow view that you specified for the run using the aggregation run type. **Note** Use only cash flow sets with one single cash flow view.
	The system displays the market data scenarios that you specified for the run.

Note

The options , , and are not relevant for cash flow hedge analysis.

4.6.3.2.7 Starting the Correction Run

Use

You use this function to recalculate the results of valuation runs from Cash Flow Hedge Analysis, and to update the existing results.

You normally use the correction run if you have changed hedging relationships, and want to correct the associated valuation run without having to schedule a new batch job. The correction run is an online job that you can start from the area menu.

Prerequisite

You may use the correction run only if the system has already completed the corresponding valuation run.

Features

The correction run updates the results of all the subruns for a valuation run. The following constraints apply:

- Only hedging instruments are recalculated.
- Only those hedging relationships are taken into account that already existed when the original valuation run was created. Therefore, the correction run cannot take into account any hedging relationships that were created after the valuation run but before the correction run was started. This applies also to the corresponding hedging instruments.

Although the system aggregates results of the valuation run as part of the correction run, you still need to carry out an aggregation run to prepare the results for reporting.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Run Administration* > *Create and Schedule* > *Start Correction Run*.
   A selection screen appears.
2. Define the scope of the corrections by entering the following information:
   - Attributes of the run
     Specify the valuation run from Cash Flow Hedge Analysis that you want to correct, and enter the time stamp that the correction run is to use to select the transaction data.
   - Data selection
     Specify whether you want the correction run to include hedging relationships and hedging instruments that were not designated in the original valuation run.
     When you have set the indicators, the system automatically adds the relevant primary data sources.
   - Run control
     Here you define which types of message the correction run is to write to the application log: just messages resulting from any termination of the transaction; success messages; information messages; warnings; error messages.

3. Choose *Execute*.
   The system starts the correction run.

4.6.3.2.8 Deleting Runs

Prerequisites

You have set the deletion flag for the runs you want to delete in run administration [page 430]. You do this as follows:

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Run Administration* > *Manage* > *Manage Runs*, *Manage Subruns*, or *Manage Aggregation Runs*.
2. Select the run that you want to delete and choose .
3. Choose .
   The fact that runs have to be selected for deletion before they can be deleted ensures that runs are deleted only after this has been approved by another user.

Context

You can delete valuation runs and aggregation runs at any time in order to reduce the memory requirement.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Run Administration* > *Delete and Archive* > *Delete Runs*.

   If you have already flagged runs for deletion, the system displays all the runs that are to be deleted on an overview screen.

   Otherwise, a selection screen appears.
2. Enter selection criteria to restrict the runs that are displayed to those you want to delete.
3. Choose .

   An overview screen of all the runs to be deleted appears.
4. You can use the following functions.

Action	Explanation
	On a detail screen, the system displays the run parameters and the administrative data for the selected run.
	The system deletes the selected run.
	The system updates the overview screen. You can track whether the system has deleted runs as required.
	The system displays a log of the deletion process.

4.6.3.3 Reporting

You can use the following functions to display the results of Cash Flow Hedge Analysis:

- Display Batch Analysis Reports [page 1140]
- Display Hedge Proposals [page 1148]
- Carry Out Online Analysis [page 1149]
  Display Reports [page 1140]

4.6.3.3.1 Reporting in Cash Flow Hedge Analysis

Use

Reporting provides you with an overview of the cash flow hedge analysis [page 1107] results.

The system allows you to display the key figures calculated for all levels of the segment hierarchy that you provided for the aggregation run. Hedged items are displayed in aggregated form; hedging instruments are displayed for each single transaction. This enables the cash flow hedges to be documented in detail and the risk reduction to be verified at the level of single transactions.

In reporting, you can simulate new hedging instruments and add existing, non-designated hedging instruments to hedging relationships, or remove them from hedging relationships. You can also generate hedge proposals.

You can display the results of aggregation runs that have already been completed.

Integration

Reporting displays the results of one aggregation run at a time. For online runs [page 1149], the system displays the data from the main memory. For aggregation runs in batch mode, the system selects the data from the Result Database (RDB). You can display the results of aggregation runs that have already been completed at any time.

Once you save the hedge proposals generated in reporting in Cash Flow Hedge Analysis, they are then available in the Hedge Management [page 976] component, where you can display and process the proposals.

If want to use the hypothetical derivative approach in your cash flow hedge analysis, you can call the fair value effectiveness test for hedging relationships [page 1036] from within the reporting function in Cash Flow Hedge Analysis.

Activities

To call up reporting directly, proceed as follows:

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Reporting* > *Result Database for Cash Flow Hedge Analysis*.
   The system displays the initial screen for reporting.
2. Choose the aggregation run for which you want to display results.
3. Choose *Execute*.
   In the dialog structure, the system displays the segment hierarchy [page 1142]. On the tab pages, it displays the key figures [page 1144] that were calculated in Cash Flow Hedge Analysis.
   You can then generate hedge proposals [page 1146] by shifting either whole hedging instruments, or just parts of them, in the segment hierarchy.
   You also have the following options:

Action	Function
Change Scenario	Displays the results data for a different market data scenario. You provide the system with scenarios and scenario progressions in a scenario set. You can define scenario sets in Customizing under *Financial Services* > *Foundation* > *Market Data* > *Market Data Scenarios* > *Edit Scenario Sets*.
Overview Screen	Displays all the calculated key figures.
Detail Screen	Displays a selection of the calculated key figures.
Fictitious Transaction	Creates a transaction in order to simulate a different hedging instrument.
Display Hedge Proposal	Displays the hedge proposal that was just processed. For each hedging relationship, the system displays which hedging instrument you have added or removed, and the amount that was added or removed.

Action	Function
Save Hedge Proposal	Saves the hedging relationships in the Result Database (RDB) and generates a hedge proposal that can be processed further in Hedge Management.
Start Effectiveness Test	Calls the fair value effectiveness test for hedging relationship that uses the hypothetical derivative approach.
Edit > *Display Original Results*	Removes the fictitious transactions from the results data display.
Goto > *Detail Log*	Displays the log for the cash flow hedge analysis.
Goto > *Evaluation Parameters*	Displays the parameters used in the valuation and aggregation runs.
Goto > *Error Messages*	Displays the error log for the cash flow hedge analysis.

4.6.3.3.1.1 Display of the Segment Hierarchy

Use

In the reporting function in Cash Flow Hedge Analysis the system assigns hedging relationships and non-designated hedging instruments to a segment hierarchy. You use the segment hierarchy to choose which results the system is to display. You also use the segment hierarchy to create fictitious transactions, which you then use when you shift hedging instruments and generate hedge proposals.

Prerequisites

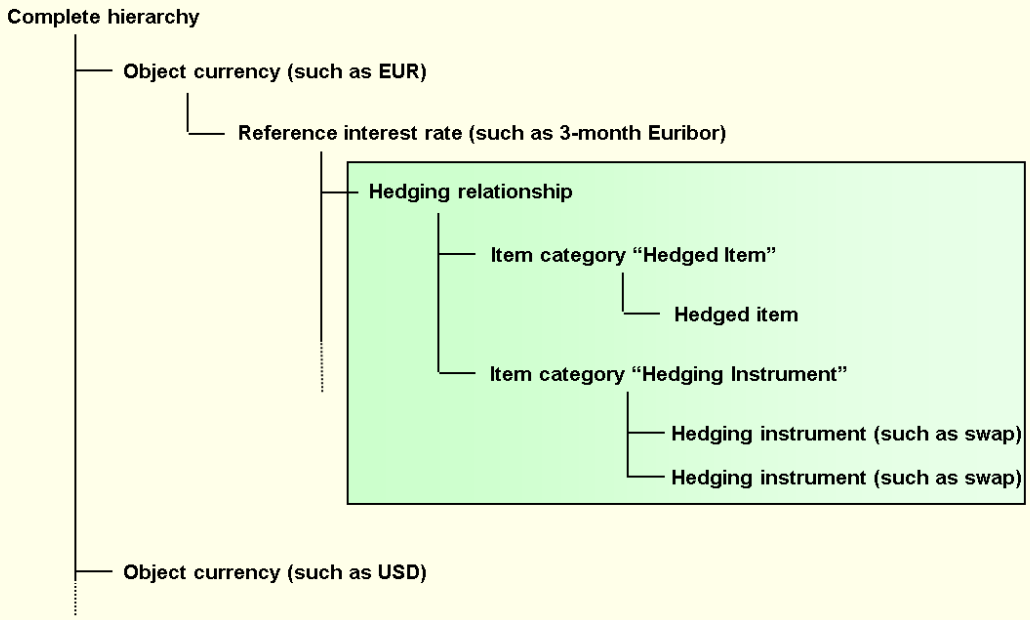
In Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Segment Hierarchies*, you have created at least one segment hierarchy, and under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Processes* > *Edit Aggregation Run Type*, you have assigned your aggregation run type.

> **Note**
>
> The last five levels of the segment hierarchy are reserved for Cash Flow Hedge Analysis. These are the *Currency*, *Reference Interest Rate*, and*Hedging Relationship ID* characteristics, the *Hedged Item* and *Hedging Instrument* item categories, and the transaction level. This reflects the fact that the hedged items and hedging instruments in a hedging relationship should have the same currency, and the hedged items in the hedging relationship should have the same reference interest rate.

To make sure that the hedged items and hedging instructions are not used more than once in a hedging relationship, you have assigned a cash flow filter and valuation rules for the hedged item and hedging instruments to your cash flow hedge environment, You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Valuation* > *Edit Cash Flow Hedge Environment*. For more information, see Cash Flow Filters [page 513] and Removal of Cash Flows Used in Fair Value Hedges [page 1119].

Features

In the reporting in Cash Flow Hedge Analysis, the system displays the segment hierarchy in the right-hand part of the screen. The hedged items and hedging instruments are sorted in the segment hierarchy by their characteristics. When you drill down to transaction level, the segment hierarchy in the dialog structure is displayed as follows:



Note the following about how the system displays the hedged items and hedging instruments:

- Underlying transaction
  The system displays the hedged items in a hedging relationship in aggregated form only. You cannot drill down to transaction level.
- Designated hedging instruments
  You can drill down to the individual transactions upon which designated hedging instruments are based.
- Non-designated hedging instruments
  You can drill down to the individual transactions upon which non-designated hedging instruments are based. Hedging instruments for which there are no values for the characteristics in the segment hierarchy are positioned in the segment hierarchy under their currency and reference interest rate.
  Since the segment hierarchy is predefined and cannot be changed, non-designated hedging instruments are always shown below the *Hedging Relationship* hierarchy level. You can recognize the corresponding nodes by the fact that they do not have IDs.

You can use the following functions.

- To display the key figures from Cash Flow Hedge Analysis [page 1144] for an aggregation level, double click a row in the hierarchy.
- To generate hedge proposals [page 1146], use Drag&Drop to shift the hedging instruments within the segment hierarchy.

4.6.3.3.1.2 Display of Key Figures from Cash Flow Hedge Analysis

Use

In the reporting function in Cash Flow Hedge Analysis, the system displays the key figures that were calculated in the valuation run and the aggregation run. This gives you an overview of the how the hedged items are hedged, and of the effectiveness of the cash flow hedging relationships that were analyzed.

The key figures are displayed differently, depending on which approach is used.

- Reinvestment risk approach
  The system displays key figures for the hedged items and the hedging instruments of the hedging relationships that were analyzed.
- Hypothetical derivative approach
  The system displays key figures for the hedged items of the hedging relationships analyzed, and compares them to the key figures for the hypothetical derivative that was generated. The key figures for the hypothetical derivative are shown in the *Hedging Instrument* columns.

The system displays the results as aggregated values. The granularity is governed by the maturity band, as the system aggregates the nominal values and the flows for each section of the maturity band. You can also influence the granularity by choosing in the navigation structure the level of the segment hierarchy [page 1142] for which the system is to display the results.

Prerequisites

- In Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Valuation* > *Edit Cash Flow Hedge Environment*, you have defined how the system is to treat the nominal values and interest cash flows of variable-rate hedged items, and the nominal values of hedging instruments. For more information, see Analysis of Variable Rate Hedged Items and Hedging Instruments [page 1121].
- In Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Valuation* > *Enter Settings for Short Term Category*, you have defined which transactions belong to the short-term category. You assign transactions to the short-term category by means of their maturity or residual maturity. The short-term category is used in reporting only, so that by assigning it, you influence only how results are displayed.
- In Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Valuation* > *Edit Cash Flow Hedge Environment*, you have defined whether the system is to display

the hedge volume for each maturity band date, or the average nominal value for the maturity band section. You make these settings by means of the *Key Date Position/Average Position* checkbox.

Features

The system displays the following tab pages:

- Overview
- Asset Side
- Liability Side

The *Asset Side* and *Liability Side* tab pages contain all the key figures that were calculated, and enable you to analyze the hedging relationships in detail. The *Overview* tab page displays the total exposure and a selection of the key figures from the other tab pages. The display is designed so that the *Overview* tab page provides a detailed insight into the effectiveness of the hedging relationships that were analyzed.

Key Figures Shown on the *Overview* Tab Page

Key Figure	Definition
Total of the incoming or outgoing payments	Total of the incoming or outgoing cash flows (total of the following categories of cash flow: fixed-rate transactions, short-term category, variable capital flows, and variable-rate flows).
Hedge volume	Nominal volume of the hedging instruments for a given section of the maturity band. Depending on the setting you made for the *Key Date Position/Average Position* checkbox, the system displays either the hedge volume for the maturity band date or the average hedge volume across the maturity band section.
Hedge portion	Quotient of the average hedge volume divided by the total of the incoming payments and interest rate adjustments. If the hedge portion is greater than 1, then the transaction is overhedged.
Forward rate	Reference interest rate on the maturity band date. For dates after the evaluation date, the system uses the specified market data scenarios to determine the interest rates.
Net present value (NPV) of hedging instruments	NPV of the hedging instruments on the maturity band date. This value is used to monitor the quality of the hedging instruments. For more information about the calculation of key figures, see Price Calculator [page 547].

Change in the NPV of hedging instruments	Difference between the NPV of the hedging instruments for the current maturity band interval and that for the previous maturity date interval. This value is a measurement of the NPV risk of the hedging instruments.
Total exposure	Total of the incoming payments and the interest rate adjustments minus the outgoing payments and interest rate adjustments.

Additional Key Figures on the *Asset Side* and *Liability Side* Tab Pages

Key Figure	Definition
Incoming payments from fixed-rate transactions (or outgoing payments)	Interest and capital cash flows of the fixed-rate hedged items.
Incoming payments in the short-term category (or outgoing payments)	Cash flows from hedged items that are in the short-term category due to their maturity or residual maturity, and which must therefore be shown separately.
Variable capital flow	Cash flows of the hedged items that are calculated on the basis of market data scenarios and forward rates.
Interest adjustments	Nominal value of the variable-rate hedged items if the rates were fixed in the previous section of the maturity band. This is a measurement of the reinvestment risk that is associated with variable-rate transactions.
Incoming (or outgoing) payments and adjustments	Total from the interest rate adjustments and the total of the incoming payments. This total represents the actual risk of the hedged items.
Current rate	Reference interest rate on the evaluation date.
Accrued forward interest	Product of the forward rate, incoming payments, and interest rate adjustments. This product represents the expected future cash flows.
Hedged interest	Product of the accrued forward interest and the hedge portion.

4.6.3.3.1.3 Generation of Hedge Proposals

Use

As well as allowing you to display the results data, the reporting function in cash flow hedge analysis enables you to simulate the impact of a new transaction and to move hedging instruments for test purposes. You can save

changed hedging relationships and generate hedge proposals that you can then process in Hedge Accounting Workplace [page 976].

Prerequisites

If you combine cash flow hedge analysis with the hypothetical derivative approach, you can call the fair value effectiveness test from the reporting function. To do this, you must have created a hypothetical derivative environment in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Valuation* > *Edit Environment for Hypothetical Derivative Approach* and assigned it to your valuation run type under *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Processes* > *Edit Valuation Run Type*.

Activities

Create Fictitious Transactions

1. Double-click the hierarchy level for which you want to create a fictitious transaction.
2. Choose *Fictitious Transaction*.
   Note that you can create fictitious transactions only if you have selected a hierarchy level that is below the currency node in the segment hierarchy.
3. Define the fictitious transaction by entering a nominal amount, currency, transaction start and end, and stating whether it belongs to the asset or liability side.
   You can create swaps and forward rate agreements (FRA) as fictitious transactions. When you create a swap as a fictitious transaction, it is always the variable leg that is meant.
   The system displays the effect of the fictitious assignment in the result display.

Move Hedging Instruments

1. In the segment hierarchy, open the directory that contains the hedging instrument that you want to move.
2. In the segment hierarchy, open the hedging relationship to which you want to assign the hedging instrument.
   For the target directory, open the hedging relationship folder with the item category hedging instruments.
3. Move the hedging instrument to the target directory, using drag and drop.

   > **Note**
   >
   > To remove hedging instruments from a hedging relationship, choose the folder for the item category *Hedging Instrument* for an empty hedging relationship (no ID specified) as the target directory.
   >
   > The system opens a dialog box and displays the amount that you can move from the hedging instrument. This amount is either the entire nominal volume of the hedging instrument, or just a part of it (if the hedging instrument has already been partly assigned elsewhere).

4. Choose the share of the hedging instrument you want to move.
   You can specify this portion of the hedging instrument as an absolute amount or a percentage rate of the available amount.
   The system displays the effect of the new assignment in the result display.

Start Effectiveness Test

Choose *Start Effectiveness Test* to run a fair value effectiveness test in the hypothetical derivative approach.

The system checks the designated hedging instruments to determine how they influence the effectiveness of the hedging relationship.

> **Caution**
>
> You can run an effectiveness test only if the cash flow hedge analysis was carried out using the hypothetical derivative approach.

Save Hedge Proposal

Choose *Hedge Proposal*.

The system saves the hedging relationships in their current form in the Result Database (RDB). A new version is created.

Note that you have to remove any fictitious transactions before you save the data. To do so, choose *Edit* > *Display Original Results*.

> **Note**
>
> To display hedge proposals, choose *Display Hedge Proposal* in the reporting function for Cash Flow Hedge Analysis. You can display the hedge proposals directly on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Reporting* > *Display Hedge Proposal*.
>
> For more information, see Display of Hedge Proposals [page 1148].

4.6.3.3.2 Display of Hedge Proposals

Use

You use this function to display the hedge proposals that you generated in the reporting function for Cash Flow Hedge Analysis. In that function, you have usually changed multiple hedging relationships by shifting the hedging instruments that were already assigned, or by assigning to a hedging relationship hedging instruments that were not already in a hedging relationship.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Reporting* > *Display Hedge Proposals*.
   A selection screen appears.
2. Enter the ID of the aggregation run for which you want the system to display hedge proposals.
3. Specify the version of the hedge proposals that you want to display.

4. Choose *Execute*.
   The system displays a list containing all the hedging instruments that were shifted, from which hedging relationship they were removed, and to which hedging relationship they were then assigned.
   If you have used the hypothetical derivative approach in your cash flow hedge analysis, and then carried out a fair value effectiveness test, you can display these results as well by choosing Choose *Results of Effectiveness Test*.

4.6.3.3.3 Using the Online Analysis of Cash Flow Hedges

Use

You can use the online analysis to analyze primary objects, aggregate the valuation results, and to display the aggregated results in one step. This is useful, for example, to test the cash flow hedge analysis process using a small dataset.

In the cash flow hedge analyses run in batch mode, the system saves the results in the database, whereas in the online analysis, it stores the results in the main memory. This has the following consequences:

- You **cannot** access the results at a later date/time.
- In reporting [page 1140], you can still redesignate hedging instruments for test purposes and create fictitious transactions, but you **cannot** save the result.

The system does not manage runs in the online analysis function. This means that you can use the online analysis function without having to have already carried out valuation runs and aggregation runs. Furthermore, the system does not read the Customizing settings in the online analysis. This means that you have to enter all the parameters required on the initial screen that would otherwise normally be stored in Customizing.

> **Note**
>
> For more information, see Calling and Processing a Cash Flow Hedge Analysis [page 1110], Creating Valuation Runs, and Creating Aggregation Runs.

Procedure

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Reporting* > *Online Analysis of Cash Flow Hedges*.

The system displays a screen that has the following tab pages:

- **Valuation settings**
  Enter the key date, and choose the field instance, a scenario set, a cash flow set, a cash flow hedge environment, a maturity band, plus the hierarchy type, the segment hierarchy, and, if required, a market data scenario.
  If you want use the hypothetical derivative approach in your cash flow hedge analysis, set the *Use Approach for Hypothetical Derivatives* checkbox.
  To display the related Customizing settings, and to edit them if required, choose *Detail*.

- **Data sources**
  Enter the time stamp and primary data sources for data selection. To restrict the amount of data selected using characteristic values, choose *Use Selection Criteria*.
  Assign primary and secondary data sources for the analysis of the transactions.
- **Logging**
  Choose whether the system is to display success, information, warning, and error messages in the application log.

> **Note**
>
> You can save your settings as a variant, and use them again later. To save the settings, choose *Save as Variant*. To reuse the variant you saved, choose *Get Variant*.

Choose *Execute*.

This takes you to reporting.

4.6.3.4 Tools

4.6.3.4.1 Display of Field Instances

Use

Field instances contain all the characteristics used internally by the system, and are used as the basis for valuation structures, which the system generates automatically in line with the settings in the Customizing for the fair value effectiveness test [page 1036] . Therefore it is helpful to be able to check the status of field instances when you are using the system operationally.

Integration

You create field instances in Customizing and assign them to the run type for the fair value effectiveness test. You have to activate field instances before you can use them. You also do this in Customizing. Once field instances have been activated, the system automatically generates the package type required for the evaluations to save results in the Result Database (RDB). When a field instance is activated, the system always uses the Customizing settings that are valid at this particular point in time. Because you can change the Customizing settings for the fair value effectiveness test at any time, the properties of the field instances used must be checked while the system is in operation.

You can find the IMG activities for creating and activating field instances in Customizing for *Bank Analyzer* by choosing *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Configuration* > *Field Instance* .

Activities

On the *SAP Easy Access screen* choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Tools* > *Display Field Instances.*

4.6.3.4.2 Display of Secondary Data Sources

Use

The fair value effectiveness test [page 1036] uses secondary data sourcesto select transaction data and derive the test rule that defines how the fair value effectiveness is tested for the transaction in question. In order to check the settings for the secondary data source while the system is in operation, you can display the secondary data source and with the selection modules. This enables you to track which transactions have been selected, and see how the system prepared the data for the tests.

Prerequisites

You have configured the secondary data source in the Customizing for the fair value effectiveness test. You can find the relevant IMG activities in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Processes* > *Secondary Data Source* .

Activities

On the *SAP Easy Access screen* , choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Fair Value Effectiveness Test* > *Tools* > *Edit Secondary Data Sources.*

4.6.3.5 Current Settings

You can change the following Customizing settings in your operational system:

- Edit Maturity Band [page 1152]
- Edit Due Date Scenarios [page 1154]
- Edit Scenarios and Scenario Progressions
  To set up scenarios, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Current Settings* > *Edit Scenarios*
  To set up scenario progressions, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Current Settings* > *Edit Scenario Progressions*.

For information about other functions, see in the Source Data Layer (SDL) documentation.

4.6.3.5.1 Editing Maturity Bands

Use

The system uses maturity bands as a period pattern for calculating and displaying results data.

From a technical point of view, maturity bands are lists of date values. You can define maturity bands as absolute or relative. For absolute maturity bands, you specify the first maturity band date. For relative maturity bands, the system uses the evaluation date as the first maturity band date. You can enter the subsequent date values in a maturity band directly, or define them by specifying the distance between two sequential date values, or by specifying the number of date values.

The system uses maturity bands in Cash Flow Hedge Analysis, for example. In the case of portfolio fair value hedging relationships, you cannot use any absolute maturity bands. You can only use relative maturity bands with a constant number of time buckets.

Procedure

In Customizing for *Bank Analyzer* choose *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Create Maturity Band*.

> **Note**
>
> You can also define maturity bands in the area menu. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Current Settings* > *Edit Maturity Band*.

In the left-hand part of the screen, the system displays a navigation structure that contains the maturity bands that have already been created.

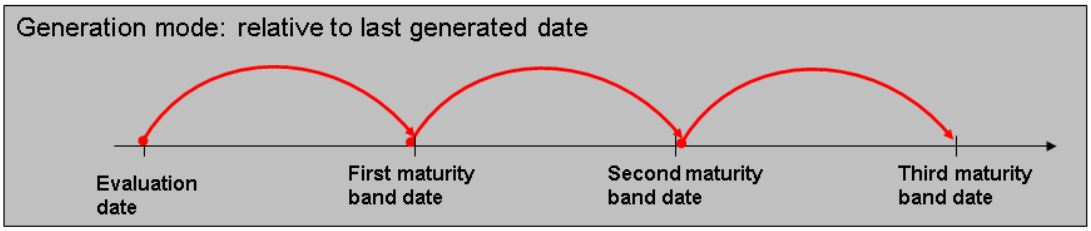
You create a maturity band as follows:

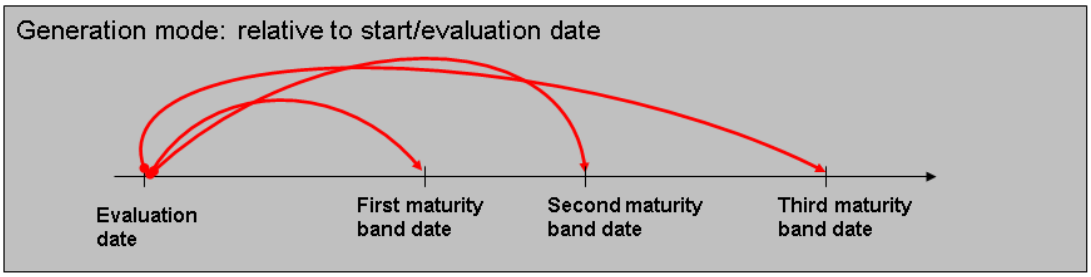
1. Choose  .
2. Enter a key to identify your maturity band.
3. Enter an appropriate short and long name for the maturity band.
4. If you want to create an absolute maturity band, enter a start date.
5. Choose a generation mode. The system distinguishes between the following generation modes:
   - *Enter dates individually*
   - *Relative to last generated date*
     The system uses the last maturity band date that was calculated as the starting point for the calculation of the next date value.
   - *Relative to start/evaluation date*

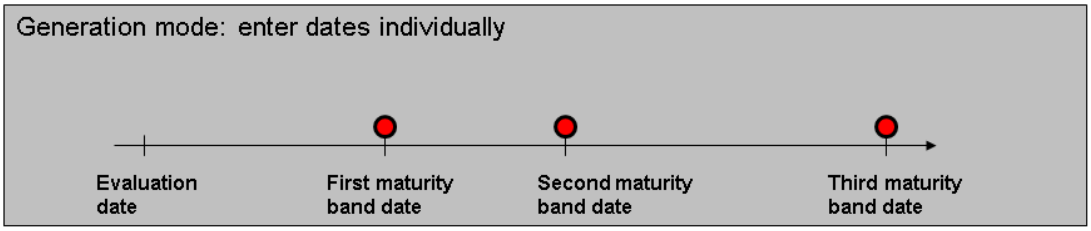
The system takes the start date or evaluation date as the starting point for the calculation of the next date value in the maturity band.

6. In the parameter table, enter details for your maturity band.
   - If you selected the value *Enter Dates Individually*, then you enter the date values here.
   - If you have chosen a different generation mode, then enter the following parameters: time unit, increment, number of date values for this time unit, and the time period for filling the date values.

   The following figure shows how the system generates maturity bands for each of the generation modes.







7. Choose .

You can use the following functions.

Action	Function
	The system switches to change mode or display mode.
	The system displays the parameters of the maturity band selected.
	The system opens a screen in which you can create a new maturity band.

Action	Function
	The system displays the date values of the maturity band selected.
	The system opens the screen for creating a new maturity band, and copies over the parameters of the maturity band that was selected.
	The system resets the last changes that were made.
	The system deletes the maturity band that was selected.
	The system displays the current status of the navigation structure.
	The system opens a screen in which you can choose the maturity bands that you want to transport.
Utilities > *Change Requests (Organizer)*	The system displays your transport requests for Customizing transports.

4.6.3.5.2 Editing Due Date Scenarios

Use

You use due date scenarios to simulate the maturity of account balances. The settings that you enter for due date scenarios cannot normally be based on a theory, so they should therefore reflect your experience regarding how balances change over time.

The system distinguishes between absolute and relative due date scenarios. For absolute due date scenarios, you specify each maturity date. For relative due date scenarios, the maturity dates are relative to the evaluation date. You use due date periods to define how the system finds the maturity dates during the evaluation, and how it simulates the associated maturities. For each due date period, you can specify the distance between each maturity date, and the number of maturity dates, plus the percentage change in the balances on these maturity dates. For relative due date scenarios, the system takes the evaluation date as the starting point for the determination of the maturity dates; the evaluation date is usually not the first maturity date.

Procedure

In Customizing for *Bank Analyzer*, choose *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Basic Settings for Methods* > *Edit Due Date Scenario*.

> **Note**
>
> You can also define due date scenarios in the area menu. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Cash Flow Hedge Analysis* > *Current Settings* > *Edit Due Date Scenarios*.

In the left-hand part of the screen, the system displays a navigation structure that contains all the due date scenarios and due date periods that have already been created.

To create **due date scenarios** proceed as follows:

1. In the navigation structure, select the *Due Date Scenarios* folder and choose *Create*.
2. Enter a three-character abbreviation to identify your due date scenario.
3. Enter an appropriate short and a long name for your due date scenario.
4. Enter the dates for the due date scenario to define whether the due date scenario is absolute or relative.
   - Absolute due date scenario: *Fixed Maturity*
   - Relative due date scenario: *Maturity Relative to Evaluation Date*
5. Enter the parameters for the due date scenario.
   - For absolute due date scenarios, enter one or more due dates and the percentage change in the balances.
   - For relative due date scenarios, proceed as follows:
     1. Choose the *Basic Data* tab page, and enter the start date and end date for your due date scenario, and the base amount as a percentage.
        If you do not enter a date for the start of the due date scenario, the system takes the evaluation date as the start date for the calculation of the maturity dates.
        If you do not enter a base amount, the system uses the value 0%.
     2. Choose the *Settings* tab page and then *Insert Row*. Enter one or more due date periods, and the number that defines how many times the system is to apply due date periods within the due date scenario. Number the rows to specify the order in which the system is to use the due date periods.
        You can use due date periods only if you have created them in the navigation structure in the left-hand part of the screen.
   - Save your entries.

To create **due date periods**, proceed as follows:

1. In the navigation structure, select the *Due Date Periods* folder and choose *Create*.
2. Enter a four-character abbreviation to identify the due date period.
3. Enter an appropriate short name and long name for your due date period.
4. Choose *Insert Row*.
5. Number each due date scenario sequentially, and choose an increment, a time unit, and specify the change in percentage.
   If you enter a positive value for the percentage change, this reduces the balances. If you enter a negative value, this increases the balance.
6. Save your entries.

You can use the following functions.

Action	Function
	The system switches to change mode or display mode.
	The system displays the parameters of the due date scenario selected.
	The system opens a screen in which you create a new due date scenario or due date period.
	The system displays the date values of the due date scenario selected.
	The system opens a screen in which you create a new due date scenario or due date period, and it copies over the parameters of the due date scenario or due date period selected.
	The system resets the last changes that were made.
	The system deletes the selected due date scenario or due date period.
	The system displays the current status of the navigation structure.
	The system opens a screen on which you can choose the due date scenarios that you want to transport.
Utilities > *Change Requests (Organizer)*	The system displays your transport requests for Customizing transports.

Example

On the evaluation date, the balance of an account is EUR 40,000, the nominal interest rate for borrowing is 2.2% p.a. Based on your experience, you could set up a due date scenario in which 40% matures in the first month, 30% in the second month, 20% in the third month, and 10% in the fourth month. The base amount is 20% of the nominal volume on the evaluation date (EUR 8,000).

4.6.4 Portfolio Fair Value Hedge (FS-BA-PM-HP-PFV)

Use

This component is used in the processing of portfolio fair value hedging relationships. Since these hedging relationships usually have a large portfolio of hedged items, the fair value effectiveness test and Hedge Accounting

do not process them as single transactions. Instead, they process portfolio items [page 1158], which have a lower granularity (are less detailed).

The Portfolio Fair Value Hedge component contains functions that you use to calculate and manage portfolio items, and to calculate the *hedge adjustment* key figure for portfolio fair value hedging relationships.

Integration

The functions of the Portfolio Fair Value Hedge component are used in the following components:

- Hedge Accounting Workplace [page 976]
  Calculation of portfolio items in preparation for the designation and redesignation of portfolio fair value hedging relationships
- Fair Value Effectiveness Test for Hedging Relationships [page 1036]
  Calculation of the hedge adjustment in the effectiveness test
- Accounting Processes [page 693]
  Calculation of the hedge adjustment for Hedge Accounting

The processes in the Portfolio Fair Value Hedge component call the Fair Value Server [page 628]. The results are stored in the Result Database (RDB).

Features

The system contains the following runs for the calculation of portfolio items [page 1160]:

- Initial generation run [page 1169]
- Mapping of hedging instruments [page 1171]
- Portfolio item runs [page 1172]
  - Period-end run for portfolio items (subledger scenario)
  - Period-start run for portfolio items (subledger scenario)
  - Portfolio item run (merge scenario)
- Simulation run for portfolio item run [page 1175]

The mapping of hedging instruments and the simulation runs are online runs, and their results are not stored permanently. In contrast, the results of the initial generation run and the portfolio item runs are stored in the Result Database (RDB). You can display the status of these runs and their results in Managing Runs [page 1188]. You can also archive and delete the runs.

The system uses the portfolio fair value calculator [page 1175] to calculate the portfolio items. This function also calculates the *hedge adjustment* key figure for portfolio fair value hedging relationships. The Portfolio Fair Value Calculator is called by the runs for the calculation of portfolio items, by the fair value effectiveness test, and by the accounting processes.

4.6.4.1 Portfolio Items

Definition

A table that contains the cash flows of a portfolio fair value hedge in an aggregated view for a time bucket. Portfolio items are the basis for all other calculations in Hedge Accounting.

Use

Aggregated amounts, rather than individual transactions, are the basis of hedge accounting for portfolio fair value hedges. The aggregated amounts reflect the portfolio of a hedging relationship for the time buckets of a maturity band. The portfolio items are the basis of this aggregated analysis of hedging relationships. They are recalculated for different processing stages of a valuation period (such as designation, key date valuation).

Structure

Each row of portfolio items contains the following information:

- ID of the time bucket
- Start and end of the time bucket
- Amount of the qualified assets
- Amount of the non-qualified assets
- Amount of the qualified liabilities
- Amount of the non-qualified liabilities

Each financial transaction and financial instrument in the portfolio of the hedging relationship contributes to one of the amounts. Whether the transaction is an asset or a liability governs to which amount it contributes. Also, whether the transaction is flagged as qualified or unqualified in the derivation of the validity [page 1165] determines the amount to which it belongs. Only qualified positions can be used as amounts that are to be hedged. The system calculates the amount of qualified and unqualified positions when it maps the cash flows to time buckets [page 1168]. In the mapping process, the system makes the cash flow structure of the transactions less detailed by assigning one portfolio item per time bucket.

The portfolio items are calculated for different process categories. Different process categories are used depending on what the data is required for. For example, the process category is determined by the selection date of the portfolio of hedged items in the hedging relationship and the maturity band that the system uses in the mapping process. The following process categories are used:

- **Processing at the Start of the Period**
  Calculates the portfolio items for the start of the current valuation period
- **Processing at the End of the Period**
  Calculates the portfolio items for the prior valuation period
- **Rollover Processing**
  Calculates the portfolio items that the system needs to update the hedge adjustments when the old valuation period ends and the new valuation period starts.

The process categories have the following parameters:

Process Category	Maturity Band	Date for the Generation of Cash Flows	Selection Date of the Portfolio of Hedged Items	Calculation Date for Portfolio Items
Processing at the Start of the Period	Maturity band of the current valuation period	Start date of the current valuation period	Start date of the current valuation period	End date of the preceding valuation period
Processing at the End of the Period	Maturity band of the prior valuation period	End date of the prior valuation period	Start date of the prior valuation period	End date of the preceding valuation period
Rollover Processing	Maturity band of the following valuation period	End date of the prior valuation period	Start date of the prior valuation period	End date of the preceding valuation period

> **Note**
>
> The system contains multiple runs for the calculation of portfolio items [page 1160]. Note that not all runs calculate portfolio items for all process categories.

Example

In a portfolio fair value hedge that contains only one loan as the hedged item, the portfolio item could be as follows:

Time Bucket	Start/end of the time bucket	Qualified Asset	Non-Qualified Asset	Qualified Liability	Non-Qualified Liability
1	1. January 2011; 31 January 2011	0	0	EUR 1,000,000	0

For example, the value of EUR 1,000,000 could be the disbursement of the loan. Depending on the maturity band, further time buckets may occur after this time bucket that contain the interest payments and the repayments, for example.

Usually, portfolios of hedging relationships contain multiple financial transactions, so that the associated portfolio items contain more than just one value. Since the amounts are aggregated amounts, it is usually not possible to break the portfolio items down into single transactions.

4.6.4.2 Runs for the Calculation of Portfolio Items

The Portfolio Fair Value Hedge component contains the following runs for the calculation of portfolio items [page 1158]:

- Initial generation run [page 1169]
- Mapping of hedging instruments [page 1171]
- Portfolio item runs [page 1172] (mass runs)
- Simulation run [page 1175]

The runs are used at different stages in Hedge Accounting, and so have different technical features. The portfolio items that are calculated by the runs are used in the designation and redesignation of hedging relationships in the Hedge Management work center. They are also used in the key date valuation.

The process that is used to calculate portfolio items is largely the same in all runs. For more information, see Calculation of Portfolio Items [page 1163].

> **Note**
>
> You must select the maturity band that you use to calculate the portfolio items in such a way that the key date of the next key date valuation is in the first time bucket of the maturity band.

For more information about the hedge accounting process for portfolio fair value hedges, see Portfolio Fair Value Hedges [page 1005] in the Hedge Accounting Workplace documentation.

Business and Technical Features

The following table gives you an overview of the features of the individual runs, and the differences between the runs.

Business Features of the Runs

Name of the Run	Call-Up Point of Run	Purpose
Initial generation run	• When hedging relationships are created • After the portfolio of hedged items has been defined	Calculation of the portfolio items before the hedging instruments are assigned
Mapping of hedging instruments	• At designation and redesignation • When the hedging instruments are assigned	• Estimation of the hedged amount • Preparation for the prospective effectiveness test

Name of the Run	Call-Up Point of Run	Purpose
Portfolio item run (mass run; merge scenario)	• Key date valuation • Before the fair value effectiveness test and before the calculation of the hedge adjustment	• Preparation for the calculation of hedge adjustments • Preparation for the fair value effectiveness test • Preparation for redesignation
Period-end run (mass run; subledger scenario)	• Key date valuation • Before the fair value effectiveness test and before the calculation of the hedge adjustment	• Preparation for the calculation of hedge adjustments • Preparation for the fair value effectiveness test • Preparation for redesignation
Period-start run (mass run; subledger scenario)	Before redesignation	• Preparation for the calculation of hedge adjustments • Preparation for the fair value effectiveness test
Simulation run	During the simulation process in the key date valuation	• Preparation for the calculation of hedge adjustments • Preparation for the fair value effectiveness test

Technical Features of the Runs

Name of the Run	Run Category in Run Administration	Processing Mode	Call Type	Number of Hedging Relationships Processed	Portfolio Items Calculated
Initial generation run	PSI	Batch run	Scheduled in Schedule Manager or called manually from the SAP Easy Access screen or in the Hedge Accounting Workplace	One hedging relationship	Period start
Mapping of hedging instruments	-	Online run	Called automatically in the Hedge Accounting Workplace	One hedging relationship	Period start

Name of the Run	Run Category in Run Administration	Processing Mode	Call Type	Number of Hedging Relationships Processed	Portfolio Items Calculated
Portfolio item run (mass run; merge scenario)	MSR	Batch run	Scheduled in Schedule Manager or called manually from the SAP Easy Access screen or in the Hedge Accounting Workplace	Multiple hedging relationships	• Period end • Rollover • Start of the next valuation period
Period-end run (mass run; subledger scenario)	MPE	Batch run	Scheduled in Schedule Manager or called manually from the SAP Easy Access screen or in the Hedge Accounting Workplace	Multiple hedging relationships	• Period end • Rollover
Period-start run (mass run; subledger scenario)	MPS	Batch run	Scheduled in Schedule Manager or called manually from the SAP Easy Access screen or in the Hedge Accounting Workplace	Multiple hedging relationships	Period start
Simulation run	-	Online run	Called automatically from the simulation of key date valuation	One hedging relationship	Period end

Information About Run Names

The system saves the results only of the initial generation run and the portfolio item runs in the Result Database (RDB). In run administration, the system creates a separate run for each hedging relationship, and each run needs a run name.

Different Granularity for Processes and Runs

When you start the initial generation run for the first time, you can choose a name for the run. The system then assigns this run name to the processed hedging relationship. It also uses this run name, together with the key date

of the run, the run category, and the run version, to store the results of the hedging relationship in the portfolio item runs.

A run in run administration corresponds to a hedging relationship. This means that the runs in run administration have a different granularity to that used for the portfolio item runs. In the initial generation run, the system creates the run with a run version run for each hedging relationship. In the portfolio item runs, which usually handle multiple hedging relationships, the system creates a separate version of the run for each hedging relationship that is to be processed. In run administration, you use the run category and the run version to identify the run results.

Restricted Length of Run Names

You can use a run name of up to 25 characters. When you start the initial generation run for the first time, if you do not enter a name for the run, the system tries to use as the run name the ID of the relevant hedging relationship in Hedge Accounting Workplace. Note that the hedging relationship ID can be up to 40 characters long. If the hedging relationship ID has more than 25 characters, the system cannot use it as the name of the run. In this case you must enter a different run name.

However, if you let the system assign hedging relationship IDs in the Hedge Accounting Workplace, then no problems should occur, as the IDs it assigns automatically are 12 characters long.

4.6.4.2.1 Calculation of Portfolio Items

Use

In this process the system calculates the portfolio items [page 1158] for the time buckets of a maturity band. The system calls this process during the runs for the calculation of portfolio items [page 1160].

Prerequisites

You have created a portfolio item environment in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Valuation* > *Edit Portfolio Item Environment*.

The portfolio item environment contains all the parameters that the system needs to calculate portfolio items. So that portfolio items are calculated consistently, you can define the portfolio item environment once only per hedging relationship. You do this the first time you start the calculation of portfolio items when you call the initial generation run [page 1169]. After this, you can no longer change the assignment of the portfolio item environment to the hedging relationship.

Process

1. **Retrieval of the hedging relationships that are to be analyzed**
   In the portfolio item runs, you can enter the hedging relationships to be processed directly on the selection screen or you can select a portfolio item run type, the worklist for which has been entered with the corresponding selection criteria in Customizing using the primary data source for hedging relationships. In the other runs the system processes one hedging relationship only. In the selection screen of the initial generation run you specify the hedging relationship. In the mapping of hedging instruments and in the simulation run, the function that calls the mapping function or simulation run specifies the hedging relationship.
2. **Retrieval of transactions**
   In the initial generation run, the portfolio item runs, and the simulation run, the system uses the information about the portfolio of the hedging relationship to select the transactions.
   In the mapping of hedging instruments, the Hedge Accounting Workplace transfers the relevant transactions.
3. **Generation of cash flows**
   The system generates cash flows for the relevant transactions.
4. **Derivation of the validity**
   For every financial transaction and financial instrument, the system derives the information about whether it contributes to the qualified positions.

   > **Note**
   >
   > In the secondary data source, the information relevant for the evaluation is selected, the cash flows are generated, and the validity is derived. For more information, see Cash Flow Generation and Derivation of Validity [page 1165].

5. Mapping of Cash Flows to Time Buckets [page 1168]
   The system maps the transactions to the time buckets of the maturity band that you defined in the calculation environment of the portfolio item environment.

In Steps 1 and 2, the system generates worklists; in Steps 3, 4, and 5, it processes single records.

Result

The system has calculated the portfolio items for the selected transactions.

When the function is executed from the initial generation run and the portfolio item runs, the system saves the results in the Result Database (RDB). You can display the results of these runs as described in Managing Runs [page 1188].

4.6.4.2.1.1 Cash Flow Generation and the Derivation of Validity

Use

The system uses this function to calculate portfolio items. It does this by selecting data, generating cash flows, filling the valuation structure with data, and deriving whether a financial transaction or financial instrument contributes to the qualified positions or to the unqualified positions.

Integration

The cash flows are generated and the validity is derived in a secondary data source in the *Portfolio Fair Value Hedge* component. You define this data source in Customizing. You can determine the validity using a derivation in the secondary data source.

> **Note**
>
> It makes sense to create separate data sources for financial transactions and financial instruments because these are processed differently.

Input Fields and Output Fields of the Secondary Data Source

When the secondary data source is called, the system fills the input fields with the financial transaction ID, or with the financial instrument and financial position ID, along with any additional data required. You can use the secondary data source to read additional information that the system needs to generate the valuation structure [page 495] and derive the validity of the positions. The output fields of the secondary data source comprise the valuation structure and the validity.

Fields and Structures in the Input, System, and Output Structure

Structure	Field/Structure	Technical Name
Input fields	Financial transaction ID or financial instrument ID	/BA1/C40FTRAN or /BA1/C41FINST
System structure	Key date for the selection of data and cash flow generation	KEYDATE
System structure	System time point for the selection of data and cash flow generation	SYSTEM_TIME
System structure	Market data area of the hedging relationship	MDCODE
System structure	Legal entity of the hedging relationship	LEGENT
System structure	Hedging relationship ID	HEDGE_ID

Structure	Field/Structure	Technical Name
System structure	Table containing the position object IDs of the financial transaction or financial instrument	FP_ID_TAB
Output fields	Validity of a position	/BA1/C57VALID
Output fields	Complex key figure of the risk basis (valuation structure)	/BA1/KX69TSSR

Prerequisites

If you want to use a derivation to determine the validity, you must have defined a derivation strategy for the validity in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Configuration* > *Derivation of Validity*.

In Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Configuration* > *Secondary Data Source* > *Edit Selection Module*, you have defined a secondary data source and added to it your derivation strategy for the derivation of the validity of positions.

> **Note**
>
> If you use financial transactions in addition to financial instruments, it may make sense to create separate secondary data sources for financial transactions and financial instruments.

You have defined a portfolio item environment in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Valuation* > *Edit Portfolio Item Environment*. In the same activity, you have also created primary data sources for the hedging instruments. The entry of a primary data source for hedging relationships is obsolete if you have entered all the required data relating to the hedging relationship in the Hedge Accounting Workplace. Instead, the system reads the detailed information about the hedging relationship via an application programming interface (API).

You have also assigned to all the primary data sources to be used to select financial transactions and financial instruments (where applicable) a secondary data source that is used by the system to further process data.

> **Note**
>
> Make sure that you store the primary data sources for hedging instruments in the portfolio item environment. You store the primary data sources for hedged items in the description of the portfolio of a hedging relationship.

Features

In the secondary data source, the system does the following:

1. It selects the complex key figure for contracts from the Source Data Layer (SDL).
2. It breaks down the contract relationships.
3. It generates cash flows.
4. It maps the complex key figure for contracts to the valuation structure.
5. If required, it adds data to the existing information ready for the derivation of the validity.
6. It derives whether the positions are valid.

Example

The following examples show the steps required. You define these steps in the sequence of secondary data sources for financial transactions and financial instruments. Note that for technical reasons additional settings and steps are usually required.

This means that in the environment for your selection module, you have to define local components in the form of fields and structures. You use these fields and structures to store field content temporarily, and as preparation for the call of function modules.

Secondary Data Source for Financial Transactions

A typical sequence for financial transactions contains the following steps, for example:

1. Primary data source for reading the complex key figure for contracts (S400000X)
2. Function module for breaking down contract relationships (/BA1/FP_API_PO_CRL_REF_GET)
3. Function module for the generation of cash flows for category 40 /BA1/R0_RO_API_CFGEN_CONTR_40)
4. Function module for mapping data to the valuation structure (/BA1/R0_RO_API_MAP_CONTR_40)
5. Derivation of the validity of the positions

Secondary Data Source for Financial Instruments

A typical sequence for financial instruments contains the following steps, for example:

1. Primary data source for reading the complex key figure for contracts (S410000X)
2. Function module for breaking down contract relationships (/BA1/FP_API_PO_CRL_REF_GET)
3. Function modules for adjusting the nominal amount and the asset-liabilities indicator in the complex key figure for contracts (/BA1/FP_API_PO_41_KFE_SET and /BA1/FP_API_PO_41_BAS_SET)
4. Function module for generating cash flows for category 41 financial instrument (/BA1/R0_RO_API_CASH FLOWGEN_CONTR_41)
5. Function module for mapping data to the valuation structure (/BA1/R0_RO_API_MAP_CONTR_40)
6. Derivation of the validity of the positions

Derivation of Validity

You can derive the validity of positions as follows, for example:

- The financial transaction contributes to the qualified positions only if it is classified as *Available for Sale* (AfS) or *Loans and Receivables* (LAR).
- If a financial transaction or financial instrument is already in a micro fair value hedge, it contributes to the unqualified positions, regardless of its holding category.

4.6.4.2.1.2 Mapping of Cash Flows to Time Buckets

Use

You use this function to map the cash flows of the transactions for a portfolio fair value hedging relationship to a maturity band. The system calls this function during the calculation of portfolio items [page 1163]. Mapping reduces the level of detail of the cash flow structure of the hedging relationship.

Prerequisites

In Customizing for *Bank Analyzer* you have created a calculation environment with a maturity band, calculation procedure, and a calculation approach under *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Valuation* > *Edit Portfolio Item Environment*.

The maturity band defines the time bucket to which the system maps the cash flows of the individual transactions.

> **Note**
>
> You use only those maturity bands that are defined relative to the start date or evaluation date.

The parameters calculation procedure and calculation approach are used to determine the mapping process. You define them for each portfolio fair value hedging relationship once only using the portfolio item environment.

Features

The system carries out the following steps:

1. The system selects the cash flows that are to be used.
   You can choose from the following calculation procedures:
   - All flows
   - Only nominal cash flows

   Usually, you configure the system to use the nominal flows option. If this is the case, the pull-to-par effect is hidden, in which the value of a financial transaction becomes tends to revert to the nominal value over time. If you use all the flows (including interest cash flows) in the mapping process, this pull-to-par effect is retained, which could influence the effectiveness measurement for the portfolio fair value hedging relationship.
2. The system calculates the amounts at the end of the time buckets
   You can choose from the following calculation approaches:
   - Simple time bucket assignment
     The system increases the amount for each existing cash flow at the end of the time bucket by the cash flow amount.
   - Mapping conforming with NPV and duration

The system increases the amounts for each existing cash flow at the end of the previous time bucket and the end of the current time bucket so that the following conditions apply:

- The NPV remains the same:

$$NPV\ (CF_{old}) = NPV\ (CF_1) + NPV\ (CF_2)$$

  Where *CFold* is the existing cash flow; *CF1* and *CF2* are the amounts interpreted as cash flows at the end of the time bucket.

- The derivation of the NPVs after the yield remains the same:

$$duration\ (CF_{old}) \cdot NPV\ (CF_{old}) = duration\ (CF_1) \cdot NPV\ (CF_1) + duration\ (CF_2) \cdot NPV\ (CF_2)$$

  Where *duration* is the modified duration.
  To calculate the NPVs, the system calls the price calculator [page 547]. To do this, the price calculator environment and derivation of the market data set are used (both of which are stored in the calculation environment). The derivation takes place at single transaction level.

> **Note**
>
> The NPV-compliant mapping process is more precise than the simple assignment procedure, as the NPV and the sensitivity to the interest rate (that is, the first NPV derivation after the yield) do not change as a result of the mapping.

4.6.4.2.2 Initial Generation Run

Use

You use this function to process the portfolio of hedged items in a portfolio fair value hedge, and to generate portfolio items (see Portfolio Items [page 1158]) in preparation for the initial designation of hedging relationships.

Integration

You start the initial generation run after you have assigned the hedged items to a hedging relationship in Hedge Management. After the initial generation run, in the Hedge Accounting Workplace you can assign hedging instruments to portfolio fair value hedges, and designate the hedging relationships. The system stores the results in the Result Database (RDB). You can display the results in run administration (see Managing Runs [page 1188]).

Prerequisites

In Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Configuration* > *Secondary Data Source* you have created a secondary data source that generates the complex key figure for Risk Basis, and derives the validity.

You have defined a portfolio item environment in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Valuation* > *Edit Portfolio Item Environment*.

In the Hedge Accounting Workplace [page 976] you have assigned hedged items to the hedging relationship and have entered a portfolio item environment on the *Basic Data* tab page.

Features

The system determines which transactions contribute to the qualified and unqualified positions, and calculates the amount of the portfolio items. For more information, see Calculation of Portfolio Items [page 1163] and Management of Hedging Relationships [page 988].

Activities

The initial generation run is an independent process. It processes one portfolio fair value hedge at one time. You can call the initial generation run from the Hedge Accounting Workplace. Alternatively, call it from the *SAP Easy Access* screen or schedule it using Schedule Manager [page 437]. The report name is /BA1/R9_VPM_BTC_ID_RUN.

To start an initial generation run for hedged items:

1. **Subledger Scenario**
   On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Run Administration* > *Create and Schedule* > *Start Initial Generation Run*.
   Merge Scenario
   On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Accounting: Merge Scenario* > *Service Functions for Risk Basis* > *Portfolio Fair Value Hedge* > *Run Administration* > *Create and Schedule* > *Start Initial Generation Run*.
   Alternatively, in the Hedge Accounting Workplace go to the *Basic Data* tab page and choose the pushbutton.
   The system displays a selection screen.
2. Enter the following data:
   - Select the hedging relationship for which you want to generate the portfolio items.
   - Specify the portfolio item environment.
     If, in run administration, you want to use a different name for the hedging relationship ID, enter that name.

> **Note**
>
> The first time you start the process you must enter the run name and the portfolio item environment. You cannot change these parameters after this and you do not have to enter them again if you start this process at a later date.

- You have two processing options:
  - **Test mode**
    The system does not save the results of a test run in the database. Test runs are not run as parallel jobs. The system writes an application log. You do not need to enter a run name for test runs.

    > **Note**
    >
    > You can use test runs to analyze errors.

  - **Update mode**
    In the update mode, the system saves the results of the run in the database. The system starts the run as a parallel job.

3. Choose  .
   The system performs an initial generation run, and saves the results in the RDB. The system versions the runs internally but it always accesses the latest version in the Hedge Accounting Workplace and in the key date valuation.

4.6.4.2.3 Mapping of Hedging Instruments

Use

The system uses this function to calculate the portfolio items [page 1158] for the hedging instruments of a portfolio fair value hedging relationship that is to be designated.

The portfolio items that are calculated for the hedging instruments are used only in the Hedge Accounting Workplace [page 976] to determine the hedge percentage for each time bucket. They make it easier to determine which hedging instruments should be assigned to the portfolio of a hedging relationship. The portfolio items calculated in the mapping of hedging instruments are not used in Hedge Accounting or in the fair value effectiveness test.

Integration

The mapping of hedging instruments is an online run, which you trigger in the Hedge Accounting Workplace. This function is used once you have assigned hedging instruments to a hedging relationship. For this to happen, you need to have already started the initial generation run for designation or the period-start run (general ledger scenario) or portfolio item run (merge scenario) for redesignation.

Once the hedging instruments have been mapped, you can designate or redesignate the hedging relationship.

Prerequisites

In Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Valuation* > *Edit Portfolio Item Environment*, in the portfolio item environment you have assigned a primary data source and a secondary data source for hedging instruments.

Features

The system calculates the portfolio items for the period-start processing for the hedging instruments that you have assigned to the portfolio fair value hedge. For more information, see Calculation of Portfolio Items [page 1163].

Activities

In Hedge Accounting Workplace, you trigger the system to map hedging instruments. Choose the *Update Time Bucket Data* pushbutton on the *Hedging Instruments* tab page. You can start the job as a sequential job as a parallel job. We recommend that you run the job as a parallel job for portfolio fair value hedging relationships that contain a large number of hedging instruments. The system calculates the portfolio items for the hedging instruments, and displays them on the *Portfolio Items* tab page. On this tab page, the system compares the results of the portfolio of hedged items with the results for the hedging instruments.

Hedge Accounting Workplace displays the results of the most recent calculation until the time bucket data is next updated manually.

The system calculates the time bucket data using 100% of the nominal of the hedging instruments. When you call the *Portfolio Items* tab page, the system automatically scales the portfolio items to the nominal that is to be designated. This means that you do not have to map the hedging instruments again.

4.6.4.2.4 Portfolio Item Runs

Use

You use this function to determine the portfolio items of a portfolio fair value hedging relationship. Portfolio items are always required when a portfolio fair value hedging relationship is to be valued, for example during designation, redesignation, or key date valuation.

For more information, see Portfolio Items [page 1158].

Subledger Scenario

The system uses a portfolio item run for the **end** of a valuation period (period-end run) to determine the portfolio items for the portfolio fair value hedging relationship that has already been designated that will be required at the end of a period.

The system uses a portfolio item run for the **start** of a valuation period (period-start run) to determine the portfolio items for portfolio fair value hedging relationships that need to be redesignated at the end of a period.

Merge Scenario

The system uses a portfolio item run to determine the portfolio items for the portfolio fair value hedging relationships that have already been designated.

Integration

Portfolio item runs are mass runs that can process multiple portfolio fair value hedging relationships.

The system stores the results in the Result Database (RDB). You can display the results in run administration (see Managing Runs [page 1188]). In the key date valuation, in the fair value effectiveness test, and in redesignation, the system reads the portfolio items from the RDB, and processes them.

Subledger Scenario

You use the period-end run to prepare the fair value effectiveness test and the key date valuation of portfolio fair value hedging relationships. You use the period-start run to prepare the redesignation of portfolio fair value hedging relationships.

Merge Scenario

You use the portfolio item run to prepare the data for the fair value effectiveness test, key date valuation, and for the redesignation of portfolio fair value hedges.

Prerequisites

You have defined a run type in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Process* > *Edit Run Types for Portfolio Items*.

Features

Subledger Scenario

During a period-end run, the system calculates the portfolio items for the valuation period to be settled in period-end and rollover processing.

During a period-start run, the system calculates the portfolio items for a valuation period in period-start processing.

Merge Scenario

The system calculates the portfolio items for the valuation period changeover in period start processing, period-end processing, and rollover processing. For more information about the calculation of portfolio items, see Calculation of Portfolio Items [page 1163]. For more information about the process categories, see Portfolio Items [page 1158].

Activities

The period-end run (/BA1/R9_VPM_BTC_MR_PE_RUN), the period-start run (/BA1/R9_VPM_BTC_MR_PS_RUN), and the portfolio item run (merge scenario) (/BA1/R9_VPM_BTC_MR_RUN) are separate processes that can be called from the Hedge Accounting Workplace. Alternatively, call these from the *SAP Easy Access* screen or schedule these using Schedule Manager [page 437].

You start the respective portfolio item run as follows:

1. To start the **period-end run**, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Run Administration* > *Create and Schedule* > *Period-End Run for Portfolio Items*.

   Alternatively, in the Hedge Accounting Workplace [page 976] go to the *Basic Data* tab page and choose the pushbutton.

   To start the **period-start run**, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Run Administration* > *Create and Schedule* > *Period-Start Run for Portfolio Items*.

   Alternatively, in the Hedge Accounting Workplace go to the *Basic Data* tab page and choose the pushbutton.

   To start the **portfolio item run in the merge scenario**, choose *Analytics* > *Accounting: Merge Scenario* > *Service Functions for Risk Basis* > *Portfolio Fair Value Hedge* > *Run Administration* > *Create and Schedule* > *Start Portfolio Item Run (Merge Scenario)*.

   Alternatively, in the Hedge Accounting Workplace go to the *Basic Data* tab page and choose the pushbutton.
   The system displays a selection screen.
2. Under *Processing Options*, choose the mode in which you want to start the portfolio item run. Each mode uses different parameters:
   - In the *Update Mode with Run Type* enter the run type to start the portfolio item run.
   - In the *Update Mode without Run Type* enter the hedging relationship ID to start the portfolio item run.
   - In the *Update Mode without Run Type* enter the hedging relationship ID to start the portfolio item run.
     In the update mode, the system saves the results of the run in the database. The system starts the run as a parallel job. The system does not save the results of a test run in the database. Test runs are not run as parallel jobs. The system calculates the data only for the process category you specify. It records an application log. You can use test runs to analyze errors.

   For test purposes, you can use the following function codes in the runs:
   - `DIAL` switches parallel processing off independently of the run mode.
   - `DEBG` switches parallel processing off independently of the run mode, and activates breakpoints at different synchronization points in single transaction processing.
3. Choose .
   The system performs the portfolio item run, and saves the results in the RDB. The system versions the runs internally but it always accesses the latest version in the Hedge Accounting Workplace and in the key date valuation.

4.6.4.2.5 Simulation Run for Portfolio Item Run

Use

In this function, the system finds the portfolio items from period end processing for a portfolio fair value hedging relationship.

This function is used in the online run for the fair value effectiveness test and the simulation of the key date valuations. The system triggers this function in the background.

Features

The system first checks whether the portfolio items required were already calculated. If so, it reads hierarchy portfolio items from the Result Database (RDB). If not, it calculates the portfolio items from period end processing. This process is an online run for which parallel processing is not used.

> **Note**
>
> To avoid this calculation run, which can have a long runtime, start a portfolio item run before you call the simulation run.

For more information about the calculation, see Calculation of Portfolio Items [page 1163].

4.6.4.3 Portfolio Fair Value Calculator

The Portfolio Fair Value Calculator is used by the Portfolio Fair Value Hedge component, and contains the following functions:

- Calculation of the Hedge Adjustment for Portfolio Items [page 1175]
- Repayment Adjustment for Hedge Adjustments [page 1179]
- Reassignment of Hedge Adjustments [page 1180]

The functions Repayment Adjustment and Reassignment are applied to the hedge adjustment to be deferred.

4.6.4.3.1 Calculation of the Hedge Adjustment for Portfolio Items

Use

You use this function to calculate the hedge adjustment for the hedged items in a portfolio fair value hedge, and its hedging instruments. The hedge adjustment always relates to precisely one time bucket.

The portfolio fair value calculator is used only once the system has determined the portfolio items for the hedging relationship in question, and mapped the cash flows of the single transactions to the maturity band.

Integration

The following applications call the portfolio fair value calculator:

- Accounting Processes (key date valuation for time bucket objects)
- Fair Value Effectiveness Test for Hedging Relationships [page 1036]

The portfolio fair value calculator calls the Fair Value Server [page 628] to calculate the auxiliary key figures *full fair value*, *hedge fair value*, and *hedge amortized cost*. The hedged items that are used to calculate the hedge adjustment are taken from the Accounting Processes.

Prerequisites

In Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Valuation* > *Edit Portfolio Item Environment*, you have defined a valuation environment, and have assigned a method schema and a derivation strategy to this environment for the market data set. This market data set is derived for the whole portfolio fair value hedge.

You have defined a calculation environment in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Valuation* > *Edit Portfolio Item Environment* and you have used the calculation procedure to define whether the system is to include all flows or just nominal flows in the calculation of the hedge adjustment. You have also assigned a valuation environment to the calculation environment, which contains a suitable method environment.

> **Note**
>
> Note that you require a separate valuation environment for each calculation procedure. If you use the *Nominal Flows Only* procedure, the corresponding valuation environment must contain a method schema that in turn contains the cash flow refinement method generation of a par bond [page 517]. We provide the method schemas PFVH0 (generation of par bonds) and PFVH1 (no generation of par bonds).

Features

The system uses the following approaches to calculate the hedge adjustment (HA) for financial transactions. The system uses the calculation procedures stored in the calculation environment to determine which approach has to be applied:

Hedged Part of the Portfolio of Hedged Items

- Par bond approach (*Nominal cash flows only* calculation procedure)

The system interprets the amount in the time bucket as the nominal amount and generates a par bond that maps the change in the value of this nominal amount. It calculates the hedge adjustment using this par bond:

$$HA(t) = HI \cdot \left(V_{parbond}(t) - V_{parbond}(d)\right)$$

Where *HI* is the share of the portfolio of hedged items that is hedged; *V par bond* is the value of the par bond, *t* is the evaluation date, and *d* is the designation date of the hedging relationship.
For more information about creating par bonds, see Generation of a Par Bond [page 517] in the documentation for the cash flow refinement method.

- Discounting approach (*All Flows* calculation procedure)
  The system interprets the amount in the time bucket as a zero bond and calculates the hedge adjustment as the difference in the full fair value between the evaluation date and the designation date of the hedging relationship.

$$HA(t) = FFV(t) - FFV(d)$$

Where *FFV* is the full fair value of the hedged part of the portfolio of hedged items, *t* is the evaluation date, and *d* is the designation date of the hedging relationship.

- Discounting approach that avoids the pull-to-par effect (calculation procedure *All Flows Avoiding the Pull-to Par Effect*)
  The system calculates the hedge adjustment as the difference between the full fair value on the evaluation date and the full fair value on the evaluation date based on the forward yield curve from the designation date onwards:

$$HA(t) = FFV(t) - FFV_{d,t}(t)$$

Where $FFV_{d,t}$ is the full fair value of the hedged part of the portfolio of hedged items based on the forward yield curve for the designation date of the hedging relationship *d* on evaluation date *t*. If there are no cash flows between the designation date and the evaluation date, this is the full fair value on the designation date compounded up to the evaluation date.

Hedging Instruments

- The system generally calculates the hedge adjustment as follows:

$$HA(t) = FFV(t) - FFV(d)$$

  Where *FFV* is the full fair value of the hedging instruments, *t* is the evaluation date, and *d* is the designation date of the hedging relationship.
- If you select the *All Flows Avoiding the Pull-to Par Effect* calculation procedure, the system calculates the hedge adjustment as follows:

$$HA(t) = FFV(t) - FFV_{d,t}(t)$$

  Where $FFV_{d,t}$ is the full fair value of the hedging instruments based on the forward yield curve for the designation date of the hedging relationship *d* on evaluation date *t*. If there are no cash flows between the designation date and the evaluation date, this is the full fair value on the designation date compounded up to the evaluation date.

4.6.4.3.2 Bottom Layer Approach

Use

In this function, the system uses the bottom layer approach to calculate the hedged amount of portfolio items.

When you designate portfolio fair value hedging relationships, you usually specify only the percentage of the hedge. Even if you enter the hedge as an amount when you designate the relationship, the system converts this amount into a percentage.

If you use the default setting, when the system calculates the hedge adjustment it retains the percentage rate of the hedge in the time buckets. If the portfolio item changes from one analysis period to the next, the hedged amount of the portfolio item also changes. To avoid large fluctuations, use the bottom layer approach.

Integration

The system calls this function during the calculation of the hedge adjustment in the Portfolio Fair Value Calculator [page 1175].

Prerequisites

In Customizing for *Bank Analyzer* you have set the *Bottom Layer Approach* checkbox in the parameters of the valuation environment under *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Valuation* > *Edit Portfolio Item Environment*.

You can define the bottom layer approach for each valuation rule [page 490].

Features

If you set the *Bottom Layer Approach* checkbox, you avoid fluctuations in the hedged amount. The system fixes the hedged amount of a portfolio item for a time bucket; the amount stays fixed even if the portfolio items in this particular time bucket increase.

If the portfolio items in this time bucket are less than the amount originally hedged, the system interprets the entire portfolio item as hedged.

4.6.4.3.3 Repayment Adjustment for Hedge Adjustments

Use

In this function, the system calculates the repayment adjustment for the hedge adjustments to be deferred for a portfolio fair value hedging relationship. This adjustment is required when a valuation period ends. Hedge adjustments reflect changes to the value of portfolio items and hedging instruments. Therefore, repayments that are made during a valuation period (for example, standard repayments, transfers) reduce the hedge adjustments that the system carries over to the next period at the end of the valuation period.

Integration

Repayment adjustment for hedge adjustments is a function of the Portfolio Fair Value Calculator [page 1175]. This function is called by the accounting processes as part of the key date valuation of the time bucket objects.

To adjust the repayments for the hedge adjustment, the system needs the portfolio items of period start processing and period-end processing and the hedge adjustments for the time bucket of the prior period. The system reads these values from the results of the portfolio item run in the Result Database (RDB).

You assign a method for calculating the repayment adjustment to the valuation environment in Customizing for *Bank Analyzer* under *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Valuation* > *Edit Portfolio Item Environment*.

Features

Hedge adjustments are adjusted on the basis of repayments throughout the entire maturity band, in the respective time buckets, or in the portions of the legal entities:

- **Standard procedure**
  The system uses the following formula to calculate the overall repayment factor for the entire maturity band:

$$\frac{\left(\sum_{i=0}^{n} CF_{iBPstart} - \sum_{i=0}^{n} CF_{iBPend}\right)}{\sum_{i=0}^{n} CF_{iBPstart}} * 100$$

  n describes the number of time buckets and CF_i denotes the cash flow or the reference nominal value within time bucket i at the start and end of the valuation period. The system applies the repayment factor calculated in this way to the hedge adjustments of all time buckets.
- **Maturity-band-specific repayment adjustment**
  The system calculates the repayment portions for each time bucket in the maturity band. It calculates the repayment factor for each time bucket by examining the outflows in the time bucket. It applies the repayment factor calculated in this way to the hedge adjustment of the time bucket.

  > **Note**
  >
  > When you calculate the repayment adjustment on a maturity-band-specific basis, you can use the bottom layer approach [page 1178] if you wish.

- **Distribution in accordance with the nominal values of hedging instruments**
  This setting is only beneficial for portfolio fair value hedging relationships with several legal entities. In this case, hedge adjustments exist for the time buckets for each legal entity.
  The system calculates repayment factors for each legal entity. The basis for this is the overall repayment factor (see standard procedure), which is scaled for each legal entity. The system calculates the portion of the overall nominal value that the hedging instruments of each legal entity have and uses this portion to scale the overall repayment factor. The resulting repayment factors for each legal entity are applied to the hedge adjustments that belong to that legal entity.

4.6.4.3.4 Reassignment of Hedge Adjustments

Use

The system uses this function to update the values of the hedge adjustments of a portfolio fair value hedging relationship for a particular time bucket in the maturity band (merge scenario) or to update the values of the hedge

adjustments of a portfolio fair value hedging relationship that are to be dissolved (subledger scenario). These values have to be updated when a new valuation period begins.

Integration

The reassignment of the hedge adjustment is a function of the Portfolio Fair Value Calculator [page 1175].

In order to reassign the hedge adjustment, the system needs the portfolio items from period-end processing and rollover processing, plus the hedge adjustments for the time bucket of the prior period. The system reads these values from the results of the portfolio item run in the Result Database (RDB).

The system calls this function during the redesignation process. You use the calculation procedure template 2133 *Hedge: Redesignation Time Bucket Objects* to reassign the hedge adjustment.

Features

Strictly speaking, the system does not calculate the hedge adjustment (HA) for the time buckets of the current valuation period. Instead, it uses the hedge adjustments of the time bucket of the prior valuation period, and assigns them to the time buckets of the new valuation period.

For qualified assets and liabilities, the system calculates the outflows and inflows (separately for assets and liabilities) between the time buckets, and updates the old values for the hedge adjustment according to these changes. The system does this as follows:

$$HA_N^{new} = HA_N^{old} + HA_N^{old} \cdot \Delta_N$$

$$HA_N^{new} = HA_n^{old} + HA_n^{old} \cdot \Delta_n - HA_{n+1}^{old} \cdot \Delta_{n+1}, \; n \leq N$$

Where HA_n is the hedge adjustment for time bucket n; the time bucket N is the last time bucket. The values $Ä_n$ represent the addition of the qualified positions to, or their removal from, time bucket n. These values are defined as follows:

$$\Delta_N = \begin{cases} \frac{1}{v_N^{old}} \cdot \left(v_N^{new} - v_N^{old} \right), & v_N^{old} \neq 0 \\ 0, & v_N^{old} = 0 \end{cases}$$

$$\Delta_n = \begin{cases} \frac{1}{v_n^{old}} \cdot \left(v_n^{new} - v_n^{old} + \Delta_{n+1} \cdot v_{n+1}^{old} \right), & v_n^{old} \neq 0 \\ 0, & v_n^{old} = 0 \end{cases} \qquad n \leq N$$

Where v_n is the qualified position (qualified asset or qualified liability) in time bucket n. The value v_{new} is taken from the results of rollover processing, and the value v_{old} from period-end processing. This ensures that the system takes the values from the same maturity band.

4.6.4.4 Run Administration

Definition

Run administration includes the following functions:

- Execute or create run
- Display an overview of runs
- Display Application Log
- Edit run
- Manage run
- Replace run
- Select run for archiving
- Delete run
- Log of the deletion function

> **Note**
>
> The above functions are not all available for each application. For more information, see the application-specific documentation.

Use

The following table lists the runs available for each application:

Application	Run
General Calculation and Valuation Methods [page 376]	Account Pooling [page 383] or Account Pooling (CVPM) Facility Distribution [page 399] or Facility Distribution (CVPM) Determination of the Free Line for Facilities [page 395] or Determination of the Free Line (CVPM) Collateral Distribution [page 404] or Collateral Distribution (CVPM) Determination of Default [page 410] Stress Runs [page 429]: • Stress run for account pooling or stress run for account pooling (CVPM) • Stress run for facility distribution or stress run for facility distribution (CVPM) • Stress run for determination of the free line or stress run for determination of the free line (CVPM) • Stress run for collateral distribution or stress run for collateral distribution (CVPM) • Stress run for determination of default
Credit Risk [page 1191]	Credit Exposure Run [page 1238] or Credit Exposure Run (CVPM) Country Risk Run [page 1336] (prototype for country risk) **Stress run:** Credit Exposure Stress Run [page 1261] or Credit Exposure Stress Run (CVPM)

Application	Run
Historical Database [page 1478]	**Version management:** Run for generating historical data for data layers [page 1481] Run for generating historical data for the bank's own models [page 1484] Uploading of Files [page 1486] **Calculation functions:** Determining Default Rates [page 1488] Determining Average Default Rates [page 1491] Determining Default Figures [page 1493] Calculation of Migration Matrices [page 1496] **Data retrieval:** Run for supplying data to models [page 1498] Downloading of Files [page 1500] **Stress tests:** Stress Run for Supplying Models with Data [page 1505] Generating Scenario Data in the Source Data Layer [page 1506]
Generic BI Data Extraction [page 1638]	Testing the BI Extractor [page 1639] BI extraction run **Note** Extraction runs are created and executed in SAP NetWeaver Business Intelligence (BI). The system displays information about extraction runs in run administration of Bank Analyzer.
Regulatory Reporting Interface	Data Extraction Runs
Limit Manager [page 1529]	Limit Utilization Run [page 1551]
Fair Value Effectiveness Test for Hedging Relationships [page 1036]	Effectiveness Test Run
Cash Flow Hedge Analysis [page 1107]	Valuation Run [page 1128] Subvaluation Run Aggregation Runs [page 1129]

Application	Run
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169] Portfolio Item Runs [page 1172] • Period-end run for portfolio items (subledger scenario) • Period-start run for portfolio items (subledger scenario) • Portfolio item run (merge scenario)

> **Note**
>
> For some of the Bank Analyzer components, you can use the Schedule Manager to schedule and control jobs. If you use multiple applications, you can define the sequence in which the runs are to be carried out. For more information, see Schedule Manager [page 437].

More Information

Status Overview for Run Administration [page 1248]

4.6.4.4.1 Schedule Manager

Definition

A tool in Bank Analyzer that you can use to execute and monitor complex business processes, such as period-end closing. When you schedule a run, you always have to specify the program with which the system calls the run. There is a separate program for each run category in an application.

The table below shows for which runs and run categories the Schedule Manager can be used, and their corresponding programs.

Application	Run Category	Program
Account Pooling	Account Pooling [page 383]	/BA1/DE_GM_MAIN_COMP
Account Pooling	Stress Run for Account Pooling [page 429]	/BA1/DE_GM_MAIN_COMP_SCEN
Determination of the Free Line for Facilities	Determination of the Free Line for Facilities [page 395]	/BA1/DE_GM_MAIN_FL
Determination of the Free Line for Facilities	Stress Run for Determination of the Free Line for Facilities [page 429]	/BA1/DE_GM_MAIN_FL_SCEN

Application	Run Category	Program
Distribution of Facilities	Distribution of Facilities [page 399]	/BA1/DE_GM_MAIN_RLV
Distribution of Facilities	Stress Run for Facility Distribution [page 429]	/BA1/DE_GM_MAIN_RLV_SCEN
Collateral Distribution	Collateral Distribution [page 404]	/BA1/DE_GM_MAIN_SV
Collateral Distribution	Stress Run for Collateral Distribution [page 429]	/BA1/DE_GM_MAIN_SV_SCEN
Determination of Default	Determination of Default [page 411]	/BA1/RA_DFLT4280_RUN
Determination of Default	Stress Run for Determination of Default [page 429]	/BA1/RA_DFLT4280_RUN_SCENARIO
Fair Value Effectiveness Test [page 1036]	Effectiveness Test Run [page 1039]	/BA1/R8_FVHE_MAIN_RUN
Fair Value Effectiveness Test	Generation of Time Series for Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MAIN_RUN
Fair Value Effectiveness Test	Modifying a Time Series for the Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MODIFY
Cash Flow Hedge Analysis [page 1107]	Subvaluation Run [page 1128]	/BA1/R5_CFHA_BTC_RUN
Cash Flow Hedge Analysis	Aggregation Run [page 1129]	/BA1/R5_START_BATCH_AGGR
Cash Flow Hedge Analysis	Joint call of Valuation Run [page 1111] and Aggregation Run [page 1123]	/BA1/R5_EXECUTE_CFHA
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169]	/BA1/R9_VPM_BTC_ID_RUN
Portfolio Fair Value Hedge	Portfolio Item Run (Merge Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_RUN
Portfolio Fair Value Hedge	Period-End Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PE_RUN
Portfolio Fair Value Hedge	Period-Start Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PS_RUN
Portfolio Fair Value Hedge	Define End of Valuation Period	/BA1HM/RAH_M_PFVH_SET_VP_END
Portfolio Fair Value Hedge	Adjust Initial Portfolio Fair Value Hedges for Next Valuation Period	/BA1HM/RAH_M_PFVH_ADJUST
Portfolio Fair Value Hedge	(Re-) Designation of Initial Portfolio Fair Value Hedges	/BA1HM/RAH_M_PFVH_DESIG

Application	Run Category	Program
Credit Exposure [page 1192]	Credit Exposure Run [page 1238]	/BA1/R2_CRA_MAIN_CRE
Credit Exposure	Credit Exposure Stress Run [page 1261]	/BA1/R2_CRA_MAIN_CRE_SCEN
Credit Exposure	Credit Exposure Run for Revolving Retail	/BA1/RR2_CRE_REVRET
Credit Exposure	Approximate Stress Run [page 1274]	/BA1/RR2_CRE_SCENARIO
Credit Exposure	Aggregation Run for Ad Hoc Calculation [page 1276]	/BA1/RR2_SCEN_AGGR
Country Risk [page 1326]	Country Risk Run [page 1336]	/BA1/R2_CRE_MAIN
Historical Database [page 1478]	Historization Run for Data Layers [page 1481]	/BA1/R6_HIST_RUN
Historical Database	Historization Run for Bank's Models [page 1484]	/BA1/R6_HIST_RUN_MODEL
Historical Database	Determine Default Rates [page 1488]	/BA1/R6_CF_DEFAULT_4210
Historical Database	Determine Average Default Rates [page 1491]	/BA1/R6_CF_DEFAULT_4220
Historical Database	Determine Default Figures [page 1493]	/BA1/R6_CF_DEFAULT_KEYFIGURES
Historical Database	Exporting Data to In-House Models [page 1498]	/BA1/R6_MOD_RUN
Historical Database	Calculation of Migration Matrices [page 1496]	/BA1/R6_CF_MIGRATIONS
Historical Database	Stress Run for Supplying Models with Data [page 1505]	/BA1/R6_MOD_RUN_SCENARIO
Historical Database	Stress Run for Supplying Data [page 1506]	/BA1/R6_FDB_RUN_SCENARIO
Regulatory Reporting Interface	Processing Framework Run Copying the Start ID	/BA1/S1_COPY_STARTID
Regulatory Reporting Interface	Processing Framework Run Creating Runs	/BA1/S1_CREATE_RUN
Regulatory Reporting Interface	Processing Framework Run Release Run	/BA1/S1_RELEASE_RUN

Application	Run Category	Program
Regulatory Reporting Interface	Processing Framework Run Creating a Worklist	/BA1/S1_CREATE_WRKL
Regulatory Reporting Interface	Processing Framework Run Processing a Run	/BA1/S1_PROCESS
Regulatory Reporting Interface	Data Extraction Runs	/BA1/S3_EXTRACT_RUN
Limit Manager [page 1529]	Limit Utilization Run [page 1551]	/BA1/R3_UTILIZATION_RUN_DPF

More Information

Schedule Manager

4.6.4.4.2 Managing Runs

Use

In run administration, you can obtain an overview of the data of individual portfolio fair value hedging relationships that was processed in the initial generation runs and in portfolio item runs. The system displays all the main attributes of the runs and run versions, including the key date, the creation date, and the current and previous status.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Run Administration* > *Manage* > *Manage Runs*.
   The system displays the initial screen for run administration.
2. Enter the external ID, business key date, version, and system status of the runs that you want to manage. You can also select a layout as the display variant.

   > **Note**
   >
   > If you did not enter a name for the run when you started the initial generation run for the first time, the system uses the ID of the hedging relationship as the external run name. For more information, see Runs for the Calculation of Portfolio Items [page 1160].

3. Choose  .

The system displays an overview of the runs.

Overview of the Selected Runs

The system displays an overview of all the runs that meet the selection criteria entered. This includes information about the business key date, creation date, and status of the runs. The runs can have the following status:

Status	Description
Crtd	Run created
Strt	Run started
Canx	Run cancelled
End	Run finished
AF	Flag for archiving
Del.	Deletion flag was set
Rvke	Deletion flag revoked
RINC	Run finished but not complete

To display more information about a run, or to flag a run for deletion, select the run and then choose  . The system displays a detail screen.

Displaying Runs Individually

The system displays detailed information about the selected run. You can use the following functions:

Action	Function
	The system switches to display mode
	The system switches to change mode
	The system flags the displayed run for deletion
Edit > *Deletion Flag* > *Undo*	The system resets the deletion flag
	The system displays the current and previous status of the run
Edit > *Flag for Archiving*	The system flags the displayed run for archiving

Display of the Results of a Run

On the *Configuration* tab page, the system displays the following data:

- Hedging relationship ID

- Run mode (`ID` for the initial generation run, `MR` for the portfolio item run, `PE` for the period-end run, `PS` for the period-start run)
- Portfolio item environment
- Start and end of the current valuation period
  The end date is not known for the initial generation run and the period-start run and is set to the start date.

To display the portfolio items that were calculated, on the *Result Database* tab page choose the following:

- Period start
- Period end
- Rollover

The system displays the portfolio items [page 1158] for period start processing, period-end processing, or rollover processing. Note that in the initial generation run and period-start run, the system calculates results for period start processing only. In the period-end run, it calculates results for period-end and rollover processing only.

In the portfolio item run, the results are relevant for two valuation periods: the portfolio items of rollover and period-end processing belong to the current valuation period, which is displayed on the *Configuration* tab page. The portfolio items of period-start processing refer to the following valuation period.

4.6.4.4.3 Deleting Runs

Use

You use this function to delete runs for individual hedging relationships that were processed in the initial generation run or in the portfolio item runs.

Prerequisites

Under Managing Runs [page 1188] you have set the deletion flag for the runs.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Run Administration* > *Delete and Archive* > *Delete Runs*.
2. Enter the external ID, business key date, type, and version of the run that you want to delete.
3. Choose to delete the run immediately or to simulate its deletion.
4. Choose .
   The system deletes the selected run.

More Information

Runs for the Calculation of Portfolio Items [page 1160]

4.6.4.4.4 Archiving of Runs

Use

You use this function to archive runs for individual hedging relationships that were processed in the initial generation run or in the portfolio item runs. You need to do this if you want to remove data from the database without deleting the data permanently.

Activities

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Run Administration* > *Delete and Archive* > *Archive Runs* .

The system displays the initial screen for SAP Archive Administration (transaction `SARA`). The *Archiving Object* field defaults to the name of the archiving object for the *Portfolio Fair Value Hedge* component (`BA1_R4_057`).

More Information

Runs for the Calculation of Portfolio Items [page 1160]

4.7 Credit Risk (FS-BA-PM-CR)

Use

Purpose

In order to manage a banks at overall level, it has to be possible to access all business transactions and organizational units that generate income, expense and risks. All financial transactions and bank positions are transferred along with all data to the Source Data Layer (SDL) [page 100]. Credit Risk is integrated into the Processes and Methods [page 366] of Bank Analyzer. The system stores the results generated by the Credit Risk application in the Results Data Layer (RDL) [page 1364] or in the Result Database (RDB) [page 1823]. The RDL and RDB contain analytical applications [page 1410] such as the Historical Database (HDB) [page 1478] that process the data further.

Features

Credit Risk contains the following components:

- Credit Exposure [page 1192]
  - Basel II
  - Counterparty/Issuer and Country Risk Calculation [page 1193]
- Portfolio Credit Risk Calculation:
  - Credit Portfolio Data Processing
  - Default Mode Credit Portfolio Model

> **Note**
>
> For more information about the availability of the portfolio credit risk calculation, see SAP Note 2316265.

4.7.1 Credit Exposure

Use

Credit Exposure contains an infrastructure that you can use to calculate exposure key figures such as the Exposure at Default (EAD), which is the basis for calculating the capital requirement (CR).

Implementation Considerations

To support compliance with the requirements from Basel II governing capital requirements, you can use the following approaches in Credit Exposure to measure credit risk:

- Standardized approach (simple standardized approach, comprehensive standardized approach)
- Internal ratings-based approach (IRB approach)
  - Foundation IRB approach
  - Advanced IRB approach

Integration

Credit Exposure is integrated into the Processes and Methods [page 366] of Bank Analyzer. Various source systems supply data to Credit Exposure. Rather than communicating directly with Credit Exposure or with the upstream General Calculation and Evaluation Methods [page 376], these source systems first store the data in the Source Data Layer (SDL) [page 100]. Credit Exposure can access master data and flow data in the SDL that was either supplied by one of the source systems or originally created in the SDL.

In Credit Exposure, you can use the results generated by the upstream general calculation and evaluation methods. The system accesses the results using the Results Data Layer (RDL) [page 1364] or the Result Database

(RDB) [page 1823], where the results data from general calculation and evaluation data is stored. You can use the results from the following methods:

- Account pooling [page 383]
- Facility distribution [page 399]
- Determination of the free line for facilities [page 395]
- Collateral distribution [page 404]
- Determination of default [page 410]

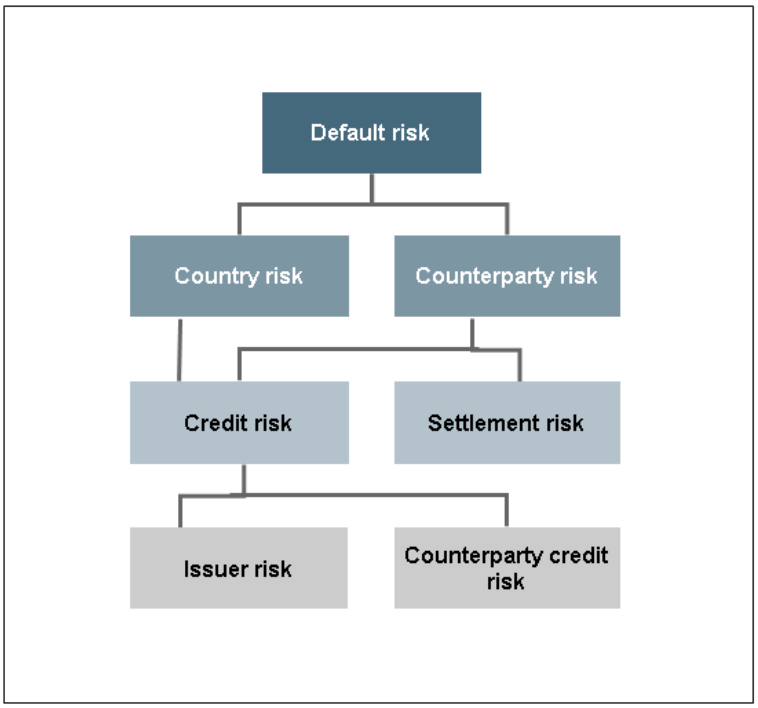
The calculated credit exposure results are also stored in either the Results Data Layer or in the Result Database. You define where results are stored for each client individually. You do this in Customizing under *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Storage of Results* > *Define Results Storage (RDB/RDL)*.

The results data can then be reported in SAP NetWeaver Business Intelligence (BI) or displayed in the run administration. Credit Exposure results can also be used in the runs for generating historical data, and stored in the Historical Database [page 1478].

4.7.1.1 Counterparty/Issuer and Country Risk Calculation

Use

Counterparty/issuer and country risks are calculated for internal default risk management. The default risk is the potential loss arising from a financial transaction should the business partner not fulfill his contractual obligations either due to specific, economic, or political reasons.



Classification of Default Risks

Counterparty/issuer risk describes the danger of a loss in the value of a receivable due to a worsening of the creditworthiness of the business partner. Counterparty/issuer risks are subdivided into credit risk and settlement risk. The existence of both these risks depends on the timing of the analysis of the transactions. Credit risk exists over the whole term of the transaction. Settlement risk exists only during the settlement period. Depending on the transaction category, credit risk can comprise just a counterparty risk or also an issuer risk (for example, in the case of forward securities transactions).

Country risk describes the danger of a loss in the value of a receivable due to a worsening of the credit standing of the relevant country for country risk. Country risk measures the inability of a business partner in the country affected or the country itself to fulfill its obligations due to political measures or economic problems.

Integration

- You can determine the counterparty/issuer risk and the country risk as part of the credit exposure calculation. You can calculate the counterparty/issuer risk and the country risk in two separate credit exposure runs, or in parallel in one calculation run.
- You can further process the results calculated for the counterparty/issuer risk and country risk in the Limit Manager (see Limit Manager [page 1529]).
- You can schedule the credit exposure run for the counterparty/issuer risk and country risk in the Schedule Manager (see Schedule Manager [page 437]).

For more information about the integration of the credit exposure calculation in Bank Analyzer, see Credit Exposure [page 1192].

Prerequisites

You have made the relevant settings for the credit exposure run in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Exposure*.

If you want to calculate the counterparty/issuer risk and country risk in parallel in one calculation run, you have to perform the IMG activity *Define Run Types* and assign a calculation method of the *Counterparty Risk* credit risk calculation category and a calculation method of the *Country Risk* credit risk calculation category to the run type. Also note that you have to define the settings for the data enrichment process step in such a way that the system reads the relevant data for both the counterparty/issuer risk and the country risk from the Source Data Layer (SDL).

Features

In calculation level 1 of the credit exposure run, you can calculate risk parameters according to the requirements of the counterparty/issuer risk and country risk. These can be risk parameters such as:

- Credit Exposure Calculation Basis
- Probability of Default (PD)
- Loss Given Default (LGD)
- Country Risk Quadrant

If you want to use your own risk parameters, you can define calculation modules in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Module Editor*.

Based on the risk parameters calculated, you can use the distribution method of the credit exposure run to carry out a general pro rata distribution of collateral. This is used to calculate the gross and net attributable amounts for counterparty/issuer and country risks.

More Information

For more information about executing the credit exposure run, see Starting the Credit Exposure Run [page 1238].

For more information about the distribution method, see Distribution of Collateral Instruments [page 1210] and Collateral Distribution in Counterparty/Issuer Risk and Country Risk [page 1215].

4.7.1.2 Calculation Function in Credit Exposure

Use

In the calculation function in Credit Exposure, the system calculates risk key figures for exposures and their collateral. By means of the settings you make in Customizing, you can configure the calculation function to support compliance with the requirements of Basel II (for example, calculation of the EAD [page 1204] and calculation of the regulatory capital requirement [page 1204]). The calculation also fulfils the requirements of the European Union and Germany using addtional, localization-specific, settings in Customizing.

The central component of the Credit Exposure calculation is the calculation method. This governs how a set of results is calculated. This is defined in Customizing by means of the combination of the Basel II approach and the calculation type, and stored in the run type so that it can be used in the credit exposure run. You start the Credit Exposure calculation function by means of the credit exposure run [page 1238].

Prerequisites

You need to have entered all the Customizing settings for *Credit Exposure* under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure*.

If you want to use the *European Union* or *Germany* localization, make the following settings in Customizing for *Credit Exposure*:

- *Master Data* > *Create Risk Weight Subclasses for Real Estate Financing*
- *Settings for Basel II* > *Define Bank-Internal Process Parameters (Germany and EU)*
- *Settings for Basel II* > *Edit Regulatory Settings for EU (European Union)*
- *Settings for Basel II* > *Display Standard Haircuts for Localization*
- *Settings for Basel II* > *Edit Localization Parameters*

If you do not use the localization, you need make no settings in these activities.

Process

Use the selection master parameters for InfoSet selection in Customizing to calculate credit exposures specifically for selected transactions. For more information, see the documentation for the individual activities under *Bank Analyzer* > *Process and Methods* > *Credit Risk* > *Credit Exposure* > *General Settings* > *Selection*.

The credit exposure calculation comprises the following steps:

1. Calculation Level 0 [page 1197]
   At calculation level 0, the system finds all the data that is required across different functions and for each approach before the actual Credit Exposure calculation is started. This is mainly for the purposes of data enrichment.
2. Calculation Level 1 [page 1199]

In calculation level 1, the system determines the parameters that are needed to calculate the capital requirements. It takes into account the Basel regulations, national discretion, and the bank's own rules (as defined by you in Customizing).

3. Calculation Level 2 [page 1209]
   In calculation level 2, the collateral is first optimally distributed, and then the capital requirement is calculated. Then the results of the distributed collateral instruments (secured and unsecured portions of the exposure) are mapped to RDL or RDB structures.

Result

The results can be reported in SAP NetWeaver Business Intelligence (BI). However, you can also display the results in the run administration by calling the logs or store them in the Historical Database (HDB) [page 1478].

> **Note**
>
> Although online detail reporting [page 1255] uses the same processes as the Credit Exposure calculation function (apart from data selection), it is purely a test function. You can use it to display additional interim results, which helps you to understand each calculation step by looking at a small amount of data. Note that the results of online detail reporting are not stored in the RDL or the RDB.

4.7.1.2.1 Calculation Level 0

Use

At calculation level 0, the system finds all the relevant data that is required across different functions and for each approach before the actual Credit Exposure calculation is started.

Process

1. **Read calculation-relevant data**
   The system uses the calculation-relevant data from the primary objects in the Source Data Layer [page 100] (SDL).
2. **Create and transform bundles**
   The bundling service [page 378] bundles all data that is linked to the objects from the SDL (such as the collateral objects for the selected exposures). It transforms the bundles so that the Credit Exposure run can process them (see Bundle Transformation [page 380]).
3. **Transfer results from General Calculation and Valuation Methods**
   You can use the results from General Calculation and Valuation Methods [page 376] in Credit Exposure Calculation.
   1. Account Pooling [page 383]
   2. Determination of the Free Line for Facilities [page 395]

3. Determination of Default [page 410]

> **Note**
>
> We recommend that you use the function for the distribution of collateral instruments [page 1210] that is integrated in the Credit Exposure calculation function instead of the collateral distribution function from General Calculation and Valuation Methods.

4. **Preliminary processing**
   1. **Determine the residual maturity**
      The residual maturity of the respective contract objects is used in particular to determine whether the maturities of the exposure and the collateral instrument match. The residual maturity is determined differently for exposures and collateral instruments, since call options must be treated differently for exposures and collateral instruments.
      - Determine the residual maturity for exposures
        The residual maturity for loans is the difference between the contract expiry date of the loan contract and the reporting key date, which is the period in which the borrower must have fulfilled his obligations at the latest. The borrower's early notice option for loans does not need to be considered. The bank's early notice option can be considered. The expiry date is then calculated as the earliest possible date of notice plus any additional periods of notice.
        If a loan is rolled over, the residual maturity is calculated as the difference between the reporting key date and the new *contract expiry date of the loan contract*. The different treatment of the call options of the bank and the call options of the borrower aims to reflect the principle of conservatism. To support Basel II requirements whilst still ensuring a certain leeway for banks, the following figures are calculated for each loan:
        - The minimum nominal residual maturity (`RLZ_NOM_MIN`), taking notice rights into consideration
        - The maximum nominal residual maturity (`RLZ_NOM_MAX`), taking into consideration the existing notice rights
        - The nominal residual maturity (`RLZ_NOM`) without taking into consideration rollover and notice options
      - Determine the residual maturity of collateral
        The residual maturity of collateral instruments is the difference between the maturity date of the collateral contract and the reporting key date. If the borrower or the collateral provider has an early notice option for the collateral contract, the date of the first notice option plus any existing period of notice is used as an input parameter and, therefore, as the minimum residual maturity of the hedging instrument.
        So that banks have some scope to use their own methods, the following figures are calculated for each collateral instrument:
        - The minimum nominal residual maturity (`RLZ_NOM_MIN`), taking notice rights into consideration
        - The nominal residual maturity (`RLZ_NOM`) without taking into consideration rollover and notice options
   2. **Calculate the original maturity**
      The original maturity of loans is the difference between the contract start date and the contract end date. The original maturity of securities with a final due date is the difference between the start and end of the term of the security.
   3. **Calculate the market value change period**
      The market value change period is determined identically for all approaches. The market value change period is used to calculate the add-on factors when the credit equivalent amount is calculated.
   4. **Determine the approach**

To measure the capital requirement for credit risks, banks can choose between the standardized approach and the internal-ratings-based (IRB) approach.
There are two variants of the standardized approach for banks, sovereigns, and enterprises: the **simplified standardized approach** and the **comprehensive standardized approach**. The approaches differ primarily in the attribution of collateral instruments.
There are two variants of the IRB approach for banks, sovereigns, and enterprises: the **foundation IRB approach** and the **advanced IRB approach**.
The advanced IRB approach is sub-divided into the following approaches
- Advanced IRB approach in which the CCF and the LGD values are estimated
- Advanced IRB approach in which only the CCF values are estimated
- Advanced IRB approach in which only the LGD values are estimated

5. **Risk differentiation**
In risk differentiation, a risk asset is assigned to the relevant risk categories, since different amounts for the capital requirement are determined for risk assets on the basis of the risk category. If the risk asset in the calculation has any collateral relationships, they are adjusted appropriately for the determined risk categories.
The following risk categories are considered:
   - Default risk
   - Specific market risk
   - Delivery risk
   - Settlement risk
   - Dilution risk
   - Residual value risk

The system temporarily saves the data enrichment data in the complex key figures, so that the data can be accessed during the calculation.

4.7.1.2.2 Calculation Level 1

Use

In Level 1 of the calculation run, the program finds all the parameters that are needed to calculate the capital requirement. It supports the Basel II regulations, national discretion, and the bank's own rules.

Process

Level 1 contains the following functions:

1. **Settings for Retail Exposures and Purchased Receivables**
   - Assignment of retail exposures to subportfolios
   Every private customer loan is assigned a subportfolio within the private customer portfolio. The accrual/deferral of subportfolios is measured according to the various criteria for the determination of equity securitization. This means that different risk weight functions apply for each subportfolio.
   - Assign Retail Receivables to Segments

Each retail loan is assigned to a segment within the retail portfolio. The segments are to be created so that mostly homogenous sub-portfolios result. No individual segment may contain an inappropriate concentration of the entire retail loans of a bank.

- Assign Purchased Receivables to Pools
  For purchased corporate and retail receivables in the IRB approaches, pool IDs are determined for the top-down approach. This assignment is not relevant for purchased receivables in the standardized approaches. No top-down approach is available for purchased receivables in the standard approaches. Therefore, a pool ID is not required.

2. **Rating**
   1. Determining the master rating class
      This function determines the object on which external, accepted rating information is stored.
      - In case of emissions, you need to differentiate whether the rating is needed to determine haircuts for financial (bond) securities or to determine the risk weight.
      - Furthermore, you need to determine the long-term and short-term external, accepted ratings separately for each business partner as they are used to
        - determine the business partner's risk weight
        - specify the eligibility of guarantees and hedging instruments
        - to check, whether a special case is applicable in case of short-term receivables.
      - The long-term and short-term external, accepted ratings must also be determined for each country because they are necessary to specify the risk weight.

      Then the external recognized ratings are mapped to a master class so that they are comparable in later stages of the calculation process.
   2. Determining the relevant rating
      The determination of a relevant rating checks whether an issuer or issue rating in the form of a short-term or a long-term rating is effective. If several ratings are available from different rating agencies, you must select the relevant rating.
3. **Risk Weight**
   The capital requirements are determined according to Basel II based on the risk-weighted assets. The system requires the risk weights r on the single transaction level in the standardized approach and in special cases in the IRB approach to determine the risk-weighted assets. The risk weights r are derived based on external or internal ratings that have been made comparable.
4. **Netting**
   This function is used to pool nettable receivables and payables. Mismatches are mapped by haircuts. To support the requirements of Basel II, the netting for on-balance-sheet, derivative and off-balance-sheet transactions is carried out in different ways.
   For more information, see Netting [page 1202].
5. **Exposure at default (EAD)**
   The Exposure at Default (EAD) refers to the expected amount of receivables at the time of default (commitment extent) without considering hedging instruments and provides the assessment basis. The EAD is determined based on calculation approach for any transaction.
   For more information, see Exposure at Default [page 1204].
6. **Eligibility**
   This function checks the relationship between the hedging instrument and the exposure depending on the calculation approach you have selected and the attributes of the exposure. The check considers credit derivatives, such as hedging instruments from specialized lending, as well as financial and physical collateral. The eligibility of hedging instruments is determined in the following steps:
   1. Set hedging instruments that do not fulfil the qualitative minimum requirements to "not eligible" (independent of calculation approach)

2. Set hedging instruments that are already included in the rating to "not eligible" (independent of calculation approach)
3. Determine the eligibility of hedging instruments for the relevant approaches

> **Note**
>
> You can also determine the eligibility of hedging instruments yourself outside Bank Analyzer, and then provide the system with that data.

7. **Haircuts**
   "Haircut" describes collateral markups or markdowns. Haircuts are calculated with respect to the volatility of exchange rates and the price volatility of the exposure amount or the exposure.
8. **Calculation of the Probability of Default (PD), Loss Given Default (LGD), and Maturity M**
   This function supplies the parameters PD, LGD, and maturity M that are required in addition to the EAD to calculate the risk-weighted assets in the IRB approaches.
   1. Calculating the Probability of Default (PD)
      When calculating the probability of default, for each combination of exposure and guarantee, the system assigns a PD to every exposure that is calculated according to either of the IRB approaches. This PD is alter used to calculate the capital requirement.
      The PD calculation is dependent on the relevant rating. The system uses the transaction to determine which rating to assign to the single transaction.
   2. Calculation of Loss Given Default (LGD)
      The Loss Given Default (LGD) is defined as the amount of loss in percent of the engagement at the time of default. The LGD is calculated differently according to the calculation approach.
      - In the standard approach, LGD is implicitly contained in the risk weight so that this function is only significant for the IRB approach.
      - For enterprises, states and banks a difference is made between the basic IRB approach and the advanced IRB approach.
      - For retail receivables, there is only a IRB approach which has the same configuration of estimate possibilities for risk parameters as the advanced IRB approach.

      There are differences in the procedure for LGD determination in the basic IRB approach for enterprises/states/banks on the one hand and advanced IRB approach for enterprises/states/banks as well as IRB retail on the other:
      - If the Foundation IRB Approach is in use, then the LGD for the portfolios *Enterprises*, *States*, and *Banks* is specified by legislators..
      - When using the advanced IRB approach or the IRB retail approach, banks can make their own estimates, though certain minimal requirements must be fulfilled.

      No LGD is calculated for the following transactions:
      - Transactions in the Standardized Approach
      - Specialized Lending in the Elementary Approach
      - Shareholdings in the Market Approach
      - Leasing Residual Value Risks
      - Securitizations
   3. Calculates maturity M
      The calculation of maturity M (effective residual maturity) depends on the calculation approach. The maturity is calculated for financial transactions such as loans, forward contracts, swaps, options, or FRAs. National discretion is permitted by law for the Foundation IRB Approach.

Maturity M is specified in years and is relevant for loans of the following risk classes:

- Sovereigns
- Banks
- Corporates
- Equity holdings
- Purchased receivables

More Information

Business Content provides detailed examples of the credit exposure calculation. For more information, see the Business Content documentation.

4.7.1.2.2.1 Netting

Use

The term *netting* describes a complex legal concept. The individual contracts between the parties involved are linked using a facility in such a way that only the balance from the individual contracts is owed if the contractual relationship is terminated due to a cash flow disturbance or bankruptcy. Netting is a means of reducing credit risk, as is the provision of collateral. The result of netting is that the delivery of securities and payment obligations from trading activities are netted to reduce the number of settlement processes.

Netting reduces the economic default risk and the capital requirement in the context of supervisory requirements. Another advantage of netting is that it reduces the drawing on the internal limits set for counterparties. As banks price individual transactions according to their level of risk, they gain a competitive advantage from netting because this leads to a reduced risk margin.

Netting is to be seen as a preparatory step for the EAD determination. From the perspective of the EAD determination you have to distinguish whether the EAD is to be determined for contracts with or without netting agreements. If contracts are offset against each other in the sense of netting, only the result is considered in the subsequent EAD determination.

Features

The credit exposure calculation distinguishes between:

- Netting of on-balance sheet transactions
- Netting of repo-style transactions
- Netting of OTC derivative transactions

4.7.1.2.2.1.1 On-Balance-Sheet Netting

Use

In on-balance-sheet netting, balance sheet transactions are netted.

Prerequisites

To permit on-balance-sheet netting, in the Source Data Layer (SDL) a netting agreement (corresponding to the netting agreement of the counterparty) is created for the on-balance-sheet netting category. You use relationships that have this netting category to link the transactions that are to be netted. After the transactions have been netted, the netting agreement contains the result of the netting process. It is this result that is used in the calculation of the capital requirement.

To net transactions for a particular Credit Exposure run, in Customizing you need to have set the **`Perform Netting`** indicator under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Calculation* > *Define Calculation Method*.

You can also define for individual transactions whether these are to be netted. Contracts that are part of a netting agreement but are not relevant for netting in a particular run are treated as though there were no netting agreement for them.

Features

If a partially disbursed loan is linked to a contract relationship by means of the netting agreement, only the disbursed part is taken into account in on-balance-sheet netting. The part that has not been disbursed is treated as a separate traditional off-balance-sheet transaction, and included in other calculation processes.

Results of Netting

- In on-balance-sheet netting, the balance sheet assets and liabilities that belong to the same netting bundle are netted. The netting result (=EAD for netting, ignoring the calculation components) is calculated on the netting agreement level.
- The longest residual maturity of all transactions that are to be netted.
- In the advanced IRB approach, maturity M is calculated for the netting agreement.

In on-balance-sheet netting, the assets and liabilities that are to be netted are linked to the netting bundle by means of relationships to the netting agreement. A netting bundle consists of a netting main contract and two or more asset and liability transactions. Within the netting bundle, an ID (which has to be provided) for the assets and liability transactions is used to split the asset transactions into *n* asset transactions and the liability transactions into *m* liability transactions. Then the relationships between all n+m transactions that to be netted and the ijth relationships are created. All ijth relationships between the ith asset transaction (with i=1,...,n) and the jth liability transaction (with j=1,...,m) are mapped.

4.7.1.2.2.2 Exposure at default (EAD)

Use

The Exposure at Default (EAD) refers to the expected amount of receivables at the time of default (commitment extent) and provides the assessment basis before hedging instruments are credited. The Exposure at Default provides the basis for determining the risk-weighted assets and, therefore, the capital requirement amount, considering borrower-specific risk parameters and possibly risk-mitigation effects using hedging instruments The EAD is determined based on calculation approach for any transaction. The EAD determination is carried out for balance-sheet, off-balance-sheet, and derivative transactions. The system performs the EAD determination differently for transactions for which a netting agreement exists and transactions that lack a netting agreement.

Features

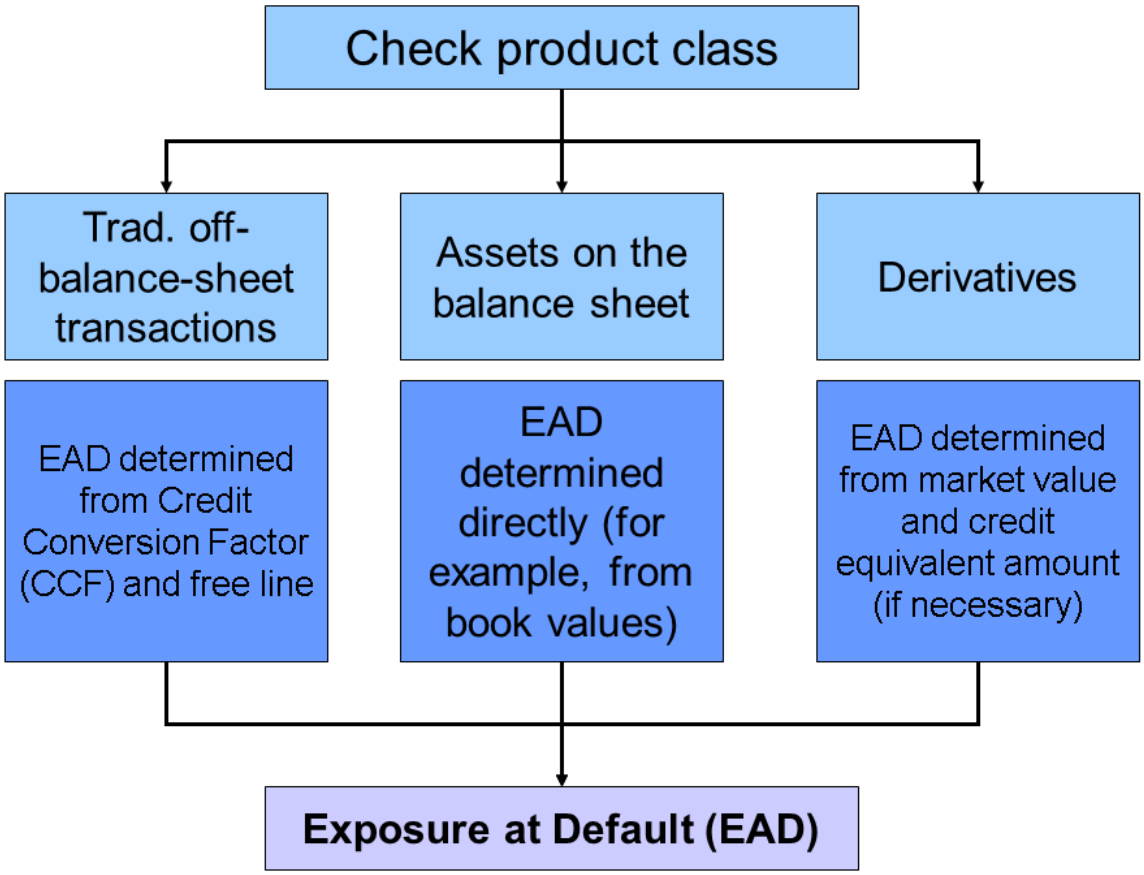
- **Calculation of the EAD for Balance Sheet Assets**
  The system performs the EAD determination for banking book and trading book asset transactions (balance sheet assets) depending on the calculation approach. Asset transactions refers to balance sheet receivables of banks payable to borrowers, that is credit or loan receivables.???
- **Calculating the EAD for off-balance sheet transactions**
  The relevant calculation approach controls how the system performs the determination of the EAD for traditional off-balance sheet. The system uses the derived or calculated CCF values or CCF classes, and haircuts Hc, He, and Hfx, to calculate the EAD. The EAD is calculated based on the maturity.
- **Calculating the EAD for derivative transactions**
  To calculate the EAD, the system uses the calculation of maturity and residual maturity; it also uses securities finance transactions by calculating the determined credit equivalent.

4.7.1.2.2.2.1 Example: Calculation of the EAD

The Exposure at Default (EAD) shows the amount of the exposure at the time of default, and is the basis for calculating the secured and non-secured parts.

> **Note**
>
> To calculate the EAD for derivatives using the mark-to-market method, you need to supply at least one assessment basis category, or enter the relevant assessment basis.

The EAD is calculated separately for each product for balance sheet assets, derivatives, and traditional off-balance-sheet transactions. This means that the credit equivalent amount is calculated for derivatives (in the mark-to-market method only), whereas for traditional off-balance-sheet transactions, such as *open commitments*, the free line for facilities is weighted by the credit conversion factor.



Calculation of the EAD for Traditional Off-Balance-Sheet Transactions

You can import the CCF value directly by means of key figures.

Calculation of the EAD for Balance Sheet Assets

For example, you enter the book value for the calculation of the EAD.

Calculation of the EAD for Derivatives

You can import the credit equivalent amount directly by means of key figures.

4.7.1.2.2.2.2 Counterparty Credit Risk Methods

The following counterparty credit risk methods are available for determining the EAD assessment basis for securities financing transactions (SFT) and OTC derivatives:

- **Mark-to-market method**: The credit equivalent value is used as the EAD assessment basis.
- **Standardized method** (for OTC derivatives only): To determine the EAD assessment basis, the financial transactions are divided into individual legs, for which the system then determines risk positions.
  The following activities in Customizing for Credit Exposure are particularly important for the standardized method. Choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure*:
  - *Settings for Basel II* > *Edit Credit Conversion Factor for Counterparty Credit Risk*
  - *Calculation* > *Counterparty Credit Risk* > *Standardized Method* > *Edit Calculation Rules*
  - *Calculation* > *Counterparty Credit Risk* > *Standardized Method* > *Assign Calculation Rules to Calculation Types*
  - *Calculation* > *Counterparty Credit Risk* > *Standardized Method* > *BAdI: Calculation of Risk Positions*
  - *Calculation* > *Counterparty Credit Risk* > *Edit Read Method for Counterparty Credit Risk*
  - You define the scaling factor *Beta* for the standardized method under *Settings for Basel II* > *Edit Bank-Internal Process Parameters*.
  - You define the standard settings for the counterparty credit risk method that you want to use under *Calculation* > *Define Calculation Method*.
- **Internal model method**: The EAD assessment basis is calculated from the expected positive replacement value. You must provide the replacement values for the system.
  The following activities in Customizing for *Credit Exposure* are particularly important for the internal-model-based method (IMM). Choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure*:
  - *Calculation* > *Counterparty Credit Risk* > *Edit Read Method for Counterparty Credit Risk*
  - You define the standard settings for the counterparty credit risk method that you want to use under *Calculation* > *Define Calculation Method*.
  - *Settings for Runs in Result Database or Settings for Runs in Results Data Layer* > *BAdI: Assignment of Characteristics and Key Figures*
  - You define the scaling factor *Alpha* for the internal-model-based under *Settings for Basel II* > *Edit Bank-Internal Process Parameters*.
  - The system uses the Fair-Value-Server to calculate Maturity M according to the internal-model-based method. For a more detailed explanation of the Customizing prerequisites for these activities, see Example: Calculation of Maturity M in the Internal Model Method [page 1207].
- **Standardized approach for counterparty credit risk (SA-CCR)** (for derivatives only):

  > **Note**
  >
  > Once it enters into force, this method will replace the mark-to-market method and the standardized method.

  To determine the EAD at the level of a netting agreement, the transactions to be netted are subdivided into the following asset classes:
  - Interest rate

- Foreign exchange
- Credit
- Equity
- Commodity

Hedging sets are formed on the basis of regulatory requirements within each asset class. Add-on amounts are determined at hedging set level and these amounts are then used to calculate the add-on amounts for the individual asset classes. The add-on amounts for the asset classes are totaled as the aggregated netting add-on, which in turn is weighted with a multiplier to determine the potential future exposures (PFE).
The replacement costs are still calculated from the existing provided collateral (including margined collateral) and the market values of the underlying derivatives.
The sum of the potential future exposures (PFE) and replacement costs is weighted by the supervisory factor *Alpha* to calculate the EAD. Note when the EAD is calculated that the EAD for a netting agreement with margin collateral must never be higher than the EAD that would have been calculated for the netting agreement if the margin collateral did not exist. This is ensured by a corresponding minimum function.

The following Customizing activities under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* are relevant for the standardized approach for counterparty credit risk:

- You define the standard settings for the counterparty credit risk method that you want to use under *Calculation* > *Define Calculation Method*.
- You define the scaling factor *Alpha SA-CCR* for the standardized approach under *Settings for Basel II* > *Edit Bank-Internal Process Parameters*. in this Customizing activity, you also specify which calendar is to be used to calculate the business days and enter the basis to be used to convert the number of business days into years.
- You enter the supervisory factors to be included in the calculation under *Settings for Basel II* > *Edit Supervisory Factors for Standardized Approach for CCR*.
- You select the supervisory factors using the supervisory factor ID, which you assign to the corresponding localization parameter under *Settings for Basel II* > *Edit Localization Parameters*.

You can apply the methods to corresponding netting agreements or to individual transactions that do not belong to a netting agreement.

In the Customizing activity *Edit Calculation Method* you can define which method is to be the default method for SFTs, and which is to be used for OTC derivatives. You can also use this activity to specify that the system use this method if no method has been defined for a transaction (partial use), or if it should apply the method regardless of the method defined for a transaction (no partial use).

4.7.1.2.2.2.1 Example: Maturity M in the Internal Model Method

In order to use the Fair Value Server to calculate maturity M under the internal model method, you need to make the following settings in Customizing for *Bank Analyzer*:

1. *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Create Cash Flow Views*.

View	ID of Cash Flow View
99	B2 CF View

1. *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Cash Flow Refinement* > *Method Schemas* > *Edit Method Schemas* .

Method Schemas

Method Schemas	Permitted Application	Method Schema Name
CCR_IM	Fair Value Server	R2: CCR Internal Method

Method Assignment

Step Number in Method Schema	ID of Method for Cash Flow Refinement	ID of Cash Flow Refinement
10	1100	RB: Market Data Assignment
20	2000	CCR Internal Method

1. *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Price Calculator* > *Edit Price Calculator Environments* .

Price Calculator Environment

Price Calculator Environment	Name of Price Calculator Environment	Evaluation Category	Bid/Ask	Transfer Type	Switch Off Check of Term of Transaction	Switch Off Check for Cash Flows
ATB2	AT: Price Calculator Environment for Basel II	STAN			X	

Settings for the Price Calculator Environment

Transfer NPV	Valuation Control	Special Model	...
Data is not transferred	Use special model	CASH-FLOW DISC	

1. *Bank Analyzer* > *Processes and Methods* > *General Calculation and Valuation Methods* > *Present Value Calculation* > *Fair Value Server* > *Configuration* > *Create Global Valuation Environments* :

Global Valuation Environment

Global Valuation Environment	Name of Global Valuation Environment	Price Calculator Environment	Method Schemas	Method Environment	Splitting Environment
AT_FVS1_B2	XXXXXX	ATB2 (see above)	CCR_IM (see above)	SIFRS	SIAS

Determination of Market Data Sets

Parameters for Method for the Transfer of Customizing	Derivation Strategy for Market Data
Initial (= general derivation of market data sets)	SDS1 (= derivation strategy for the market data set for IFRS)

> **Note**
>
> You must enter the data above for the determination of the market data set. This information is required for the discount factors.

1. *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *General Settings* > *Enter Market Data Settings*:

Global valuation environment: **AT_FVS1_B2** (see above: Create Global Valuation Environment).

4.7.1.2.3 Calculation Level 2

Use

At level 2 of the calculation run, depending on the calculation approach, the system determines the capital requirement, taking collateral instruments (collateral, guarantees, credit derivatives) into account. While collateral instruments in the standard approach reduce the exposure or the risk weight, in the IRB approach they reduce the loss given default (LGD) or the probability of default (PD). The system also distinguishes between exposure and collateral instrument in terms of the control values such as risk category, credit standing, collateral instrument type or the match or mismatch of currency and terms.

Process

1. Distribution of Collateral Instruments [page 1210]
   The core content of the capital calculation in the SAP Basel II Analyzer is the determination of the optimal distribution of collateral instruments to exposures. This considers all restrictions and liberties specified by Basel regulations. The system uses constructive or numerical algorithms (such as simplex algorithms) to optimize the distribution, depending on complexity. A choice of solution method for treating very large bundles ensures the high-performance optimization for all imaginable cases .

2. Determination of the Capital Requirement [page 1236]
   The key figures risk-weighted assets, Expected Loss (EL) and capital requirement (CR) are determined from the risk parameters and the EAD. The special features for the standard and IRB approach and for diverse special cases like specialized lending are taken into account.

More Information

Business Content provides detailed examples of the credit exposure calculation. For more information, see the Business Content documentation.

4.7.1.2.3.1 Distribution of Collateral Instruments

Use

You control how collateral instruments are distributed in the credit exposure calculation by means of the distribution method. You define the distribution method in the Customizing for Credit Exposure. The distribution method is used to calculate the capital requirement by distributing collateral instruments, and is also used in on-balance-sheet netting.

In addition to the collateral distribution function in the General Calculation and Valuation Methods [page 376], the system also contains a function for distributing collateral that is integrated in the credit exposure calculation.

> ➔ Recommendation
>
> We recommend that you use the function for the distribution of collateral instruments that is integrated in the Credit Exposure calculation function instead of the collateral distribution function from General Calculation and Valuation Methods.
>
> To map collateral instruments in the Source Data Layer (SDL), we recommend that you use the extended collateral mapping that includes portions.

$$\overline{EK} = \sum_{i=1}^{n} \left[\kappa_i^{us} \cdot EAD_i^{us} + \sum_{j=1}^{m} \kappa_{ij} \cdot EAD_{ij} \right]$$

General Formula for Distribution of Collateral Instruments Using the Distribution Method

In the standardized approach, the minimizing of the capital requirement is the equivalent of the minimizing of the risk weighted assets. This gives the following formula:

$$\kappa_i^{us} = r_i^{us}, \kappa_{ij} = r_{ij}$$

Minimizing the Capital Requirement Is the Equivalent of Minimizing Risk-Weighted Assets

In the IRB approaches, the minimizing of the capital requirement is the equivalent of the simultaneous minimizing of the expected loss and the unexpected loss. This is achieved by using optimization parameter λ in the following formulas:

$$\kappa_i^{us} = 12.5 \cdot \left[\lambda \cdot LGD_i^{us} \cdot PD_i^{us} + (1-\lambda) \cdot LGD_i^{us} \cdot f\left(PD_i^{us}\right)\right]$$

$$\kappa_{ij} = 12.5 \cdot \left[\lambda \cdot LGD_{ij} \cdot PD_{ij} + (1-\lambda) \cdot LGD_{ij} \cdot f\left(PD_{ij}\right)\right]$$

Minimizing the Capital Requirement Is the Equivalent of Simultaneously Minimizing the Expected and Unexpected Loss

For more information about this, see the document Simultaneous Minimization of Expected and Unexpected Loss in the IRB Approach [page 1222].

> **Note**
>
> Usually, there are *n* exposures and *m* hedging instruments.
>
> $$i \in I = [1, \ldots, n]$$
>
> **Run Index for Exposures**
>
> $$j \in J = [1, \ldots, m]$$
>
> **Run Index for Collateral Instruments**

Prerequisites

In Customizing for *Credit Exposure*, you have made the required settings under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *General Settings* > *Settings for Distribution Method*. In addition to defining the distribution method, here you can also store BAdI implementations that contain the following steps for the distribution function:

- Distribution methods
- Sorting
- Linear optimization

> **Note**
>
> For more information, see the documentation for the relevant Customizing activities.

Features

The distribution coefficient a_{ij} defines how collateral instruments are distributed. The distribution coefficient is the portion of the ith exposure that is secured by the jth collateral instrument.

$$\alpha_{ij} = \frac{EAD_{ij}}{EAD_i}$$

Distribution Coefficient

For the general formula given above, this results in:

$$EK = \sum_{i=1}^{n} \left[\kappa_{i0} \cdot \alpha_{i0} \cdot EAD_i + \sum_{j=1}^{m} \kappa_{ij} \cdot \alpha_{ij} \cdot EAD_i \right]$$

General Formula for Distribution of Collateral Instruments Taking Distribution Coefficients Into Account

The distribution coefficients have to be positive, and smaller than or equal to 1. The distribution coefficient is set to zero for relationships that are not part of a bundle. Each collateral instrument can be used completely, and the exposures cannot be over-collateralized.

As stipulated in the Basel Consultative Paper, the value of collateral is adjusted by a haircut. This haircut comprises:

- Maturity mismatch (*τ*)
- Currency mismatch (*Hc*)
- Volatility of the collateral instrument (*Hfx*)

In normal circumstances, the value of the collateral is adjusted by the haircut *β*:

$$\beta = \max \left[(1 - Hfx - Hc) \cdot \tau, 0 \right]$$

Haircut β

The actual valuation of the figures *Hc*, *Hfx*, and *τ* takes place in previous steps, as described in the section Determination of the Capital Requirement [page 1236].

In Credit Exposure, collateral instruments are distributed differently based on the cardinality of the bundle. The bundle cardinality describes the ratio of the exposures (n) to the collateral instruments (m).

A bundle can have one of the following cardinalities:

Bundle Cardinality	Processing of the Bundle Cardinality
1:0	Simple calculation using the general formula
1:1	Simple calculation using the general formula
1:m	• Use of an optimization algorithm • Optional: Use of a sort algorithm The advantage of using the sort algorithm is that it improves the run time of the program considerably. Note, however, that the result produced can be suboptimal.

Bundle Cardinality	Processing of the Bundle Cardinality
n:1	• Exposures are sorted before they are assigned to collateral instruments. By sorting the exposures, the optimum solution can be found. • Optional: Use of an optimization algorithm
n:m	• Use of an optimization algorithm (linear optimization) • Optional: Use of a sort algorithm The advantage of using the sort algorithm is that it improves the run time of the program considerably. Note, however, that the result produced can be suboptimal.

Recommendation

We recommend that you use optimization for bundle cardinalities 1:m and n:m.

Optimization

In a linear optimization process, the system uses the following objective function to find the coefficients of distribution:

$$\min \overset{!}{=} \sum_{i=1}^{n} \left[\kappa_{i0} \cdot \alpha_{i0} \cdot EAD_i + \sum_{j=1}^{m} \kappa_{ij} \cdot \alpha_{ij} \cdot EAD_i \right]$$

Distribution Coefficient as Part of a Linear Optimization Process

With the following constraints:

- The distribution coefficients a_{ij} have to be positive, and smaller than or equal to 1.
- Each collateral instrument can be assigned to exposures to a maximum of 100 %.
- The exposures must not be over-collateralized.

In addition, priorities and seniorities are taken into account, and in the IRB approach the level of over-collateralization C^{**} and the threshold value C^{*} are also included.

Note

If the standard setting delivered by SAP is used, the Simplex algorithm is applied to solve the linear optimization problem. You can use BAdI definition `/BA1/R2_RCR_SIMPLEX` to implement alternative algorithms.

Sorting for Bundle Category 1:m (Standardized and IRB Approach)

For bundle category 1:m, in the sort process the system determines the order in which collateral instruments are to be assigned to the exposure in the following way:

- Determination of the order
  1. Priorities in ascending order
  2. Risk weight κ_{ij} in ascending order

In a separate step, the system allocates the collateral instruments successively to the exposure in the order determined. The allocation process ends when all collateral instruments have been assigned or when the exposure is 100 % collateralized.

Example

For a more detailed explanation of this process, see Example: Sorting for Bundle Category 1:m [page 1223].

Sorting for Bundle Category n:1 (Standardized and IRB Approach)

For bundle category n:1, in the sort process the system determines the order in which the collateral instrument is to be assigned to the exposures in the following way:

- Determination of the order
  Sorting in descending order by the product of *β* and the difference in the risk weight of exposure *i* and the risk weight of collateral *j*.

$$\left(\kappa_i - \kappa_{ij}\right) \cdot \beta_{ij}$$

Sorting for Bundle Category n:1

In a separate step, the system allocates the collateral instrument successively to the exposures in the order determined. The allocation process ends when the collateral instrument has been 100% assigned, or when all exposures are 100 % collateralized.

Example

For a more detailed explanation of this process, see Example: Sorting for Bundle Category n:1 [page 1229].

Sorting for Bundle Category n:m (Standardized and IRB Approach)

For bundle category n:m, in the sort process the system determines the order in which the collateral instruments are to be assigned to the exposures in the following way:

Sorting of the relationships between collateral instrument *j* and exposure *I* as per the following formula:

- Determination of the order
  Sorting in descending order by the product of *β* and the difference in the risk weight of exposure *I* and the risk weight of collateral *j*.

$$\left(\kappa_i - \kappa_{ij}\right) \cdot \beta_{ij}$$

Sorting for Bundle Category n:m

In a further step, the system allocates the collateral instruments successively to the exposures in the order determined. The allocation process ends when all collateral instruments have been 100 % assigned, or when all exposures are 100 % collateralized.

4.7.1.2.3.1.1 Collateral Distribution in Counterparty/Issuer Risk and Country Risk

Use

You can use the distribution method of the credit exposure run to carry out a generalized pro rata distribution of collateral instruments. On the basis of the collateral distribution, the system calculates the secured and unsecured gross and net portions, as well as the gross and net attributable amounts for the counterparty/issuer risk or country risk.

Prerequisites

- You manage the collateral distribution using the *BAdI: Distribution Method* in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *General Settings* > *Settings for Distribution Methods* > *BAdI: Distribution Method*.
  For this purpose, you have defined your own implementation or selected one of the following standard BAdI implementations:
  - /BA1/R2_SLV_EXP_CALC_ECO – Economic
  - /BA1/R2_SLV_EXP_CALC_POL – Political
  - /BA1/R2_SLV_EXP_CALC_MAX – Maximum
- The standard implementations of the *BAdI: Distribution Method* take into account general priorities at relationship level between collateral, portions, and exposures. You have already determined these priorities in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *General Settings* > *Settings for Distribution Methods* > *BAdI: General Priority*.
- The standard implementations of the *BAdI: Distribution Method* can calculate the collateral distribution for the counterparty/issuer risk or country risk. In Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Calculation* > *Define Calculation Methods*, you have specified whether the system is to calculate the collateral distribution for the *Counterparty Risk* or *Country Risk* credit risk calculation category.
- The distribution coefficients calculated are converted into attributable amounts by means of modules. You have defined corresponding modules in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Module Editor*.
- You have assigned these modules to the calculation methods in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Calculation* > *Assign Modules to Calculation Methods*.

Features

The system distributes collateral on a pro rata basis to the exposures and attributes the assigned collateral pro rata for the exposures. The system first calculates distribution coefficients at relationship level by comparing the

net or gross amount distributed to the relationship with the credit exposure calculation basis. It then calculates both the net and gross attributable amounts from the distribution coefficients.

You can use the following calculation methods:

- Economic
  The system distributes the collateral based on the economic collateral agreement amount (collateral value) specified in the collateral agreement contract.
- Political
  The system distributes the collateral based on the political collateral agreement amount specified in the collateral agreement contract.
- Maximum
  The system distributes collateral first with economic and then with political collateral agreement amounts. The system uses the maximum of the attributable amounts for the secured portions and reduces the attributable amount of the unsecured portion accordingly.

You can distribute collateral for the counterparty/issuer risk and country risk. In the case of country risk, the following applies:

In net distribution, the system does not recognize collateral with the same country risk country as the exposure. In gross distribution, calculations are at country group level.

More Information

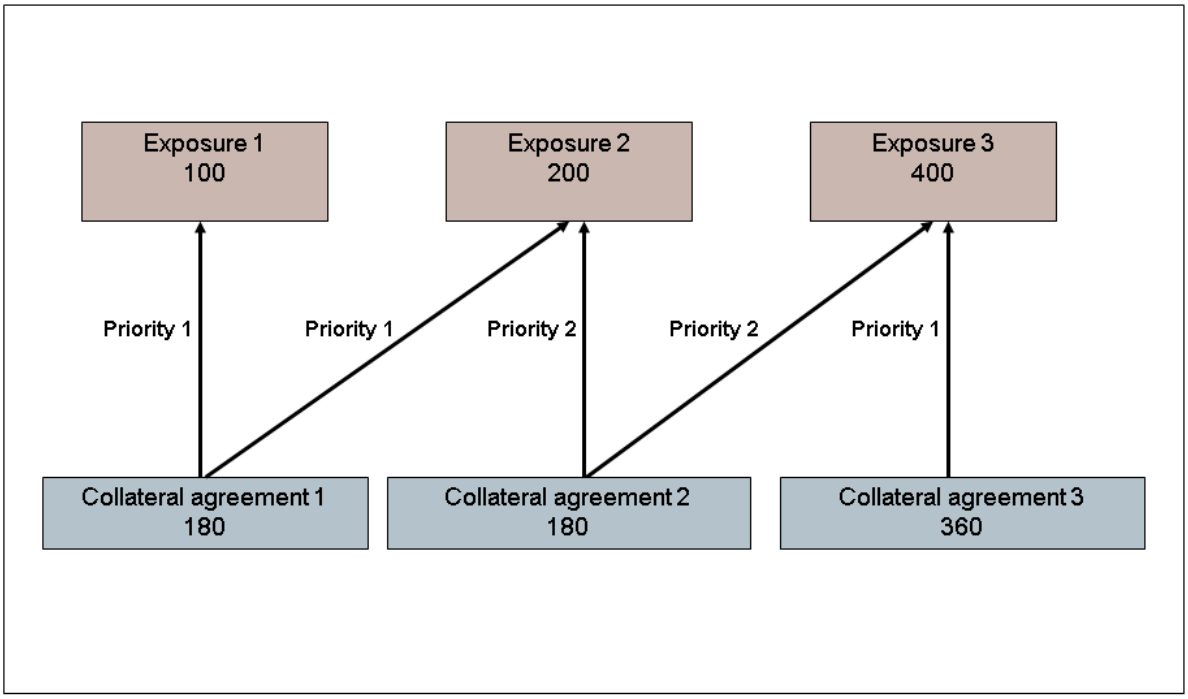
Example: Collateral Distribution in Counterparty/Issuer Risk and Country Risk [page 1216].

4.7.1.2.3.1.1.1 Example: Collateral Distribution in Counterparty/ Issuer Risk and Country Risk

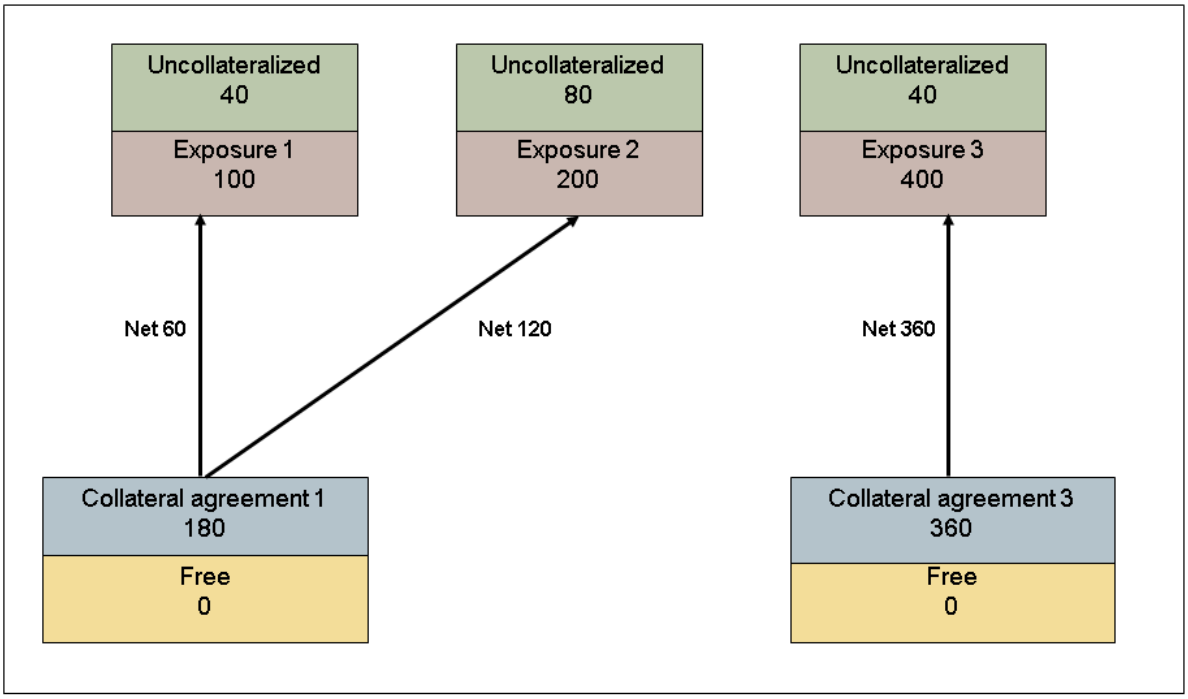
This simplified example of a collateral-exposure relationship without assets and portion granularity illustrates the logical process of pro rata collateral distribution, taking into account priorities at relationship level.

> **Note**
>
> The general collateral model of Bank Analyzer is more complex and has multilevel structures: The collateral comprises assets, collateral agreement contracts, and portions. During the calculation, the system generates elementary collateral instruments (ECIs).

The following figure shows three collateral agreements. Their values are to be distributed proportionally to three exposures in order to secure these:

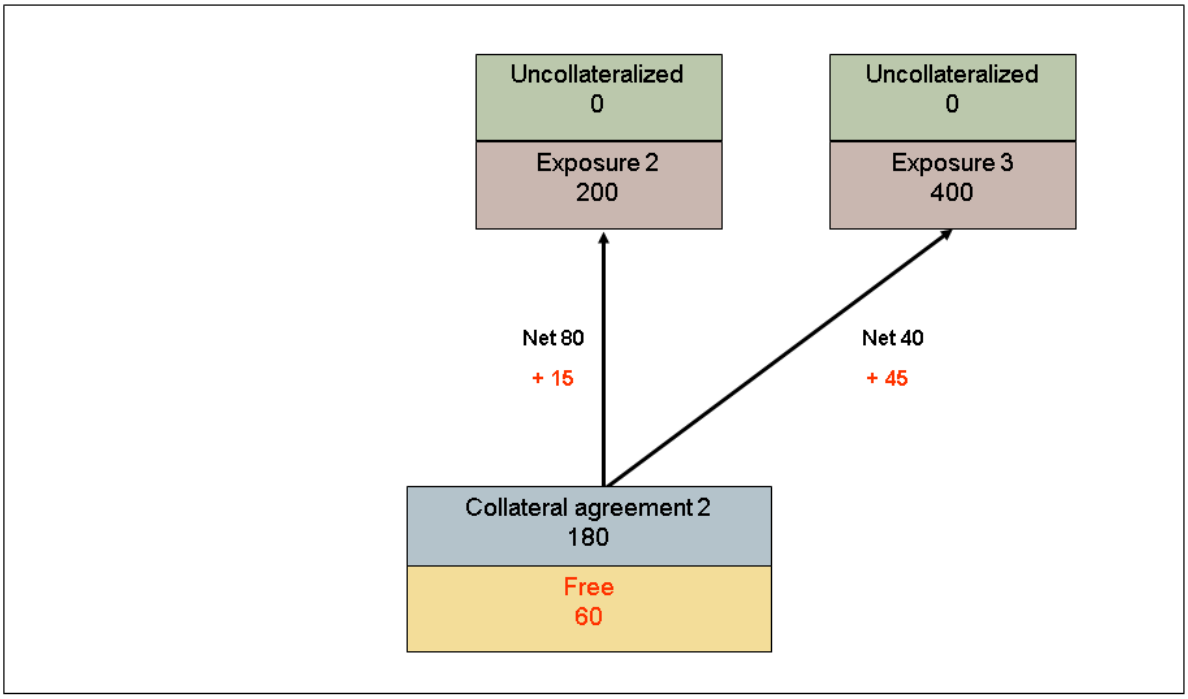


The distribution method first takes into account only the relationships between collateral agreements and exposures with the highest priority. The next figure examines only the priority 1 relationships and the **net amounts** of the collateral that the system distributes to these relationships:



Since collateral agreement 3 is only assigned to exposure 3, the system assigns the complete collateral value (360) to this relationship. Since collateral agreement 1 is assigned to multiple exposures, the system distributes the collateral value (180) to the relationships based on the amount of the exposures involved, proportionally with a ratio of 1 to 2.

The three exposures remain partially uncollateralized after the collateral distribution to the priority 1 relationships. The system now distributes the collateral that is linked with exposures through relationships of priority 2. The figure below explains this scenario:



If a collateral value is higher than the uncollateralized exposure, the net amount is automatically given a maximum limit at relationship level. The net amount that the system distributes to the relationship is no higher than the value of the exposure to be collateralized. In this example, the system distributes only the portion of the collateral value 2 to the partially collateralized exposures 2 and 3 that is needed to fully collateralize these (120). The distribution ratio of the net amounts is 2:1, in accordance with the portions of the second exposure (80) and third exposure (40) that are still uncollateralized.

> **Note**
>
> If an exposure is assigned multiple collateral items that would collateralize the exposure by too much overall, the system also limits the total net amounts distributed. The relevant collateral contributions are distributed proportionally according to the ratio of the collateral.

Since the net amounts to be distributed are limited in this example, a collateral value of 60 still remains. The system distributes this amount to both relationships as **gross amounts**. Before the gross amounts can be

calculated at relationship level, the total gross amount of the collateral must be calculated. This calculation is based on the following formula:

$$G = \min\left(C, \sum_i E_i\right)$$

Gross Risk

The gross amount of a collateral item corresponds to collateral value C. However, if the total of the exposures to be collateralized (E_i) is less than the collateral value, the gross amount (G) is reduced to the total of the exposure.

The gross amount (G_j) of a specific relationship between a collateral item and an exposure (E_j) is calculated based on the following formula:

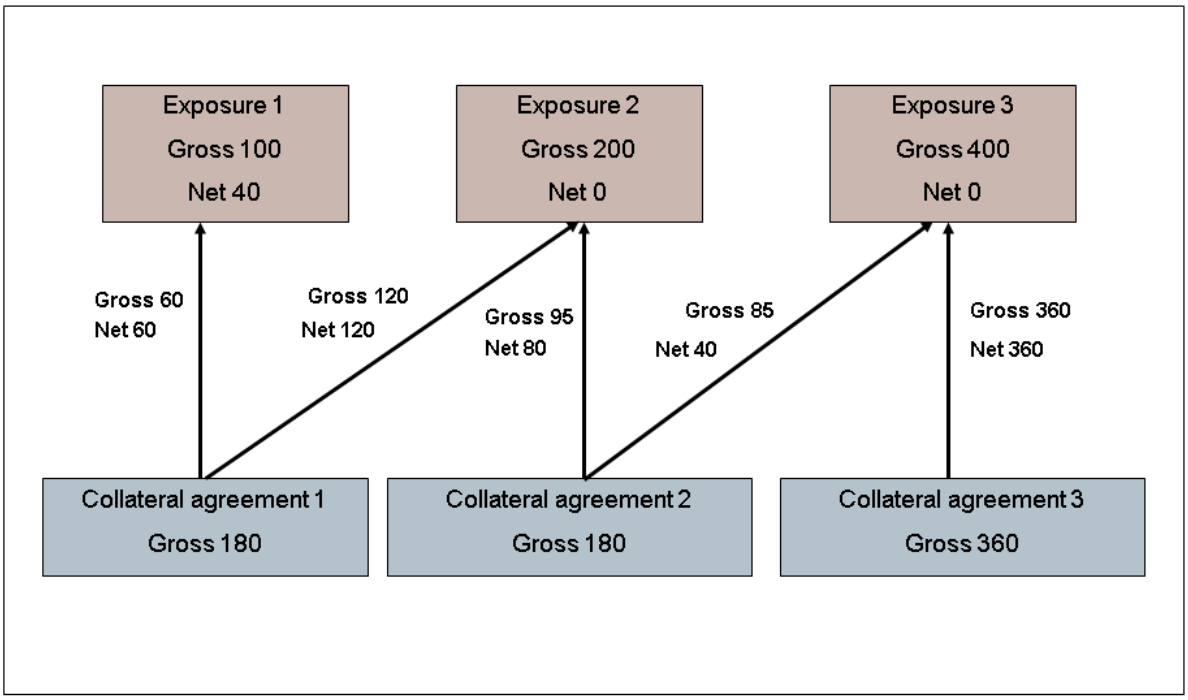
$$Gj = Nj + \left(G - \sum_i Ni\right) \frac{Ej - Nj}{\sum_i \left(Ei - Ni\right)}$$

Gross Attributable Amount

N_j is the net amount of the relationship. N_i is all distributed net amounts of the collateral. E_i is all exposures related to the collateral.

> **Note**
>
> If the total (N_i) of the distributed net amounts already equals the total of all the exposures assigned to this collateral, you cannot distribute a gross amount (G_j) to a relationship.

The following figure shows the resulting net and gross amounts:



The gross amount of an **exposure** is the original value of the credit exposure calculation basis. Its net amount is the uncollateralized portion.

The net amount of a **collateral-exposure relationship** is the portion of the collateral that the system has distributed to this relationship pro rata. The gross amount equals the net amount, unless the collateral still has collateral value available after all the assigned exposures have been fully collateralized. This is distributed to the relationship as gross amounts.

The gross amount of **collateral** is its original collateral value. This value is reduced to the total of the assigned exposures if it is lower than the collateral value.

Note

In an exposure ECI bundle, the distributed amount of each relationship between exposure, portion, and asset is calculated as follows: If no asset exists or the asset is 0, the distributed amount is equal to the distributed amount of the exposure-portion relationship. Otherwise, it is equal to the product value of the distributed amount of the asset-collateral relationship and the distributed amount of the portion-exposure relationship, divided by the total of the capital invested to secure the collateral.

4.7.1.2.3.1.2 Simultaneous Minimizing of EL/Unexpected Loss in IRB Approaches

The purpose of optimization is to maximize the capital ratio:

$$Capital\,ratio = \frac{(Tier\,1 + Tier\,2 - expected\,loss + provisions)}{(unexpected\,loss + market\,risk)\cdot 12.5} > 8\%$$

(1)

or

$$x = \frac{(Tier\,1 + Tier\,2 - expected\,loss + provisions)}{unexpected\,loss + market\,risk} > 1$$

(2)

In this process, the expected loss influences the capital ratio in the numerator, and the unexpected loss influences the capital ratio in the denominator.

Note the following:

- The smaller the expected loss, the larger the capital ratio. (If the expected loss can be offset against the provisions, the capital ratio remains unaffected. If the expected loss is greater than the provisions, then capital is deducted from the difference between Tier 1 and Tier 2. However, this is not implemented in Bank Analyzer.)
- The smaller the unexpected loss, the larger the capital ratio.

The principal objective is to maximize the capital ratio by minimizing the expected loss and unexpected loss. To be able to minimize the expected and unexpected loss at the same time, the following objective function can be constructed:

$$Min. \overset{!}{=} (1-\lambda)\cdot unexpected\,loss + \lambda \cdot expected\,loss$$

(3)

The parameter *lambda* defines the weighting of the expected loss and unexpected loss in the objective function.

Any value between 0 and 1 can be used for *lambda*. You can define the following:

- *lambda = 0*: This minimizes the unexpected loss only
- *lambda = 1*: This minimizes the expected loss only

If *lambda = 0.5*, the expected loss and the unexpected loss are equally weighted in the objective function (2). If the capital ratio (1) is 8 %, this gives the exact minimum of the capital ratio (1). The default value in Customizing is *0.5*. If the actual capital ratio is not 8%, then formula (4) delivers the minimum capital ratio as per equation (1):

$$\lambda = \frac{1}{1} + x$$

(4)

Where *X* is the result of equation (2).

This enables you to use an iterative method to determine the maximum capital. The iteration can start from a value of *x = 1* (corresponds to a capital ratio of 8 % or *lambda = 0.5*) for example. For this value of *x*, objective function (3) is minimized in the optimized distribution of collateral. The result of this minimization is a capital ratio of *x'*. You can then use *x'* to trigger the next iteration in which objective function (3) is minimized. In this way, you use iteration to obtain the required maximum capital ratio.

Due to the positive correlation between the expected and unexpected loss, this iteration method converges very quickly, which means that usually only a few iteration steps are required.

4.7.1.2.3.1.3 Example: Sorting for Bundle Category 1:m

Example showing the distribution of collateral instruments for sorting the data for bundle category 1:m (standardized approach and IRB approach).

Example

Your bank has lent EUR 500,000 to company A Borrower Inc, which has an annual sales figure of EUR 10m. Standard & Poor's has given the company a borrower rating of A, and your bank's own rating method gives a rating of VI. The loan still has 5 years to run, and an effective residual maturity of 3.45 years. The loan is a senior claim. The company has provided two collateral instruments: one is financial, and the other is physical collateral. The first collateral instrument is a bond to the amount of USD 80,000, which has an issue rating of AA and a residual maturity of 3.5 years. The second collateral instrument is residential real estate valued at EUR 450,000, which has a residual maturity of 2 years. The current exchange rate is EUR/USD = 0.98.

Business Parameters

Business parameters for the exposure (RM = residual maturity; M = maturity; S = annual sales; RCL = risk class of borrower; S&P = Standard & Poor's; IRM = internal rating model; Co = company; Sn = seniority)

Exposure	Amount	RM [Years]	M [Years]	S	RCL	Rating [S&P]	Rating [IRM]	Sn
E1	EUR 500,000	5	3,45	EUR 10,000,000	U	A	VI	Senior

Business parameters for the collateral instruments (CIT = collateral instrument; RM = residual maturity)

CIT	Collateralization	RM [Years]	Rating [S&P]	Exchange Rate [EUR/USD]
Financial collateral C1	USD 80,000	3.5	AA	0.98
Physical collateral C2	EUR 450,000	2	-	-

Formulas

Cardinality of the Problem

Number of Exposures *n*	Number of Collateral Instruments *m*
1	2

In this case, there are now two *collateral instruments* to take into account.

The following applies for all Basel II approaches:

$$\sigma_j \in \{\text{fin}, \text{RE}\} \quad : \quad j = 1,2 \quad : \quad \sigma_1 = \text{fin}, \sigma_2 = \text{RE}$$

$$\beta_{1j} = \left(1 - H_{1j}^{C} - H_{1j}^{FX}\right) \text{ (Equation 1)}$$

The expressions are as follows:

Standardized approach

$$EAD_{10} = \text{Max}\left(EAD_1 - \sum_{j=1}^{m} \beta_{1j} \cdot P_j, 0\right) = \text{Max}\left(EAD_1 - \sum_{j=1}^{2} \beta_{1j} \cdot P_j, 0\right)$$
$$= \text{Max}\left(EAD_1 - \beta_{11} \cdot P_1 - \beta_{12} \cdot P_2, 0\right)$$
$$EAD_{11} = \text{MAX}(\beta_{11} \cdot P_1, 0)$$
$$EAD_{12} = \text{MAX}(\beta_{12} \cdot P_2, 0) \quad \text{(Equation 2)}$$

$$RWA = r_{10} \cdot EAD_{10} + \sum_{j=1}^{m} r_{1j} \cdot EAD_{1j} = r_{10} \cdot EAD_{10} + \sum_{j=1}^{2} r_{1j} \cdot EAD_{1j}$$
$$= r_{10} \cdot EAD_{10} + r_{11} \cdot EAD_{11} + r_{12} \cdot EAD_{12} \quad \text{(Equation 3)}$$

IRB approach

$$EAD_{100} = \text{Max}\left(EAD_1 - \sum_{j=1}^{m} \frac{\beta_{1j} \cdot P_j}{C^{**,\sigma_j}}, 0\right) = \text{Max}\left(EAD_1 - \sum_{j=1}^{2} \frac{\beta_{1j} \cdot P_j}{C^{**,\sigma_j}}, 0\right)$$

$$= \text{Max}\left(EAD_1 - \frac{\beta_{11} \cdot P_1}{C^{**,fin}} - \frac{\beta_{12} \cdot P_2}{C^{**,RE}}, 0\right)$$

$$EAD_{110} = \text{Max}\left(\frac{\beta_{11} \cdot P_1}{C^{**,fin}}, 0\right)$$

$$EAD_{120} = \text{Max}\left(\frac{\beta_{12} \cdot P_2}{C^{**,RE}}, 0\right)$$

(Equation 4)

$$RWA = 12{,}5 \cdot LGD_{100} \cdot f(PD_{100}) \cdot EAD_{100} + \sum_{k=1}^{2} \sum_{l=1}^{0} 12{,}5 \cdot LGD_{1kl} \cdot f(PD_{1kl}) \cdot EAD_{1kl}$$

$$= 12{,}5 \cdot LGD_{100} \cdot f(PD_{100}) \cdot EAD_{100}$$

$$+ 12{,}5 \cdot LGD_{110} \cdot f(PD_{110}) \cdot EAD_{110} + 12{,}5 \cdot LGD_{120} \cdot f(PD_{120}) \cdot EAD_{120}$$

(Equation 5)

Calculation

The following applies for all Basel II approaches:

Haircuts (CIC = collateral instrument category; haircuts for fluctuations in value HC; haircuts for currency mismatch Hfx; haircuts for maturity mismatch t; total haircut b)

CIT	*H* C	*H* FX	τ	β
Financial collateral C1	4%	8%	0.7	0.616
Physical collateral C2	0%	0%	2/5	2/5

The haircuts for maturity mismatch

,

,

plus the haircuts for currency mismatch and value fluctuations give rise to the following:

$$\beta_{11} = \left(1 - H_1^C - H_{11}^{FX}\right) \cdot \tau_{11} = \left(1 - 4\% - 8\%\right) \cdot 0{,}7 = 0{,}616 \text{ ,}$$

$$\beta_{12} = \left(1 - H_2^C - H_{12}^{FX}\right) \cdot \tau_{12} = \left(1 - 0\% - 0\%\right) \cdot 0{,}4 = 0{,}4 \text{ .}$$

The *financial collateral* is listed in a foreign currency and relative to the reporting currency, which is euro. The currency translation is as follows:

$$P_1 = 0{,}98 \frac{EUR}{USD} \cdot 80.000\ USD = 78.400\ EUR$$

Standardized approach

Risk weights of the portions of the overall exposure

CIT	Risk weight	CIT
Uncollateralized	50%	Uncollateralized
Financial collateral C1	0%	Financial collateral C1
Physical collateral C2	40%	Physical collateral C2

First a check can be made to see if the transaction is over-collateralized. This is the case if

$$\sum_{j=1}^{2} \beta_{1j} \cdot P_j = \beta_{11} \cdot P_1 + \beta_{12} \cdot P_2 > EAD_1$$

applies. In our example, the following applies:

$$\sum_{j=1}^{2} \beta_{1j} \cdot P_j = 0{,}616 \cdot 78.400\ EUR + 0{,}4 \cdot 450.000\ EUR$$

$$= 228.294{,}4\ EUR \leq EAD_1 = 500.000\ EUR$$

Here the transaction is not over-collateralized, so both *collateral instruments* can be attributed to it.

Step 1:

The starting point is the total exposure, = *EAD 1*. If the adjusted value of the *financial collateral* is larger than the *EAD*, then the recursion has already finished, as in this case only the *financial collateral* has to be attributed. If the adjusted value of the *financial collateral* is smaller than the EAD, which is the case here, then the adjusted value of the *financial collateral* is deducted from the *EAD*, though it has to be ensured that the amount that is attributed is not greater than the exposure itself. This is done by using the maximum function:

$$\Delta_1 = \mathrm{Max}\left(\Delta_0 - \beta_{11} \cdot P_1, 0\right) = \mathrm{Max}\left(500.000\ EUR - 0{,}616 \cdot 78.400\ EUR, 0\right) = 451.705{,}6\ EUR$$

This amount is now the part of the exposure that is still unsecured after the *collateral* has been attributed to it. The secured portion of the exposure can be expressed as follows:

$$EAD_{11} = \Delta_0 - \Delta_1 = 0{,}616 \cdot 78.400\ EUR = 48.294{,}4\ EUR$$

Step 2:

Since there is a residual part that is unsecured, and an additional *collateral instrument* is available, the second *collateral item* is attributed to this unsecured residual part as follows:

$$\Delta_2 = \mathrm{Max}\left(\Delta_1 - \beta_{12} \cdot P_2, 0\right) = \mathrm{Max}\left(451.705{,}6\ EUR - 0{,}4 \cdot 450.000\ EUR_2, 0\right) = 271.705{,}6\ EUR$$ The value of the part of the exposure that is secured by the *physical collateral* is therefore the difference between the unsecured part of the exposure before the *physical collateral* was attributed to it, and the unsecured part after the physical collateral was attributed to it.

$$EAD_{12} = \Delta_1 - \Delta_2 = 451.705{,}6\ EUR - 271.705{,}6\ EUR = 180.000\ EUR$$

There is still a residual unsecured portion even after the second *collateral* item was attributed. This is treated as an exposure:

$$EAD_{10} = \Delta_2 = 271.705{,}6\ EUR$$

The unsecured portion of the total exposure, and the portions secured by the collateral instruments, as they result from the recursion process in the standardized approach

EAD 10	*EAD* 11	*EAD* 12
EUR 271,705.6	EUR 48,294.4	EUR 180,000

The *risk weights* are still to be calculated, and these are read from standard tables. This means that all the information needed to calculate the capital requirement has been collected, and the last step can now be carried out (equation 3):

$$RWA = 50\% \cdot 271.705{,}6\ EUR + 0\% \cdot 48.294{,}4\ EUR + 40\% \cdot 180.000\ EUR = 207.852{,}8\ EUR$$

IRB approach

Probability of default (PD); loss given default (LGD); level of over-collateralization (C**,s) and threshold values (TV) in the IRB approach

CIT	*PD*	*LGD*	*C* **,σ	*TV*
Unsecured	0.07%	45%	-	-
Financial collateral C1	0.07%	0%	100%	0%
Physical collateral C2	0.07%	35%	140%	30%

Additional parameters for function f

Exposure	A	B	Z
E1	0.12	0.24	50

The fact that the values of the collateral instruments adjusted by the haircuts have to exceed certain thresholds so that they can be attributed has to be reflected in the IRB approach. Specifically:

$$\frac{\beta_{12} \cdot P_{12}}{EAD - \beta_{11} \cdot P_1} = \frac{0{,}4 \cdot 450.000\ EUR}{500.000\ EUR - 0{,}616 \cdot 78.400\ EUR} = 39{,}85\% >= 30\%$$

Since the threshold value has been exceeded, the *physical collateral* can be attributed. When the program checks whether the threshold has been exceeded, the *levels of over-collateralization* have to be included as an additional adjustment to the current values of the *collateral instruments*.

$$\sum_{j=1}^{2} \frac{\beta_{1j} \cdot P_{1j}}{C^{**,\kappa_j}} = \frac{\beta_{11} \cdot P_1}{C^{**,\mathrm{fin}}} + \frac{\beta_{12} \cdot P_2}{C^{**,\mathrm{RE}}}$$
$$= \frac{0{,}616 \cdot 78.400\ EUR}{100\%} + \frac{0{,}4 \cdot 450.000\ EUR}{140\%}$$
$$= 176.865{,}83\ EUR < EAD = 500.000\ EUR$$

Therefore the exposure is not over-collateralized. This means that both *collateral instruments* are to be attributed. The process in which the *collateral instruments* are attributed to the exposure is the same as in the *standardized approach*. The only difference is that the *levels of over-collateralization* are now taken into account as well.

Step 1:

The starting point is the total exposure, = *EAD*. If the adjusted value of the *financial collateral*, inclusive of the respective *level of over-collateralization*, is larger than the *EAD*, then the recursion has already finished, as in this case only the *financial collateral* has to be attributed. If the adjusted value of the *financial collateral* is smaller, which is the case here, then the adjusted value of the *financial collateral* is deducted from the *EAD*.

$$\Delta_1 = \text{Max}\left(\Delta_0 - \frac{\beta_{11} \cdot P_1}{C^{**,m}}, 0\right) = 500.000\ EUR - 0{,}616 \cdot 78.400\ EUR = 451.705{,}6\ EUR$$

The secured but unguaranteed portion of the exposure results from the portion of the exposure that is still unsecured and unguaranteed after the collateral has been attributed, and the total exposure.

$$EAD_{110} = \Delta_0 - \Delta_1 = \frac{0{,}616 \cdot 78.400\ EUR}{100\%} = 48.294{,}4\ EUR$$

The second *collateral instrument*, the physical collateral, is then attributed to the portion that is still unsecured and unguaranteed:

$$\Delta_2 = \text{Max}\left(\Delta_1 - \frac{\beta_{12} \cdot P_2}{C^{**,RE}}, 0\right) = 451.705{,}6\ EUR - \frac{0{,}4 \cdot 450.000\ EUR}{140\%} = 323.134{,}17\ EUR$$

The portion of the total exposure that is secured by the second collateral instrument is then calculated as the difference between the amount that remains after the attribution of the first collateral item, and the amount that remains after the attribution of the second collateral item.

$$EAD_{120} = \Delta_1 - \Delta_2 = 128.571{,}43\ EUR$$

There is still a residual unsecured portion even after the second collateral item was attributed. This is treated as an exposure:

$$EAD_{100} = \Delta_2 = 323.134{,}17\ EUR$$

Unsecured portion, and the portions secured by the collateral instruments in the total exposure in the IRB approach

Exposure	*EAD* 100	*EAD* 110	*EAD* 120
E1	EUR 323,134.17	EUR 48,294.4	EUR 128,571.43

Now all the information is available that is required for calculating the capital requirement in the *IRB approach*. The result is (equation 5):

$$\begin{aligned} RWA &= 12{,}5 \cdot 45\% \cdot f(0{,}07\%) \cdot 323.134{,}17\ EUR + 12{,}5 \cdot 0\% \cdot f(0{,}07\%) \cdot 48.294{,}4\ EUR \\ &\quad + 12{,}5 \cdot 35\% \cdot f(0{,}07\%) \cdot 128.571{,}43\ EUR = 80.284{,}08\ EUR + 24.845{,}43\ EUR \\ &\quad = 105.129{,}51\ EUR \end{aligned}$$

4.7.1.2.3.1.4 Example: Sorting for Bundle Category n:1

Example showing the distribution of collateral instruments for the sorting of data for bundle category 1:m (standardized approach).

Example

Your bank has lent EUR 500,000 to the Berlin branch of company A Borrower Inc, which has an annual sales figure of EUR 10m. The loan has a residual maturity of 5 years, an effective residual maturity of 3.45 years, and is of senior priority. Standard & Poor's has given this branch of the company a borrower rating of A, and your bank's own rating model gives it a rating of VI. The Cologne branch of the company has annual sales totaling EUR 5m, and has borrowed EUR 250,000, which is a senior priority loan. The loan still has 2.5 years to run, and has an effective residual maturity of 2.05 years. Standard & Poor's has given this branch of the company a borrower rating of AA-, and your bank's own rating model gives it a rating of IV. The Munich branch, which has annual sales of EUR 1m and is rated by Standard & Poor's as BBB (your bank's own rating is IX) receives a loan of EUR 100,000, which is a subordinated claim. The loan has a residual maturity of 0.5 years, and an effective residual maturity of 0.2 years. The company provides a bond valued at EUR 800,000 as collateral for the three loans. Standard and Poor's rates this bond as AA. The residual maturity of the bond is 3.5 years.

Parameter

Business parameters for the exposure (RM = residual maturity; M = maturity; S = annual sales; RCL = risk class of borrower; S&P = Standard & Poor's; IRM = internal rating model; Co = company; Sn = seniority)

Exposure	Amount	RM [Years]	M [Years]	S	RCL	Rating [S&P]	Rating [IRM]	Sn
E1	EUR 500,000	5	3.45	EUR 10,000,000	U	A	VI	Senior
E2	EUR 250,000	2.5	2.05	EUR 5,000,000	U	AA-	IV	Senior
E3	EUR 100,000	0.5	0.2	EUR 1,000,000	U	BBB	IX	Junior

Business parameters for the collateral instruments (CIT = collateral instrument; RM = residual maturity)

CIT	Collateralization	RM [Years]	Rating [S&P]	Exchange Rate [EUR/USD]
Financial collateral C1	EUR 800,000	3.5	AA	-

Formulas

Cardinality of the Problem

Number of Exposures *n*	Number of Collateral Instruments *m*
3	1

There is only one collateral instrument for all three exposures.

The following applies for all Basel II approaches:

$$\sigma_j = \text{fin} : j = 1 : \sigma = \sigma_1 = \text{fin}$$

$$\beta_{i1} = \left(1 - H_1^C - H_{i1}^{FX}\right) \quad : \quad i = 1,\ldots,3 \text{ (equation 1)}$$

In both approaches, the collateral instrument is assigned to each exposure in succession, meaning that is assigned first to one exposure, and then the program checks whether any of the value of the collateral instrument is left over. If the collateral instrument can secure *i*-1 exposures, then there is a residual amount of:

Standardized approach

$$\text{Rest nach Anrechnung auf } i-1 \text{ Forderungen} = \text{Max}\left(P_1 - \sum_{k=1}^{i-1} \frac{EAD_s}{\beta_{s1}}, 0\right)$$

IRB approach

This residual amount can then be attributed to the *i*th exposure. However, it is only the amount adjusted by the haircuts (and possibly also the levels of over-collateralization) that is attributed, which gives:

Standardized approach

$$\text{Korrigierter Rest nach Anrechnung auf } i-1 \text{ Forderungen} = \text{Max}\left(\beta_{i1} \cdot \left(P_1 - \sum_{s=1}^{i-1} \frac{EAD_s}{\beta_{s1}}\right), 0\right)$$

IRB approach

$$\text{Korrigierter Rest nach Anrechnung auf } i-1 \text{ Forderungen} = \text{Max}\left(\frac{\beta_{i1}}{C^{**,F_1}} \cdot \left(P_1 - \sum_{s=1}^{i-1} \frac{EAD_s \cdot C^{**,F_1}}{\beta_{s1}}\right), 0\right)$$

Hence the unsecured part of the *i*th exposure is:

Standardized approach

$$EAD_{i0} = \text{Max}\left(EAD_i - \text{Max}\left(\beta_{i1} \cdot \left(P_1 - \sum_{s=1}^{i-1} \frac{EAD_s}{\beta_{s1}}\right), 0\right), 0\right)$$

IRB approach

$$EAD_{i00} = \text{Max}\left(EAD_i - \text{Max}\left(\frac{\beta_{i1}}{C^{**,F_1}} \cdot \left(P_1 - \sum_{s=1}^{i-1} \frac{EAD_s \cdot C^{**,F_1}}{\beta_{s1}}\right), 0\right), 0\right)$$ (equation 2)

And the unsecured portion is as follows:

Standardized approach

$$EAD_{i1} = EAD_i - \text{EAD}_{i0} = \text{EAD}_i - \text{Max}\left(EAD_i - \text{Max}\left(\beta_{i1} \cdot \left(P_1 - \sum_{s=1}^{i-1} \frac{EAD_s}{\beta_{s1}}\right), 0\right), 0\right)$$

IRB approach

$$EAD_{i10} = EAD_i - \text{EAD}_{i00} = \text{EAD}_i - \text{Max}\left(EAD_i - \text{Max}\left(\frac{\beta_{i1}}{C^{**,F_1}} \cdot \left(P_1 - \sum_{s=1}^{i-1} \frac{EAD_s \cdot C^{**,F_1}}{\beta_{s1}}\right), 0\right), 0\right)$$ (equation 3)

The risk weighted assets are now calculated as follows:

Standardized approach

$$RWA = \sum_{i=1}^{3} \left(r_{i0} \cdot EAD_{i0} + r_{i1} \cdot EAD_{i1}\right)$$ (equation 4)

IRB approach

$$RWA = \sum_{i=1}^{3} \left[12{,}5 \cdot LGD_{i00} \cdot f\left(PD_{i00}\right) \cdot EAD_{i00} + 12{,}5 \cdot LGD_{i10} \cdot f\left(PD_{i10}\right) \cdot EAD_{i10}\right]$$ (equation 5)

Calculation

The following applies for all Basel II approaches:

Haircuts (CIC = collateral instrument category; haircuts for fluctuations in value HC; haircuts for currency mismatch Hfx; haircuts for maturity mismatch t; total haircut b)

CIT	*H* C	*H* 1 ; *H* 2 ; *H* 3	*t t t*	*b b b*
Financial collateral C1	4%	-	0.7; 1; 1	0.672; 0.96; 0.96

The haircuts for this case can be determined beforehand. The results are as follows:

$$\beta_{11} = \left(1 - H_1^C - H_{11}^{FX}\right) \cdot \tau_{11} = \left(1 - 4\% - 0\%\right) \cdot 0{,}7 = 0{,}672$$

$$\beta_{21} = \left(1 - H_2^C - H_{21}^{FX}\right) \cdot \tau_{21} = \left(1 - 4\% - 0\%\right) \cdot 1 = 0{,}96$$

$$\beta_{21} = \left(1 - H_3^C - H_{21}^{FX}\right) \cdot \tau_{13} = \left(1 - 4\% - 0\%\right) \cdot 1 = 0{,}96$$

Standardized approach

Risk weights of the borrowers

Exposure	Risk weight *r*
E1	50%
E2	20%
E3	100%

Risk weights of the portions of the overall exposure

CIT	Risk weight *r*
Uncollateralized	50%
Financial collateral C1	0%

Two particular factors here point to a problem with the optimization process:

The secured portion of the *i*th exposure and the total exposure determine the amount of the unsecured part.

The risk weights by which the unsecured portions of the individual exposures are multiplied are too large

The collateral instrument is now to be distributed to each exposure in order to minimize the capital requirement. A more suggestive form of the equation for the capital requirement (equation 4) can be used here to be able to understand the problem:

$$RWA = \sum_{i=1}^{3} \left[r_{i0} \cdot EAD_{i0} + r_{i1} \cdot EAD_{i1}\right] = \sum_{i=1}^{3} \left[r_{i0} \cdot \left(EAD_i - EAD_{i1}\right) + r_{i1} \cdot EAD_{i1}\right]$$
$$= \sum_{i=1}^{3} r_{i0} \cdot EAD_i + \sum_{i=1}^{3} \left(r_{i1} - r_{i0}\right) \cdot EAD_{i1} = \sum_{i=1}^{3} r_{i0} \cdot EAD_i + \sum_{i=1}^{3} \left(r_{i1} - r_{i0}\right) \cdot \alpha_{i1} \cdot EAD_i$$

Where

$$\alpha_{10} + \alpha_{11} = \alpha_{20} + \alpha_{21} = \alpha_{30} + \alpha_{31} = 1$$

and

$$P_1 - \alpha_{11} \frac{EAD_1}{\beta_{11}} - \alpha_{21} \frac{EAD_2}{\beta_{21}} - \alpha_{31} \frac{EAD_3}{\beta_{31}} \geq 0$$

When the total is broken down into two subtotals, you can see that the first of the two subtotals is not based on the distribution of the collateral to the individual exposures. Therefore, the problem of minimizing the collateral requirement means that ij (portion of the exposure secured by the *i*th collateral instrument) is to be chosen in such

a way that the second total is minimized, whereby minimization here does not mean a minimization of the amount itself. This is clearly the case if those terms are given the greatest weighting for which

$$\Delta r_i := (r_{i1} - r_{i0}) < 0$$

applies, and for which the product $\Delta r_i \cdot \beta_{i1}$ is maximal. This is because this gives the largest negative contribution (the haircut "wastes the least amount of the collateral for capital *b* 's). Our example demonstrates this solution, in which the collateralization of each exposure is to be based on the plus or minus sign of the total summand and the value of the product specified. Firstly, for each exposure, the product of the risk weight and the total haircut of the collateral is to be calculated. This figure is then used in subsequent calculations.

Difference between the risk weights, the product of the risk weight and the haircut, and the resulting order in which the exposures are secured by the collateral instrument (highest priority = 1)

Exposure	*r r*	Δr_i x Total Haircut	Sequence
E1	0%-50%<0	50% x 0.672 = 0.336	2
E2	0%-20%<0	20% x 0.96 = 0.192	3
E3	0%-100%<0	100% x 0.96 = 0.96	1

Step 1:

$$EAD_{30} = \text{Max}\left(EAD_3 - \beta_{31} \cdot P_1\ , 0\right) = \text{Max}\left(100.000\ EUR - 0{,}96 \cdot 800.000\ EUR, 0\right) = 0$$
$$EAD_{31} = EAD_3 - EAD_{30} = 100.000\ EUR$$

The third exposure could be secured completely. The amount of the residual portion of the collateral instrument is:

$$\text{Rest}_3 = \text{Max}\left(P_1 - \frac{EAD_3}{\beta_{31}}, 0\right) = \text{Max}\left(800.000\ EUR - \frac{100.000\ EUR}{0{,}96}, 0\right) = 695.833{,}33\ EUR$$

Step 2:

$$EAD_{10} = \text{Max}\left(EAD_1 - \beta_{11} \cdot \text{Rest}_3, 0\right)$$
$$= \text{Max}\left(500.000\ EUR - 0{,}672 \cdot 695.833{,}33\ EUR, 0\right)$$
$$= \text{Max}\left(500.000\ EUR - 467.600\ EUR, 0\right) = 32.400\ EUR$$
$$EAD_{11} = EAD_1 - EAD_{10} = 500.000\ EUR - 32.400\ EUR = 467.600\ EUR$$

The second exposure is not completely secured. There is a residual amount of:

$$\text{Rest}_1 = \text{Max}\left(P_1 - \frac{EAD_{11}}{\beta_{11}} - \frac{EAD_{31}}{\beta_{31}}, 0\right)$$

$$= \text{Max}\left(800.000\ EUR - \frac{470.600\ EUR}{0{,}672} - \frac{100.000\ EUR}{0{,}96}, 0\right) = \text{Max}\left(-4.464{,}29\ EUR, 0\right) = 0$$

for securing the last exposure. This means that the last exposure remains completely unsecured. The last step should be performed anyway in order to be able to demonstrate the formula completely.

Step 3:

$$EAD_{20} = \text{Max}\left(\text{EAD}_2 - \beta_{21} \cdot \text{Rest}_1, 0\right) = \text{Max}\left(250.000\ EUR - 0{,}96 \cdot 0, 0\right)$$
$$= \text{Max}\left(250.000\ EUR - 0, 0\right) = 250.000\ EUR$$
$$EAD_{21} = EAD_2 - EAD_{20} = 250.000\ EUR - 250.000\ EUR = 0$$

The process for distributing the portions of the collateral instrument to the individual exposures is complete. All the information required for the determination of the capital requirement has been obtained, and the values can be placed in the formula (equation 4):

$$RWA = 50\% \cdot 32.400\ \text{EUR} + 0\% \cdot 467.600\ EUR + 20\% \cdot 250.000\ EUR + 0\% \cdot 0\ EUR$$
$$+ 100\% \cdot 0 + 0\% \cdot 100.000\ EUR = 66.200\ EUR$$

IRB foundation approach

Probability of default (PD); loss given default (LGD); level of over-collateralization (C**,s) and threshold values (TV) in the IRB approach

CIT	PD	LGD	C **,σ	TV
Unsecured (E1)	0.07%	45%	-	-
Unsecured (E2)	0.04%	45%	-	-
Unsecured (E3)	0.22%	75%	-	-
Financial collateral C1	0.07%	0%	100%	100%

Additional parameters for function f

CIT	A	B	Z
Financial collateral C1	0.12	0.24	50

In the IRB approach, the method for calculating the capital requirement uses the same formula as that of the standardized approach. The differences are that the levels of over-collateralization now have to be taken into account, and the default probabilities and default amounts are used instead of risk weights. Product $LGD_{100} \cdot PD_{100} \cdot \beta_{11}$ is to be taken into account in the determination of the order in which collateral is to be allocated to the exposures. The order that is determined is the same as the one used in the standardized approach. In our example, in which the collateral instrument is financial collateral, the level of over-collateralization is 100%. This means that the portions of the individual exposures are identical. However, the method is to be applied once again in its entirety for the IRB approach in order to show the effect the level of over-collateralization has on the calculations.

Step 1:

$$EAD_{300} = \text{Max}\left(\text{EAD}_3 - \frac{\beta_{31}}{C^{**,m}} \cdot P_1, 0\right) = \text{Max}\left(100.000\ EUR - \frac{0{,}96}{100\%} \cdot 800.000\ EUR, 0\right) = 0$$

$$EAD_{310} = EAD_3 - EAD_{300} = 100.000\ EUR$$

The third exposure could be secured completely. The amount of the residual portion of the collateral instrument is:

$$\text{Rest}_3 = \text{Max}\left(P_1 - \frac{EAD_{310}}{\beta_{31}/C^{**,m}}, 0\right)$$

$$= \text{Max}\left(800.000\ EUR - \frac{100.000\ EUR}{0{,}96/100\%}, 0\right) = 695.833{,}33\ EUR$$

Step 2:

$$EAD_{100} = \text{Max}\left(\text{EAD}_1 - \frac{\beta_{11}}{C^{**,m}} \cdot \text{Rest}_3, 0\right) = \text{Max}\left(500.000\ EUR - \frac{0{,}672}{100\%} \cdot 695.833{,}33\ EUR, 0\right)$$

$$= \text{Max}\left(500.000\ EUR - 467.600\ EUR, 0\right) = 32.400\ EUR$$

$$EAD_{110} = EAD_1 - EAD_{100} = 500.000\ EUR - 32.400\ EUR = 467.600\ EUR$$

This exposure is not completely secured. There is a residual amount of:

$$\text{Rest}_1 = \text{Max}\left(P_1 - \frac{EAD_{110}}{\beta_{11}/C^{**,m}} - \frac{EAD_{310}}{\beta_{31}/C^{**,m}}, 0\right)$$

$$= \text{Max}\left(800.000\ EUR - \frac{467.600\ EUR}{0{,}672/100\%} - \frac{100.000\ EUR}{0{,}96/100\%}, 0\right) = \text{Max}\left(-4.464{,}29\ EUR, 0\right) = 0$$

for securing the last exposure. This means that the last exposure remains completely unsecured. The last step should be performed anyway in order to be able to demonstrate the formula completely.

Step 3:

$$EAD_{200} = \text{Max}\left(\text{EAD}_2 - \frac{\beta_{21}}{C^{**,m}} \cdot \text{Rest}_1, 0\right) = \text{Max}\left(250.000\ EUR - \frac{0{,}96}{100\%} \cdot 0, 0\right)$$

$$= \text{Max}\left(250.000\ EUR - 0, 0\right) = 250.000\ EUR$$

$$EAD_{210} = EAD_2 - EAD_{200} = 250.000\ EUR - 250.000\ EUR = 0$$

The process for distributing the portions of the collateral instrument to the individual exposures is complete. All the information required for the determination of the capital requirement is now available (equation 5):

$$RWA = 12{,}5 \cdot 45\% \cdot f\left(0{,}07\%\right) \cdot 32.400\ EUR + 12{,}5 \cdot 0\% \cdot f\left(0{,}07\%\right) \cdot 467.600\ EUR$$

$$+ 12{,}5 \cdot 45\% \cdot f\left(0{,}04\%\right) \cdot 250.000\ EUR + 12{,}5 \cdot 0\% \cdot f\left(0{,}04\%\right) \cdot 0\ EUR$$

$$+ 12{,}5 \cdot 45\% \cdot f(0{,}04\%) \cdot 250.000\ EUR + 12{,}5 \cdot 0\% \cdot f(0{,}04\%) \cdot 0\ EUR$$

$$= 8.049{,}92\ EUR + 29.900{,}52\ EUR = 37.950{,}44\ EUR$$

4.7.1.2.3.2 Determination of the Capital Requirement

The capital requirement is determined for every uncollateralized and non-guaranteed part of the exposures and for every collateralized and/or guaranteed portion of the exposure. The Basel Committee has defined the following exceptions: Equity holdings in the IRB approach; defaulted transactions in the Advanced IRB Approach; Securitizations whose capital requirements must be calculated separately at the single deal level or for the securitization as a whole.

The requirements of the Basel consultation paper often leave room for decisions. Hence, there is some freedom in the determination for the capital requirement (especially in the distribution of the hedging instruments to exposures [page 1210]).

The determination of the capital requirement distinguishes between the standardized and IRB approaches. A further subdivision into the corresponding subapproaches is usually not necessary for describing the determination of the capital requirement.

- According to the requirements of the Basel Committee, no distinction is made between the expected loss and the unexpected loss in the standardized approach. Instead, the risk-weighted assets include the expected loss as well as the unexpected loss. Hence the solutions aims at distributing the collateral instruments in the standardized approach to the assigned exposures in such a way that the risk-weighted assets and thus the regulatory capital requirements are minimized. This minimization leads to the highest capital ratio possible.
- In the IRB approach, in contrast, the separation makes it possible to individually control to which extent the distribution of the collateral instruments is to be used to reduce the expected loss and to which extent it is to reduce the unexpected loss.

To determine the capital requirement, the system primarily uses the key figures and risk parameters determined in calculation level 1:

- Eligibility of Hedging Instruments
- Formation of bundles
- Determination of haircut Hc for collateral instruments
- Determination of haircut Hfx for currency mismatches
- Determination of the maturity mismatch factor
- Calculation of the EAD
- Determination of the collateral value *P*
- Determination of the PD and LGD for the IRB approaches
- Adjustment of the risk weight for the standardized approaches.
- Determination of the credit value adjustment (CVA)

4.7.1.2.3.2.1 Credit Value Adjustment (CVA)

Definition

During credit value adjustment (CVA), the valuation of a portfolio of transactions is adjusted to the medium market value of a counterparty valuation.

To illustrate the effects of netting agreements to the supervisory authority, you can determine the CVA in parallel with netting agreements and also without including netting agreements. When you determine the CVA without including netting agreements, the system handles all transactions assigned to a netting agreement as if they were not assigned to this netting agreement. You control the parallel determination of the CVA in the Customizing settings for the calculation method.

Prerequisites

You have made the required settings to determine the CVA in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Settings for Basel II* > *Edit CVA Weighting Factors*.

Procedure

A preliminary run and a main run are executed to determine the CVA. These runs use different calculation methods.

Preliminary Run

The system determines the following figures in the preliminary run:

- You define the number of CVA-relevant transactions by importing a corresponding key figure for the transaction. You can exclude certain transactions (such as transactions with central counterparties) in the system.
- Relevant master rating for the determination of the weighting factor `w(i)`
  - To calculate the weighting factor `w(i)` at business partner level, the system derives the master rating of the business partner from its external rating (provided an external rating exists). If there is more than one external rating for a business partner, the second-worst rating is used. In this case, the system analyzes the external ratings regardless of the selected calculation approach.
  - If there is no external rating for the business partner, an internal rating can be used if you are using the IRB approach. You control this using a regulatory setting.
- Weighting factor `w(i)` for each business partner
  You can enter a weighting factor ID in Customizing for the localization parameters. In the Customizing activity "Edit CVA Weighting Factors", you can assign a master rating grade to weighting factor `w(i)` for each weighting factor ID.
- Weighting factor `w(ind)` for index CDS
- Maturity `M_CVA` for each exposure, individual CDS business partner, and CDS index
- Unsecured EAD for each exposure

- Nominal values for each individual CDS business partner and CDS index
- Discount factor for each exposure, individual CDS business partner, and CDS index
- Redistribution values for the redistribution of CVA capital requirements from portfolio level to single transaction level for each exposure

A credit default swap (CDS) that is used to hedge CVA risks is itself handled as a CVA-relevant exposure. A split is used for this purpose. This split divides the CDS into two parts:

- A hedge part (rating and business partner information is copied for this part from substructure CD_REF_DEB)
- An exposure part (the CVA/CDS type is deleted for this part)

The fixed characteristic `/BA1/C62BUPCVA` (*Relevant Business Partner for CVA Calculation*) was assigned to the result node category `HKCL2`. The CVA determination in the main run uses this characteristic to determine the relevant business partner. This means that a different business partner can be used for the exposure part than is used for the hedge part. You make the relevant Customizing settings to ensure that the EAD and the redistribution value for the exposure part are determined in the preliminary run.

Main Run

The main run uses the values determined in the preliminary run to determine the aggregated key figures and the capital requirements for CVA at portfolio level and at single transaction level.

4.7.1.3 Starting the Credit Exposure Run

Prerequisites

If you want to include one of the general calculation and valuation methods (*Account Pooling*, *Facility Distribution*, *Determination of the Free Line for Facilities*, *Collateral Distribution*, or *Default Determination*), you need to have applied these functions to your data beforehand, so that the results are available for further processing. The results of the general calculation and valuation methods are stored in the Results Data Layer (RDL) [page 1364] or in the Result Database (RDB) [page 1823] and are made available to the Credit Exposure run by means of the primary data sources [page 1593] and secondary data sources. *Account Pooling, Facility Distribution* and *Determination of the Free Line for Facilities* provide results at object level (account) and *Collateral Distribution* provides results at object/collateral level.

Instead of using the collateral distribution function provided in the calculation and valuation methods in the Credit Exposure run you can use a BAdI implementation to distribute collateral. To do so, you need to have entered the relevant settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *General Settings* > *Settings for Distribution Methods*.

> **Recommendation**
>
> We recommend that you use the collateral distribution function that is part of Credit Exposure.

In Customizing for *Bank Analyzer*, you have already entered the settings required for data selection under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *General Settings* > *Selection*. When you enter these

settings, you define which data the worklist for the existing credit exposure runs is to contain. The selection process includes direct and indirect links between objects, such as the links that exist between collateral and exposures.

So that the results of the credit exposure run can be stored in the RDL or RDB, you need to have made the relevant settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Settings for Runs in Results Data Layer* and *Settings for Runs in Result Database*.

Context

In the credit exposure run (CE run), the program selects all the transactions that match the valuation date and that are also relevant for Credit Exposure. The exposure at default, for example, is then calculated for these transactions.

During the credit exposure run, contracts (or objects) are processed, such as loans or trading transactions. Some of the contracts might have complex relationships with one another that have to be taken into account during data processing and selection. In the credit exposure run, those transactions are taken into account that meet the selection criteria.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Start Credit Exposure Run*.

   The system displays the *Credit Exposure Run* screen.
2. Enter a business record date. The default value is today's date.
3. Enter the run type.
4. If required, change the time stamp of the access to the data in the Source Data Layer (SDL) and Results Data Layer (RDL). The default values are today's date and the current system time.
5. Enter a filter variant ID to define how messages are stored and displayed in the run administration function.

   You define filter variants in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages*.
6. Choose  .

Results

You can display the result in the overview of credit exposure runs [page 1251].

4.7.1.4 Postprocessing of Credit Exposure Runs

Use

In the credit exposure run, it may be the case that the input data that was taken from the Source Data Layer (SDL) and used in the calculation process was incorrect. If so, you need to correct the data in the SDL and recalculate the transaction in question. You can use the postprocessing function to do this without having to start the entire run again. The system deletes the results data stored in the Results Data Layer (RDL) or Result Database (RDB) and replaces it with the new results data.

You can also include additional transactions for a run that was already completed. This enables you to add the data for important transactions not included in those selected in the original run to the results already stored in the RDB or RDL.

> **Note**
>
> Note that there is a difference between postprocessing a run and restarting a run. You can use the postprocessing function to correct a complete run by changing just a subset of the data or transactions. If you restart the run, the system continues calculating a (mass) run that terminated due to technical reasons.

Prerequisites

To be able to postprocess a run, it must have the status *Run Completed Successfully* (End). Runs that are completed (Cmtd) or that are flagged for deletion (Del.) cannot be postprocessed.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Method* > *Credit Risk* > *Credit Exposure* > *Postprocess Credit Exposure Run*.
2. In the *Selection Criteria for Run data* group enter the following data:
    1. By entering the relevant combination of external run ID, key date, run category, and run version, you define which run you want to postprocess.
    2. Specify the time stamp for the start of the time period for which data is to be selected for postprocessing. If you choose *Run Admin*, the system uses the timestamp of the last change made the run specified above.
    3. If required, change the time stamp of the access to the data in the Source Data Layer (SDL) and Results Data Layer (RDL). The default values are today's date and the current system time.
    4. If required, set the *Delta Selection* indicator. The delta selection function selects from the SDL only those transactions that were changed between the start time stamp and the SDL time stamp in the SDL. Note that the results for data that has been deleted in the meantime are not changed.
    If you have not set the indicator, the postprocessing function selects the data for the start time point from the Source Data Layer (SDL), and sets the corresponding results in the Results Data Layer (RDL) to *invalid* or deletes them in the Result Database (RDB). The data is selected again and calculated for the SDL time stamp. The new results are stored in the RDL or RDB. This includes any deleted objects.

For more information, see the document Key Date Query and Delta Query [page 267].

3. In the *Settings for Data Selection* data group, enter the InfoSet. The InfoSet defines which data is to be selected. and hence enables you to override the Customizing settings for the selection group ID stored in the run type. Once you have selected an InfoSet, choose Enter. You can then choose the *SelectionCharacteristics* pushbutton to enter additional selection criteria for the InfoSet specified.
4. Choose *Execute* to start the postprocessing run.

The system stores the changed data in the RDL or RDB. The run then has the status *Run in Postprocessing* (InPP). You can display the result in the overview of credit exposure runs [page 1251].

You can also start postprocessing directly from the overview of credit exposure runs.

4.7.1.5 CVPM Run for Credit Exposure

Use

You can use this function to schedule the credit exposure run or credit exposure stress run as a process and have it executed by the system.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. The same prerequisites apply to executing the CVPM run as apply for the Credit Exposure Run [page 1238].
2. You have also made the necessary settings for CVPM in Customizing under *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager*.
3. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *CVPM Run for Credit Exposure*, or *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Support for Stress Tests* > *CVPM Stress Run for Credit Exposure*.

2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Run Type
     Specify the run type.
   - Combined Scenario
     Specify the combined scenario. This is required only if you start the CVPM run for a stress run.
   - Process partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. The standard system enters the time at which you call the transaction.
   - Technical Settings
     You can use a filter variant or start the CVPM run as a test run.
3. Choose  .

> **Note**
>
> To display the CVPM run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.7.1.6 Schedule Manager

Definition

A tool in Bank Analyzer that you can use to execute and monitor complex business processes, such as period-end closing. When you schedule a run, you always have to specify the program with which the system calls the run. There is a separate program for each run category in an application.

The table below shows for which runs and run categories the Schedule Manager can be used, and their corresponding programs.

Application	Run Category	Program
Account Pooling	Account Pooling [page 383]	/BA1/DE_GM_MAIN_COMP
Account Pooling	Stress Run for Account Pooling [page 429]	/BA1/DE_GM_MAIN_COMP_SCEN
Determination of the Free Line for Facilities	Determination of the Free Line for Facilities [page 395]	/BA1/DE_GM_MAIN_FL
Determination of the Free Line for Facilities	Stress Run for Determination of the Free Line for Facilities [page 429]	/BA1/DE_GM_MAIN_FL_SCEN

Application	Run Category	Program
Distribution of Facilities	Distribution of Facilities [page 399]	/BA1/DE_GM_MAIN_RLV
Distribution of Facilities	Stress Run for Facility Distribution [page 429]	/BA1/DE_GM_MAIN_RLV_SCEN
Collateral Distribution	Collateral Distribution [page 404]	/BA1/DE_GM_MAIN_SV
Collateral Distribution	Stress Run for Collateral Distribution [page 429]	/BA1/DE_GM_MAIN_SV_SCEN
Determination of Default	Determination of Default [page 411]	/BA1/RA_DFLT4280_RUN
Determination of Default	Stress Run for Determination of Default [page 429]	/BA1/RA_DFLT4280_RUN_SCENARIO
Fair Value Effectiveness Test [page 1036]	Effectiveness Test Run [page 1039]	/BA1/R8_FVHE_MAIN_RUN
Fair Value Effectiveness Test	Generation of Time Series for Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MAIN_RUN
Fair Value Effectiveness Test	Modifying a Time Series for the Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MODIFY
Cash Flow Hedge Analysis [page 1107]	Subvaluation Run [page 1128]	/BA1/R5_CFHA_BTC_RUN
Cash Flow Hedge Analysis	Aggregation Run [page 1129]	/BA1/R5_START_BATCH_AGGR
Cash Flow Hedge Analysis	Joint call of Valuation Run [page 1111] and Aggregation Run [page 1123]	/BA1/R5_EXECUTE_CFHA
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169]	/BA1/R9_VPM_BTC_ID_RUN
Portfolio Fair Value Hedge	Portfolio Item Run (Merge Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_RUN
Portfolio Fair Value Hedge	Period-End Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PE_RUN
Portfolio Fair Value Hedge	Period-Start Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PS_RUN
Portfolio Fair Value Hedge	Define End of Valuation Period	/BA1HM/RAH_M_PFVH_SET_VP_END
Portfolio Fair Value Hedge	Adjust Initial Portfolio Fair Value Hedges for Next Valuation Period	/BA1HM/RAH_M_PFVH_ADJUST
Portfolio Fair Value Hedge	(Re-) Designation of Initial Portfolio Fair Value Hedges	/BA1HM/RAH_M_PFVH_DESIG

Application	Run Category	Program
Credit Exposure [page 1192]	Credit Exposure Run [page 1238]	/BA1/R2_CRA_MAIN_CRE
Credit Exposure	Credit Exposure Stress Run [page 1261]	/BA1/R2_CRA_MAIN_CRE_SCEN
Credit Exposure	Credit Exposure Run for Revolving Retail	/BA1/RR2_CRE_REVRET
Credit Exposure	Approximate Stress Run [page 1274]	/BA1/RR2_CRE_SCENARIO
Credit Exposure	Aggregation Run for Ad Hoc Calculation [page 1276]	/BA1/RR2_SCEN_AGGR
Country Risk [page 1326]	Country Risk Run [page 1336]	/BA1/R2_CRE_MAIN
Historical Database [page 1478]	Historization Run for Data Layers [page 1481]	/BA1/R6_HIST_RUN
Historical Database	Historization Run for Bank's Models [page 1484]	/BA1/R6_HIST_RUN_MODEL
Historical Database	Determine Default Rates [page 1488]	/BA1/R6_CF_DEFAULT_4210
Historical Database	Determine Average Default Rates [page 1491]	/BA1/R6_CF_DEFAULT_4220
Historical Database	Determine Default Figures [page 1493]	/BA1/R6_CF_DEFAULT_KEYFIGURES
Historical Database	Exporting Data to In-House Models [page 1498]	/BA1/R6_MOD_RUN
Historical Database	Calculation of Migration Matrices [page 1496]	/BA1/R6_CF_MIGRATIONS
Historical Database	Stress Run for Supplying Models with Data [page 1505]	/BA1/R6_MOD_RUN_SCENARIO
Historical Database	Stress Run for Supplying Data [page 1506]	/BA1/R6_FDB_RUN_SCENARIO
Regulatory Reporting Interface	Processing Framework Run Copying the Start ID	/BA1/S1_COPY_STARTID
Regulatory Reporting Interface	Processing Framework Run Creating Runs	/BA1/S1_CREATE_RUN
Regulatory Reporting Interface	Processing Framework Run Release Run	/BA1/S1_RELEASE_RUN

Application	Run Category	Program
Regulatory Reporting Interface	Processing Framework Run Creating a Worklist	/BA1/S1_CREATE_WRKL
Regulatory Reporting Interface	Processing Framework Run Processing a Run	/BA1/S1_PROCESS
Regulatory Reporting Interface	Data Extraction Runs	/BA1/S3_EXTRACT_RUN
Limit Manager [page 1529]	Limit Utilization Run [page 1551]	/BA1/R3_UTILIZATION_RUN_DPF

More Information

Schedule Manager

4.7.1.7 Run Administration

Definition

Run administration includes the following functions:

- Execute or create run
- Display an overview of runs
- Display Application Log
- Edit run
- Manage run
- Replace run
- Select run for archiving
- Delete run
- Log of the deletion function

> **Note**
>
> The above functions are not all available for each application. For more information, see the application-specific documentation.

Use

The following table lists the runs available for each application:

Application	Run
General Calculation and Valuation Methods [page 376]	Account Pooling [page 383] or Account Pooling (CVPM) Facility Distribution [page 399] or Facility Distribution (CVPM) Determination of the Free Line for Facilities [page 395] or Determination of the Free Line (CVPM) Collateral Distribution [page 404] or Collateral Distribution (CVPM) Determination of Default [page 410] Stress Runs [page 429]: • Stress run for account pooling or stress run for account pooling (CVPM) • Stress run for facility distribution or stress run for facility distribution (CVPM) • Stress run for determination of the free line or stress run for determination of the free line (CVPM) • Stress run for collateral distribution or stress run for collateral distribution (CVPM) • Stress run for determination of default
Credit Risk [page 1191]	Credit Exposure Run [page 1238] or Credit Exposure Run (CVPM) Country Risk Run [page 1336] (prototype for country risk) **Stress run:** Credit Exposure Stress Run [page 1261] or Credit Exposure Stress Run (CVPM)

Application	Run
Historical Database [page 1478]	**Version management:** Run for generating historical data for data layers [page 1481] Run for generating historical data for the bank's own models [page 1484] Uploading of Files [page 1486] **Calculation functions:** Determining Default Rates [page 1488] Determining Average Default Rates [page 1491] Determining Default Figures [page 1493] Calculation of Migration Matrices [page 1496] **Data retrieval:** Run for supplying data to models [page 1498] Downloading of Files [page 1500] **Stress tests:** Stress Run for Supplying Models with Data [page 1505] Generating Scenario Data in the Source Data Layer [page 1506]
Generic BI Data Extraction [page 1638]	Testing the BI Extractor [page 1639] BI extraction run **Note** Extraction runs are created and executed in SAP NetWeaver Business Intelligence (BI). The system displays information about extraction runs in run administration of Bank Analyzer.
Regulatory Reporting Interface	Data Extraction Runs
Limit Manager [page 1529]	Limit Utilization Run [page 1551]
Fair Value Effectiveness Test for Hedging Relationships [page 1036]	Effectiveness Test Run
Cash Flow Hedge Analysis [page 1107]	Valuation Run [page 1128] Subvaluation Run Aggregation Runs [page 1129]

Application	Run
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169] Portfolio Item Runs [page 1172] • Period-end run for portfolio items (subledger scenario) • Period-start run for portfolio items (subledger scenario) • Portfolio item run (merge scenario)

> **Note**
>
> For some of the Bank Analyzer components, you can use the Schedule Manager to schedule and control jobs. If you use multiple applications, you can define the sequence in which the runs are to be carried out. For more information, see Schedule Manager [page 437].

More Information

Status Overview for Run Administration [page 1248]

4.7.1.7.1 Status Overview for Run Administration

Run Status	Description	Technical Key
Crtd	Run created	IR001
End	Run finished	IR002
Canx	Run cancelled	IR003
Del.	Deletion flag	IR004
Rvke	Deletion flag revoked	IR005
Strt	Run started	IR006
Cmtd	Run completed	IR007
RFIN	Run finished	IR008
RINC	Run finished but not complete	IR009
Rel.	Run released for limit	IR010
PPrc	Run was postprocessed	IR011

Spcd	Reset release of limit	IR012
InPP	Run in postprocessing	IR013
LVAZ	Archiving flag was cleared	IR014
Inct	Run potentially inconsistent	IR015
RCA	Run was completely archived	IR016
RPA	Run was partially archived	IR017
AF	Flag for archiving	IR018
RCM	Run was changed manually	IR020
ROBS	Run is obsolete	IR021
ROFC	*Run Obsolete* flag was cleared	IR022
RRHA	Released for Hedge Accounting	IR023
CRHA	Release for Hedge Accounting was reset	IR024
FRHA	Hedge Accounting run was fetched	IR025
RRRR	Released for restart	IR026
RRMC	Run is ready for manual change	IR030
LRST	Run was started	IR031
RCR	Completion of run was reset	IR050
RI01	Released for PA costing	IR101
RI02	Released for PA planning	IR102
RIBW	Released for BI extraction	IR109
SEL1	Selection in process	IR110
SEL2	Selection finished	IR111
PRP1	Preprocessing in process	IR112
PRP2	Preprocessing complete	IR113
PFM1	Portfolio model in process	IR114
PFM2	Portfolio model finished	IR115
POP1	Postprocessing in process	IR116

POP2	Postprocessing complete	IR117
CPR1	Counterparty risk run created	IR118
CPR2	Counterparty risk run complete	IR119
PFM3	Portfolio model canceled	IR120
DLFL	Deletion flag	IR121
DLRV	Clear deletion flag	IR122
TYPC	Run type changed manually	IR123
RSIM	Results were imported	IR124
CPRL	Counterparty risk run sealed	IR125
INCM	Not deleted completely	IR126

4.7.1.7.2 Special Features of Run Administration in Credit Exposure

Use

Run administration manages information about runs. This information can include, for example, the date of the run, the version of the run based on a key date, the user, the time and date the user started the run, the name of the user who made the latest changes to the run, and the time and date the user made those changes, the run category, and the current status of the run. From the overview of runs, you can also branch to the detail logs of the results stored in the Results Data Layer (RDL) or Result Database (RDB) and to the application log for the run in question. In Credit Exposure, the system creates a detail log for each calculation level and for each run.

Prerequisites

You need to have started at least one run in Credit Exposure.

Features

Run administration in Credit Exposure contains the functions described below. To access these functions, on the SAP Easy Access screen choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure*, and the following functions:

- Starting the Credit Exposure Run [page 1238]
- *Run Administration* > *Overview of Credit Exposure Runs*
- *Run Administration* > *Deletion Report*
- *Run Administration* > *Log of the Deletion Function*: Provides an overview of the runs that were deleted.

4.7.1.7.3 Overview of Credit Exposure Runs

Use

This report enables you to use the to display the runs you have carried out. In the detail log for a run, you can change the status of the run manually. For example, you can set a run to completed, or flag a run for deletion. The detail log contains additional information, and you can navigate from the detail log to the master data in the Source Data Layer (SDL) [page 100].

Prerequisites

You have already started at least one run.

Activities

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Run Administration* > *Overview of Credit Exposure Runs* and display the overview for the runs required.

To display the overview of Credit Exposure stress runs, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Support for Stress Tests* > *Overview of Credit Exposure Stress Runs*.

Selection

Use the following input fields to select the run you require:

- External ID of the run
- Key date
- Run category

- Version of run
- System status

Action

Double-click a run to display the information about the run.

Depending on the run, the status of the run, and where you store results data, the system displays different pushbuttons that you can use to navigate to other functions.

> **Note**
>
> You define whether results data is stored in the Results Data Layer (RDL) [page 1364] or in the Result Database (RDB) [page 1823] in Customizing for *Bank Analyzer* under *Processes and Methods* > *General Calculation and Valuation Methods* > *Storage of Results* > *Define Results Storage*.

Navigation Options in the Detail Log

Navigation Using Pushbuttons	Use
	Switches from change mode to display mode.
	Switches from display mode to change mode.
	Change the status of the run to *Deletion Flag* (`Del.`). You have to flag a run for deletion before it can be deleted (see also Deletion of Credit Exposure Runs [page 1254]).
	Changes the status of the run to *Run Completed* (`Cmtd`).
	Resets the status of a run.
	Can be used to navigate to the overview of the result objects stored in the RDL. In the overview of the result objects you can display more information about each result object, branch to online detail reporting [page 1255] and change the status of a result object.
	Displays the application log for a run. The filter variant ID you entered before you started the run defines which types of messages are displayed in the application log.
	You can use a filter in the log to restrict the number of messages that are displayed. This is particularly useful if the log contains a large number of messages and you want to display a certain subset of them.
	Restarts the run.

Navigation Using Pushbuttons	Use
	Displays the timestamp collections that were used when the run was started.
	Displays additional information about the results data stored in the RDL.
	Displays information about postprocessing [page 1240], such as the timestamp collection and the selection characteristics.
	Enables you to start postprocessing.
•  •  • 	Displays the detail views for the respective calculation level of the runs stored in the RDB. You can enter selection criteria in the dialog box that appears so that the system displays the transaction required. **Note** To be able to switch between the individual levels, you need to have assigned the predefined role `Valuation Object (KEY 10)` to the characteristics. For more information, see Roles of Characteristics [page 23].
Selection Criteria (in the detail view for a calculation level)	Choose *Selection Criteria* to call the detail log again, and change the selection criteria as required. The selection criteria are first applied to the current calculation level.
 (in the detail view for a calculation level)	Select a characteristic to which you assigned an appropriate role. You can then use this characteristic as a key to identify a primary object from the Source Data Layer. For more information, see Roles for Characteristics [page 23]. Then choose the pushbutton or the characteristic. The system then branches to the display of data for the primary object. **Example** To display the basic data for the financial transaction, click on the ID of the financial transaction.
 (in the detail view for a calculation level)	Select one or multiple rows in the detail log, and choose the pushbutton. For each row selected, the system displays an overview of all three levels.
 (in the detail view for a calculation level)	You can also use the normal processing functions in the ALV.

4.7.1.7.4 Deletion of Credit Exposure Runs

Use

You use the deletion report to delete individual runs, or to delete all the runs that you have flagged for deletion. You can also carry out a test run for the report. The system records a log of the deletion process for each run.

The deletion function works differently, depending on where you save results data:

- If you defined that results data is to be stored in the Results Data Layer (RDL), the system deletes only the administrative data for the runs in Credit Exposure. The results of the runs are retained in the RDL.
- If you defined that results data is to be stored in the Result Database (RDB), the system deletes the run data completely.
  Note that once you have deleted the data, the data is deleted permanently and cannot be recovered.

Prerequisites

The runs that you want to delete have to have the system status *Flagged for Deletion*. You can set and clear the deletion flag on the Overview of Credit Exposure Runs [page 1251] screen.

Activities

1. You call the report for actual runs on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Run Administration* > *Deletion Function*.
   You call the report for stress runs on the *SAP Easy Access* under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Support for Stress Tests* > *Deletion Report for Stress Runs*.
2. Make the following settings in the *Selection Criteria of Run* data group:
   - External ID of the run
   - Key date
   - Version of run
3. Choose one of the following *Control Parameters*:
   - Set the *Delete Immediately* checkbox to delete all the runs that are selected straight away.
   - Set the *Test Run* checkbox to display all the runs that can be deleted.
4. Choose *Execute*.
   - If you deleted the selected runs immediately, these are displayed in the .
   - If the run was a test run, the runs that can be deleted are shown in the ALV. You cannot delete any of these runs from the list.
   - If you did not set any checkbox, the runs that are to be deleted are shown in the ALV. From this list, you can delete the entire worklist or just individual runs. Once the runs are deleted, the system regenerates the list.

4.7.1.8 Online Detail Reporting for Credit Exposure

Use

Online detail reporting enables you to check the calculation of the credit exposure by analyzing a smaller set of data online. Looking at a smaller volume of data makes it easier to understand the calculation process. The online run uses the Customizing for the run types but with the following differences:

- In online detail reporting, you can use your own selection criteria to define which transactions are to be analyzed, rather than having to use the criteria stored in the Customizing for selection groups, as is the case for the credit exposure runs scheduled as batch jobs. These selection criteria are completely independent of those stored in Customizing for the selection group. For example, you can restrict the selection of data to that for particular transactions.
- You start the online detail reporting for one calculation method at a time.
- Results are calculated for one bundle at a time. For more information, see the documentation about the bundling service. Once a bundle has been processed by the bundling transformation function, the system displays the transformed bundle in the calculation screen.
- Since the analysis is based on a small volume of data, you do not need to use parallel processing here.
- Unlike the logs for credit exposure runs, those for online detail reporting contain detailed interim results from the calculation run.
- The results of online detail reporting are displayed only, and are not stored in the Results Data Layer (RDL) **or** Result Database (RDB).

Prerequisites

You need to have entered all the Customizing settings for Credit Exposure under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure*. The settings in the sections *Settings for Runs in Results Database* and *Settings for Runs in Results Data Layer* are not required for online detail reporting.

Procedure

Selection

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Online Reporting* > *Online Detail Reporting for Credit Exposure*.
2. Enter a business record date. The default value is today's date.
3. If required, change the time stamp for access to the data of the Source Data Layer (SDL) and the RDL. The default values are today's date and the current system time.
4. If required, specify a filter variant ID for message storage in the application log. You define filter variants in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages*.
5. If you want to start the online detail reporting for a stress test [page 426] you must also enter the ID of the relevant scenario.

6. Enter the run type and the calculation method.
7. In the *Settings for Data Selection* data group, enter the InfoSet. The InfoSet defines which data is to be selected. and hence enables you to override the Customizing settings for the selection group ID stored in the run type.
   Once you have selected an InfoSet, choose `Enter`. You can then choose the *Selection Characteristics* pushbutton to enter additional selection criteria for the InfoSet specified.

   > **Note**
   >
   > Before you execute the run, you can display the Customizing settings relevant for the selection by choosing *Customizing*.

8. Choose *Execute* to start data selection.

   > **Note**
   >
   > Online detail reporting uses internal variants so that you can call up the analyses again at a later point in time. The system automatically saves the parameters that were used. Note, however, that the internal variants are client-dependent.

   The system displays an overview of the selected data.

Action

In the overview list of the selected data, you have the following options:

Action	Use
Application Log	Displays the application log. The application log contains data only if you have started at least one calculation run.
Fair Value Server Log	Displays the detail log [page 652] of the valuations in the Fair Value Server (FVS).
Customizing	Displays the relevant Customizing settings for Credit Exposure.
Display SDL Objects	Displays the data of the SDL object, such as the basic data of the financial instrument.
Display Preliminary Runs	Displays the relevant preliminary runs. By double clicking a preliminary run you can display an overview of the results objects.
Double-click an object (for example, *bundle* or *contract*) in the dialog structure	Starts the calculation for the selected object.
Context menu (right mouse button), for example on the transaction ID in the dialog structure	Displays the data of the SDL object, such as the basic data of the financial instrument.

Display Eligible Collateral Only *Display All Collateral* (top right-hand part of the screen)	If you switch the display mode from *Display Eligible Collateral Only* to *Display All Collateral*, you can display the relationships of all financial transactions on the tab pages, and not just those for eligible collateral.
The tab pages in the results screen are based on the approach used, and contain interim results.	• Level 0 contains the interim results from calculation level 0. • Level 1 contains the interim results from calculation level 1. • Level 2 contains the final results from calculation level 2. If, for example, balance-sheet netting has taken place for the selected bundle, then the tab pages *Level 1: Contract* (interim results from the netting of balance-sheet assets) and *Level 1: Relations* are relevant.
Details (on the tab page)	Branches to the detailed view for an individual row.
Test Run (on the tab page)	Where appropriate, you can branch from the results view to the test mode of the Module Editor. In this function, based on the calculation level and the underlying primary object, the system branches to the data enrichment and calculation modules defined in the Customizing for the relevant combination of run type and calculation method.
Display Complex Key Figures (on the tab page)	Displays an overview of the complex key figures that were used.
Display for Level (on the tab page)	Displays the delta of the complex key figures for the current level.
Display for Module (on the tab page)	Displays the delta of the complex key figures for the relevant module.

You can use the display function in online detail reporting to display the interim results of a Credit Exposure run as follows: Proceed as follows:

1. In the overview of credit exposure runs [page 1251] choose the detail view for a run.
2. Choose *RDL Viewer*.
3. In the overview of results objects for Credit Exposure select a row and then choose *Branch to Online Detail Reporting*.

> **Note**
>
> You can use this function only if you store results in the Results Data Layer (RDL) [page 1364].

4.7.1.9 Stress Tests

Use

To meet the requirements of Basel II, banks that use the IRB approach must have sound stress test methods that enable them to assess their total capital adequacy. The banks are responsible for configuring the stress tests. The tests must be both meaningful and conservative.

Stress tests are also of interest to banks that use the standardized approach if, for example, migration analyses are to be carried out for external ratings. Banks that use the standardized approach are also required to assess their capital adequacy. Stress tests can be used for this purpose.

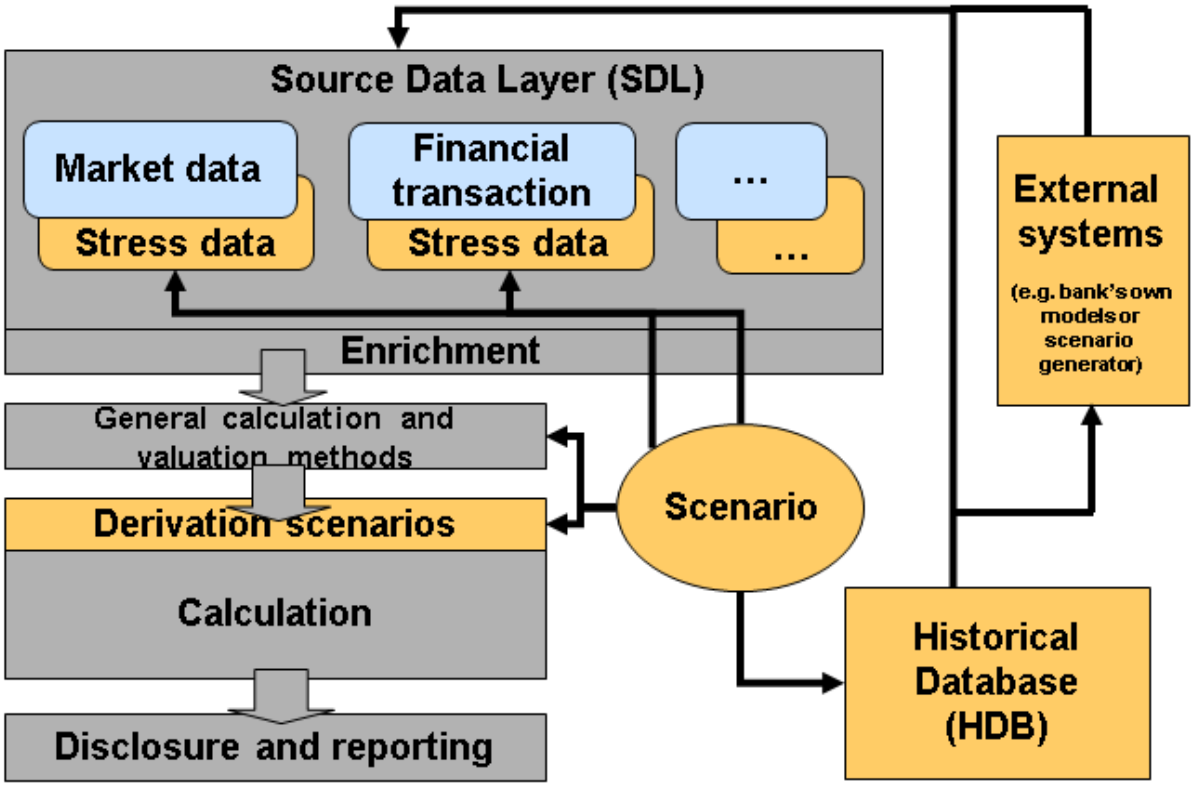
Stress tests use input data and control parameters that have been modified to reflect the (possibly hypothetical) result of a negative change in the economic conditions.

The following are examples of stress tests for Basel II:

- Modification of classic market data, such as exchange rates, interest rates, or security prices, and the effects of these on the market value of derivatives
- Modification of the PD and LGD as the result of a crisis in a particular sector or country
- Worsening of the ratings of counterparties in particular countries

Integration

The figure below shows which Bank Analyzer functions are used in stress tests.



Framework for Mapping Stress Tests

- Stress test scenarios in the Source Data Layer (SDL), for example:
  - Assumption of a worsening credit rating
  - Modified maturities
  - Modified positions
  - Modified market data

  For more information, see Scenario [page 1731] and Retrieving Stress Data [page 1734].
- Stress test scenarios in the calculation process (using derivation or using modules from the module editor), for example:
  - Country-specific rating changes

  You can carry out stress runs for the general functions and for the credit exposure calculation. For more information, see Support for Stress Tests in General Calculation and Valuation Methods [page 429] and Credit Exposure Stress Run [page 1261].
- Stress test scenarios are supported by the Historical Database (HDB), for example:
  - Modification of the input data used in banks' own models
  - Modification of position data in the Source Data Layer

  The HDB contains stress runs for banks' own models. For more information, see Support for Stress Tests in the Historical Database [page 1505].

You define which Bank Analyzer functions are involved in a stress test by creating a stress test scenario (see Stress Test Scenarios [page 428]). Here you can also combine multiple scenarios. By specifying whether factors are to be stressed implicitly or explicitly, for example, you define which scenario is depicted in a stress run.

4.7.1.9.1 Stress Test Scenarios

Definition

Different scenarios can be used for stress tests. These can be categorized as follows:

- Market data scenarios
  These are based on the use of modified classic or generic market data.
- Class determination scenarios
  These are based on the use of primary object data that has been modified.
- Modification of the input data used in banks' own models

When market data scenarios and class determination scenarios are used, the calculation process uses modified input data, which is stored along with the original data in the Source Data Layer (SDL). For more information, see Scenario [page 1731].

Use

To be able to map scenarios that reflect crises in particular regions or sectors, the scenarios do not always have to include all transactions and business partners during a calculation run. Instead, they can be limited to particular groups or individual entities. The following application areas are available for scenarios:

- **Local**
  Scenarios that are applied locally use data that is manipulated locally. This means that attributes are changed at *single transaction* or *business partner* level.
- **Global**
  Scenarios that are used globally apply modified values to the entire portfolio.
- **Subportfolio**
  Scenarios that are used for subportfolios apply modified values to a group of single transactions.

Integration

Bank Analyzer provides an infrastructure that enables you to combine stress test scenarios You define the combination of scenarios in scenario administration by assigning multiple single or composite scenarios to a composite scenario.

You define composite scenarios in Customizing for *Bank Analyzer* under *Infrastructure* > *General Scenario Administration* > *Edit Composite Scenarios*. You can still change these settings after your system has gone live. You do this on the *SAP Easy Access* screen under *Bank Analyzer* > *Infrastructure* > *General Scenario Administration*.

A composite scenario can contain only one scenario from each data category for scenarios, for example a market data scenario and a business partner scenario.

4.7.1.9.2 Credit Exposure Stress Run

Use

The Basel II requirements stipulate that banks using the IRB approach must have sound methods for stress testing. The Basel II calculation process provides stress test results that are not based on real data, but on simulated data (stress data).

A Basel II calculation run whose results are not used to calculate the capital requirement for reporting purposes is therefore referred to as a **simulation run**. A calculation run whose results are used for regulatory reporting is referred to as an **update run**. A simulation run is different from an update run in that the calculation itself is based on input data or control parameters that have been modified.

Features

If you want to carry out stress tests as part of the credit exposure calculation, you can use either stressed initial data (stressed market data and stressed primary object data) from the Source Data Layer (SDL) or use derivation rules to stress the data. You specify which option you want to use by means of the stress test scenario entered (see Stress Test Scenarios [page 428]).

> **Recommendation**
>
> Use modified primary objects from the SDL and modified market data for stress tests.

Stress Tests Based on Stressed Market Data and Stressed SDL Data

The data you use in your stress tests can be modified fictitious market data (market data scenarios) or modified fictitious data for primary objects in the SDL (class determination scenarios). You can also use a combination of the two. For more information, see Scenario [page 1731].

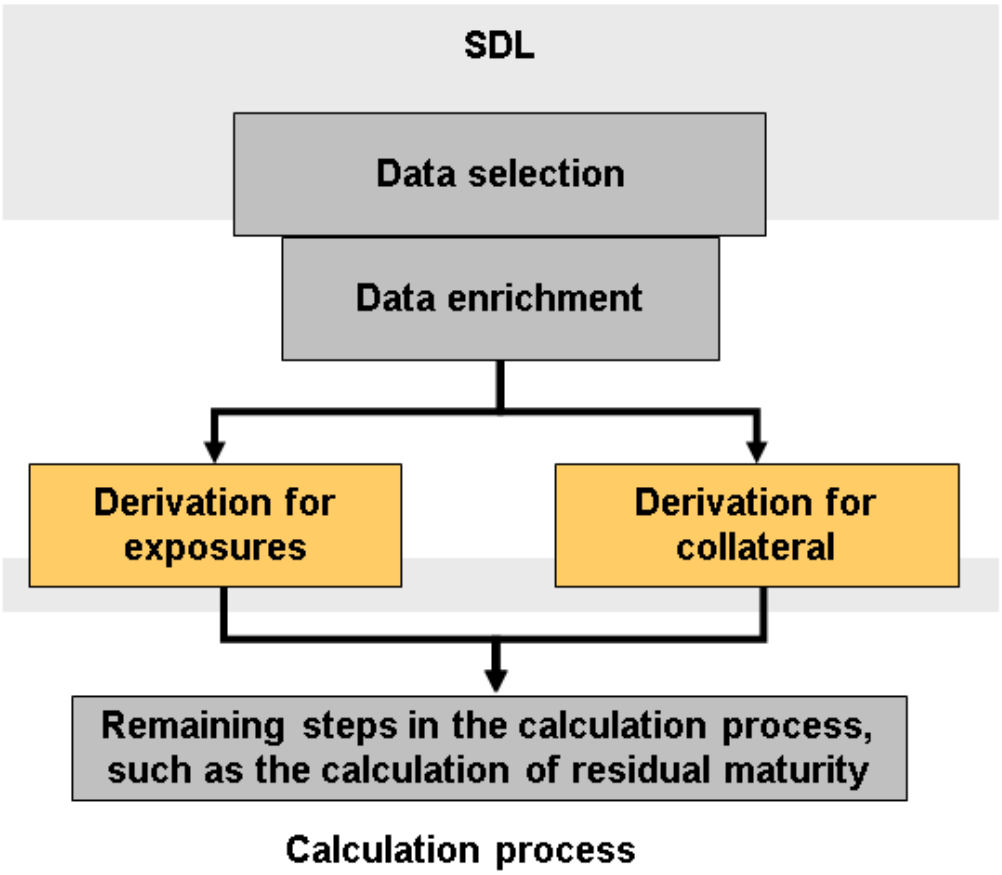
Prerequisites

Make sure that the following prerequisites are met:

- The Foundation component contains stress data for market data. The SDL contains stress data for primary objects. For more information, see Retrieving Stress Data [page 1734].
- You have made the following settings in Customizing for *Bank Analyzer*:
  1. Under *Infrastructure* > *General Scenario Administration*, you have created scenarios and assigned these to a composite scenario.
  2. Optional: Under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Module Editor* > *Edit Modules*, you have created the modules that you want to use in the credit exposure stress run.
  3. Optional: Under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Scenario* > *Define Modules and Localization Categories for Scenarios*, you have assigned modules or a localization category (or both) to the scenarios that you want to use in the credit exposure stress run.
- These prerequisites also apply to the update run for calculating the credit exposure. For more information, see Starting the Credit Exposure Run [page 1238].

Stress Tests Based On Derivation Rules

When stress tests are based on derivation rules, data such as transaction data and partner data is first selected from the SDL. Then the attributes of the primary objects (characteristics) are modified according to simple derivation rules.



Process of a Stress Test Based on Derivation Rules

Prerequisites

You have made the following settings in Customizing for *Bank Analyzer*:

1. Under *Infrastructure* > *General Scenario Administration*, you have created scenarios and assigned these to a composite scenario. Note that the scenario assigned to the modules is assigned to this composite scenario by means of data category `CRE_D` (scenarios in CRE calculation).
2. Optional: Under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Module Editor* > *Edit Modules*, you have created the modules that you want to use in the credit exposure stress run.
3. Optional: Under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Scenario* > *Define Modules and Localization Categories for Scenarios*, you have assigned modules or a localization category (or both) to the scenarios that you want to use in the credit exposure stress run.

4. Under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Scenario* > *Derivation for Scenarios*, you have made the appropriate settings for the derivation process.
   You can still change the rule entries after your system has gone live. You do this on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Support for Stress Tests* > *Edit Rule Entries for Derivation Steps*.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Support for Stress Tests* > *Start Credit Exposure Stress Run*.
2. Start a stress run for the credit exposure calculation in the same way as for an update run (see Starting the Credit Exposure Run [page 1238]).
3. Enter the ID of the composite scenario that contains the scenario that you want the system to apply.

The results of the stress calculation run are stored in the Results Data Layer (RDL) or in the Result Database (RDB) in the same way as those generated by a normal calculation run. However, the results from the stress tests can still be identified. You can display the results in run administration. You do this on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Support for Stress Tests* > *Overview of Credit Exposure Stress Runs*, as described in the Overview of Credit Exposure Runs [page 1251].

You can also delete the stress calculation runs in the same way as credit exposure runs You do this on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Support for Stress Tests* > *Deletion Report for Stress Runs*, as described in the Deletion of Credit Exposure Runs [page 1254].

The results of the stress calculation run on level 2 of the credit exposure calculation are then used to create the worklist for extracting stressed results from Bank Analyzer to SAP NetWeaver Business Intelligence (BI).

4.7.1.9.2.1 Credit Exposure Stress Run with Online Selection

Use

Using this function, you can specify an online selection for the credit exposure stress run. This allows you to specify any selection, without first having to enter selection settings in Customizing. In this way you can simulate the effect of a commercial event on a given region, for example.

Prerequisites

The same prerequisites apply to the credit exposure stress run with online selection as apply to the credit exposure stress run and the credit exposure run (see Credit Exposure Stress Run [page 1261] and Starting the Credit Exposure Run [page 1238]). The only difference is that you need not make **any** settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *General Settings* > *Selection*.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Support for Stress Tests* > *Credit Exposure Stress Run with Online Selection*.

1. Enter a business key date. The default value is today's date.
2. If required, change the time stamp for access to the data of the Source Data Layer (SDL) and the Result Data Layer (RDL). The default values are today's date and the current system time.
3. If required, specify a filter variant ID for message storage in the application log. You define filter variants in Customizing for *Bank Analyzer*under *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages*.
4. Enter the ID of a composite scenario.
5. Specify the run type.
6. In the *Settings for Data Selection* data group, enter the InfoSet. The InfoSet defines which data is to be selected. The InfoSet allows you to select specific transactions without making settings in Customizing for selection.

   Once you have selected an InfoSet, choose *Enter*. You can then choose the pushbutton to enter additional selection criteria for the InfoSet specified.

   > **Note**
   >
   > Before you execute the run, you can display the Customizing Settings relevant to the run type by choosing .

7. Choose to start data selection.

Result

The system stores the results of a credit exposure stress run with online selection in the same way that the results of a normal calculation run are stored - in the Results Data Layer (RDL) or the Result Database (RDB). However, the results can still be identified as stress test results.

You can display the results in run administration. You do this on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Support for Stress Tests* > *Overview of Credit Exposure Stress Runs*, as described in the Overview of Credit Exposure Runs [page 1251].

4.7.1.9.3 Stress Tests for Credit Risk Scenarios

Use

You can use the following processes to prepare and run stress tests for credit risk scenarios and to display the results of these tests:

- Approximate stress run
- Ad hoc calculation
- Credit exposure stress run (Basel II)

- Credit exposure stress run with online selection (Basel II)

> **Note**
>
> For more information about the availability of *Credit Risk Stress Tests*, see SAP Note 2316265.

The following table shows which scenarios can be processed by the stress runs:

	Approximate Stress Run	Ad Hoc Calculation	Credit Exposure Stress Run	Credit Exposure Stress Run with Online Selection
Credit Risk Scenario	Yes	Yes	Yes, if the credit risk scenario is part of a composite scenario.	Yes, if the credit risk scenario is part of a composite scenario.
Credit Risk Scenario Set	Yes	No	Yes, if the credit risk scenario set is part of a composite scenario.	Yes, if the credit risk scenario set is part of a composite scenario.
Composite Scenario	Yes, but the run processes only the credit risk scenario contained in the composite scenario.	No	Yes	Yes
Non-Credit Risk Scenario (scenarios with other data categories, such as market data, business partner)	No	No	Yes	Yes

Prerequisites

General Prerequisites

You have made the following settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Risk Stress Tests* > *Configuration*:

- *Define Characteristics for Scenario Segments* (optional)
- *Edit Risk Parameter Shift Types* (optional)
- *Assign Key Figures to Risk Parameter Shift Types*

Process-Specific Prerequisites

Credit Exposure Stress Run/Credit Exposure Stress Run with Online Selection

- You have included the credit risk scenario or the credit risk scenario set in a composite scenario. You do this in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Risk Stress Tests* > *Configuration* > *Edit Composite Scenarios*.
  For more information about composite scenarios, see the *Integration* section in Stress Test Scenarios [page 428].
- Optional: You have created the modules that you want to use in the credit exposure stress run in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Module Editor* > *Edit Modules*.
- Optional: You have assigned to the scenarios the modules or localization category (or both) that you want to use in the credit exposure stress run in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Scenario* > *Define Modules and Localization Categories for Scenarios*.

For more process-specific prerequisites, see the documentation for the relevant stress run.

Process

1. **Definition**
   Create a credit risk scenario [page 1267] and the corresponding entities:
   - Scenario segments
   - Scenario segmentations
   - Risk parameter specifications

   You can group a single credit risk scenario with other credit risk scenarios in a credit risk scenario set [page 1273].

   Make these settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Risk Stress Tests* > *Configuration* or on the user interface in SAP NetWeaver Business Client or SAP NetWeaver Portal under *Credit Risk Stress Tests* > *Definition*.
2. **Calculation**
   To run a stress test for a credit risk scenario, execute one of the following stress runs as described in the accompanying documentation:
   - Approximate stress run [page 1274]
   - Ad hoc calculation [page 1276]
   - Credit exposure stress run [page 1261]
   - Credit exposure stress run with online selection [page 1263]

   Credit risk scenarios are calculated with the help of the credit exposure stress run or the credit exposure stress run with online selection based on the same processes as the Basel II stress test solution. However, you can also calculate credit risk scenarios if you have the license for *SAP Interactive Credit Risk Analysis*.

The following table shows where you can execute stress runs:

	Approximate Stress Run	Ad Hoc Calculation	Credit Exposure Stress Run	Credit Exposure Stress Run with On-line Selection
SAP Easy Access (Bank Analyzer system)	Yes	No	Yes	Yes
Background Processing (Bank Analyzer system)	Yes	No	Yes	Yes
NetWeaver Portal (UI)	No	Yes	No	No
NetWeaver Business Client (UI)	No	Yes	No	No

3. **Results**
   For more information about displaying and storing the results of credit risk stress tests, see the documentation for the corresponding stress run.

More Information

For more information, see the Business Content documentation for Bank Analyzer under **Credit Risk Stress Tests** and the NetWeaver Portal Content documentation under Business Package for Financial Services Analytical Banking 8.0 [page 1280].

For more information about the relevant roles, see Credit Risk Analyst in SAP NetWeaver Business Client [page 1278] and Credit Risk Analyst in SAP NetWeaver Portal [page 1281].

4.7.1.9.3.1 Credit Risk Scenario

Definition

You can use a credit risk scenario to simulate the effects of stressed risk parameters on a bank's portfolio, for example, counterparty default or a lowering of the rating of the government bonds of certain countries.

You edit credit risk scenarios on the user interface in SAP NetWeaver Business Client or SAP NetWeaver Portal under *Credit Risk Stress Tests* > *Definition* > *Manage Credit Risk Scenarios* or in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Risk Stress Tests* > *Configuration* > *Edit Credit Risk Scenarios*.

Structure

A credit risk scenario comprises at least one risk parameter specification, to which you can assign one scenario segmentation. A scenario segmentation consists of scenario segments.

The CEBS stress scenarios from 2010 are an example of this. These scenarios were segmented by country and country group, and the stressed risk parameter values for probability of default (PD) and loss given default (LGD) were provided explicitly.

Scenario Segment

You edit scenario segments on the user interface in SAP NetWeaver Business Client or SAP NetWeaver Portal under *Credit Risk Stress Tests* > *Definition* > *Manage Scenario Segments* or in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Risk Stress Tests* > *Configuration* > *Edit Scenario Segments*.

Scenario segments define a subset of transactions on the basis of selection criteria that are based on characteristics. The characteristic values of a transaction determine the scenario segment or segments in which this transaction falls. Scenario segments are not necessarily without overlaps.

In the Customizing activity *Define Characteristics for Scenario Segments*, you specify the characteristics for which selection options can be entered on the *Manage Scenario Segments* user interface in SAP NetWeaver Portal or SAP NetWeaver Business Client.

You can define scenario segments based on characteristics such as *Country*, *Industry*, *Product Category* or *Customer Class*:

- Germany
- Euro area countries
- Automobile industry in the EU
- Real estate financing
- Consumer loans in the good book

Scenario Segmentation

You edit scenario segmentations on the user interface in SAP NetWeaver Business Client or SAP NetWeaver Portal under *Credit Risk Stress Tests* > *Definition* > *Manage Scenario Segmentations* or in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Risk Stress Tests* > *Configuration* > *Edit Scenario Segmentations*.

You assign a scenario segment to one or more scenario segmentations in order to structure the scenario portfolio in more detail: For example, you can create a scenario segment as a combination of different characteristics (such as food industry in France) rather than selecting individual characteristics.

Since a transaction can meet the selection criteria of several scenario segments you have to define the priority of a scenario segment within a segmentation. On the user interface in SAP NetWeaver Business Client or SAP NetWeaver Portal, the sequence in which you assign scenario segments to a scenario segmentation determines their priority. The first (or highest) scenario segment has the highest priority. In Customizing, the number of the scenario segment determines its priority. In this case, the lower the number, the higher the priority.

The system assigns the transactions in the scenario portfolio to one scenario segment in order of priority: The transaction is assigned to the first scenario segment whose selection criteria it meets. This means that the scenario portfolio is divided into non-overlapping scenario segments.

For example, you can define a scenario segmentation as follows:

- Scenario segment 1: automobile industry in Germany
- Scenario segment 2: European Union
- Scenario segment 3: U.S.
- Scenario segment 4: complementary segment (residual)

The transactions that are included in the scenario segment with the highest priority (*Automobile Industry in Germany*) could also be included in the second-ranked scenario segment (*European Union*) because they have the characteristic value *Germany*. However, the system assigns these transactions only to the first scenario segment because this has the higher priority in the segmentation.

The complementary segment contains all the transactions in the scenario portfolio that could not be assigned by the system to any of the other scenario segments. This ensures that all the transactions are assigned to scenario segments. The use of a complementary segment is optional.

Risk Parameter Specification

You edit risk parameter specifications on the user interface in SAP NetWeaver Business Client or SAP NetWeaver Portal under ▶ *Credit Risk Stress Tests* ▶ *Definition* ▶ *Manage Risk Parameter Specifications* ◀ or in Customizing for *Bank Analyzer* under ▶ *Processes and Methods* ▶ *Credit Risk* ▶ *Credit Risk Stress Tests* ▶ *Configuration* ▶ *Edit Risk Parameter Specifications* ◀.

The **risk parameter shift type** determines which risk parameter is stressed, such as the probability of default (PD) or the exposure at default (EAD). For each risk parameter shift type, you can assign only one risk parameter specification to a credit risk scenario.

A risk parameter specification corresponds to one of the following **categories**:

- A **risk parameter shift** deals with the change in the value of a risk parameter in a credit risk scenario. This change is entered as a **relative** or **absolute** shift. The system increases or reduces the original value of the risk parameter being stressed through multiplication (relative risk parameter shift) or addition (absolute risk parameter shift):
  - Example 1:
    PD = 0.10 * 1.5 = 0.15 = 15%
    (relative shift of the original PD of 10% by 1.5)
  - Example 2:
    PD = 0.10 + 0.05 = 15%
    (absolute shift of the original PD of 10% by 5%)
- A **risk parameter value** is the value of a stressed risk parameter that is explicitly provided in a credit risk scenario.
  Example: LGD = 40%

The combination of specific risk parameter shift types in a credit risk scenario is problematic if these affect the same key figure. Therefore, the following combinations are *not* permitted:

- Currency and simulation of new business
- Currency and EAD
- Currency and collateral value
- Currency and Credit Conversion Factor (CCF)
- Simulation of new business and EAD
- Simulation of new business and collateral value
- Simulation of new business and CCF

- EAD and CCF
- Master rating and PD
- Master rating and risk weight

> **Note**
>
> The system cannot check the compatibility of customer-specific risk parameter shift types.

> **Note**
>
> Currency shifts and simulations of new business can be executed only in the ad hoc calculation and the approximate stress run, but not in the credit exposure stress run.

The credit risk scenarios are calculated based on the changes to the risk parameter. You can map these changes in different ways:

- Entry of values (generic market data)
  Risk parameter specifications entered as values in SAP NetWeaver Business Client or SAP NetWeaver Portal are mapped in the system based on generic market data. When you select the data structure of the risk parameter specification, you define the generic market data class.
  Before you can select a data structure in SAP NetWeaver Business Client or SAP NetWeaver Portal, the corresponding market data class must have been created in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Generic Market Data* > *Edit Generic Market Data Classes*.
  If multiple market data areas have been created in Customizing you need to select the required market data area in SAP NetWeaver Business Client or SAP NetWeaver Portal.
  The generic market data is used to store the risk parameters in the specifications. You can also use this data to define additional dimensions for differentiating between individual scenario segments.
- Derivation of values (module editor)
  Risk parameter specifications derived as values in SAP NetWeaver Business Client or SAP NetWeaver Portal are mapped in the system based on the module editor. You can use the module editor to define very complex formulaic risk parameter specifications.
  When you create a new risk parameter specification based on the derivation of values, you must enter a module and a module environment. You must first create both parameters in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Risk Stress Tests* > *Configuration* > *Module Editor*.
  They can then be selected on the user interface in SAP NetWeaver Business Client or SAP NetWeaver Portal.

You can assign one scenario segmentation to a risk parameter specification. You can assign the same scenario segmentation to different risk parameter specifications.

A scenario segmentation does not have to cover all the transactions in the scenario portfolio. If you are not using a scenario segmentation or if there are no suitable transactions in any of the scenario segments, you can still stress the affected transactions by configuring the data structure (or the generic market data class) of the risk parameter specification accordingly.

Integration

The *Audit Support* checkbox controls the traceability of credit risk stress tests. The system only supports the auditing of a credit risk scenario if this checkbox has been set for all the corresponding scenario segments,

scenario segmentations, and risk parameter specifications. The checkbox is set automatically for the scenario entities defined in Customizing.

> **Note**
>
> If the *Audit Support* checkbox has been set you can still change or delete the scenario entity in Customizing, but not on the user interface in SAP NetWeaver Business Client and SAP NetWeaver Portal.

You can include a credit risk scenario in a credit risk scenario set. For more information, see Credit Risk Scenario Set [page 1273].

You can include a credit risk scenario or a credit risk scenario set in Customizing in a composite scenario. For more information, see the *Integration* section under Stress Test Scenarios [page 428].

> **Note**
>
> We recommend that you do not include a credit risk scenario and a credit risk scenario set in the same composite scenario.

When you create a credit risk scenario, a credit risk scenario set, and a risk parameter specification, the system creates an entry with the same name in the table for simple scenarios used for general scenario management (see Scenario [page 1731]). You can change or display the generic market data of a risk parameter specification on the *SAP Easy Access* screen under *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Generic Market Data* > *Edit Scenario Version* or *Display Scenario Version*. Enter the ID of the risk parameter specification in the *Scenario* field.

Example

Example: Credit Risk Scenario [page 1271]

More Information

NetWeaver Portal Content documentation:

- Manage Credit Risk Scenarios [page 1286]
- Manage Risk Parameter Specifications [page 1283]
- Manage Scenario Segmentations [page 1285]
- Manage Scenario Segments [page 1284]

4.7.1.9.3.1.1 Example: Credit Risk Scenario

The following example illustrates a hypothetical scenario for a euro debt crisis.

In the IRB approach, all the risk parameter specifications are applied in the specified countries to the country portfolio only; in the standardized approach for credit risk, they are also applied to multilateral development banks

and other public sector entities. These specifications are characterized by their own asset classes in each approach.

- **PD stress: scenario for the segmentation of the PIIGS countries**

Country	Risk Parameter Specification
Greece	PD = 15% (stressed PD value)
Portugal	Downgrading of current rating by 3 notches
Ireland	Downgrading of current rating by 2 notches
Spain	Downgrading of current rating by 2 notches
Italy	Relative increase of current PD value by 10%

The unstressed ratings apply to all other countries and asset classes.

- **Loss given default (LGD)**
An LGD scenario is defined in the same segmentation as for the PD stress scenario. This is restricted to the same countries and asset classes:

Country	Risk Parameter Specification
Greece	Absolute shift of the current LGD by 10%
Portugal	Relative shift of the current LGD by 10%
Ireland	Default stressed LGD value of 43.5%
Spain	Default stressed LGD value of 41.2%
Italy	Absolute shift of the current LGD by 5%

The unstressed LGDs apply to all other countries and asset classes.

- **Financial collateral/reductions in the value of government bonds**
The stress refers to a reduction in the value of government bonds, which are used as collateral for credit exposures, by the specified percentage:

Country	Risk Parameter Specification
Greece	50 %
Portugal	15 %
Ireland	15 %
Spain	10 %

Country	Risk Parameter Specification
Italy	3 %

The securities are not stressed for all other countries and asset classes.

4.7.1.9.3.2 Credit Risk Scenario Set

Definition

Group of credit risk scenarios [page 1267].

You edit credit risk scenario sets on the user interface in SAP NetWeaver Business Client or SAP NetWeaver Portal under *Credit Risk Stress Tests* > *Definition* > *Manage Credit Risk Scenario Sets* or in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Risk Stress Tests* > *Configuration* > *Edit Credit Risk Scenario Sets*.

Use

You have the **option** of grouping multiple credit risk scenarios in a credit risk scenario set and triggering the calculation – usually in a background job – of the stressed results of these scenarios. Since the system calculates several credit risk scenarios in one run, data is selected and enriched only once. This improves the system's performance because scenarios are calculated in one run as opposed to several individual runs.

Structure

You can assign one or more credit risk scenarios to a credit risk scenario set. However, you cannot assign scenarios of other data categories, such as market data or business partner scenarios.

Integration

You must have multiple credit risk scenarios to be able to create a credit risk scenario set.

When you create a credit risk scenario set, the system creates an entry with the same name in the table for simple scenarios used for general scenario management (see Scenario [page 1731]).

You can include credit risk scenario sets directly in the approximate stress run [page 1274] or indirectly in the credit exposure stress run [page 1261] and credit exposure stress run with online selection [page 1263] in a composite scenario. The ad hoc calculation cannot process any credit risk scenario sets.

To include a credit risk scenario set in a composite scenario, choose Customizing for *Bank Analyzer* ▶ *Processes and Methods* ▶ *Credit Risk* ▶ *Credit Risk Stress Tests* ▶ *Configuration* ▶ *Edit Composite Scenarios* ▶. For more information about composite scenarios, see the *Integration* section in Stress Test Scenarios [page 428].

> **Note**
>
> We recommend that you do not include a credit risk scenario and a credit risk scenario set in the same composite scenario.

More Information

Manage Credit Risk Scenario Sets [page 1287] (NetWeaver Portal Content documentation)

4.7.1.9.3.3 Approximate Stress Run

Use

You use the approximate stress run to apply credit risk scenarios to portfolios that have already been calculated using the credit exposure run.

The approximate stress run is based on the results of a credit exposure calculation run using real, unstressed data. The system reads these results from the Results Data Layer (RDL). More specifically, the approximate run uses the method of collateral distribution of the update run, which improves system performance.

The use of this method of collateral distribution means that the results of the approximate stress run are approximately correct because the use of a credit risk scenario could change the optimal collateral distribution. This means that the capital requirement at portfolio level, which is a result of the approximate stress run, is always the same or higher than if optimal collateral distribution is used.

The approximate stress run is particularly suited to certain portfolios, in particular retail portfolios and portfolios containing the receivables from small- and medium-sized entities (SMBs).

> **Note**
>
> The approximate stress run allows you to calculate **credit risk scenarios** only and not scenarios of other data categories. If you enter a composite scenario in the approximate stress run, the system processes only the credit risk scenario contained in the composite scenario.

You can use credit risk scenario sets [page 1273] to calculate several risk scenarios in one run.

You can use the approximate stress run in parallel jobs.

Prerequisites

- You have defined a credit risk scenario and the corresponding entities:
  - Scenario segments
  - Scenario segmentations
  - Risk parameter specifications

  For more information, see Credit Risk Scenario [page 1267].

  > **Note**
  >
  > You do not need to create composite scenarios for approximate stress runs. The use of credit risk scenario sets is optional.

- You have made the required settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Risk Stress Tests* > *Run Settings* > *Approximate Stress Run*.
- The results of a credit exposure calculation run are available. These results are read from the Results Data Layer (RDL) and are the basis for the real data for the approximate stress run.
- The prerequisites specified in Stress Tests for Credit Risk Scenarios [page 1264] also apply.

Process

To execute an approximate stress run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Support for Stress Tests* > *Credit Risk Stress Tests* > *Approximate Stress Run*.

You can also schedule the approximate stress run as a background job.

In Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*, you can adjust the step sequence of this CVPM process, for example, by adding a module editor step or extracting the results to BI instead of to the RDL.

When it executes the process, the system performs the following steps:

1. Parameter enrichment
   The system copies the Customizing settings that it needs to create a worklist, for example, the Customizing settings for the run type.
2. Worklist creation
   The system reads the transactions in the scenario portfolio from the Results Data Layer (RDL).
3. Data enrichment using scenario data
   The system reads the configuration settings for the credit risk scenario and stresses the scenario portfolio data accordingly.
4. Save results in the RDL
   The results of a test run are not saved.

You can analyze the approximate stress runs in the CVPM process monitor [page 1635], for example, by displaying the detailed logs.

4.7.1.9.3.4 Ad Hoc Calculation

Use

You use the ad hoc calculation to test a representative section of a scenario portfolio and to validate the credit risk scenarios used. Since the ad hoc calculation does not work as a parallel job it is suitable for small, clearly defined portfolios only.

The ad hoc calculation uses calculation steps from the approximate stress run [page 1274], including customer-specific enhancement steps. Unlike the approximate stress run, the ad hoc calculation cannot be scheduled as a background job.

> **Note**
>
> The ad hoc calculation allows you to calculate **credit risk scenarios** only and not scenarios of other data categories.

Prerequisites

- You have defined a credit risk scenario and the corresponding entities:
  - Scenario segments
  - Scenario segmentations
  - Risk parameter specifications

  For more information, see Credit Risk Scenario [page 1267].

  > **Note**
  >
  > You do not need to create composite scenarios and credit risk scenario sets for ad hoc calculations.

- You have made the required settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Risk Stress Tests* > *Run Settings* > *Ad Hoc Calculation*.
- The results of a **CVPM aggregation process** or a **credit exposure calculation run** are available. These results are read from the Results Data Layer (RDL) and are the basis for the real data for the ad hoc calculation. For more information about the aggregation process, see Aggregation Run for Ad Hoc Calculation [page 1278].

The prerequisites specified in Stress Tests for Credit Risk Scenarios [page 1264] also apply.

Procedure

You execute ad hoc calculations on the user interface in SAP NetWeaver Business Client or SAP NetWeaver Portal under *Credit Risk Stress Tests* > *Ad Hoc Calculation* > *Execute Ad Hoc Calculations*.

When you execute this process, you can re-use the entries saved for previous ad hoc calculations.

Perform the following steps:

1. **Select the preliminary process**
   - Select an ad hoc run type.
   - Use the input help (F4) for the key date to choose a credit exposure run or a CVPM run.
2. **Restrict the scenario portfolio**
   Specify the transaction worklist for which you want to execute an ad hoc calculation. You can use filter criteria to do this.

   > **Note**
   >
   > You define which characteristics are suitable filter criteria for the ad hoc calculation in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Risk Stress Tests* > *Run Settings* > *Ad Hoc Calculation* > *Edit Filter Settings*:
   >
   > - Characteristics are suitable if the same characteristics have been entered for all portions of the exposure.
   > - Characteristics are not suitable if different characteristics have been entered for some portions or if no characteristics have been entered.
   >
   > The correct functioning of the ad hoc calculation can only be ensured if suitable filter characteristics are used. You enter the ID for the filter settings in the ad hoc run type.
3. **Load the scenario portfolio**
   Choose *Load Scenario Portfolio*.
   A new screen appears on which the aggregated, unstressed result is displayed.
4. **Define the credit risk scenario**
   Create a new credit risk scenario or select an existing credit risk scenario in the *Scenario ID* field for which you want to execute an ad hoc calculation.
5. **Calculate the credit risk scenario**
   Choose *Execute Calculation*. The system calculates the stressed results for the selected credit risk scenario.

You can repeat the calculation with other credit risk scenarios and compare the results.

Result

The system displays the results as aggregated values at the level of the general scenario portfolio. You can also display the results at single transaction level (detailed view). A diagram is also available.

You can compare the results data of the ad hoc calculation with real data and with the results data of other credit risk scenarios. If you have calculated stressed results for several credit risk scenarios, you can use one of these results as the reference for a delta calculation.

The results of an ad hoc calculation are not saved but can be exported to Microsoft Excel.

More Information

NetWeaver Portal Content documentation:

- Ad Hoc Calculation [page 1289]
- Execute Ad Hoc Calculations [page 1289]

4.7.1.9.3.4.1 Aggregation Run for Ad Hoc Calculation

Use

You can use this function to accelerate the ad hoc calculation for large, homogenous portfolios. This usually results in a loss in accuracy. You can also use this aggregation run as a preliminary run for the approximate stress run.

The aggregation run is especially suitable for revolving retail that consists of unsecured checking accounts and credit card exposures.

The aggregates are created from single transactions that have the same predefined risk characteristics, for example, the same credit score or assignment to the same rating level. The system handles the summarized data in subsequent processes in the same way as an *entity*. Data is aggregated using the aggregation tool. You define the rules for generating aggregates in Customizing.

Prerequisites

You have made the required settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Risk Stress Tests* > *Run Settings* > *Aggregation*.

These Customizing settings are fixed and cannot vary on the user interface in SAP NetWeaver Portal and SAP NetWeaver Business Client.

Activities

To execute an aggregation run, on the *SAP Easy Access* screen choose *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Support for Stress Tests* > *Credit Risk Stress Tests* > *Aggregation Run for Ad Hoc Calculation*.

4.7.1.9.3.5 Credit Risk Analyst in SAP NetWeaver Business Client

`SAP_BA1_NWBC_CRST, SAP_BA1_SHOW_CRST, SAP_BA1_USER_CRST, SAP_BA1_EXPERT_CRST`

Use

Credit risk analysts work in the area of credit risk management in banks. It is their job to use stress tests to check and analyze a bank's capital adequacy.

The following typical activities are associated with this role:

- Defining credit risk scenarios
- Performing ad hoc calculations for credit risk stress tests with different credit risk scenarios
- Comparing, analyzing, and exporting stressed results

For example, credit risk analysts can use ad hoc calculations to check the plausibility of a scenario definition before they run a credit exposure stress run or an approximate stress run in the Bank Analyzer system. For more information, see Stress Tests for Credit Risk Scenarios [page 1264].

Integration

This role belongs to the *SAP Interactive Credit Risk Analysis* application in the FS-BA-AN-CRA *Credit Risk Analytics* application component.

The role `SAP_BA1_NWBC_CRST` corresponds to the role of Credit Risk Analyst in SAP NetWeaver Portal [page 1281].

Configuration

Define the authorizations for the role `SAP_BA1_NWBC_CRST` by assigning an additional role based on one of the following templates to the user in the Bank Analyzer system:

- Credit risk analyst (display role) – `SAP_BA1_SHOW_CRST`
  In this role you can display existing credit risk scenarios and all their entities in Customizing and on the user interface in SAP NetWeaver Business Client.
- Credit risk analyst (user role) – `SAP_BA1_USER_CRST`
  In this role you can manage credit risk scenarios and all their entities on the user interface in SAP NetWeaver Business Client and perform ad hoc calculations. You can also display the Customizing settings in the Bank Analyzer system.
- Credit risk analyst (expert role) – `SAP_BA1_EXPERT_CRST`
  In this role you can manage credit risk scenarios and all their entities on the user interface in SAP NetWeaver Business Client and in Customizing, and perform ad hoc calculations. Users with this role are also authorized to run approximate stress runs in the Bank Analyzer system. This role is typically assigned to IT experts.

> **Note**
>
> The authorization for credit exposure runs and credit exposure stress runs is linked to the role `SAP_BA1_EXPERT_CE` and, if required, must be assigned additionally.

Constraints

This role does not contain an integrated function for reporting. However, you can add a link to a customer-specific reporting solution.

4.7.1.9.3.6 Business Package for Financial Services Analytical Banking 8.0

Use

The *Business Package for Financial Services Analytical Banking 8.0* provides credit risk analysts with a portal-based interface that supports their daily tasks.

Technical Data

Availability	NetWeaver 7.00 SP23
Data Source	FSPP4.0
Browser Recommendation	For more information about the system prerequisites for these components, see the Product Availability Matrix on SAP Service Marketplace at http://service.sap.com/pam.
Languages Available	• CS Czech • DE German • EN English • ES Spanish • FR French • HE Hebrew • HU Hungarian • IT Italian • JA Japanese • KO Korean • PT Portuguese • RO Romanian • RU Russian • SK Slovakian • UK Ukrainian • ZH Chinese

Support	FS-BA-PM-CR

4.7.1.9.3.6.1 Credit Risk Analyst in SAP NetWeaver Portal

`com.sap.pct.fsp.ba.creditriskofficer, SAP_BA1_SHOW_CRST, SAP_BA1_USER_CRST, SAP_BA1_EXPERT_CRST`

Use

Credit risk analysts work in the area of credit risk management in banks. It is their job to use stress tests to check and analyze a bank's capital adequacy.

The following typical activities are associated with this role:

- Defining credit risk scenarios
- Performing ad hoc calculations for credit risk stress tests with different credit risk scenarios
- Comparing, analyzing, and exporting stressed results

For example, credit risk analysts can use ad hoc calculations to check the plausibility of a scenario definition before they run a credit exposure stress run or an approximate stress run in the Bank Analyzer system. For more information, see Stress Tests for Credit Risk Scenarios [page 1264].

Integration

This role belongs to the *SAP Interactive Credit Risk Analysis* application in the FS-BA-AN-CRA *Credit Risk Analytics* application component.

The role `com.sap.pct.fsp.ba.creditriskofficer` corresponds to the role of Credit Risk Analyst in SAP NetWeaver Business Client [page 1278].

Configuration

Define the authorizations for the role `com.sap.pct.fsp.ba.creditriskofficer` by assigning an additional role based on one of the following templates to the user in the Bank Analyzer system:

- Credit risk analyst (display role) – `SAP_BA1_SHOW_CRST`
  In this role you can display existing credit risk scenarios and all their entities in Customizing and on the user interface in SAP NetWeaver Portal.
- Credit risk analyst (user role) – `SAP_BA1_USER_CRST`
  In this role you can manage credit risk scenarios and all their entities on the user interface in SAP NetWeaver Portal and perform ad hoc calculations. You can also display the Customizing settings in the Bank Analyzer system.

- Credit risk analyst (expert role) – `SAP_BA1_EXPERT_CRST`
  In this role you can manage credit risk scenarios and all their entities on the user interface in SAP NetWeaver Portal and in Customizing, and perform ad hoc calculations. Users with this role are also authorized to run approximate stress runs in the Bank Analyzer system. This role is typically assigned to IT experts.

  > **Note**
  >
  > The authorization for credit exposure runs and credit exposure stress runs is linked to the role `SAP_BA1_EXPERT_CE` and, if required, must be assigned additionally.

Constraints

This role does not contain an integrated function for reporting. However, you can add a link to a customer-specific reporting solution.

4.7.1.9.3.6.1.1 Definition

Technical Data

Technical Name	com.sap.pct/specialist/ com.sap.pct.fsp.ba.bp_folder/ com.sap.pct.fsp.ba.creditriskstresstest/ com.sap.pct.fsp.ba.worksets/ com.sap.pct.fsp.ba.definition

Use

You use this workset to define the parameters of a credit risk stress test.

Integration

You can subject the credit risk scenarios that you have defined here to a stress test in the Ad Hoc Calculation [page 1289] workset or in the Bank Analyzer system.

For more information, see Stress Tests for Credit Risk Scenarios [page 1264].

Features

You use this workset to define credit risk scenarios, which comprise the following parameters:

- Scenario segments
- Scenario segmentations
- Risk parameter specifications

You can also define credit risk scenario sets. However, these can be calculated in the Bank Analyzer system only.

For more information, see Manage Credit Risk Scenario Sets [page 1287].

4.7.1.9.3.6.1.1.1 Manage Risk Parameter Specifications

Technical Data

Technical Name of the iView	`com.sap.pct/specialist/`
	`com.sap.pct.fsp.ba.bp_folder/`
	`com.sap.pct.fsp.ba.creditriskstresstest/`
	`com.sap.pct.fsp.ba.iviews/`
	`com.sap.pct.fsp.ba.adminrpsp`

Technical Name of the Application	`wda_fi_qaf_rpsp`
Runtime Technology	ABAP/Web Dynpro
Data Source or Data Store	FSPP4.0

Use

You use this iView to create, check, and change risk parameter specifications, and to configure audit support.

Integration

In SAP NetWeaver Portal you can also edit risk parameter specifications that were created in the Bank Analyzer system.

You can include risk parameter specifications in credit risk scenarios using the iView Manage Credit Risk Scenarios [page 1286].

Prerequisites

You have defined the authorizations for the role of credit risk analyst by assigning an additional role to the user in the Bank Analyzer system.

For more information, see Credit Risk Analyst in SAP NetWeaver Portal [page 1281].

Configuration

We provide the configuration `/BA1/WDA_FI_QAF_RPSP`.

More Information

For more information, see Credit Risk Scenario [page 1267].

4.7.1.9.3.6.1.1.2 Manage Scenario Segments

Technical Data

Technical Name of the iView	`com.sap.pct/specialist/`
	`com.sap.pct.fsp.ba.bp_folder/`
	`com.sap.pct.fsp.ba.creditriskstresstest/`
	`com.sap.pct.fsp.ba.iviews/`
	`com.sap.pct.fsp.ba.adminsegments`

Technical Name of the Application	`wda_fi_qaf_part`
Runtime Technology	ABAP/Web Dynpro
Data Source or Data Store	FSPP4.0

Use

You use this iView to create, check, and change scenario segments, and to configure audit support.

Integration

In SAP NetWeaver Portal you can also edit scenario segments that were created in the Bank Analyzer system.

You can include scenario segments in scenario segmentations using the iView Manage Scenario Segmentations [page 1285].

Prerequisites

You have defined the authorizations for the role of credit risk analyst by assigning an additional role to the user in the Bank Analyzer system.

For more information, see Credit Risk Analyst in SAP NetWeaver Portal [page 1281].

Configuration

We provide the configuration `/BA1/WDA_FI_QAF_PART`.

More Information

For more information, see Credit Risk Scenario [page 1267].

4.7.1.9.3.6.1.1.3 Manage Scenario Segmentations

Technical Data

Technical Name of the iView	`com.sap.pct/specialist/`
	`com.sap.pct.fsp.ba.bp_folder/`
	`com.sap.pct.fsp.ba.creditriskstresstest/`
	`com.sap.pct.fsp.ba.iviews/`
	`com.sap.pct.fsp.ba.adminsegmentation`

Technical Name of the Application	`wda_fi_qaf_area`
Runtime Technology	ABAP/Web Dynpro
Data Source or Data Store	FSPP4.0

Use

You use this iView to create, check, and change scenario segmentations, and to configure audit support.

Integration

In SAP NetWeaver Portal you can also edit scenario segmentations that were created in the Bank Analyzer system.

You can include scenario segmentations in risk parameter specifications using the iView Manage Risk Parameter Specifications [page 1283].

Prerequisites

You have defined the authorizations for the role of credit risk analyst by assigning an additional role to the user in the Bank Analyzer system.

For more information, see Credit Risk Analyst in SAP NetWeaver Portal [page 1281].

Configuration

We provide the configuration `/BA1/WDA_FI_QAF_AREA`.

More Information

For more information, see Credit Risk Scenario [page 1267].

4.7.1.9.3.6.1.1.4 Manage Credit Risk Scenarios

Technical Data

Technical Name of the iView	`com.sap.pct/specialist/`
	`com.sap.pct.fsp.ba.bp_folder/`
	`com.sap.pct.fsp.ba.creditriskstresstest/`
	`com.sap.pct.fsp.ba.iviews/`
	`com.sap.pct.fsp.ba.adminscenarios`

Technical Name of the Application	`wda_fi_qaf_scen`
Runtime Technology	ABAP/Web Dynpro
Data Source or Data Store	FSPP4.0

Use

You use this iView to create, check, and change credit risk scenarios, and to configure audit support.

Integration

In SAP NetWeaver Portal you can also edit credit risk scenarios that were created in the Bank Analyzer system.

You can include credit risk scenarios in credit risk scenario sets using the iView Manage Credit Risk Scenario Sets [page 1287].

Prerequisites

You have defined the authorizations for the role of credit risk analyst by assigning an additional role to the user in the Bank Analyzer system.

For more information, see Credit Risk Analyst in SAP NetWeaver Portal [page 1281].

Configuration

We provide the configuration `/BA1/WDA_FI_QAF_SCEN`.

More Information

For more information, see Credit Risk Scenario [page 1267].

4.7.1.9.3.6.1.1.5 Manage Credit Risk Scenario Sets

Technical Data

Technical Name of the iView	com.sap.pct/specialist/ com.sap.pct.fsp.ba.bp_folder/ com.sap.pct.fsp.ba.creditriskstresstest/ com.sap.pct.fsp.ba.iviews/ com.sap.pct.fsp.ba.adminscenariosets
Technical Name of the Application	wda_fi_qaf_sset

Runtime Technology	ABAP/Web Dynpro
Data Source or Data Store	FSPP4.0

Use

You use this iView to create, check, and change credit risk scenario sets. You assign at least one credit risk scenario to a credit risk scenario set.

Integration

In SAP NetWeaver Portal you can also edit credit risk scenario sets that were created in the Bank Analyzer system.

Credit risk scenario sets can be included in the following processes:

- Approximate stress run [page 1274]
- Credit exposure stress run [page 1261]
- Credit exposure stress run with online selection [page 1263]

You cannot determine stress test results for a credit risk scenario set using the iView Execute Ad Hoc Calculations [page 1289].

Prerequisites

You have defined the authorizations for the role of credit risk analyst by assigning an additional role to the user in the Bank Analyzer system.

For more information, see Credit Risk Analyst in SAP NetWeaver Portal [page 1281].

Configuration

We provide the configuration `/BA1/WDA_FI_QAF_SSET`.

More Information

For more information, see Credit Risk Scenario Set [page 1273].

4.7.1.9.3.6.1.2 Ad Hoc Calculation

Technical Data

Technical Name	com.sap.pct/specialist/ com.sap.pct.fsp.ba.bp_folder/ com.sap.pct.fsp.ba.creditriskstresstest/ com.sap.pct.fsp.ba.worksets/ com.sap.pct.fsp.ba.calculation

Use

You use this workset to perform ad hoc calculations for credit risk stress tests using different credit risk scenarios and to compare the results. For example, you can check the plausibility of a scenario definition.

Features

- You can use the entries saved for previous ad hoc calculations.
- You can restrict the scenario portfolio using filters.
- You can compare the results data of the ad hoc calculation with real data and with the results data of other credit risk scenarios. The system displays the results as aggregated values. A detailed view and a diagram are also available.

More Information

For more information, see Ad Hoc Calculation [page 1276] and Aggregation Run for Ad Hoc Calculation [page 1278].

4.7.1.9.3.6.1.2.1 Execute Ad Hoc Calculations

Technical Data

Technical Name of the iView	com.sap.pct/specialist/ com.sap.pct.fsp.ba.bp_folder/ com.sap.pct.fsp.ba.creditriskstresstest/ com.sap.pct.fsp.ba.iviews/ com.sap.pct.fsp.ba.adhocstresstests

Technical Name of the Application	wda_fi_qaf_adhoc
Runtime Technology	ABAP/Web Dynpro
Data Source or Data Store	FSPP4.0

Use

You use this iView to perform ad hoc calculations for credit risk stress tests using different credit risk scenarios and to compare the results. For example, you can check the plausibility of a scenario definition.

Prerequisites

- You have defined the authorizations for the role of credit risk analyst by assigning an additional role to the user in the Bank Analyzer system (see Credit Risk Analyst in SAP NetWeaver Portal [page 1281]).
- For more information, see Ad Hoc Calculation [page 1276].

Configuration

We provide the configuration /BA1/WDA_FI_QAF_ADHOC.

4.7.1.10 Archiving of Credit Exposure Data

Use

You use this function to remove obsolete Credit Exposure data that is stored in the Result Database (RDB) from the system. Over time, the volume of data in your productive system increases. So that the evaluations that you run in the system are not slowed down by too large a volume of data, we recommend that you use the archiving tool.

> **Note**
>
> For more information about the results data stored in the Results Data Layer, see Archiving and Destroying Results Data Using the Archiving Engine [page 1395].

Prerequisites

In run administration, you need to have already flagged the runs that you want to archive.

Features

The archiving function removes the data stored in the Result Database (RDB data) for the runs flagged for archiving. This process comprises the steps Write Archive (archive) and Delete RDB Data from the Database.

Since three RDB packages are created for each Credit Exposure run, in the current release it is not possible to use the (standard) deletion job in the technical Customizing of the archiving function. Use Schedule Manager (CA) to schedule the deletion job.

Once a run has been archived, it then has the status *Run Completely Archived* in run administration. You cannot delete runs in run administration that have the status *Run Completely Archived*.

The archiving functions include an information system that you can use to display the archived data. Displaying this archived data can seriously impair system performance, however. If required, you can use the reload program to reload the data back into the system.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Method* > *Credit Risk* > *Credit Exposure* > *Archiving* > *Archive Credit Exposure Data*.
   The *Archive Administration: Initial Screen* appears. The relevant archiving object is `BA1_R4_063`, and this value is entered in the field by default.
2. You can use the following functions:

User Actions for Data Archiving

Action	Function
Write	Archives data that is contained in the database. For more information, see .
Delete	Deletes archived data from the database. For more information, see .
Management	Displays and edits administrative information about archiving runs. For more information, see .
Job Overview	Displays the archiving jobs and their statuses. For more information, see .

Action	Function
Goto > *Reload*	Reloads archived data into the database. For more information, see .

> **Note**
>
> For more information about archiving, see the documentation about .

4.7.2 Credit Portfolio Data Processing

Use

You can use this component to connect a credit portfolio model to Bank Analyzer. If you use SAP Basel II, Credit Portfolio Data Processing (CPDP) contains an integrated run that you can use to compare the more precise internal risk key figures calculated by CPDP with the external credit risk calculations required by Basel II. A PI infrastructure is available, which allows you to send the credit risk figures to a credit portfolio model (such as RiskFrontier from Moody's KMV) for further calculation.

> **Note**
>
> For more information about the availability of *Credit Portfolio Data Processing*, see SAP Note 2316265.

Implementation Considerations

You make the required settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing*.

You also make the required Customizing settings for Credit Exposure under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure*.

Integration

With Other SAP Components

Function Required	Component Necessary
Data Provision	Source Data Layer (FS-BA-SD) [page 100] for master data, flow data, and generic market data Foundation for (FS-FND-MKD)
Data Storage	Results Data Layer (FS-BA-RD) [page 1364]
Credit exposure calculation process	Credit Exposure (FS-BA-PM-CR-CE) [page 1192] In particular Credit Exposure Calculation [page 1196]
CPDP Runs	Calculation and Valuation Process Manager (FS-BA-IF-CVP) [page 1622]

Features

Credit Portfolio Data Processing prepares master data, flow data, generic market data and market data for calculation in a credit portfolio model. CPDP obtains master data, flow data and generic market data from the Source Data Layer (SDL) of Bank Analyzer, and market data from the Foundation.

The following steps comprise Bank Analyzer's role in a credit portfolio calculation:

1. Aggregation
2. Extraction
3. Field mapping
4. Disaggregation

The system calculates the risk figures in a credit exposure run, and stores this data in Bank Analyzer's Results Data Layer (RDL). Market data and generic market data are an exception to this, since the system exports them to the connected credit portfolio model without any preliminary calculation in a credit exposure run. After an aggregation run [page 1295] and an extraction run for initialization [page 1297], the following extraction runs are available:

Data Extraction Run	Data Source
Extraction Run for Business Partners [page 1298]	Master data
Extraction Run for Collateral Providers [page 1299]	Master data
Extraction Run for Exchange Rates [page 1301]	Market data
Extraction Run for Yield Curves [page 1302]	Market data
Extraction Run for Spread Matrices [page 1304]	Generic market data
Extraction Run for PD Matrices [page 1305]	Generic market data
Transition Matrix Extraction Run [page 1307]	Generic market data

Data Extraction Run	Data Source
Extraction Run for Exposures [page 1308]	Master data
Extraction Run for Aggregated Exposures [page 1310]	Master data

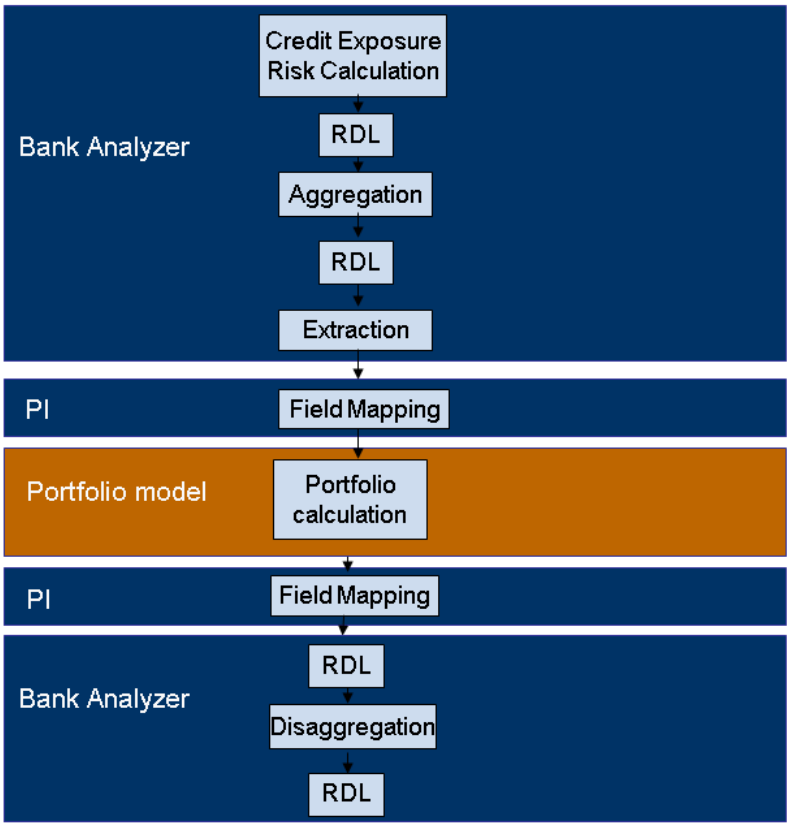
After extraction, the system uses field mapping and the PI infrastructure to send the data to a credit portfolio model for further processing. Once the connected system has concluded the credit portfolio calculation and returned the data, you need to start a disaggregation run [page 1311]. The system disaggregates the transactions that were previously aggregated, and adds the transactions that had defaulted before extraction. It then stores this complete data record in the Results Data Layer (RDL) of Bank Analyzer. The data is then available for further processing.

Note

For test purposes, you can use ad hoc processes.

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Ad Hoc Calculation*:

- *Ad hoc Extraction*
- *Ad hoc Postprocessing*



Overview of a Credit Portfolio Calculation

4.7.2.1 Aggregation Runs

Use

You can use this function to have the system execute the aggregation run.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. You have made the necessary settings for Credit Portfolio Data Processing in Customizing under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Aggregation*.
2. You have created at least one step sequence in Customizing for the analytical process `/BA1/R2_CPM_AGGR_CVPM`. You can find the required IMG activity in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*. You have created at least the following steps for a step sequence:
   1. Parameter Enrichment
   2. Worklist Creation
   3. Data Enrichment Using Module Editor
   4. Aggregation
   5. Save
3. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Aggregation Run*
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Run Types for CPDP
   - Scenario
   - Process Partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Step Sequence
     If you have created multiple step sequences, you can select the one you want to use from among them.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. This field is blank by default. If you leave the field blank, the system uses the current time stamp when the process starts.
   - Technical Settings
     You can also call filter variants, create new run groups, define a sequential execution or execute the CVPM run as a test run.
3. Choose  .

> **Note**
>
> To display the run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.7.2.2 Extraction Run for Initialization

Use

You can use this function to have the system execute the extraction run for initialization.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

You have created at least one step sequence in Customizing for the analytical process `/BA1/R2_REQ_TRAN_INIT`. You can find the required IMG activity in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*. A step sequence requires at least one parameter enrichment step that, when executed, triggers the initialization of communication between the system and the connected credit portfolio model.

Activities

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Extraction* > *Extraction Run for Initialization*.
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Process Control ID for CPDP
     Specify an ID to be transferred to the connected credit portfolio model at initialization.
   - Run Types for CPDP
   - Scenario
   - Step Sequence
     If you have created multiple step sequences, you can select the one you want to use from among them.
   - Technical Settings
     You can also call filter variants, create new run groups, define a sequential execution or execute the CVPM run as a test run.
3. Choose  .

> **Note**
>
> To display the run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.7.2.3 Extraction Run for Business Partners

Use

You can use this function to have the system execute the extraction run for business partners.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. You have created at least one step sequence in Customizing for the analytical process `/BA1/R2_BUPA_DAT_PRO`. You can find the required IMG activity in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*. You have defined at least the following steps for a step sequence:
    1. Parameter Enrichment
       You can use this step to restrict the worklist's selection. For example, you can restrict the selection so that the system only selects data records that have a blank aggregation ID and are flagged as CPDP-relevant. For this type of selection, you can use the class `/BA1/CL_AL_R2_EXTR_BUPA_PE` delivered by SAP for step execution.
    2. Worklist Creation
       You have defined a primary data source that reads the business partner ID, the aggregation ID, and the CPDP relevance indicator from the results of the credit exposure run.
    3. Data Enrichment Using Module Editor
       You have created a data enrichment module that the system can use to enrich data records from the worklist with data from other data sources (such as the Source Data Layer). This data enrichment module uses the complex key figure for business partners `/BA1/KX62CKEXB` as an output structure.
    4. Repackaging
       During step execution, the system exports data to the connected credit portfolio model. You can use the class `/BA1/CL_AL_R2_SEND_CPM` delivered by SAP for the step execution.

2. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Extraction* > *Extraction Run for Business Partners*
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - CPDP Data Transfer ID
     Specify the data transfer ID that as assigned during the initialization run of the connected credit portfolio model.
   - Process Control ID for CPDP
   - Process partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Step Sequence
     If you have created multiple step sequences, you can select the one you want to use from among them.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. This field is blank by default. If you leave the field blank, the system uses the current time stamp when the process starts.
   - Technical Settings
     You can also call filter variants, create new run groups, define a sequential execution or execute the CVPM run as a test run.
3. Choose  .

> **Note**
>
> To display the run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.7.2.4 Extraction Run for Collateral Providers

Use

You can use this function to have the system execute the extraction run for collateral providers.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that

are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. You have created at least one step sequence in Customizing for the analytical process `/BA1/R2_GUAR_DAT_PRO`. You can find the required IMG activity in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*. You have defined at least the following steps for a step sequence:
   1. Parameter Enrichment
      You can use this step to restrict the worklist's selection. For example, you can restrict the selection so that the system only selects data records that have a blank aggregation ID and are flagged as CPDP-relevant. For this type of selection, you can use the class `/BA1/CL_AL_R2_EXTR_GUAR_PE` delivered by SAP for step execution.
   2. Worklist Creation
      You have defined a primary data source that reads the business partner ID of the collateral provider, the aggregation ID, and the CPDP relevance indicator from the results of the credit exposure run.
   3. Data Enrichment Using Module Editor
      You have created a data enrichment module that the system can use to enrich data records from the worklist with data from other data sources (such as the Source Data Layer). This data enrichment module uses the complex key figure for business partners `/BA1/KX62CKEXB` as an output structure.
   4. Repackaging
      During step execution, the system exports data to the connected credit portfolio model. You can use the class `/BA1/CL_AL_R2_SEND_CPM` delivered by SAP for the step execution.
2. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Extraction* > *Extraction Run for Collateral Providers*.
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Process Control ID for CPDP
   - Process Partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Step Sequence
     If you have created multiple step sequences, you can select the one you want to use from among them.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. This field is blank by default. If you leave the field blank, the system uses the current time stamp when the process starts.
   - Technical Settings

You can also call filter variants, create new run groups, define a sequential execution or execute the CVPM run as a test run.

3. Choose  .

> **Note**
>
> To display the run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.7.2.5 Extraction Run for Exchange Rates

Use

You can use this function to have the system execute the extraction run for exchange rates.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. You have created at least one step sequence in Customizing for the analytical process `/BA1/R2_FXRATE_DAT_PRO`. You can find the required IMG activity in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*. You have defined at least the following steps for a step sequence:
    1. Creation of Worklist
       You have created a primary data source that reads the exchange rates from market data. To do this, you must create an application in Customizing for *Financial Services* under *Foundation* > *Market Data* > *Create Application* You have specified the application as an attribute in the primary data source so that the system can use it. You do this in Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.
    2. Data Enrichment Using Module Editor
       You have created a data enrichment module that the system can use to enrich data records from the worklist with data from other data sources (such as the Source Data Layer). This data enrichment module uses the complex key figure for exchange rates `/BA1/KX62CKEXF` as an output structure.

3. Repackaging
   During step execution, the system exports data to the connected credit portfolio model. You can use the class `/BA1/CL_AL_R2_SEND_CPM` delivered by SAP for the step execution.

2. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Extraction* > *Extraction Run for Exchange Rates*
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Process Control ID for CPDP
   - Process Partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Step Sequence
     If you have created multiple step sequences, you can select the one you want to use from among them.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. This field is blank by default. If you leave the field blank, the system uses the current time stamp when the process starts.
   - Technical Settings
     You can also call filter variants, create new run groups, define a sequential execution or execute the CVPM run as a test run.
3. Choose .

> **Note**
>
> To display the run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.7.2.6 Extraction Run for Yield Curves

Use

You can use this function to have the system execute the extraction run for yield curves.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that

are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. You have created at least one step sequence in Customizing for the analytical process `/BA1/R2_YLDCV_DAT_PRO`. You can find the required IMG activity in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*. You have defined at least the following steps for a step sequence:
   1. Creation of Worklist
      You have created a primary data source that reads the yield curves from market data. To do this, you must create an application in Customizing for *Financial Services* under *Foundation* > *Market Data* > *Create Application* You have specified the application as an attribute in the primary data source so that the system can use it. You do this in Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.
   2. Data Enrichment Using Module Editor
      You have created a data enrichment module that the system can use to enrich data records from the worklist with data from other data sources (such as the Source Data Layer). This data enrichment module uses the complex key figure for yield curves `/BA1/KX62CKEXY` as an output structure.
   3. Repackaging
      During step execution, the system exports data to the connected credit portfolio model. You can use the class `/BA1/CL_AL_R2_SEND_CPM` delivered by SAP for the step execution.
2. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Extraction* > *Extraction Run for Yield Curves*
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Process Control ID for CPDP
   - Process Partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Step Sequence
     If you have created multiple step sequences, you can select the one you want to use from among them.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. This field is blank by default. If you leave the field blank, the system uses the current time stamp when the process starts.
   - Technical Settings
     You can also call filter variants, create new run groups, define a sequential execution or execute the CVPM run as a test run.

3. Choose  .

> **Note**
>
> To display the run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.7.2.7 Extraction Run for Spread Matrices

Use

You can use this function to have the system execute the extraction run for spread matrices.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. You have created at least one step sequence in Customizing for the analytical process `/BA1/R2_SPRDMTRX_DAT_PRO`. You can find the required IMG activity in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*. You have defined at least the following steps for a step sequence:
   1. Worklist Creation
      You have created a primary data source that reads the spread curves from market data. To do this, you must create an application in Customizing for *Financial Services* under *Foundation* > *Market Data* > *Create Application* You have specified the application as an attribute in the primary data source so that the system can use it. You do this in Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.
   2. Data Enrichment Using Module Editor
      You have created a data enrichment module that the system can use to enrich data records from the worklist with data from other data sources (such as the Source Data Layer). This data enrichment module uses the complex key figure for spread matrices `/BA1/KX62CKEXS` as an output structure.
   3. Repackaging

During step execution, the system exports data to the connected credit portfolio model. You can use the class `/BA1/CL_AL_R2_SEND_CPM` delivered by SAP for the step execution.

2. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Extraction* > *Extraction Run for Spread Matrices*
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Process Control ID for CPDP
   - Process partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Step Sequence
     If you have created multiple step sequences, you can select the one you want to use from among them.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. This field is blank by default. If you leave the field blank, the system uses the current time stamp when the process starts.
   - Technical Settings
     You can also call filter variants, create new run groups, define a sequential execution or execute the CVPM run as a test run.
3. Choose .

> **Note**
>
> To display the run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.7.2.8 Extraction Run for PD Matrices

Use

You can use this function to have the system execute the PD matrix extraction run.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that

are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. You have created at least one step sequence in Customizing for the analytical process `/BA1/R2_PDMTRX_DAT_PRO`. You can find the required IMG activity in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*. You have defined at least the following steps for a step sequence:
   1. Worklist Creation
      You have created a primary data source that reads the PD matrices from generic market data, for example. To do this, you must create a market data class in Customizing for *Bank Analyzer* under *Source Data Layer* > *Primary Objects* > *Generic Market Data* > *Edit Generic Market Data Classes*. To ensure that the system can use it, you have specified the generic market data class and the market data area as an attribute in the primary data source. You do this in Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.
   2. Data Enrichment Using Module Editor
      You have created a data enrichment module that the system can use to enrich data records from the worklist with data from other data sources (such as the Source Data Layer). This data enrichment module uses the complex key figure for PD matrices `/BA1/KX62CKEXP` as an output structure.
   3. Repackaging
      During step execution, the system exports data to the connected credit portfolio model. You can use the class `/BA1/CL_AL_R2_SEND_CPM` delivered by SAP for the step execution.
2. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Extraction* > *Extraction Run for PD Matrices*
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Process Control ID for CPDP
   - Process partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Step Sequence
     If you have created multiple step sequences, you can select the one you want to use from among them.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. This field is blank by default. If you leave the field blank, the system uses the current time stamp when the process starts.
   - Technical Settings

You can also call filter variants, create new run groups, define a sequential execution or execute the CVPM run as a test run.

3. Choose  .

> **Note**
>
> To display the run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.7.2.9 Transition Matrix Extraction Run

Use

You can use this function to have the system execute the extraction run for transition matrices.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. You have created at least one step sequence in Customizing for the analytical process `/BA1/R2_RTTRNSMTRX_DAT_PRO`. You can find the required IMG activity in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*. You have defined at least the following steps for a step sequence:
    1. Creation of Worklist
       You have created a primary data source that reads the transition matrices from the historical database (HDB), for example. To do so, you must create a data layer in Customizing for *Bank Analyzer* under *Analytics* > *Historical Database* > *Set Up Data Layers* > *Configure Data Layers*. You have specified the data layer ID as an attribute in the primary data source so that the system can use the data layer. You do this in Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.
    2. Data Enrichment Using Module Editor
       You have created a data enrichment module that the system can use to enrich data records from the worklist with data from other data sources (such as the Source Data Layer). This data enrichment module uses the complex key figure for transition matrices `/BA1/KX62CKEXR` as an output structure.

3. Repackaging
   During step execution, the system exports data to the connected credit portfolio model. You can use the class `/BA1/CL_AL_R2_SEND_CPM` delivered by SAP for the step execution.

2. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Extraction* > *Extraction Run for Transition Matrices*
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Process Control ID for CPDP
   - Process Partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Step Sequence
     If you have created multiple step sequences, you can select the one you want to use from among them.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. This field is blank by default. If you leave the field blank, the system uses the current time stamp when the process starts.
   - Technical Settings
     You can also call filter variants, create new run groups, define a sequential execution or execute the CVPM run as a test run.
3. Choose .

> **Note**
>
> To display the run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.7.2.10 Extraction Run for Exposures

Use

You can use this function to have the system execute the extraction run for exposures.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that

are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. You have created at least one step sequence in Customizing for the analytical process `/BA1/R2_EXPOSURE_DAT_PRO`. You can find the required IMG activity in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*. You have defined at least the following steps for a step sequence:
   1. Parameter Enrichment
      You can use this step to restrict the worklist's selection. For example, you can restrict the selection so that the system only selects data records that have a blank aggregation ID and are flagged as CPDP-relevant. For this type of selection, you can use the class `/BA1/CL_AL_R2_EXTR_EXP_PE` delivered by SAP for step execution.
   2. Creation of Worklist
      You have defined a primary data source taht reads all exposure data from the results of the credit exposure run.
   3. Data Enrichment Using Module Editor
      You have created a data enrichment module that the system can use to enrich data records from the worklist with data from other data sources (such as the Source Data Layer). This data enrichment module uses the complex key figures for exposures `/BA1/KX62CKEXE` and portfolio data `/BA1/KX62CKEXA` as an output structure.
   4. Repackaging
      During step execution, the system exports data to the connected credit portfolio model. You can use the class `/BA1/CL_AL_R2_SEND_CPM` delivered by SAP for the step execution. You have specified the characteristic `/BA1/C62SVOPTY` as a repackaging field.
2. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Extraction* > *Extraction Run for Exposures*
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Process Control ID for CPDP
   - Run Types for CPDP
   - Process Partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Step Sequence
     If you have created multiple step sequences, you can select the one you want to use from among them.
   - Time Stamp for Selection of Results Data and Analytical Data

Enter the time stamp. This field is blank by default. If you leave the field blank, the system uses the current time stamp when the process starts.

- Technical Settings
  You can also call filter variants, create new run groups, define a sequential execution or execute the CVPM run as a test run.

3. Choose  .

> **Note**
>
> To display the run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.7.2.11 Extraction Run for Aggregated Exposures

Use

You can use this function to have the system execute the extraction run for aggregated exposures.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. You have created at least one step sequence in Customizing for the analytical process `/BA1/R2_EXPOSURE_AGG_DAT_PRO`. You can find the required IMG activity in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*. You have defined at least the following steps for a step sequence:
   1. Creation of Worklist
      You have defined a primary data source taht reads all exposure data from the results of the aggregation run.
   2. Data Enrichment Using Module Editor
      You have created a data enrichment module that the system can use to enrich data records from the worklist with data from other data sources (such as market data). This data enrichment module uses the complex key figures for exposures `/BA1/KX62CKEXE`, business partners `/BA1/KX62CKEXB` and portfolio data `/BA1/KX62CKEXA` as an output structure.

3. Repackaging
   During step execution, the system exports data to the connected credit portfolio model. You can use the class `/BA1/CL_AL_R2_SEND_CPM` delivered by SAP for the step execution.

2. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Extraction* > *Extraction Run for Aggregated Exposures*
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Process Control ID for CPDP
   - Process Partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Step Sequence
     If you have created multiple step sequences, you can select the one you want to use from among them.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. This field is blank by default. If you leave the field blank, the system uses the current time stamp when the process starts.
   - Technical Settings
     You can also call filter variants, create new run groups, define a sequential execution or execute the CVPM run as a test run.
3. Choose .

> **Note**
>
> To display the run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.7.2.12 Disaggregation Run

Use

You can use this function to have the system execute the disaggregation run.

Integration

The system manages this process using Calculation and Valuation Process Manager [page 1622] (CVPM). CVPM provides a framework with which you can create your own analytical processes, or use analytical processes that

are preconfigured by SAP. You can track and analyze all the runs for analytical processes in the CVPM process monitor [page 1635]. An orchestration tool enables you to integrate CVPM processes into a process chain.

Prerequisites

1. You have created at least one step sequence in Customizing for the analytical process `/BA1/R2_CPM_DAGGR_CVPM`. You can find the required IMG activity in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*. You have defined at least the following steps for a step sequence:
   1. Creation of Worklist
   2. Data Enrichment Using Module Editor
   3. Save
2. You have executed the function Set the SDL Time Stamp [page 367].

Activities

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Disaggregation Run*
2. On the selection screen, make the following entries:
   - Business Record Date
     Enter the time stamp.
   - Credit Portfolio Analysis Run ID
     Specify the ID that the connected credit portfolio model returns with the results of the analysis run.
   - Run Types for CPDP
   - Scenario
   - Process Partition
     Specify the process partition that you created when you set the SDL time stamp.
   - Step Sequence
     If you have created multiple step sequences, you can select the one you want to use from among them.
   - Time Stamp for Selection of Results Data and Analytical Data
     Enter the time stamp. This field is blank by default. If you leave the field blank, the system uses the current time stamp when the process starts.
   - Technical Settings
     You can also call filter variants, create new run groups, define a sequential execution or execute the CVPM run as a test run.
3. Choose .

> **Note**
>
> To display the run, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*. Select the process and display the application log of the run required.

4.7.3 Default Model Credit Portfolio Model (DMCPM)

Use

You use this component to quantify credit risks for portfolios of financial products. You can calculate risk measures such as expected loss, value at risk, and expected shortfall for the financial transactions of a credit portfolio, as well as the risk contributions derived from these measures. The component thus provides tools that you can use for your own management of credit risks. You can use the calculated key figures, for example, to actively manage credit risks or define limits by industries and countries.

For the default mode credit portfolio model (DMCPM), the loss distribution of loan losses is the starting point for deriving the risk key figures. The model can be displayed mathematically in a self-contained, analytical form and does not need many input parameters. Its simplicity and transparency make it particularly suitable for homogeneous portfolios of borrowers with a small to moderate default probability.

> **Note**
>
> For more information about the availability of the *Default Mode Credit Portfolio Model*, see SAP Note 2316265.

Integration

The system reads the initial data from the Source Data Layer (SDL) and saves the calculated key figures in the Results Data Layer (RDL). For reporting purposes, you can extract the results data to SAP NetWeaver Business Intelligence (BI).

Features

You can calculate the following standard key figures:

- Expected loss (EL)
- Risk measures
  - Standard deviation
  - Value at risk (VaR)
  - Expected shortfall (ES)
- Risk contributions
  - Risk contribution for standard deviation
  - Risk contribution for value at risk
  - Risk contribution for expected shortfall

To calculate the key figures, you can use a separate process (see DMCPM Process [page 1317]). For more information about the key figures calculated, see Risk Measures and Risk Contributions [page 1316]. You can also calculate the key figures for scenario data.

You can use the following standard calculation methods:

- Stable recursion
- Panjer recursion
- Fourier inversion
- Saddle point method

The first two methods carry out the recursion on the basis of the discretized exposure (loss at default). You can use the other two calculation methods on the basis of the discretized loss at default or on a continuous basis. For information about using the calculation methods, see Modeling and Calculation in DMCPM [page 1314].

You can create your own calculation methods using Business Add-Ins (BAdIs). You can do so, for example, to calculate additional key figures.

More Information

The Business Content for the Bank Analyzer component contains detailed examples on using DMCPM. For more information, see the Business Content documentation.

4.7.3.1 Modeling and Calculation in DMCPM

Use

The default mode credit portfolio model (DMCPM) describes the potential losses of a credit portfolio using the loss distribution of loan losses. The basic assumption is that the borrowers have a predominantly low to moderate default probability. The default indicator is described based on a Poisson distribution. To take into account macroeconomic factors on the loan losses, the model is extended to include Gamma-distributed random variables. These stand for sectors such as countries or industries to which the individual borrowers are assigned. It is assumed that the defaults determined by these macroeconomic factors are independent of one another.

From a mathematical perspective, the DMCPM is a simple model since it is self-contained and analytically solvable (for a detailed description of the model, see Credit Suisse First Boston: CreditRisk+: A Credit Risk Management Framework, 1997).

Input Parameters

Since only the number and amount of loan losses are taken into account and not the reasons for loan losses, the model does not need many input parameters. The following parameters must be provided:

- Exposures of the transaction
- Loss unit (for discretized calculation only)
- Default probability of the business partners
- Weighting vector of the business partners across the sectors
- Sector volatilities for the systematic sectors

The Customizing settings for the DMCPM process also include additional parameters that specify, for example, market data, the aggregation of key figures at business partner level, and the confidence level of the value at risk.

Calculation Method

Determine Loss Distribution

Using the input parameters, the methods can determine the loss distribution, from which the distribution of losses for a portfolio can be derived. The following standard calculation methods can be used for this:

- Stable recursion
  This algorithm traces back to Haaf, Reiß, and Schoenmakers, and in contrast to Panjer recursion described below, is numerically stable; in other words, rounding errors do not accumulate (for details on how this algorithm works, see: H. Haaf, O. Reiss, J. Schoenmakers: Numerically Stable Computation of CreditRisk+, in F. Lehrbass, M. Gundlach (ed.): CreditRisk+ in the banking industry, 2004).
- Panjer recursion
  This algorithm, traced back to Panjer, is a standard method for calculating loss distribution for "compound distributions" (for details on how this algorithm works, see Credit Suisse First Boston: CreditRisk+: A Credit Risk Management Framework, 1997).
- Fourier inversion
  This algorithm is an alternative to the aforementioned recursions and can be used on the basis of both discretized and non-discretized lost-at-default exposure sizes (for details on how this algorithm works, see O. Reiss: Fourier Inversion Techniques for CreditRisk+, in F. Lehrbass, M. Gundlach (ed.): CreditRisk+ in the banking industry, 2004).

Calculate Risk Measures and Risk Contributions

From the default distribution, you can calculate the standard distribution, quantiles such as value at risk and expected shortfall, as well as the corresponding risk contributions (see Risk Measures and Risk Contributions [page 1316]).

> **Note**
>
> The system calculates the expected loss independent of the calculation methods for DMCPM in the DMCPM preliminary run (see DMCPM Preliminary Run [page 1318]).

In addition to Panjer recursion, stable recursion, and Fourier inversion, you can also use the **saddle point method** to calculate the risk measures and risk contributions (for details on how this algorithm works, see M. B. Gordy: Saddlepoint Approximation, in F. Lehrbass, M. Gundlach (ed.): CreditRisk+ in the banking industry, 2004).

Note the following information with regard to the calculation methods:

- Stabile recursion and Panjer recursion
  You can calculate all the standard risk measures and risk contributions available.
- Saddle point method
  You can calculate all the standard risk measures and risk contributions available, except for the risk contribution for the expected shortfall.
- Fourier inversion
  - In a discretized calculation, you can calculate all the standard risk measures and risk contributions available.
  - For continuous calculation, we recommend that you calculate the loss distribution only.

More Information

DMCPM Process [page 1317]

4.7.3.2 Risk Measures and Risk Contributions

Use

Risk Measures

Risk measures describe the degree of uncertainty of losses in a credit portfolio.

You can calculate the following standard risk measures:

- Standard deviation
  The standard deviation is also used as a key figure for unexpected loss.
- Value at risk
  With a probability of **a**, this is the maximum loss value to be expected, where **a** represents the confidence level (**a**-quantile of the probability of default).
  The value at risk is also used as a key figure for the risk capital that is required to cover unexpected losses.
- Expected shortfall
  Expected value of loss when the loss exceeds the value at risk.
  The expected shortfall is particularly useful for analyzing extreme potential losses. It is also called the conditional value at risk.

> **Note**
>
> In contrast to the value at risk, the expected shortfall is a coherent risk measure, and in particular, subadditive: Splitting a portfolio into subportfolios so that the intersection between two subportfolios is empty and the union of all subportfolios is equal to the entire portfolio leads to the values for the expected shortfall, for which the sum is greater than or equal to the expected shortfall of the entire portfolio. In contrast, the value at risk can increase through diversification, which is why the expected shortfall is often the preferred risk measure for credit portfolios.

Whereas the standard deviation can be calculated directly from the input parameters of the default mode credit portfolio model (DMCPM), the value at risk and the expected shortfall are calculated based on the loss distribution determined in the DMCPM process.

Risk Contributions

Risk contributions indicate to what extent a borrower or a single transaction contributes to the entire risk of a credit portfolio. This is measured based on the risk measure. The risk contributions therefore help you ascertain which borrowers are the largest risk drivers in the credit portfolio, for example.

You can calculate the following standard risk contributions:

- Risk contribution for standard deviation
- Risk contribution for credit value at risk
- Risk contribution for expected shortfall

More Information

Modeling and Calculation in DMCPM [page 1314]

4.7.3.3 DMCPM Process

Use

You use this process to calculate risk measures and risk contributions for portfolios and financial products as part of the default mode credit portfolio model (DMCPM).

Prerequisites

You have carried out all IMG activities in Customizing for Bank Analyzer under *Processes and Methods* > *Credit Risk* > *Default Mode Credit Portfolio Model*. You have already made the following settings:

- Define BAdI Settings
  The calculation methods of the DMCPM process are implemented in Business Add-Ins (BAdIs). You specify which methods you want to use in the BAdI settings.
- Make Read/Write Settings
  The system stores the results of the subprocesses of the DMCPM process in the Results Data Layer (RDL). In the read/write settings, you specify where the system can find the data for the subprocesses and where it saves the results data of the DMCPM process.
- Define Run Types for DMCPM
  The DMCPM run type comprises the key parameters of the DMCPM process. In addition to the BAdI settings and the read/write settings, there are also other settings, for example, that specify the market data area and exchange rate category, the sectors to be used, the confidence level for the risk measures, and whether the calculation is to be discretized.

You have also specified the following input parameters in the system:

- Default probabilities
  You have imported the default probability from the source systems or entered it manually for the business partner. To enter the parameters, go to the *SAP Easy Access* screen and choose *Financial Services* > *Business Partner* > *Maintain Business Partner*. From there, you can enter the parameters under the *BP: Counterparty* business partner role on the *Credit Risk* tab page.
- Volatilities of the systematic sectors
  In Customizing for Bank Analyzer under *Processes and Methods* > *Credit Risk* > *Default Mode Credit Portfolio Model* > *Make Read/Write Settings*, you have specified the key figure that you want to use to deliver the sector volatilities. You have also specified the primary data source that the system can use to read the data from the generic data in the Source Data Layer (SD).
  Alternatively, you can import volatilities of the default probabilities of the business partners from source systems or enter them manually in the position for the business partner. These are converted into sector volatilities by the system in the DMCPM process. The sector volatility of a sector is equal to the total of the weighted PD volatilities of all business partners that contribute to this sector.

Process

The DMCPM process comprises the following subprocesses, which from a technical viewpoint, are not dependent on each other. With the exception of the DMCPM preliminary run, you manage the following subprocesses with the Calculation and Valuation Process Manager (CVPM).

1. Set SDL Time Stamp [page 367]
   CVPM process: /BA1/RB0_CLOSE_SDL
2. DMCPM Preliminary Run [page 1318] (credit exposure run)
   CVPM process: /BA1/R2_CRA_MAIN_CRE_CVPM
3. DMCPM Preliminary Processing [page 1320] (aggregation, enrichment, and distribution of data records)
   CVPM process: /BA1/RR2_DMCPM_PREP
4. DMCPM Calculation [page 1322] (calculation of loss distribution and risk measures)
   CVPM process: /BA1/RR2_DMCPM_CALC
5. DMCPM Postprocessing [page 1324] (calculation and distribution of risk contributions at transaction level)
   CVPM process: /BA1/RR2_DMCPM_POSTP

The process steps save their results data in the Results Data Layer (RDL). From there, you can access the results for reporting.

> **Note**
>
> For test purposes, you can use the following ad hoc processes:
>
> - DMCPM Ad-Hoc Calculation (CVPM process: /BA1/RR2_DMCPM_ADHOC)
> - DMCPM Ad-Hoc Calculation (CVPM process: /BA1/RR2_DMCPM_AD_PP)

More Information

- Default Model Credit Portfolio Model (DMCPM) [page 1313]
- Calculation and Valuation Process Manager (CVPM) [page 1622]

4.7.3.3.1 DMCPM Preliminary Run

Use

You use this function to calculate exposures at transaction level as well as other key figures required by the default mode credit portfolio model (DMCPM). You carry out the DMCPM preliminary run before DMCPM preliminary processing.

Integration

The Calculation and Valuation Process Manager (CVPM) handles this function, which is one of the analytical processes. You can track and analyze all the runs for analytical processes in the CVPM process monitor. For more information, see Calculation and Valuation Process Manager (CVPM) [page 1622].

> **Note**
>
> For this subprocess, you use the credit exposure run, which can also be used by other credit risk applications. For more information, see Credit Exposure Calculation [page 1196].

Prerequisites

You have made the settings for the credit exposure calculation in Customizing for Bank Analyzer under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Calculation*. In particular, under *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *Calculation* > *Define Calculation Methods*, you have defined a calculation method of the **Default Mode Credit Portfolio Model** credit risk calculation category.

Features

The system carries out the following steps:

1. Read financial transactions from the Source Data Layer (SDL)
2. Calculate the following key figures for the individual financial transactions:
   - Exposure at default of credit
   - Expected loss
   - Default probability
3. Save results in the RDL

Activities

Schedule the process in CVPM (analytical process `/BA1/R2_CRA_MAIN_CRE_CVPM`), or on the SAP Easy Access screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Exposure* > *CVPM Run for Credit Exposure* (transaction `/BA1/R2_CRE_M_CVPM`).

More Information

DMCPM Process [page 1317]

4.7.3.3.2 Preliminary Processing in Default Mode Credit Portfolio Model (DMCPM)

Use

You use this function to prepare data for the calculation of risk measures and the risk contribution for the Default Mode Credit Portfolio Model (DMCPM). You start the preliminary processing for DMCPM after you have calculated risk key figures in the credit exposure run, and before the DMCPM calculation.

You can use this function to read parameter values, such as sector weights from the transaction level, or to aggregate risk key figures at the business partner level. This is necessary since the subsequent subprocesses in the DMCPM process require key figures at the business partner level.

Integration

The Calculation and Valuation Process Manager (CVPM) handles this function, which is one of the analytical processes. You can track and analyze all the runs for analytical processes in the CVPM process monitor. For more information, see Calculation and Valuation Process Manager (CVPM) [page 1622].

Prerequisites

- Data Enrichment
  1. In Customizing for Bank Analyzer under *Processes and Methods* > *Credit Risk* > *Default Mode Credit Portfolio Model* > *Module Editor* > *Edit Modules*, you have created a module that can be used to enrich data, discretize, and distribute result records to the relevant sectors.
  2. In Customizing for Bank Analyzer under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*, you have assigned the module to the analytical process `/BA1/RR2_DMCPM_PREP`. You do this for the *Enrich Module Editor Data* step.

  > **Note**
  >
  > You can use environment `DMCPM_PRE_PROC` and module `DMM_PREPROCESSING` as examples for your Customizing settings, and adjust them to suit your purposes.

  You have also defined the relevant input parameters, such as sector weights, probabilities of default, and variances of probabilities of default for the business partners.
- Aggregation at Business Partner Level
  You have the following options:
  - Aggregation in the Results Data Layer
    The system performs aggregation in the Results Data Layer (RDL) before you execute the DMCPM preliminary processing. For more information, see the Aggregation [page 1374] document in the RDL documentation.

- Aggregation During DMCPM Preliminary Processing
  1. You have created an aggregation ID in Customizing for Bank Analyzer under *Processes and Methods* > *Credit Risk* > *Default Mode Credit Portfolio Model* > *Aggregation* > *Edit Aggregation*.
  2. In Customizing for Bank Analyzer under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*, you have assigned the aggregation ID to the analytical process `/BA1/RR2_DMCPM_PREP`. You do this at process step level.

Features

DMCPM preliminary processing comprises the following steps:

> **Note**
>
> Note that your Customizing settings may change the steps in the process, or add further steps to it.

1. Parameter Enrichment
2. The system reads the worklist from the Results Data Layer (RDL)
3. Aggregation at Business Partner Level
   If the exposures are available at transaction level, the system adds up the exposures for each business partner. All other characteristics and key figures are unchanged.
4. Data Enrichment
   The system reads the following data to obtain business partner data:
   - Sector weights
   - Variance of the probabilities of default
5. Discretization
   If the DMCPM calculation is discretized, the system performs the following calculations:
   - Distribute the exposures to the relevant exposure bands
   - Adjust the probabilities of default for business partners
     The system adjusts the probability of default such that the expected loss for each business partner is unchanged, even though the system rounds the exposure values when distributing exposures across the exposure band.

     > **Note**
     >
     > Note that this step is optional, since it may be unnecessary for a highly-detailed discretization.
6. Distribution of Result Records
   The system divides the result records so that there are exposure values for business partners for each sector. It weights all the key figures, apart from the loss amount, using the sector weights. Result records that have a zero weighting are removed.

   > **Note**
   >
   > You can also refine the result records by using the portfolio assignment. Dividing the result records in this way is optional, and is done by specifying granularity characteristics in the aggregation tool.
7. Saving of data in the Results Data Layer (RDL)

Activities

Schedule the process in CVPM (analytical process `/BA1/RR2_DMCPM_PREP`), or on the SAP Easy Access screen choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Default Mode Credit Portfolio Model* > *Default Mode for Credit Portfolio Model: Preliminary Processing* (transaction `/BA1/DMCPM_PREP`).

If you want to execute the calculation for scenario data, enter the scenario ID used for the data in the Source Data Layer (SDL) or the Resource Data Layer (RDL).

More Information

DMCPM Process [page 1317]

4.7.3.3.3 DMCPM Calculation

Use

You use this function to calculate the loss distribution and risk measures in the default mode credit portfolio model (DMCPM). You carry out the DMCPM calculation after DMCPM preprocessing and before DMCPM postprocessing.

Integration

The Calculation and Valuation Process Manager (CVPM) handles this function, which is one of the analytical processes. You can track and analyze all the runs for analytical processes in the CVPM process monitor. For more information, see Calculation and Valuation Process Manager (CVPM) [page 1622].

Prerequisites

- Calculation steps
  The calculation steps are implemented in Business Add-Ins (BAdIs). For the standard calculation methods, you have defined separate BAdI implementations and assigned these to the DMCPM run type as follows:
  1. In Customizing for Bank Analyzer under *Processes and Methods* > *Credit Risk* > *Default Mode Credit Portfolio Model* > *Define BAdI Settings*, you have compiled the BAdI implementations for the calculation steps in an ID for BAdI settings.
  2. In Customizing for Bank Analyzer under *Processes and Methods* > *Credit Risk* > *Default Mode Credit Portfolio Model* > *Define Run Types for DMCPM*, you have assigned the ID for BAdI settings to the DMCPM run type.

- Discretized calculation
  - Aggregation
    1. You have created an aggregation ID in Customizing for Bank Analyzer under *Processes and Methods* > *Credit Risk* > *Default Mode Credit Portfolio Model* > *Aggregation* > *Edit Aggregation*.
    2. In Customizing for Bank Analyzer under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Step Sequences for Analytical Processes*, you have assigned the aggregation ID to the analytical process `/BA1/RR2_DMCPM_PREP`. You do this at process step level.
  - Discretization width of the loss distribution

    You have specified a loss unit in Customizing for Bank Analyzer under *Processes and Methods* > *Credit Risk* > *Default Mode Credit Portfolio Model* > *Define Run Types for DMCPM*. The system uses it as the width of the exposure bands when distributing loss to the exposure bands.
- Use of Fourier inversion

  In Customizing for Bank Analyzer under *Processes and Methods* > *Credit Risk* > *Default Mode Credit Portfolio Model* > *Define Run Types for DMCPM*, you have specified the number of grid points that the system uses for the fast Fourier transform.

  Alternatively, in Customizing for Bank Analyzer under *Processes and Methods* > *Credit Risk* > *Default Mode Credit Portfolio Model* > *Define Run Types for DMCPM*, you can specify a loss unit with which the system then calculates the grid points for the fast Fourier transform.

Features

The system carries out the following steps:

> **Note**
>
> Note that your Customizing settings may change the steps in the process, or add further steps to it.

1. Enrich parameters
2. Read worklist from the Results Data Layer (RDL)
3. Aggregate (for discretized calculation only)
   For all business partners, the system aggregates the data records that lie in the same sector and are assigned to the same exposure band.
4. Calculate loss distribution (not for saddle-point method)
   When using Fourier inversion, the system calculates the entire loss distribution.
   When using stable recursion and Panjer recursion, the system terminates the calculation as soon as the predefined confidence level (defined in the DMCPM run type) has been reached for the value at risk or the expected shortfall.
5. Caculate risk measures
   The system calculates the value at risk and the expected shortfall, for example.
6. Save results in the RDL

   > **Note**
   >
   > Note that the system stores the calculated key figures in the RDL by default, without information about the calculation methods used. However, you can view the calculation methods used in the application logs of the DMCPM calculation.

> **Note**
>
> If you use Fourier inversion, a separate step sequence is available in which the system carries out individual calculation steps in parallel, which can lead to faster calculations.

Activities

Schedule the process in CVPM (analytical process `/BA1/RR2_DMCPM_CALC`), or on the SAP Easy Access screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Default Mode Credit Portfolio Model* > *DMCPM Calculation* (transaction `/BA1/DMCPM_CALC`).

You can choose from the following options:

- Scenarios
  If you want to execute the calculation for scenario data, enter the scenario ID used for the data in the Source Data Layer (SDL) or the Resource Data Layer (RDL).
- Credit Portfolio ID
  If you want to perform the calculation for subportfolios, specify the credit portfolio ID. The system then takes into account only those financial transactions for which the selected credit portfolio ID was derived in the DMCPM preliminary run (credit exposure calculation).

More Information

DMCPM Process [page 1317]

4.7.3.3.4 DMCPM Postprocessing

Use

You use this function to calculate risk contributions at business partner level and distribute them at transaction level. You execute this function for the default mode credit portfolio model (DMCPM) after the DMCPM calculation.

Integration

The Calculation and Valuation Process Manager (CVPM) handles this function, which is one of the analytical processes. You can track and analyze all the runs for analytical processes in the CVPM process monitor. For more information, see Calculation and Valuation Process Manager (CVPM) [page 1622].

Prerequisites

In Customizing for Bank Analyzer under *Processes and Methods* > *Credit Risk* > *Default Mode Credit Portfolio Model* > *Define Run Types for DMCPM*, you have specified a key figure that the system uses to redistribute the risk contribution calculated for a business partner to the individual transactions.

Features

The system carries out the following steps:

> **Note**
>
> Note that your Customizing settings may change the steps in the process, or add further steps to it.

1. Enrich parameters
2. Read worklist from the Results Data Layer (RDL)
3. Calculate risk contributions at business partner level
   The system calculates the risk contribution for the value at risk and the expected shortfall, for example.
4. Distribute risk contributions at transaction level
   The system distributes the risk contributions pro rata to the relevant transactions. To do so, it uses the key figure defined in the DMCPM run type, for example, the expected loss or the loss at default. The system then saves the results in the Results Data Layer (RDL).
5. Save results in the RDL

> **Note**
>
> Note that the system stores the calculated key figures in the RDL by default, without information about the calculation methods used. However, you can view the calculation methods used in the application logs of the DMCPM calculation.

Activities

Schedule the process in CVPM (analytical process `/BA1/RR2_DMCPM_POSTP`), or on the SAP Easy Access screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Default Mode Credit Portfolio Model* > *DMCPM Calculation* (transaction `/BA1/DMCPM_POSTP`).

If you want to perform the calculation for subportfolios, specify the credit portfolio ID. The system then takes into account only those financial transactions for which the selected credit portfolio ID was derived in the DMCPM preliminary run (credit exposure calculation).

> **Note**
>
> If multiple runs of the DMCPM calculation exist with the same parameters, you also have to specify a credit portfolio analysis run ID. This ID is generated by the system during the DMCPM calculation and is used to uniquely identify the results data stored in the RDL.

More Information

DMCPM Process [page 1317]

4.7.4 Prototype for Country Risk

Use

The prototype function for *country risk* provides an infrastructure for calculations and can be defined by the customer as required. Calculations are primarily used to determine attributable amounts for individual transactions.

> **Note**
>
> You can also calculate risk parameters and attributable amounts for counterparty risk and country risk as part of the credit exposure run. In contrast to the prototype for country risk calculation, enhanced functions are provided here.
>
> For more information, see Calculation of Counterparty and Country Risk [page 1193].

Integration

The Country Risk component is part of Bank Analyzer. You can use the results generated by the General Calculation and Valuation Methods as input for the Country Risk component. You can process the attributable amounts calculated in the Country Risk component in Limit Manager [page 1529].

For more information, see the following:

Architecture of Country Risk [page 1327]

Interaction Between Country Risk and Limit Manager [page 1347]

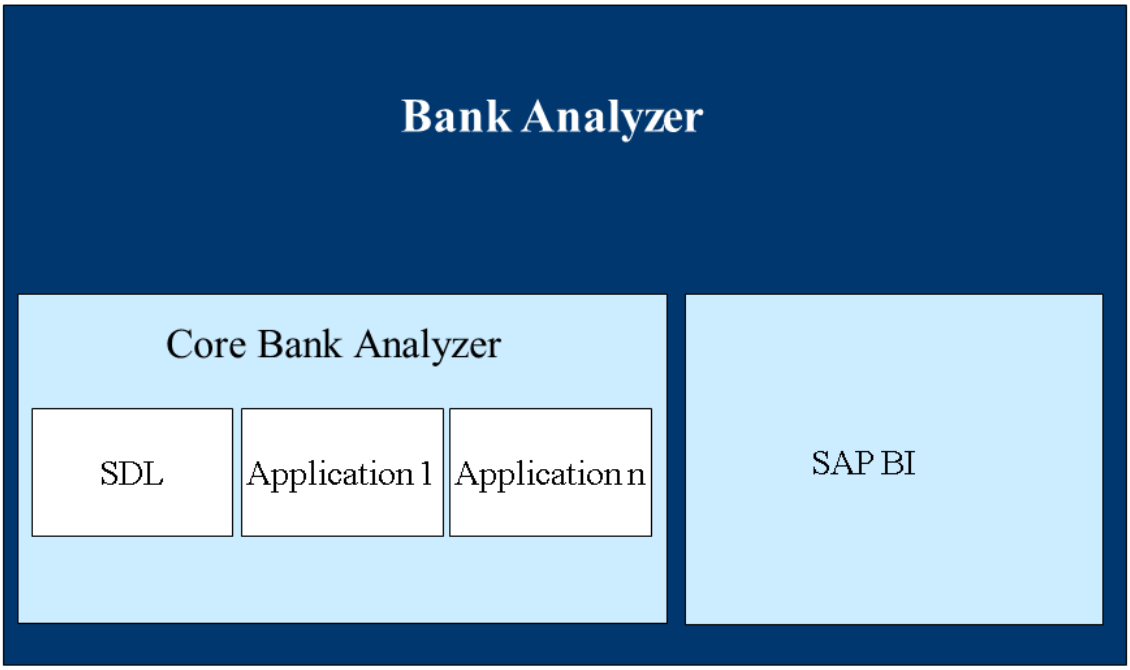
Features

Since in practice a large number of methods are used to measure the exposure to default risk, a flexible and customizable interface is provided in the Country Risk component for the analysis of financial transactions such as

loans and facilities. For each transaction entered in the system, the system calculates attributable amounts that disclose the risk content of each transaction. Formulas are assigned for each combination of determination procedure and default risk rule defined in Customizing. The formulas are stored in each transaction.

4.7.4.1 Architecture of Country Risk

The country risk component is integrated in the Bank Analyzer architecture, which comprises the core Bank Analyzer system and SAP NetWeaver Business Intelligence (BI). Core Bank Analyzer contains different applications and the Source Data Layer (SDL). The SDL provides functions and services for managing bank data. The BI technology supports data administration and reporting, for example.

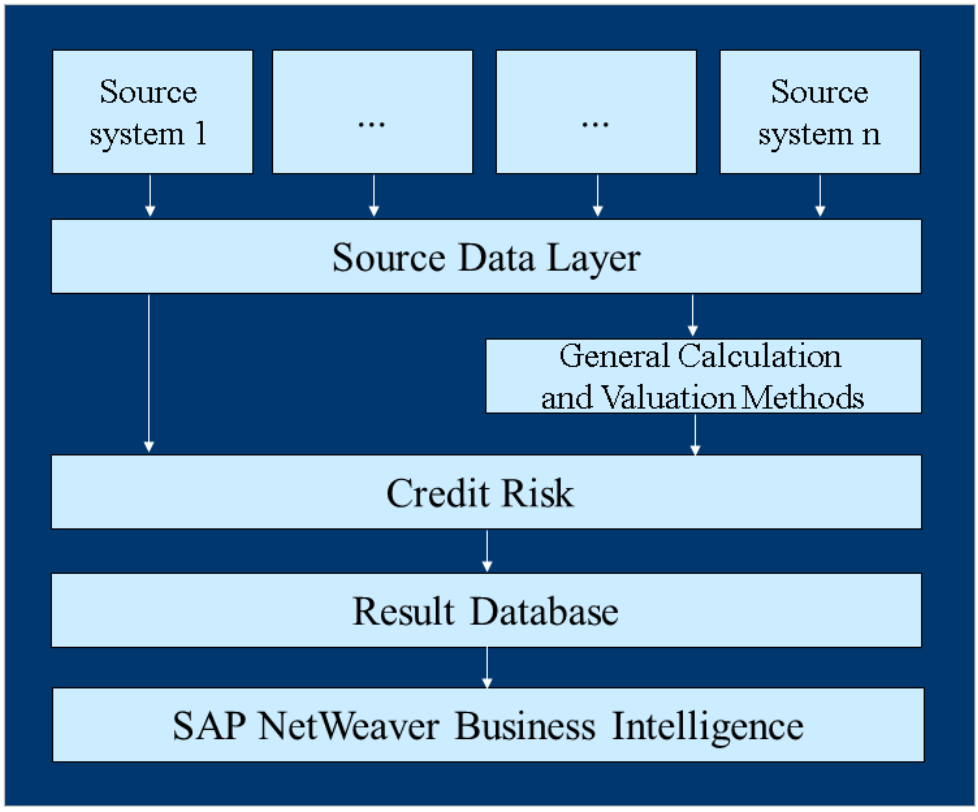


Data Flows

Various source systems supply data to the Country Risk component. Rather than communicating directly with Country Risk or with the upstream General Calculation and Valuation Methods, the data from these source systems is first stored in the SDL Country Risk can access master data and flow data in the SDL that was either supplied by one of the source systems or originally created in the SDL.

Country Risk can also use results data from the upstream General Calculation and Valuation Methods *Account Pooling*, *Facility Distribution*, and *Collateral Distribution*. It accesses this data by means of the Result Database (RDB), which contains the results data from the General Valuation and Calculation Methods.

The attributable amounts calculated in the country risk component are updated in the RDB and reporting can then be carried out for these amounts using the SAP List Viewer (ALV) or BI, or they can be displayed in run administration.



Prerequisites for Implementing the Data Model in the System

All the required characteristics and key figures [page 1331] must be available for Country Risk.

4.7.4.2 Calculations in the Country Risk Component

Use

In the Country Risk component, attributable amounts are calculated for each transaction type and risk category using a particular combination of determination procedure and default risk rule. You can then process the amounts determined in Limit Manager by assigning them as utilization amounts to limits.

Prerequisites

In Customizing, you have made the required settings summarized below:

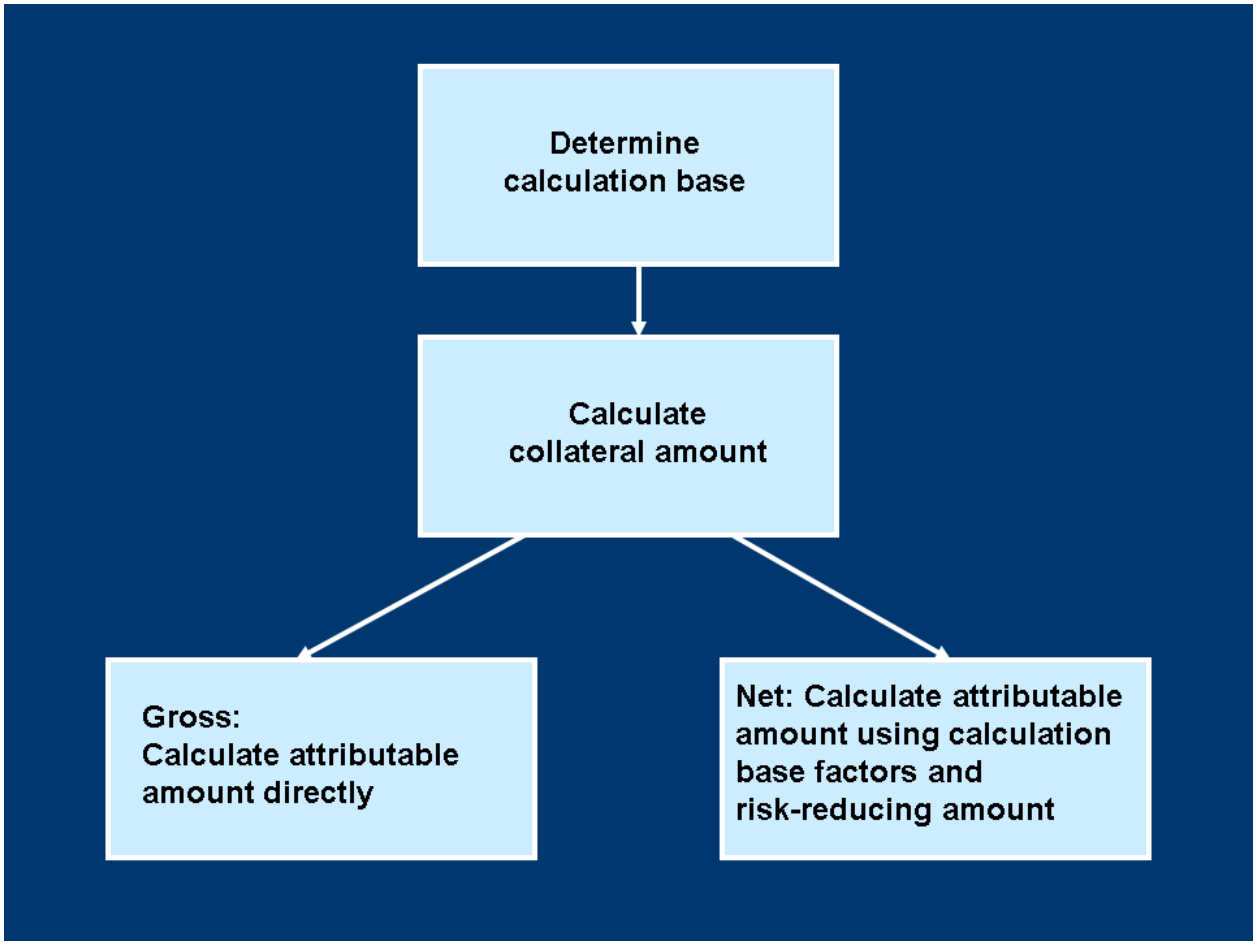
- You have made settings for the derivation that enable the system to use the derivation tool to derive the required characteristics and key figures [page 1331]. For example, the system uses the relevant characteristics to derive the CEQ class from the Source Data Layer (SDL).
- You have made settings for the Module Editor that enable the system to use this tool to create the secondary data source by selecting and enriching data from the primary data source [page 1333].
- You have made the appropriate valuation settings:
  - Edit Valuation Factor Determination
  - Edit Collateral Valuation Rule
  - Edit Determination Procedures
  - Edit Default Risk Rule
  - Edit Risk Category
- You have configured the run and RDB settings so that you can start a country risk run and the system can subsequently update the results in the Result Database (RDB).

For more information about Customizing for country risk, see *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Prototype for Country Risk*.

Process

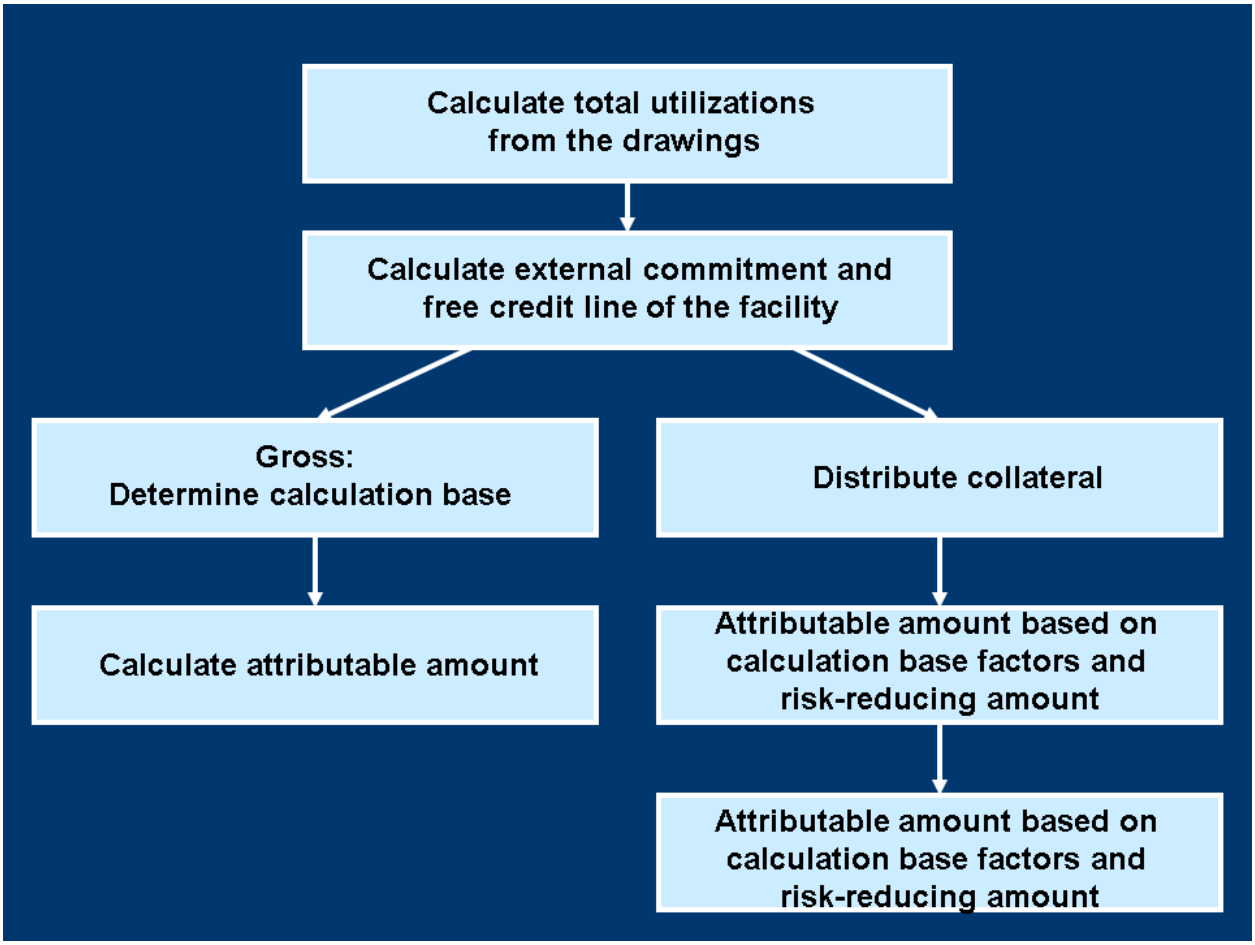
In the calculation process for individual contracts, the system does the following:

1. It determines the calculation base.
2. It calculates the collateral amount.
3. It calculates the attributable amount.
   - Gross: The attributable amount is calculated directly.
   - Net: The attributable amount is calculated using calculation base factors and a risk-reducing amount.



In the calculation process for facilities, the system does the following:

1. It calculates the total utilization resulting from the drawings.
2. It calculates the external commitment of the facility and the free line.
3. It calculates the attributable amount.
   - Gross: The attributable amount is calculated using calculation base factors.
   - After collateral has been distributed: The attributable amount is calculated on the basis of calculation base factors and a risk-reducing amount.



> **Note**
>
> Some of the calculations in the Country Risk component also involve the calculation of collateral amounts. This, however, involves the collateral of facilities, for example, and not the amount-based collateral for multiple transactions that have already been dealt with in the general method *Collateral Distribution*.

4.7.4.2.1 Characteristics and Key Figures in Country Risk

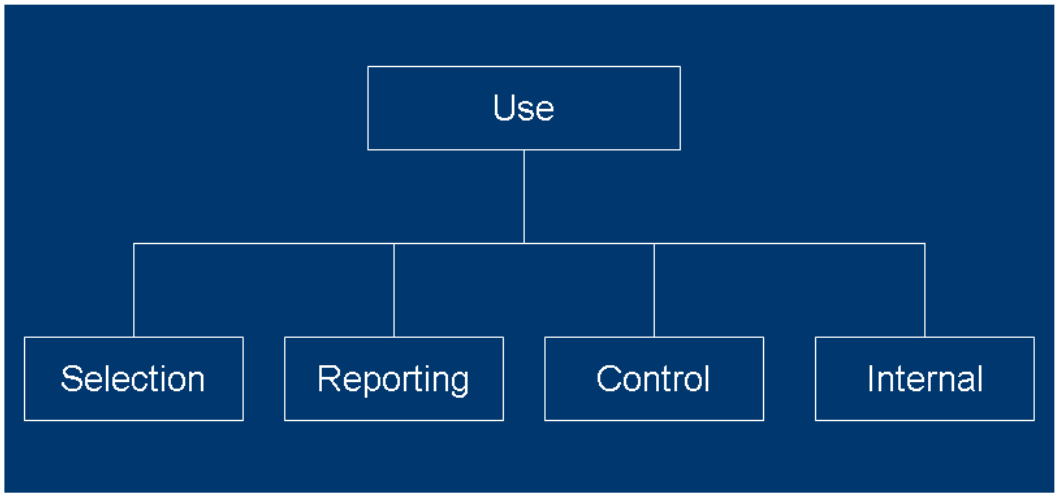
Definition

The main entities used in Country Risk are characteristics and key figures. Key figures are numerical values, such as amounts in specific currencies. Typical examples of key figures in Country Risk are *attributable amounts* and *add-on factors*. Characteristics describe key figures and are used to select these. Typical examples of characteristics in Country Risk are the *product type* and the *relevant country for country risk*.

Use

Credit Risk subdivides key figures and characteristics into those used for selection, reporting, and control. It also classifies key figures and characteristics as internal key figures and characteristics.

- Selection: Selection characteristics are the characteristics that are used to define processing packages (such as the product type).
- Reporting: Key figures and characteristics that are used in reporting or for which reporting can be carried out (such as relevant country for country risk).
- Control: In internal derivation rules in Credit Risk, the input characteristics are used for control purposes. Value ranges are usually defined for these input characteristics. For example, the LEQ class as an input parameter for the LEQ value is a control characteristic in Credit Risk.
- Internal: Derivation rules are used in Credit Risk to determined key figures and characteristics. These output parameters cannot be accessed via the Source Data Layer or the result database (RDB). The LEQ value is one example of a characteristic that is derived in Credit Risk.



Structure

The key figures and characteristics used in Country Risk can be supplied in different ways: They can be supplied as follows:

From the Source Data Layer

From other applications or from the General Calculation and Valuation Methods component

By means of derivation rules You define derivation rules in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Prototype for Country Risk* > *Basic Settings* > *General Settings* > *Derivation*.

> **Note**
>
> In addition to the characteristics and key figures that you can define as required, specific characteristics and key figures that are predefined in the system and that cannot be changed have to be used to enable communication with the core part of the country risk component. For this reason, in Customizing check that the output structure contains the fields beginning with the prefix `/BA1/F62`. For more information see the documentation in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Prototype for Country Risk* > *Basic Settings* > *General Settings* > *Module Editor*.

4.7.4.2.2 Data Selection

Use

The primary data source is used to transfer selected data to Country Risk. The primary data source for the country risk component is usually the Source Data Layer (SDL) or the Result Database (RDB) in which the results of the General Calculation and Valuation Methods are stored. The Module Editor is used to select Country Risk data.

Prerequisites

You have made the following settings in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Prototype for Country Risk* > *Basic Settings* > *General Settings*:

1. You have entered the global settings.
2. You have made the necessary Module Editor settings.
3. You have analyzed the contract relationships for the selection.
4. You have made the selection settings.

For more information, see the documentation for the relevant activities in Customizing.

Features

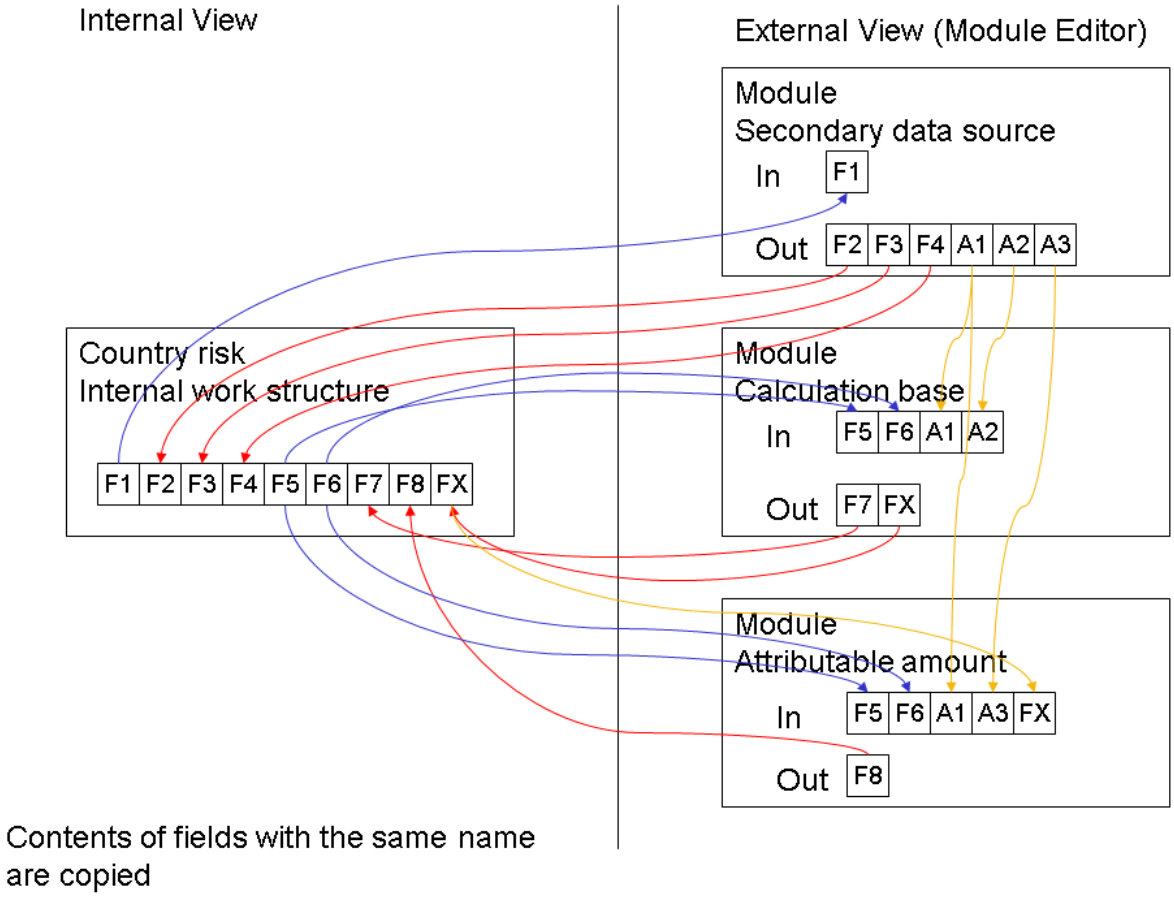
In Country Risk, data is selected for various purposes:

1. To enrich financial transaction data (contracts and collateral) from the SDL
2. To determine the calculation base
3. To determine the attributable amount

The individual steps build on one another. In Customizing for the *Module Editor*, enter the name of each of the modules that is to be called up.

You can organize the structures in the Module Editor (external view) as required. During the enrichment process, you can also add any additional data to the financial transaction data. Internally, however, Country Risk uses a predefined work structure that cannot be changed (internal view).

The following diagram provides an overview of the interaction between the internal and external selection views.



1. Secondary data source module (enrichment of financial transaction data)
   Field F1 has already been filled by the internal view and is transferred to the secondary data source in the module editor (external view) by the primary data source. Fields F2, F3, and F4 are filled in the secondary data source during the enrichment process and are transferred to the internal work structure. Fields A1, A2, and A3 are filled in the secondary data source during the enrichment process and are forwarded directly to either the calculation base module or the attributable amount module.
2. Calculation base module

Fields F5 and F6 are transferred from the internal work structure. Fields A1 and A2 are transferred by the secondary data source module. Fields F7 and FX are filled in the calculation base module and are transferred to the internal work structure.

3. Attributable amount module
   Fields F5 and F6 are transferred from the internal work structure to the attributable amount module; fields A1 and A3 are transferred directly by the secondary data source module. Field F8 is filled in the attributable amount module and is transferred to the internal work structure.

4.7.4.2.3 Figures Used to Calculate Attributable Amounts

Use

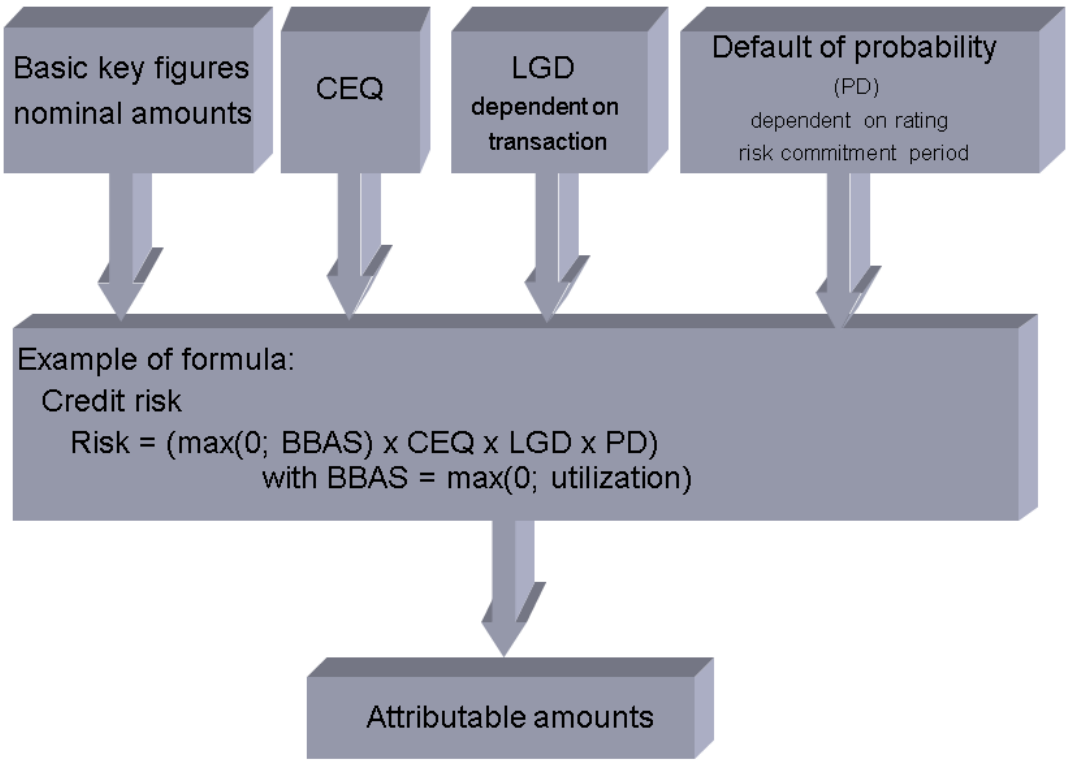
The figures used to calculate attributable amounts include the characteristics, key figures [page 1331], and the market data. In the Country Risk component, market data is mapped as positions in the Source Data Layer.

In Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Prototype for Country Risk* > *Attributable Amount Determination* > *Market Data* you can use the following market data for calculations in Country Risk:

- Default probability
- Loss given default (LGD)
- Loss equivalent factor (LEQ)
- Credit equivalent factor (CEQ)
- Add-on factor (AOF)
- Weighting factors

For more information, see the documentation for the relevant activities in Customizing.

Example



4.7.4.3 Starting Country Risk Runs

Prerequisites

If you also want to take into account the general calculation and valuation methods *Account Pooling*, *Facility Distribution*, and *Collateral Distribution*, you need to have already run these transactions, so that their results are available for further processing. The results of the general calculation and valuation methods are stored in the RDB and are made available to the Country Risk run by means of primary and secondary data sources. *Account Pooling* and *Facility Distribution* provide results at contract level (account); *Collateral Distribution* provides results at contract/collateral relationship level.

So that the results of the country risk run can be written to the RDB, you need to have made the following settings in Customizing for *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Prototype for Country Risk* > *Basic Settings* > *Settings for Runs and RDB*:

- Assign Determination Procedure to Key Figures and Characteristics
- Edit Run Types
- Generate Package Type for RDB Run

Context

In the Country Risk run (CR run), the transactions relevant for Country Risk and that are active on the valuation date are selected. Attributable amounts are determined for these transactions. The results are then stored in the Result Database (RDB).

> **Recommendation**
>
> You can also use the Schedule Manager to start the Country Risk run. To do so, enter the program /BA1/R2_CRE_MAIN in the Schedule Manager. If you want to use Limit Manager, we recommend that you start the limit utilization run using the Schedule Manager immediately after the Country Risk run. For more information, see Schedule Manager [page 437].

During the Country Risk run, contracts, such as loans or trading transactions, are processed. Some of the contracts might have complex relationships with one another that have to be taken into account during data processing and selection. Only those transactions that meet the selection criteria are processed in the Country Risk run.

> **Recommendation**
>
> For performance reasons, you should choose your selection criteria for country risk runs carefully to avoid situations where the data of the entire transaction position has to be retained in the memory.

Procedure

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Country Risk* > *Start Country Risk Run*.

   The system displays the *Country Risk Run* screen.
2. Enter a business record date. The default value is today's date.
3. Enter the run type. The only run type available in Country Risk is `CE01 – CREDIT EXPOSURE STS 01`.
4. Choose *Execute*.

Results

The system notifies you as soon as the job has finished. You can then display the results in run administration. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Country Risk* > *Tools* > *Run Administration* > *Overview of Runs*.

4.7.4.4 Reporting

4.7.4.4.1 Master Data Report for Facilities

Prerequisites

In Customizing for Country Risk you have made the following settings:

- You have created an appropriate module that provides the required characteristics and key figures in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Prototype for Country Risk* > *Basic Settings* > *General Settings* > *Module Editor*.
- You have made the settings required in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Prototype for Country Risk* > *Reporting* > *Configuration for Facilities Reporting*.

For more information, see the relevant sections in the Customizing documentation.

Context

You use the master data report for facilities to display complex contract structures. The structure is displayed as a hierarchy.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Method* > *Credit Risk* > *Prototype Functions* > *Country Risk* > *Reporting* > *Master Data Report for Facilities*.

   The system displays the *Master Data Report for Facilities* screen.

2. In the first data group, enter the following selection criteria:
   - The report that contains the secondary data source that is to be used. You define the report in Customizing for *Bank Analyzer* under *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Prototype for Country Risk* > *Reporting* > *Configuration for Facilities Reporting*.
   - The InfoSet that is to be used to select the data.
3. In the second data group, enter a key date and an exact system time up to which changes are to be considered in the process. This date must not be in the future. The default values are today's date and the current system time.
4. Choose *Execute*.

Results

The system displays the *Master Data Reporting for Facilities REPORT NAME* (name of the report chosen) screen.

At the top of the screen, the facilities are displayed in a hierarchy along with the individual drawings and collateral. Double-click a facility to display its master data. On the screen that is then displayed, you can choose *Cash Flow Generation* to display the associated cash flow.

At the bottom of the screen, the financial transaction ID is displayed in list form. Simply click the financial transaction ID to display the transaction's master data.

4.7.4.5 Tools

4.7.4.5.1 Run Administration

Definition

Run administration includes the following functions:

- Execute or create run
- Display an overview of runs
- Display Application Log
- Edit run
- Manage run
- Replace run
- Select run for archiving
- Delete run
- Log of the deletion function

> **Note**
>
> The above functions are not all available for each application. For more information, see the application-specific documentation.

Use

The following table lists the runs available for each application:

Application	Run
General Calculation and Valuation Methods [page 376]	Account Pooling [page 383] or Account Pooling (CVPM) Facility Distribution [page 399] or Facility Distribution (CVPM) Determination of the Free Line for Facilities [page 395] or Determination of the Free Line (CVPM) Collateral Distribution [page 404] or Collateral Distribution (CVPM) Determination of Default [page 410] Stress Runs [page 429]: • Stress run for account pooling or stress run for account pooling (CVPM) • Stress run for facility distribution or stress run for facility distribution (CVPM) • Stress run for determination of the free line or stress run for determination of the free line (CVPM) • Stress run for collateral distribution or stress run for collateral distribution (CVPM) • Stress run for determination of default
Credit Risk [page 1191]	Credit Exposure Run [page 1238] or Credit Exposure Run (CVPM) Country Risk Run [page 1336] (prototype for country risk) **Stress run:** Credit Exposure Stress Run [page 1261] or Credit Exposure Stress Run (CVPM)

Application	Run
Historical Database [page 1478]	**Version management:** Run for generating historical data for data layers [page 1481] Run for generating historical data for the bank's own models [page 1484] Uploading of Files [page 1486] **Calculation functions:** Determining Default Rates [page 1488] Determining Average Default Rates [page 1491] Determining Default Figures [page 1493] Calculation of Migration Matrices [page 1496] **Data retrieval:** Run for supplying data to models [page 1498] Downloading of Files [page 1500] **Stress tests:** Stress Run for Supplying Models with Data [page 1505] Generating Scenario Data in the Source Data Layer [page 1506]
Generic BI Data Extraction [page 1638]	Testing the BI Extractor [page 1639] BI extraction run **Note** Extraction runs are created and executed in SAP NetWeaver Business Intelligence (BI). The system displays information about extraction runs in run administration of Bank Analyzer.
Regulatory Reporting Interface	Data Extraction Runs
Limit Manager [page 1529]	Limit Utilization Run [page 1551]
Fair Value Effectiveness Test for Hedging Relationships [page 1036]	Effectiveness Test Run
Cash Flow Hedge Analysis [page 1107]	Valuation Run [page 1128] Subvaluation Run Aggregation Runs [page 1129]

Application	Run
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169] Portfolio Item Runs [page 1172] • Period-end run for portfolio items (subledger scenario) • Period-start run for portfolio items (subledger scenario) • Portfolio item run (merge scenario)

> **Note**
>
> For some of the Bank Analyzer components, you can use the Schedule Manager to schedule and control jobs. If you use multiple applications, you can define the sequence in which the runs are to be carried out. For more information, see Schedule Manager [page 437].

More Information

Status Overview for Run Administration [page 1248]

4.7.4.5.1.1 Features of Run Administration in Country Risk

Use

Run administration manages information about runs. This information can include, for example, the date of the run, the version of the run based on a key date, the user, the time and date the user started the run, the name of the user who made the latest changes to the run, and the time and date the user made those changes, the run category, and the current status of the run. You can branch from the overview screen to the detail logs for the results that were written to the Result Database (RDB), and to the application log for the run.

Prerequisites

You have already carried out at least one country risk run.

Features

Run administration in country risk contains the functions described below. You access these functions on the *SAP Easy Access* screen under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Country Risk* > *Tools* > *Run Administration*:

- Overview of Runs: You choose this function to display the runs, and to change the status of a run manually. For example, you can complete a run, flag a run for deletion, or set the status of a run to "completed successfully".
- Deletion Function: You use the deletion function to delete individual runs, or to delete all the runs that you have flagged for deletion. You can set and clear the deletion flag on the *Overview of Runs* screen. Note that if you delete a run you cannot reinstate it. The deletion is final.
- Log of Deletion Function: Provides an overview of the runs that were deleted.

4.7.4.5.2 Schedule Manager

Definition

A tool in Bank Analyzer that you can use to execute and monitor complex business processes, such as period-end closing. When you schedule a run, you always have to specify the program with which the system calls the run. There is a separate program for each run category in an application.

The table below shows for which runs and run categories the Schedule Manager can be used, and their corresponding programs.

Application	Run Category	Program
Account Pooling	Account Pooling [page 383]	/BA1/DE_GM_MAIN_COMP
Account Pooling	Stress Run for Account Pooling [page 429]	/BA1/DE_GM_MAIN_COMP_SCEN
Determination of the Free Line for Facilities	Determination of the Free Line for Facilities [page 395]	/BA1/DE_GM_MAIN_FL
Determination of the Free Line for Facilities	Stress Run for Determination of the Free Line for Facilities [page 429]	/BA1/DE_GM_MAIN_FL_SCEN
Distribution of Facilities	Distribution of Facilities [page 399]	/BA1/DE_GM_MAIN_RLV
Distribution of Facilities	Stress Run for Facility Distribution [page 429]	/BA1/DE_GM_MAIN_RLV_SCEN
Collateral Distribution	Collateral Distribution [page 404]	/BA1/DE_GM_MAIN_SV
Collateral Distribution	Stress Run for Collateral Distribution [page 429]	/BA1/DE_GM_MAIN_SV_SCEN

Application	Run Category	Program
Determination of Default	Determination of Default [page 411]	/BA1/RA_DFLT4280_RUN
Determination of Default	Stress Run for Determination of Default [page 429]	/BA1/RA_DFLT4280_RUN_SCENARIO
Fair Value Effectiveness Test [page 1036]	Effectiveness Test Run [page 1039]	/BA1/R8_FVHE_MAIN_RUN
Fair Value Effectiveness Test	Generation of Time Series for Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MAIN_RUN
Fair Value Effectiveness Test	Modifying a Time Series for the Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MODIFY
Cash Flow Hedge Analysis [page 1107]	Subvaluation Run [page 1128]	/BA1/R5_CFHA_BTC_RUN
Cash Flow Hedge Analysis	Aggregation Run [page 1129]	/BA1/R5_START_BATCH_AGGR
Cash Flow Hedge Analysis	Joint call of Valuation Run [page 1111] and Aggregation Run [page 1123]	/BA1/R5_EXECUTE_CFHA
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169]	/BA1/R9_VPM_BTC_ID_RUN
Portfolio Fair Value Hedge	Portfolio Item Run (Merge Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_RUN
Portfolio Fair Value Hedge	Period-End Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PE_RUN
Portfolio Fair Value Hedge	Period-Start Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PS_RUN
Portfolio Fair Value Hedge	Define End of Valuation Period	/BA1HM/RAH_M_PFVH_SET_VP_END
Portfolio Fair Value Hedge	Adjust Initial Portfolio Fair Value Hedges for Next Valuation Period	/BA1HM/RAH_M_PFVH_ADJUST
Portfolio Fair Value Hedge	(Re-) Designation of Initial Portfolio Fair Value Hedges	/BA1HM/RAH_M_PFVH_DESIG
Credit Exposure [page 1192]	Credit Exposure Run [page 1238]	/BA1/R2_CRA_MAIN_CRE
Credit Exposure	Credit Exposure Stress Run [page 1261]	/BA1/R2_CRA_MAIN_CRE_SCEN
Credit Exposure	Credit Exposure Run for Revolving Retail	/BA1/RR2_CRE_REVRET
Credit Exposure	Approximate Stress Run [page 1274]	/BA1/RR2_CRE_SCENARIO
Credit Exposure	Aggregation Run for Ad Hoc Calculation [page 1276]	/BA1/RR2_SCEN_AGGR

Application	Run Category	Program
Country Risk [page 1326]	Country Risk Run [page 1336]	/BA1/R2_CRE_MAIN
Historical Database [page 1478]	Historization Run for Data Layers [page 1481]	/BA1/R6_HIST_RUN
Historical Database	Historization Run for Bank's Models [page 1484]	/BA1/R6_HIST_RUN_MODEL
Historical Database	Determine Default Rates [page 1488]	/BA1/R6_CF_DEFAULT_4210
Historical Database	Determine Average Default Rates [page 1491]	/BA1/R6_CF_DEFAULT_4220
Historical Database	Determine Default Figures [page 1493]	/BA1/R6_CF_DEFAULT_KEYFIGURES
Historical Database	Exporting Data to In-House Models [page 1498]	/BA1/R6_MOD_RUN
Historical Database	Calculation of Migration Matrices [page 1496]	/BA1/R6_CF_MIGRATIONS
Historical Database	Stress Run for Supplying Models with Data [page 1505]	/BA1/R6_MOD_RUN_SCENARIO
Historical Database	Stress Run for Supplying Data [page 1506]	/BA1/R6_FDB_RUN_SCENARIO
Regulatory Reporting Interface	Processing Framework Run Copying the Start ID	/BA1/S1_COPY_STARTID
Regulatory Reporting Interface	Processing Framework Run Creating Runs	/BA1/S1_CREATE_RUN
Regulatory Reporting Interface	Processing Framework Run Release Run	/BA1/S1_RELEASE_RUN
Regulatory Reporting Interface	Processing Framework Run Creating a Worklist	/BA1/S1_CREATE_WRKL
Regulatory Reporting Interface	Processing Framework Run Processing a Run	/BA1/S1_PROCESS
Regulatory Reporting Interface	Data Extraction Runs	/BA1/S3_EXTRACT_RUN
Limit Manager [page 1529]	Limit Utilization Run [page 1551]	/BA1/R3_UTILIZATION_RUN_DPF

More Information

Schedule Manager

4.7.4.5.3 Postprocessing

Prerequisites

The system already contains at least one country risk run with a single transaction you want to change subsequently or update to this run.

Context

You can use the postprocessing function to update manually corrected erroneous transactions that were not analyzed and hence not updated in the country risk run. You can also use this function to update transactions that were analyzed correctly in the country risk run and that are to be changed.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Processes and Method* > *Credit Risk* > *Prototype Functions* > *Country Risk* > *Tools* > *Postprocessing*.

   The *Country Risk: Postprocessing of Single Transactions* screen appears.
2. In the *Sel. Criteria of Run* data group, enter the relevant selection criteria to identify the country risk run.
3. In the section below this, enter the object ID in the *External Number* field, and enter the unique ID for the single transaction that is to be updated for the run selected above during postprocessing.
4. Enter a date up to which changes are to be considered. This date must not be in the future. The default values are today's date and the current system time.
5. Choose *Execute*.

Results

The system displays a postprocessing log. You can display the results records that have changed as a result of postprocessing in run administration [page 1342].

4.7.4.5.4 Archiving of Country Risk Data

Over time, the volume of data in your productive system increases. So that the evaluations that you run in the system are not slowed down by too large a volume of data, we recommend that you use the archiving tool. You can use the archiving tool to delete obsolete country risk data from the system.

Archiving removes data from the system and stores it elsewhere. The archiving functions include an information system that you can use to display the archived data. Displaying this archived data can seriously impair system performance, however.

For more information about archiving, see the documentation about .

4.7.4.6 Interaction Between Country Risk and Limit Manager

Use

In Bank Analyzer, you can use the Country Risk component in conjunction with Limit Manager. This enables you to assign the attributable amounts calculated in Country Risk as limit utilization amounts to limits in Limit Manager. Limit Manager can also access the characteristics derived in Country Risk and use these as limit characteristics.

You can use Schedule Manager [page 437] to ensure that the limit utilization run is started immediately after the Country Risk run (that is, the attributable amount determination).

Prerequisites

In Customizing, you have made the necessary settings to enable Limit Manager to process data from Country Risk. The key Customizing settings that enable Country Risk and Limit Manager to communicate are given below.

Bank Analyzer > *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Prototype for Country Risk* > *Basic Settings* > *Valuation Settings* > *Edit Determination Procedures*

Bank Analyzer > *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Prototype for Country Risk* > *Basic Settings* > *Settings for Runs and RDB*

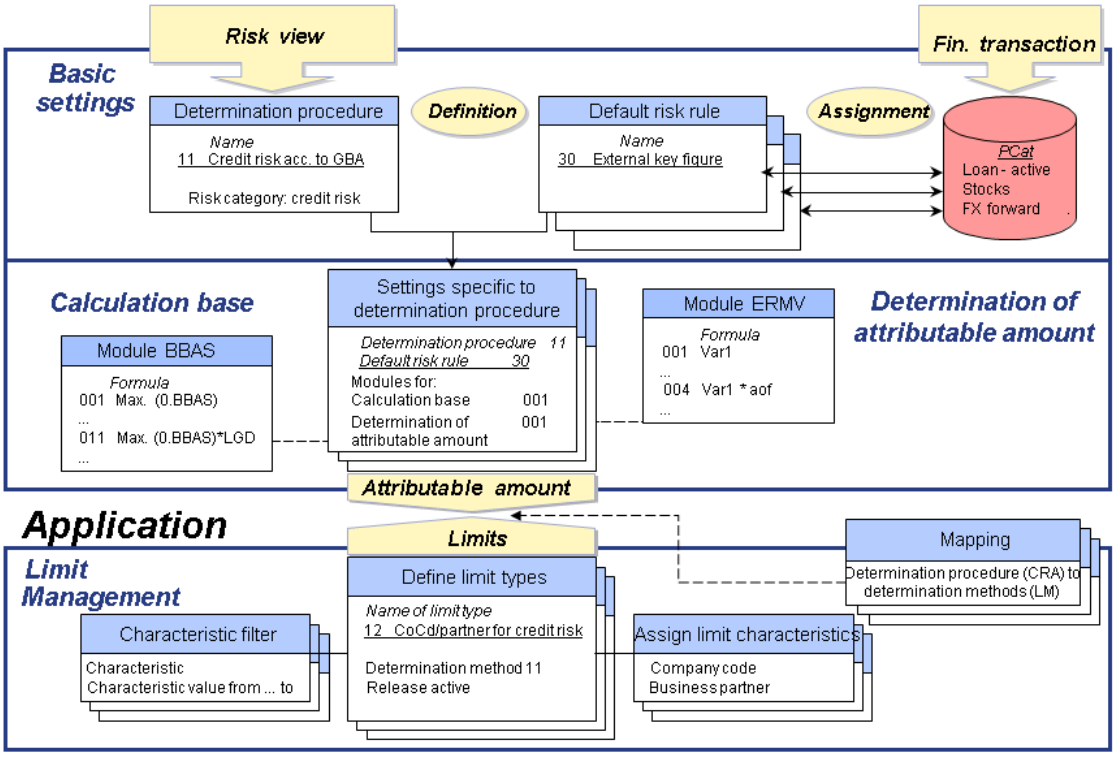
By making these settings, you define where and how the results of the Country Risk run (the attributable amounts) are stored in the Result Database (RDB).

Bank Analyzer > *Analytics* > *Limit Manager* > *Characteristics in Limit Manager* > *Edit Characteristics of the Limit Manager*

In this Customizing activity you can assign the characteristics derived in Country Risk to Limit Manager.

Bank Analyzer > *Analytics* > *Limit Manager* > *Limit Levels in Limit Manager* > *Edit Determination Methods*

You create determination methods and assign the determination method for Country Risk to them and to the limit area.



Overview of Customizing Settings

4.8 Calculation Base Data

You can use calculation base data to save information relating to the financial position object and to key figures. You need this information for target values in calculation procedures.

The calculation base data is stored in the Result Data Layer (RDL) and corresponds to the target values in present value calculations and on the Fair Value Server.

4.8.1 Storage of Calculation Base Data in RDL

Calculation base data is stored in the Result Data Layer (RDL). You can use the result category `HFBAS` (calculation base data of financial position) and the following result node categories for this purpose:

- `POSB1` for one-dimensional calculation base data dependent on positions
  This calculation base data is intended for fair value and hedge target values.
- `POSB2` for two-dimensional calculation base data dependent on positions
  This calculation base data is intended for accrual/deferral target values within the valuation of amortized costs.

- KYFB1 for one-dimensional calculation base data dependent on (positions and) key figures
- KYFB2 for two-dimensional calculation base data dependent on (positions and) key figures
  The calculation base data in KYFB1 and KYFB2 is intended for the accruals and deferrals of the remainder from fair value and hedge valuations.

In the Customizing activity *Edit Data Structures in Results Data Area*, create your own result type with the result category HFBAS for each of the above result node categories. Assign the relevant result node category in the Customizing activity at *Result Node Types*. You do this in Customizing for *Bank Analyzer* > *Results Data Layer* > *Basic Settings*.

4.8.2 Fields in the Calculation Base Data

Calculation base data consists of characteristics and key figures. We provide numerous characteristics and key figures in the individual result node types, which are then used in the calculation methods supplied. While the key figures are all provided for usable load, some of the characteristics are used as administrative information (for example, key fields for access or redundant information from the financial position object).

The following table lists the characteristics and key figures delivered by SAP. The first column contains the technical key and short text of the characteristic or key figure. The second column contains a description. Columns 3 to 6 indicate whether the result node category (POSB1, POSB2, KYFB1, or KYFB2) contains the characteristic or key figure.

Characteristics

Characteristic	Description	POSB1	POSB2	KYFB1	KYFB2
Administrative Information					
/BA1/C55FP_ID (Position)	The system uses the ID of the financial position object as the key in **all** calculation base data. If the calculation base data relates to the key figures then it also uses the posting key figure.	X	X	X	X

Characteristic	Description	POSB1	POSB2	KYFB1	KYFB2
/BA1/C55ACCKF (Posting Key Figure)	If the calculation base data relates to key figures, the system uses the posting key figure together with the ID of the financial position object as the key.			X	X
/BA1/C40FTRAN (Financial Transaction ID)	The system saves this characteristic of the financial position object redundantly in the calculation base data. Since this characteristic is derived from the key, it cannot be changed.	X	X	X	X
/BA1/C41FINST (Financial Instrument ID)	See the description for /BA1/C40FTRAN.	X	X	X	X
/BA1/C43CLACC (Securities Position Account ID)	See the description for /BA1/C40FTRAN.	X	X	X	X
/BA1/C4AAGO (Aggregation Object ID)	See the description for /BA1/C40FTRAN.	X	X	X	X
General Characteristics					

Characteristic	Description	POSB1	POSB2	KYFB1	KYFB2
/BA1/C55POSTD (Posting Date)	The system uses this characteristic in two-dimensional calculation base data to save the posting date. You can find the value date in the RDL-internal characteristic /BA1/CR0KEYDAT (key date). If the calculation base data is one-dimensional, the system saves the posting date in the RDL-internal characteristic /BA1/CR0KEYDAT (key date). The value date is not used here because of the one-dimensional nature of the data.		X		X
One-Dimensional Calculation Base Data Dependent on Financial Positions (for Fair Value)					
/BA1/C55FVCALC (FV Calculation Date)	On this calculation date the system calculates the full fair value stored in this calculation base data record.	X			
/BA1/C55TSFV (Time Stamp for Last Valuation Document)	On this system date the system calculates the fair value stored in this calculation base data record.	X			

Characteristic	Description	POSB1	POSB2	KYFB1	KYFB2
/BA1/C55FINEWF (Control Indicator for FI Spread Redetermination)	The system sets this indicator in Accounting if, for example, business transactions of the type *Disbursement* or *Position Change* are supplied. It then analyzes the indicator on the Fair Value Server when it calculates the fair value, provided this has been activated in the Customizing settings for the calculation rule.	X			
/BA1/C55MDS_FI (FI Spread Market Data Set)	Key for market data Customizing (for the price calculator during fair value calculation).	X			
/BA1/C55MDS_FU (Uncovered Market Data Set for FI Spread)	Key for market data Customizing (for the price calculator during fair value calculation).	X			
One-Dimensional Calculation Base Data Dependent on Financial Positions (for Hedges)					
/BA1/C55HACALC (Hedge Adjustment Calculation Date)	On this calculation date the system calculates the hedge fair value stored in this calculation base data record or the hedged amortized costs.	X			

Characteristic	Description	POSB1	POSB2	KYFB1	KYFB2
/BA1/C55TSHEDG (Time Stamp of Last Hedge Adjustment Document)	On this system date the system calculates the hedge fair value stored in this calculation base data record or the hedged amortized costs.	X			
/BA1/C55HIRRNEWF (Control Indicator: Redetermination Hedge Effective Interest Rate)	This indicator controls the recalculation of the hedge effective interest rate (from the Accounting side). See the description for /BA1/C55ACNEWF.	X			
/BA1/C55HSPNEWF (Control Indicator: New Hedge Spread)	The system sets this indicator in Accounting if, for example, business transactions of the type *Disbursement* or *Position Change* are supplied. It then analyzes the indicator on the Fair Value Server when it recalculates the hedge fair value.	X			
/BA1/C55MDS_HS (Market Data Set for Hedge Spread)	Key for market data Customizing (for the price calculator during hedge fair value calculation).	X			
One-Dimensional Calculation Base Data Dependent on Financial Positions (for Valuation Remnants)					

Characteristic	Description	POSB1	POSB2	KYFB1	KYFB2
/BA1/C55TSACVR (Time Stamp: Last Amortized Cost Calculation incl. Valuation Remnants)	On this system date, the system calculates the amortized cost incl. valuation remnants stored in this calculation base data record (time stamp for accessing SDL/CF).	X			
/BA1/C55ACVRCA (Date of Last Amortized Cost Calculation incl. Valuation Remnants)	On this posting date, the system calculates the amortized cost incl. valuation remnants stored in this calculation base data record.	X			
/BA1/C55VRAG8F (AG8 Control Indicator: Recalculation of Effective Interest Rate for Amortized Cost incl. Valuation Remnants)	If AG8 has been entered as the valuation rule in the Customizing settings for calculating the present value, the system freezes the effective interest rate. You can use this control indicator to override this Customizing setting so that the effective interest rate is used instead of AG8.	X			

Characteristic	Description	POSB1	POSB2	KYFB1	KYFB2
/BA1/C55VREYNF (Control Indicator: Redetermination of Effective Interest Rate for Amortized Cost incl. Valuation Remnants)	This indicator controls the recalculation of the effective interest rate (from the Accounting side). The system sets the indicator for new inflows to valuation remnants, for example as a result of a reclassification or the dissolution of a hedging relationship, or if business transactions of the type *Disbursement* or *Position Change* are imported. You can override the indicator in the Customizing settings for the valuation rules (AG8). The system also recalculates the present value if the indicator has not been set in Accounting but there is a new interest rate because the variable interest has been fixed.	X			
/BA1/C55VREYPD (Posting Date Effective Interest Rate for Amortized Cost incl. Valuation Remnants)	Posting date on which the effective interest rate was last calculated	X			
One-Dimensional Calculation Base Data Dependent on Financial Positions (for Impairment)					

Characteristic	Description	POSB1	POSB2	KYFB1	KYFB2
/BA1/C55UNCALC (Last Unwinding Calculation Date)	Date on which the unwinding was last calculated	X			
Two-Dimensional Calculation Base Data					
/BA1/C55AC_PD (AC Posting Date)	On this calculation date the system calculates the amortized cost stored in this calculation base data record. The system uses the fields /BA1/C55AC_PD and /BA1/C55ACCALC to calculate the accrued or deferred discounts based on the straight-line method and on the effective interest rate.		X		
/BA1/C55ACCALC (AC Value Date)	On this value date date the system calculates the amortized cost stored in this calculation base data record. The system uses the fields /BA1/C55AC_PD and /BA1/C55ACCALC to calculate the accrued or deferred discounts based on the straight-line method and on the effective interest rate.		X		

Characteristic	Description	POSB1	POSB2	KYFB1	KYFB2
/BA1/C55IRRPD (IRR (Internal Rate of Return) Posting Date)	Posting date on which the effective interest rate was last calculated		X		
/BA1/C55ACNEWF (Control Indicator: Redetermination of Effective Interest Rate)	This indicator controls the recalculation of the effective interest rate (from an accounting system perspective). The system sets this indicator if business transactions of the type *Loan Disbursement*, *Unscheduled Repayment*, *Rollover* or *Condition Change* are supplied. You can override the indicator in the Customizing settings for the valuation rules (AG8). The system also recalculates the present value if the indicator has not been set in Accounting but there is a new interest rate because the variable interest has been fixed.		X		

Characteristic	Description	POSB1	POSB2	KYFB1	KYFB2
/BA1/C55ACAG8F (AG8 Control Indicator: Recalculation of Effective Interest Rate)	If **AG8** has been entered as the valuation rule in the Customizing settings for calculating the present value, the system freezes the effective interest rate. You can use this control indicator to override this Customizing setting so that the effective interest rate is used instead of **AG8**.		X		
/BA1/C55TSAC (Time Stamp of the Last Amortization)	On this system date the system calculates the amortized cost stored in this calculation base data record (time stamp for accessing SDL/CF)		X		
/BA1/C55ACBODF (Amortized Cost at Start of Day/End of Day)	The setting made for the indicator specifies whether the amortized cost value (AC, key figure &55ACVAL) contains the accrued interest (AI) and/or the effective capital (EC) of the value date.		X		
One- and Two-Dimensional Calculation Base Data Dependent on Key Figures					

Characteristic	Description	POSB1	POSB2	KYFB1	KYFB2
/BA1/C55DFIACC (First Day in Accrual Period)	The date field /BA1/C55DFIACC describes the start of the accrual/deferral period and remains the same throughout the period regardless of how often you calculate accruals or deferrals during this period. The system uses this field for straight-line deferrals only. This is always the case when you want to calculate deferrals for the processing categories *BF_DEF. The system usually accesses the field /BA1/C55DFIACC during the first accrual/deferral run only. During further runs, the system uses the *Date of the Last Accrual/Deferral Run* (/BA1/C55DLDEFC). The calculation of accruals or deferrals for periods with a fixed start date is an exception here.			X	X

Characteristic	Description	POSB1	POSB2	KYFB1	KYFB2
/BA1/C55FDDEFP (First Day of Deferral Period)	This date field contains the start date of the next accrual/deferral period.			X	X
/BA1/C55DLDEFC (Date of Last Deferral Calculation)	The system uses this date field to restart the calculation of accruals or deferrals at the most recently calculated value.			X	X

Key figures

Key Figure	Description	POSB1	POSB2	KYFB1	KYFB2
One-Dimensional Calculation Base Data Dependent on Financial Positions (for Non-Hedges)					
&55FVVAL (Fair Value at End of Calculation Date)	Value of the fair value at the end of the day (on posting date /BA1/C55FVCALC).	X			
&55FISPR (Financial Instrument Spread)	You can multiply this value by 100 and use it as a percentage.	X			
&55ACVR (Amortized Cost plus Valuation Remnants)	Value of amortized cost including valuation remnants based on the posting date /BA1/C55ACVRCA	X			

Key Figure	Description	POSB1	POSB2	KYFB1	KYFB2
&55EYACV (Effective Interest for Amortized Cost plus Valuation Remnants)	You can multiply this value by 100 and use it as a percentage. A current effective interest rate is a prerequisite for calculating the amortized cost including valuation remnants.	X			
One-Dimensional Calculation Base Data Dependent on Financial Positions (for Hedges)					
&55HAC (Hedged Amortized Costs)	Value of the hedged amortized costs at the end of the day in relation to the date /BA1/C55HACALC.	X			
&55HFVAL (Hedge Fair Value)	Value of the hedge fair value at the end of the day in relation to the date /BA1/C55HACALC.	X			
&55HNOM (Hedged Nominal Amount)	Specifies the nominal amount of a transaction that is used to assign a transaction to a hedging relationship.	X			
&55HQUAN (Hedged Quantity)	Specifies the quantity of a transaction that is used to assign a transaction to a hedging relationship.	X			

Key Figure	Description	POSB1	POSB2	KYFB1	KYFB2
&55IRRHA (Effective Interest Rate for Hedged Amortized Cost)	You can multiply this value by 100 and use it as a percentage. A current hedge effective interest rate is a prerequisite for calculating the hedged amortized cost.	X			
&55HSPR (Hedge Spread)	You can multiply this value by 100 and use it as a percentage.	X			
Two-Dimensional Calculation Base Data Dependent on Financial Positions					
&55ACVAL (Amortized Cost)	Value of amortized cost based on the value date /BA1/C55ACCALC		X		
&55IRRAC (Effective Interest Rate for Amortized Cost)	You can multiply this value by 100 and use it as a percentage. A current effective interest rate is a prerequisite for calculating the amortized costs.		X		
One- and Two-Dimensional Calculation Base Data Dependent on Key Figures					
&55AMDEF (Deferral Amount)	Value of deferral based on the value date /BA1/C55FDDEFP				X

4.8.3 Use of Calculation Base Data in CM Methods

The methods in Calculation Management (CM) use ABAP Dictionary structures to write or read calculation base data (CBD). Each of these structures contains a selection of CBD fields that is meaningful from a business perspective. Some of the structures also contain non-CBD fields. This means that the CM result structure is not

overfilled with fields that are rarely used (for deferred taxes, for example). The structures do not determine which of the four CBD stores are referred to during an actual query. The following logic is implemented:

- The transferred generic structure must contain a posting date characteristic. If it also contains a value date characteristic and this is not empty, the system interprets the data record as two-dimensional. If it does not contain a value date characteristic, the system interprets the data record as one-dimensional.
- If the structure contains the posting key figure characteristic and this is not empty, the system copies the data record via "move-corresponding" to the calculation base data for the financial position and the key figures. If the structure does not contain the posting key figure characteristic, the system copies the data record to the calculation base data for the financial position only.
- If you are using your own structures, you must use the characteristics provided by SAP in the structures at least for the posting date, value date, financial position ID, and posting key figure (if applicable).
- In the case of financial position balances, you can ignore any initial fields in the transferred structure. However, we do not recommend that you do this. Instead:
  1. Query the calculation base data.
  2. Overwrite this data.
  3. Return the overwritten calculation base data with the activated "Overwrite" indicator to the financial position balances.

> **Note**
>
> Note the following: Only Calculation Management writes and reads calculation base data. However, this data is usually used to calculate the present value. This information is received or returned in the form of target values.

4.8.4 Display of Calculation Base Data in RDL Viewer

You can display the calculation base data (CBD) in the RDL viewer.

First select the results data area and the result view. Then enter an upper limit for the key date of the two-dimensional versioning. The semantics of this key date are different for one- and two-dimensional calculation base data:

- In one-dimensional calculation base data, the key date is the posting date. This means that the RDL viewer provides the current version of the calculation base data for the selection posting date.
- In two-dimensional calculation base data, the key date is the value date.
  You can enter the posting date in the form of characteristic `/BA1/C55POSTD` using "dynamic selections". The system then displays the version that is current for the value date on the specified key date - together with a history (restricted if necessary) for the posting date.

  > **Example**
  >
  > There are two versions of the calculation base data: one has the posting date January 1 and the value date January 15, the other has the posting date January 1 and the value date January 16.
  >
  > - To view the first version you must select the key date January 15.
  > - To view the second version you must select the key date January 16.
  >
  > You cannot display both versions at once.

5 Results Data Layer (FS-BA-RD)

Use

You can use this component to store, display, and edit results data. This results data is based on accounting-related or risk-related analyses of financial transactions or financial instruments in Bank Analyzer (Basel II, IAS Financial Reporting), or on analyses using other analysis tools. Results data is stored in the Results Data Layer (RDL) in results data areas in the form of result types.

The RDL is part of the Integrated Finance and Risk Architecture (IFRA). By means of common dimensions (for example, financial transaction ID, financial instrument ID, or legal entities) that are shared by results within a results data area, the RDL provides a basis for the integration of results data. It stores data in an infrastructure that is semantically and technically standardized, which enables standardized usage for existing and future applications that are integrated in the system.

The RDL provides the following functions:

- Storage of results in results data areas [page 1366]
- Aggregation [page 1374]
- Versioning [page 1378]
- Archiving and Destroying Results Data Using the Archiving Engine [page 1395]
- External Interfaces [page 1398]
- User Interfaces [page 1406]

Example

You have started a process that generates documents for financial transactions and financial instruments. These documents are stored in the RDL. The financial reporting process reads the data from the RDL and generates results that are used for the profit and loss statement or for the annual financial statement.

5.1 Results Metadata

Definition

Results metadata form the area within the Results Data Layer (RDL) where you can define the structure of results, and make Customizing settings for the RDL (for example, aggregation and archiving).

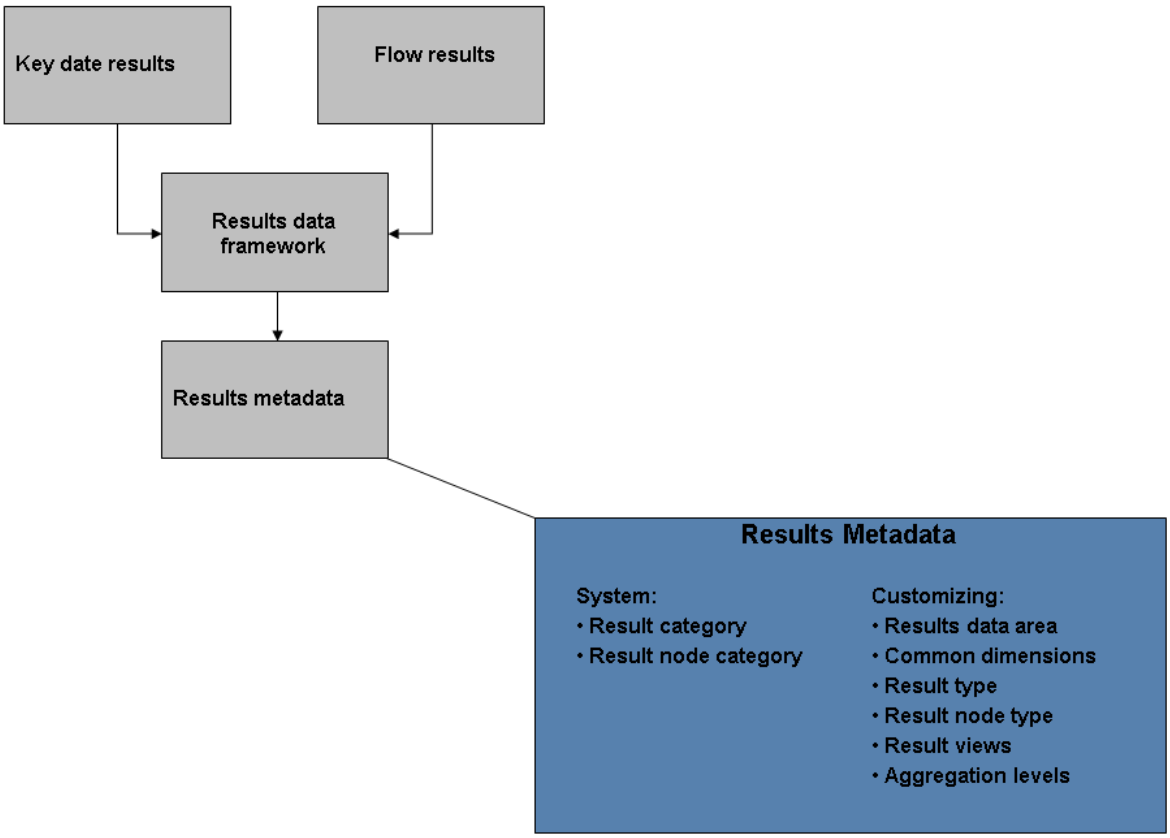
Use

Results metadata form the Customizing framework for:

- Defining results data areas [page 1366]
- Consistent reporting and consistent processing of results data by the definition of common dimensions [page 1367] of a results data area.
- Managing result types [page 1367] and the characteristics and key figures that are assigned to them.
- Defining result views [page 1373]
- Defining aggregation levels [page 1374] for storing preaggregated data.

The following settings are shipped with the system for results metadata:

- Result category [page 1367]
- Result node category [page 1370]



5.1.1 Results Data Area

Definition

The results data area is the top structuring element for all results stored in the Results Data Layer (RDL). It is a logical data pool for semantically integrated results data with common dimensions [page 1367]. The results data is provided by different information production processes for the information requestors. When you create internal balance sheets, you can define dimensions of financial statement entities for each results data area. These dimensions of financial statement entities are used to meet the requirements for credit-debit parity.

The technical separation of information production and information requests means that you can provide reusable results for creating specific analysis procedures for banks using a common information basis.

Use

Several results data areas can exist simultaneously, but only one of these can act as the central results data area.

> **Note**
>
> There is no Customizing setting for determining a central results data area. Instead, the central results data area is per definition agreed upon by those who access it. In this respect, the results data there is central.

Apart from the central results data area (if this is defined), all other results data areas are mainly used to store interim results, or results that are generated for a special information request from a specific analysis process.

For more information, see Displaying Results Data [page 1407].

Structure

A results data area is structured by result types [page 1367] that act as independent data stores for results data. All results within the RDL are stored in these result types.

The following components can also be configured in a results data area:

- Result Node Types [page 1370]
- Result Views [page 1373] and Filters [page 1373]
- Aggregation Levels [page 1374]

You can create results data areas in Customizing for Bank Analyzer under *Results Data Layer* > *Basic Settings* > *Edit Results Data Area*.

5.1.1.1 Common Dimensions

Definition

In the Results Data Layer (RDL), the term common dimensions describes a set of characteristics that is shared by all results of a results data area [page 1366], and that has to be delivered by all processes that generate information.

Common dimensions are defined per results data area. This ensures that all information requestors can request results on the level of the common characteristics. In this sense the definition of common dimensions can be seen as a contract between information production and information requestors.

Use

Common dimensions (for example, financial transaction ID, financial instrument ID, or legal entities) define the minimal granularity of results to be provided by any process within information production.

You can assign common dimensions to the results data area in Customizing for Bank Analyzer under *Results Data Layer* > *Basic Settings* > *Edit Results Data Area*.

5.1.2 Result Types and Result Categories

The list below contains the `Bank Analyzer` result categories. Also see . (Result categories that start with `HI` are only relevant for insurance companies and for Solvency II).

Definition

Result types are used to store results data in results data areas [page 1366]. All results within Results Data Layer (RDL) [page 1364] are stored in result types.

The result type defines the semantics and the actual technical properties of the data, such as the versioning schema [page 1378] and the persistence schema [page 1393], as well as the hierarchical structure of the results that it contains. A result type is assigned to exactly one result category, and takes on its properties. Result categories determine the system properties and some of the semantics of results (for example, versioning schema and persistence schema).

Integration

You create result types with the result category in Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*. You cannot add additional result categories or change the existing result categories. RDL in Bank Analyzer includes the following result categories:

Smart AFI

Smart AFI Result Categories

For an overview of the Smart AFI result categories, see . Note that the creation of aggregation levels, preaggregation, and the use of RDL data sources is not supported for Smart AFI result categories.

All Applications

- Posting document (HFSPD)[1]
  Result types that are based on the *Posting Document* result category can be configured as subledger documents or general ledger documents.
  - A subledger document has three levels and comprises the result node categories SPDHD (subledger document header), SPDIT (subledger document item) and SPDGP (subledger document GAAP data). In central GAAP documents, the posting values for all GAAP are the same. On the third document level, there are items for characteristics that can differ for the different GAAP. The number of items is the same as the number of GAAP. In delta GAAP documents, the posting values are GAAP-specific. Only one accounting system is permitted for each posting; the third document level therefore has only one item. For more information about GAAP data, see .
  - A general ledger document has two levels and comprises the result node category GPDHD (general ledger document header) and GPDIT (general ledger document item).

  This result category has an optimized database table model. The implementation of the assigned persistence schema *1:Result Category-Specific* supports only one database table. SGS, header and version tables are not supported.
- Holding category (HKHCT)[1]
  Result types that are based on this result category are used to store the holding category. The result category has an optimized database table model and uses the persistence schema *'02' Flat Table with Direct Primary Key*. Archiving, data sources, views and aggregation are not supported.

  > **Note**
  >
  > This result category is contained in the result category *Smart AFI Analytical Status (HKAAS)*. You can continue to use this. However, if you want to create new result types we recommend you use HKAAS.

- Flow results (HF)
  Result types that are based on the result category Flow Result [page 717] are used to store accounting documents that are generated in Bank Analyzer under *Processes and Methods*. They have a three-level structure comprising a header and a freely-definable number of flows, which is comparable to accounting documents. Each of them contains two components comparable to document line items. These components contain the sender and the receiver information for the posting. The layout of a flow with the relevant amount of information assigned to it and exactly two components ensures that debit/credit parity is maintained.
- Imported subledger documents (HFISD)
  Result types that are based on the *Imported Subledger Documents* result category are used to store posting information that is provided by an external system. This means that accounting data that is calculated internally and accounting data that is imported can be mapped in a standard way in the Analytics Layer and

especially in the General Ledger Connector. These results on two levels represent the standard structure used for accounting documents that comprises a header and at least two document line items assigned to it.

- Balance carryforwards (HFBCF)
  Result types that are based on the result category *Balance Carryforwards* are used to save balance carryforwards. They have a two-tiered structure with header and item.
  For more information, see Balance Carryforward [page 947].
- Main run and ABS preliminary run (HKCRE)
  Result types that are based on the result category *Main Run and ABS Preliminary Run* are used to store risk data that is determined in the Basel II calculation methods. They have a three-level structure. The exposure is on level 0, and is the root node of the structure, and there are any number of collateral items as leaf nodes.
- Expected exposure (HKEET)
  Result types that are based on the result category *Expected Exposure* are used in the Basel II calculation methods. This results data stores the necessary input data for the counterparty credit risk method for OTC derivatives and for securities financing transactions. These result types have a two-level structure containing the contract ID and the maximum maturity on level 0, and a series of expected exposures and data on level 1.
- Generic key date results (HKFLG / HWFLG / HKFLH)
  Result types that are based on the result category *Generic Key Date Results* do not have a hierarchical structure, and there are no requirements concerning which characteristics and key figures they have to contain. These result types can be used to store results generated during processes in the Bank Analyzer system (accounting, Basel II), as well as results generated by customer processes and imported into the Bank Analyzer system (see RFC interfaces [page 1398]). The only difference between HKFLG and HWFLG is that HWFLG results can be reversed.
  As opposed to HKFLG, HKFLH is used only to save results generated by HANA-based analytical processes and can only be used by SAP-internal processes and applications.
- Combined preliminary run (HKPRE)
  The same Customizing settings as for expected exposures can be used for the preliminary runs for retail exposures, calculation of the risk weight, and purchased receivables. Compared to the main run, simplified Customizing can be used that only calculates up to the EAD (level 1). The preliminary run requires some Customizing modifications regarding cyclical requirements in Basel II. The general structure is the same as for the category main run and ABS preliminary run, although only the first two levels are needed.
- Risk positions (HKRPC)
  Result types that are based on the result category *Risk Positions* are used in the Basel II calculation process. In the standardized method for the counterparty credit risk for OTC derivatives, risk positions can be delivered by the customer for exposures and collateral.
- Calculation basis for financial position (HFBAS)
  Result types that are based on the result category *Calculation Basis for the Financial Position* contain additional information about the calculation of financial positions. Accounting Processes save this data based on key dates along with a technical time stamp. Flow results also have some of these fields and represent the value of a field at the time the record was created. The system updates the field values using calculation steps in calculation management.
- Valuation results for Accounting (HKAVR)
  Result types that are based on result category *Valuation Result for Accounting* are used to store valuation results such as fair value from the initial recognition, amortized cost, and hedge adjustments. The system stores these result types in a header and node structure and receives the data via Services.
- Cash flow results (HKCFD/HKCFR)[1]
  Result types that are based on the result category *Cash Flow Results* are used to save cash flows. These can be contractual cash flows imported from an operational system, cash flows calculated in Bank Analyzer or cash flows calculated in external analytical systems. The result node categories for HKCFD and HKCFR are different.
- Market value (HKMVL)

Results used to store market values in the RDL. The results can be stored for single transactions or for portfolios. You can also store simulated market values for various business and hedging strategies. This result category can be imported using a service, or calculated internally in the fair value production process. The fair value is a type of market value, and is used in Accounting scenarios.

- Global fair value (HKGFV)
  Fair value results come from the determination of the fair value of an item for a financial instrument. The fair value is the amount for which the asset in question can be exchanged. or a liability settled. You can enter it as a dirty price or a clean price. The difference between these two values is the accrued interest.
- Preliminary run for the look-through approach (HKLTA)
  Result types that are based on the *Preliminary Run for Look-Through Approach* result category are used to store results data that is calculated in a special preliminary run for mutual funds. These mutual funds are subject to look-through treatment. The results data has a three-level structure. Level 0, the root node, represents the exposure.
- PA net present values (HKNPV)
  Result types that are based on the *PA Net Present Values* result category are used to save other results from the present value approach in addition to the funding information. This depends on the application event category that the system processes using the Determination of Net Present Values and Calculation Bases component (EICC).
- Accrual results (HKACR)
  Accrual results come from the calculation of the accrual amount for an item in a financial transaction. Accrual amounts are determined from the calculation of accruals and are recognized as income and expense.
  An accrual is a partial recognition of future income or expense.

[1] For result types for this result category, the following functions are not supported:

- Creation of aggregation levels
- Preaggregation
- Use of RDL data sources

5.1.2.1 Result Node Type and Result Node Category

Definition

Result node types form the substructure of a result type [page 1367]. A result node type refers to exactly one result node category that in turn belongs to the result category of its own result type. Result node types consist of a set of characteristics and key figures.

Use

A number of result node categories are provided along with the result categories shipped by SAP. These provide the detailed semantics for the result categories. The system can generate result node types for result node categories only within a result type of a defined result category.

You can define result node types in Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*.

The following fixed result categories and result node categories are shipped by SAP:

Result Category	Result Node Category	Higher-Level Node Based on Result Node Categories
HF **Flow Results**	FLBD FLOW FLCMP	- FLBD FLOW
HFBAS **Calculation Basis for the Financial Position**	KYFBA POSBA PCBID	- - -
HFISD **Imported Subledger Documents**	DOC DOCLN	- DOC
HFBCF **Balance Carryforwards**	BCF BCFLN	- BCF
HKAVR **Valuation Results for Accounting**	HKAVH HKAVN	- HKAVH
HKCRE **Main Run and ABS Preliminary Run**	HKCL0 HKCL1 HKCL2	- HKCL0 HKCL1
HKEET **Expected Exposure**	HKEE0 HKEE1	- HKEE0
HKFLG / HWFLG **Generic Key Date Results**	HKFLG / HWFLG	-
HKPRE **Combined Preliminary Run**	HKPL0 HKPL1	- HKPL0
HKRPC **Risk Positions**	HKRP0 HKRP1	- HKRP0
HKFRT **Funding Results**	HKFRH HKFRR HKFRS HKFRC	- HKFRH HKFRH HKFRS

Result Category	Result Node Category	Higher-Level Node Based on Result Node Categories
HKGFV **Global Fair Value**	HKFVH HKFVN	- HKFVH
HKACR **Accrual Results**	HKARH HKARI	- HKARH
HKLTA **Preliminary Run for Look-Through Approach**	HKLL0 HKLL1 HKLL2	- HKLL0 HKLL1
HKNPV **PA Net Present Values**	HKNPH HKNPN	- HKNPH
HKRCF **Agreed cash flow**	HKRCH HKRCN HKRCI	- HKRCH HKRCN
HKSCR **Standard cost rates**	HKCRH HKCRN HKPVS	- HKCRH HKPVD
HPAPT **Totals for Posting Period**	HPPTH HPPTN	- HPPTH
HPPAV **Average Volume for a Period**	HPAVH HPAVN	- HPAVH
HKCFR **Cash Flow Result**	CFRH CFRI	- CFRH

Predefined characteristics and custom characteristics that are assigned to a result node type are classified as follows:

- Data separator: A technical division of results containing results that are not harmonized or that cannot be harmonized.
- Key: A key field in the database. Different results have to differ in at least one key field.
- Result characteristic: Result that is stored in a characteristic and not in a key figure.
- Display field: A display field is not provided directly by the information production process but is derived internally and offered as an additional field when reading results.
- Required entry field: A required entry field has to be filled with non-initial values when writing results.
- Master data check: The value of the field is checked against the permitted characteristic values when writing results.

5.1.3 Result View

Definition

A result view is a flat, two-dimensional table of a set of result types [page 1367]. It refers to exactly one result view category. Whereas in the RDL the main task of result types and aggregation levels is the storage of data, result views are used to retrieve result data.

Use

Each result view is a flat representation of one or more result types that consists of characteristics and key figures. This flat representation is the key to mass data processing and to the use of Bank Analyzer's data processing tools (for example, the Bank Analyzer data source for results data is defined based on result views).

On the one hand, result views can restrict the amount of information that is generated by a result type by restricting the number of characteristics and key figures. On the other hand, result views can enrich information by including results data from different result types (for example, when providing a standardized view of business transactions that are generated in the internal system and imported subledger documents).

Result views are required whenever you want to read results data from the Results Data Layer (RDL). A result view refers to a specific result view category that provides the information about the combinations of result categories that are supported by the system within one view. For example, you can define a result view based on the result view category accounting view across flow results and imported subledger documents to provide a homogenous access method to all of the accounting documents stored in the system.

You can configure result views as required by assigning specific result types, characteristics, and key figures to them. In Customizing for Bank Analyzer, choose *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*.

When you set up a result type, the system automatically generates the ad hoc result view, unless you specify otherwise. The ad hoc result view is made up of all of the characteristics and key figures from the underlying result type.

The ad hoc result view enables you to access results data stored in the RDL without having to make any additional Customizing settings. Each result type and each aggregation level [page 1374] can deliver a result view in a flat, canonical structure that consists of either all characteristics and key figures of all result node types [page 1370] or of the assigned characteristics and key figures. The term "result views" refers to custom result views and not to ad hoc result views.

5.1.3.1 Filter

Definition

A restriction of data in a value range for one or more characteristic of a result view [page 1373]. The value range is defined in Customizing.

Use

When an aggregation query [page 1374] is made, the selection criteria for results data is specified by means of a filter. A filter always refers to a result view and its characteristics. Filters are also used in combination with aggregation levels.

1. You can define a filter in Customizing for Bank Analyzer under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*.

5.1.4 Aggregation

Use

Results Data Layer (RDL) of Bank Analyzer can aggregate related results data within a result view [page 1373]. If several result types [page 1367] are assigned to the result view, then the system can aggregate results data across several result types.

Integration

In the aggregation levels, you can define granularity characteristics, as well as characteristics and key figures that are to be aggregated and their functions. You can store selection criteria in filters as appropriate. The aggregation levels and the filters are assigned to a result view.

The RDL provides its own data source server. The primary data sources [page 1593] for this data source server refer explicitly to a result view or an aggregation level and can be used for aggregation queries without being saved permanently.

You make the settings for aggregation levels and filters in Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*.

Features

Aggregated results data has a flat structure. When an aggregation query is made, the system groups the results data of the result view according to one or more granularity characteristics (unit fields of the key figures to be aggregated on granularity level).

The following assignments of aggregation functions are possible:

	SUM (Total)	AVG (Average)	MIN (Minimum)	MAX (Maximum)
Key Figures	Yes	Yes	Yes	Yes

Characteristics:	No	No	Yes	Yes

You can use selection criteria to restrict the amount of results data that is to be aggregated.

In an aggregation level, you can define granularity characteristics, as well as characteristics and key figures that are to be aggregated and their functions. In RDL of Bank Analyzer, aggregation levels are always assigned to a result view, which is called the source result view of the aggregation level. Characteristics and key figures of this aggregation level must be assigned to this source result view.

Aggregation in Result Views

Aggregation in result views refers to result views that are defined in Customizing, and ad hoc result views from result types.

You can define the following parameters (optional):

- Scenario
- Filter (for result views that are defined in Customizing only)
- Timestamp
- Timestamp from (for a delta query)
- Parameters that depend on the versioning schema
- Granularity characteristics
- Characteristics and key figures and their aggregation functions
- Other selection criteria (conditions for the values of characteristics)

If a filter is used, the aggregation is restricted to the results that meet the filter conditions.

The result view of the aggregation query must contain characteristics and key figures. In the case of an ad hoc result view of a result type, this includes all characteristics and key figures of the corresponding result node types.

Aggregation Using Aggregation Levels

In Results Data Layer of Bank Analyzer, you can use three aggregation level categories:

- **Aggregation level**
  Aggregation using aggregation levels refers to aggregation levels that are created in Customizing. When an aggregation query is made, you can define the following parameters (optional):
  - Scenario
  - Filter (If you use a filter, the system aggregates only the results in the source result view that are assigned to the aggregation level. The results must meet the filter conditions).
  - Timestamp
  - Parameters that depend on the versioning schema
  - Granularity characteristics (Granularity characteristics must correspond to the aggregation level. The other characteristics and key figures must be contained in the aggregation level).
  - Characteristics and key figures and their aggregation functions
  - Other selection criteria (conditions for the values of characteristics)

Aggregation using aggregation levels takes place in two steps:

1. The system aggregates the results of the source result view that is assigned to the aggregation level according to the metadata of the aggregation level (for example, granularity characteristics, and characteristics and key figures with assigned aggregation functions). Results are also selected based on the selection criteria scenario, filter, timestamp, and on the parameters that depend on the versioning schema.

2. The aggregated results are aggregated further in line with the settings for granularity characteristics, characteristics and key figures and their aggregation functions, and other selection criteria. You can further restrict the set of results that were aggregated in the first step, or use a less detailed level of granularity to aggregate.

- **Balance level**
  The system uses aggregation at balance level to process results of a result view for balances. The system creates these balances for a specific date characteristic. To use this aggregation category you must use the *single versioning with reversal* versioning schema [page 1378] for the underlying results.
  The balances that are created use the *two-dimensional versioning* versioning schema. This means that, for a query at balance level, you have to enter a key date. As with other aggregation levels, the system can preaggregate balance levels, so that only new results are added when a query is run. However, you cannot enter a key date for a preaggregation run.
  You can use balance levels to calculate the totals for key figures, but not to aggregate characteristics.
- **Horizontalization level**
  As with aggregation levels, you can define the following parameters (optional):
  - Scenario
  - Filter (If you use a filter, the system aggregates only the results in the source result view that are assigned to the aggregation level. The results must meet the filter conditions).
  - Timestamp
  - Parameters that depend on the versioning schema
  - Granularity characteristics (Granularity characteristics must correspond to the aggregation level. The other characteristics and key figures must be contained in the aggregation level).
  - Characteristics and key figures and their aggregation functions
  - Other selection criteria (conditions for the values of characteristics)

  If you aggregate using the horizontalization level, the system converts key figures from an implicitly vertical container structure (name and value of the key figure) to an explicitly horizontal structure (name and key figure correspond to the columns). The horizontalization level determines which key figures are converted.

Preaggregation

You start preaggregation by means of a report in the application menu. The report creates aggregated results for an aggregation level and a filter, and saves them permanently on the database. Saved, preaggregated data is reused for single versioning with reversal when an aggregation query is made. This data is merged with data from the underlying result view to provide up-to-date aggregated results (delta aggregation). You can run the preaggregation in parallel. You can cancel incorrect preaggregation runs using the run administration function. For more information, see Preaggregation [page 1408].

5.1.5 Opening and Closing Posting Periods (Subledger Scenario)

Use

You use this transaction to determine whether postings can be made in periods of the fiscal year. This applies to both automatic and manual postings. When posting, the system uses the posting date to automatically assign the document to the corresponding posting period.

Prerequisites

You have defined at least one legal entity, one fiscal year variant, and one accounting system and assigned them to one another in the following Customizing activities under *Bank Analyzer* > *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Accounting Systems and Legal Entities*:

- *Define Accounting Systems*
- *Define Fiscal Year Variant*
- *Define Legal Entities*

Creating special periods when defining a fiscal year variant, for example, for year-end postings, is presently **not** supported by Bank Analyzer.

Procedure

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Results Data Layer* > *Periodic Tasks* > *Open and Close Periods*.

1. Choose an accounting system and a legal entity.
2. Specify a fiscal year and a period.
3. Use the indicator to specify whether the posting period is open or closed.
4. Save your entries.

> **Note**
>
> You should reopen closed posting periods in exceptional cases only. This can arise if you have reversed posting after a posting period has closed. To correct the balances in the posting period, reverse the late reversal, open the posting period, and repeat the reversal. The balances are then correct from the reversal posting date. You can close the posting period again. For more information, see Reversal [page 795].

5.2 Results Data Framework

Use

The results data framework is part of the Results Data Layer (RDL). As a generic framework, it provides the infrastructure, including the interfaces [page 1397], for storing and accessing results data.

Integration

To enable other business processes to be run, results that are defined in the results metadata [page 1364] are transferred to the results data framework.

Prerequisites

To use the functions of the results data framework for results data, you first have to have defined a results data area and a versioning schema for the results data. You make these settings in Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*.

Features

- Reading results
  - You can read all versions of results provided they do not have the status *not valid*. Reading is unlimited, unless you use result views [page 1373] or aggregates [page 1374] to restrict the data displayed. You can define characteristics and key figures of result views and aggregates in Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*.
  - If there are no results available for a key date, you can read older versions of these results. You can define the time span before the key date for which the results are to be read in Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.
  - You can also read results from a time interval (delta read). You can define the time interval for which the results are to be read in Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.
- Writing results
  You can write new results or aggregates of results. New results depend on the versioning schema [page 1378] selected.
- Setting results to invalid
  You can only set results to invalid with the versioning schema result group versioning [page 1387]. The system then generates a new version of the result that has the status invalid.
- Reversing results
  You can reverse only results that have the versioning schema single versioning with reversal [page 1385].
- Versioning [page 1378]
- Aggregation [page 1374]

5.2.1 Versioning

Definition

The versioning concept in Results Data Layer (RDL) is designed so that when you change or correct an object, the system automatically creates a new version of the object. Object versions that already exist cannot be overwritten here. Versioning enables you to view specific version statuses, and to compare different version statuses.

Use

By means of versioning in RDL, you can store data for a result without having to overwrite the existing data for the same result. To this end, a result version is given a timestamp and an internal key (globally unique identifier, or GUID). These two key characteristics are then used during the selection of results to determine the version that is currently valid.

To define how often data can be delivered for a result, and which result version is valid at which time, RDL provides six versioning schemas:

- No Versioning [page 1381]
- Single Versioning with Reversal [page 1385]
- One-Dimensional Versioning [page 1382]
- Two-Dimensional Versioning [page 1383]
- Result Group Versioning [page 1387]
- One-Dimensional Versioning with Logical Delete [page 1387]
- Two-Dimensional Versioning with Logical Delete [page 1389]

> **Note**
>
> When you create a result type, you must assign a specific versioning schema to it.
>
> You cannot change the assignment of a versioning schema.

The following table shows a selection of result categories [page 1367] and which versioning schemas can be assigned to them.

Result Category	Abbreviation of Result Category	No Versioning	One-Dimensional Versioning	Single Versioning with Reversal	Two-Dimensional Versioning	Result Group Versioning	One-Dimensional Versioning with Logical Delete	Two-Dimensional Versioning with Logical Delete
Posting Document	HFSPD	Yes	No	No	No	No	No	No
Flow Results	HF	No	No	Yes	No	No	No	No
Imported Subledger Documents	HFISD	No	No	Yes	No	No	No	No

Result Category	Abbreviation of Result Category	No Versioning	One-Dimensional Versioning	Single Versioning with Reversal	Two-Dimensional Versioning	Result Group Versioning	One-Dimensional Versioning with Logical Delete	Two-Dimensional Versioning with Logical Delete
Main Run and ABS Preliminary Run	HKCRE	No	No	No	No	Yes	No	No
Expected Exposure	HKEET	No	No	No	Yes	Yes	No	No
Generic Key Date Results	HKFLG HWFLG HKFLH	No	No No No	No No No	Yes Yes No	Yes Yes Yes	No Yes No	No Yes No
Combined Preliminary Run	HKPRE	No	No	No	No	Yes	No	No
Risk Positions	HKRPC	No	No	No	Yes	Yes	No	No
Calculation Basis for the Financial Position	HFBAS	No	No	No	Yes	No	No	No
Valuation Results for Accounting	HKAVR	No	No	No	No	No	Yes	No
Funding Results	HKFRT	No	No	No	No	No	No	Yes
Global Fair Value	HKGFV	No	No	No	No	No	Yes	No

Result Category	Abbreviation of Result Category	No Versioning	One-Dimensional Versioning	Single Versioning with Reversal	Two-Dimensional Versioning	Result Group Versioning	One-Dimensional Versioning with Logical Delete	Two-Dimensional Versioning with Logical Delete
Preliminary Run for Look-Through Approach	HKLTA	No	No	No	No	Yes	No	No
PA Net Present Values	HKNPV	No	No	No	No	No	Yes	No
Agreed Cash Flow	HKRCF	No	No	No	No	No	No	Yes
Standard Cost Rate	HKSCR	No	No	No	No	No	No	Yes
Totals for Posting Period	HPAPT	No	No	No	No	No	Yes	No
Average Volume of a Period	HPPAV	No	No	No	No	No	Yes	No
Accrual Results	HKACR	No	No	No	No	No	Yes	No
Cash Flow Result	HKCFR	No	No	No	Yes	No	No	No

5.2.1.1 No Versioning

This schema is supported only for the *Posting Document* result category. Unlike the versioning schema Single Versioning with Reversal [page 1385], with this schema a reversal is executed by creating a new document with the reversal indicator and a reference to the document to be reversed. The document to be reversed is not changed, and the consistency of the reversal is ensured by the process that generates it.

5.2.1.2 One-Dimensional Versioning

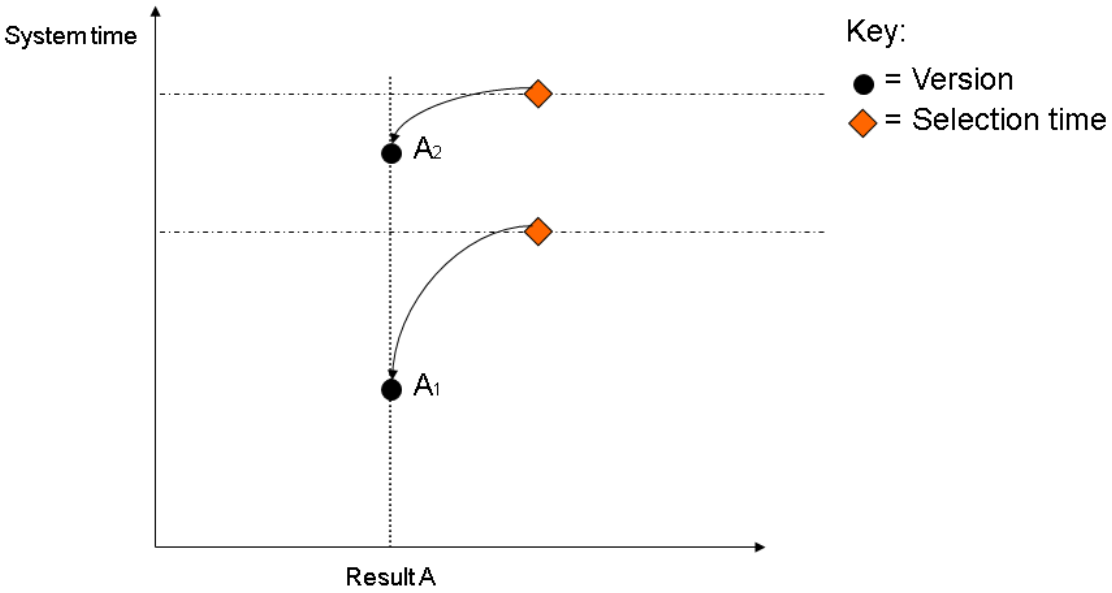
Use

The versioning schema *one-dimensional versioning* is used within aggregation levels only. It is used to store result aggregates that use the versioning schema single versioning with reversal [page 1385].

The internal ID and the timestamp are used as keys. If the result does not exist, the system generates a result header in which the internal ID is stored. The system creates a version using the timestamp given and the version is valid from this timestamp on. The results data is stored with a link to the version.

If the result already exists, the system creates a new version and the data is stored. The currently valid version is updated, because its validity ends with the timestamp of the new version.

One-Dimensional Versioning



Activities

You can assign the versioning schema in Customizing for Bank Analyzer under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*. Select a results data area and, in the dialog structure, double-click on *Result Types*.

5.2.1.3 Two-Dimensional Versioning

Use

The two-dimensional versioning schema in the Results Data Layer (RDL) of Bank Analyzer is used to store results that use the dimensions "key date" and "system time (timestamp)" for identification.

A result can have several versions. These are identified by different timestamps and key dates. The system generates these versions when changes are made to results data. The results data must be saved as a new and complete version due to these changes. An example of this is a value at risk key figure that is recalculated.

There are result types that have a unique key date, which influences the validity of the result. The creator of the result sets this date. When the system reads the result, this date has a higher priority than the timestamp. This means that the system selects a version that has a later key date even if a version with a later timestamp, but an earlier key date exists.

The RDL reads by default in reverse chronological order and selects an older key date if a key date for which there is no version was used in the selection criteria. The two-dimensional versioning schema is used by most of the primary objects of the Source Data Layer (SDL) and by the RDL for, for example, flat generic key date results.

Two-Dimensional Versioning

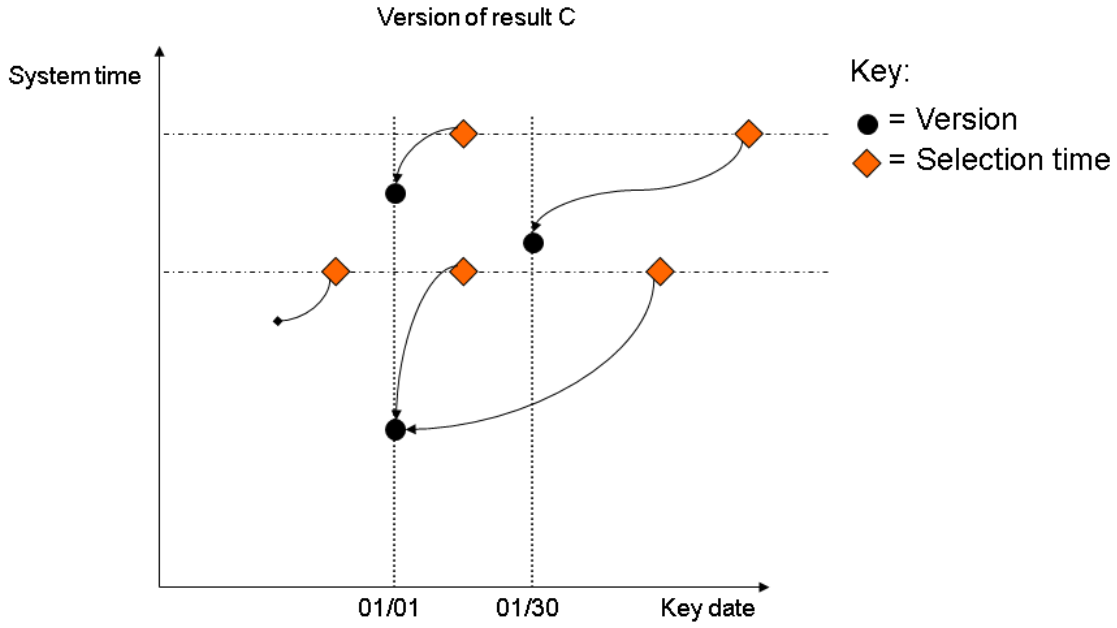


Figure 1

If the new version is within the validity period of an existing version, but has a later key date, it takes on the "valid-to key date" of the existing version. The "valid-to key date" of the existing version is changed to the day before the key date of the new version. The system creates a 'shadow' version (see figures two and three) for the existing version,

with the key date of the new version and the timestamp of the existing version. The validity of the shadow version ends with the timestamp of the new version.

Two-Dimensional Versioning

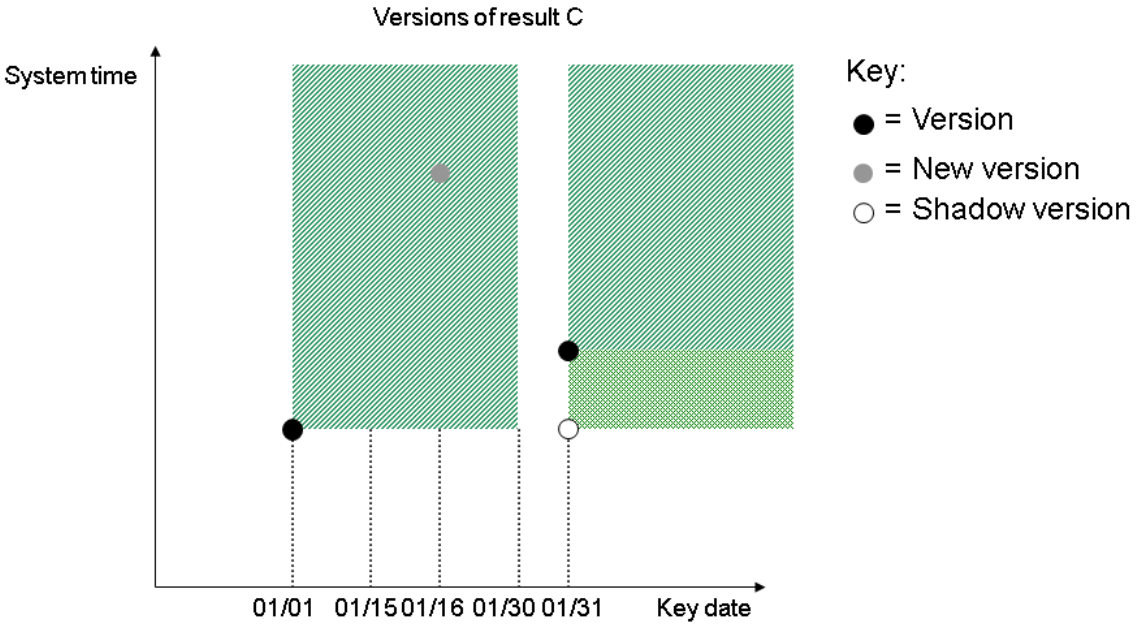


Figure 2

Two-Dimensional Versioning

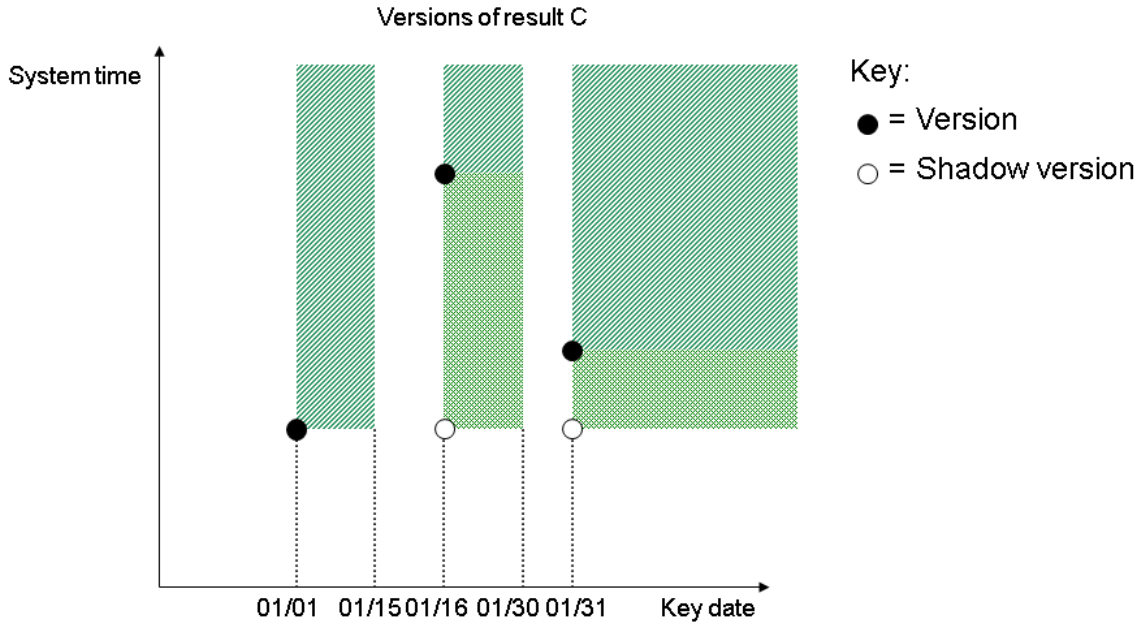


Figure 3

Activities

You can assign the versioning schema in Customizing for Bank Analyzer under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*.

5.2.1.4 Single Versioning with Reversal

Results data can be delivered only once for certain results (for example, in the accounting area). If the data delivered is incorrect or invalid, it has to be reversed. The system generates a new result where the values are the opposite of those in the original result. This result uses the schema *single versioning with reversal*.

A version of this result requires an internal key and a timestamp for identification because the status of the result can be changed independent of the delivery of results data.

If data is delivered for a result, there are two possibilities:

- If the result exists, the system refuses to store the data again because it can only de delivered once.

- If the result does not exist, the system generates a result header, a version with the current timestamp, and the data with a link to the version.

If a result is to be reversed, the following prerequisites have to be met:

- The result that is to be reversed exists
- The result that is to be reversed has not already been reversed
- The result that is to be reversed is not the reversal of a reversal

Reversal of an original:

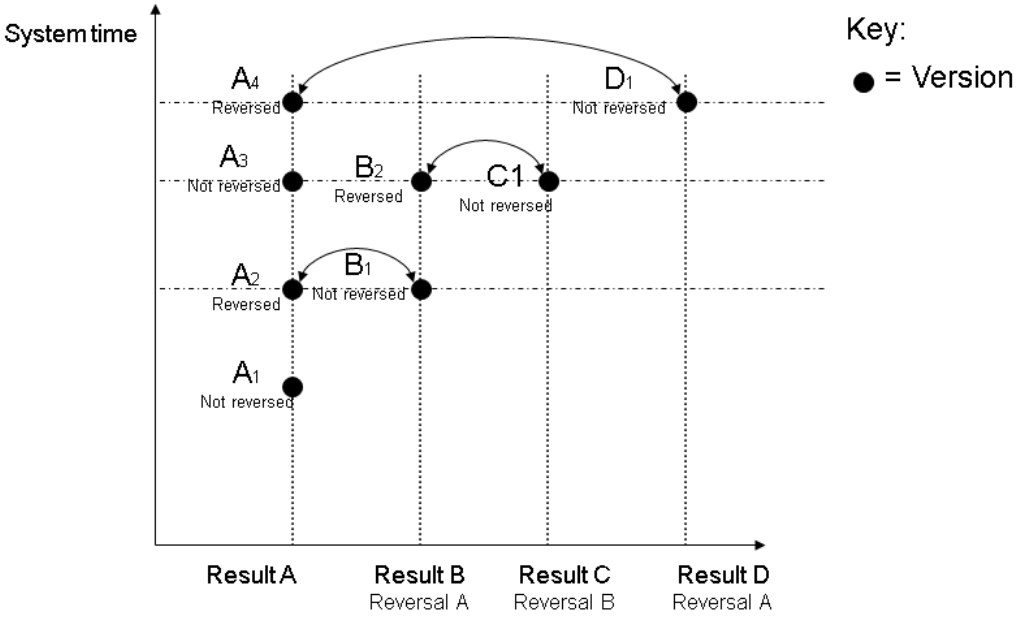
Reversal: If all prerequisites are met, the system generates the header, version, and data for the reversal. The version of the reversal refers to the reversed result.

Reversed result: The reversed result receives a new version, due to the status change and in order to store the link to the reversal. The validity of the currently valid version is limited to the timestamp of the reversal.

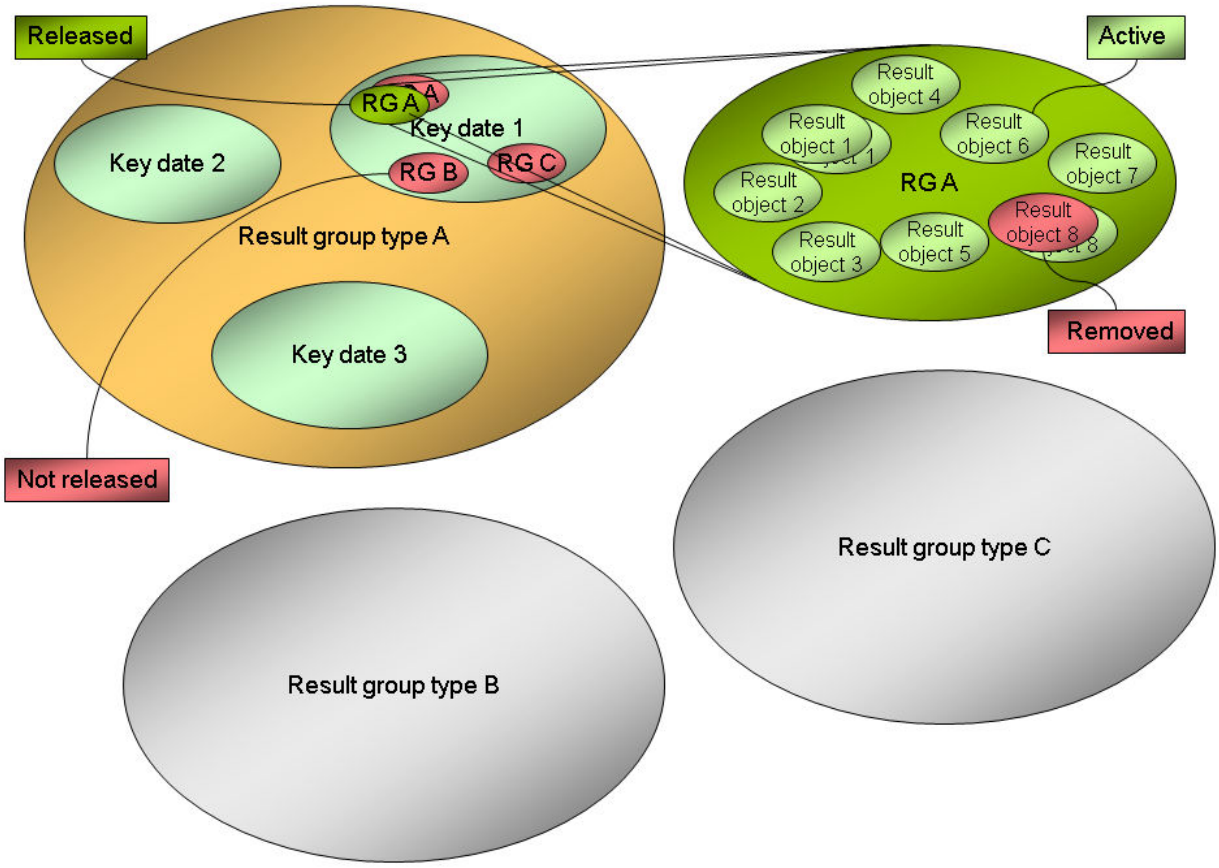
Reversal of a reversal:

If the reversed result is a reversal, it is not reversed. It is given a new version that has the timestamp of the new reversal and the status "not reversed". The link to the reversal is removed from this version and the validity of the version that is currently valid is limited to the timestamp of the new reversal.

Single Versioning with Reversal



5.2.1.5 Result Group Versioning



Relationship Between Result Group Type, Result Group, Key Date and Result Object

5.2.1.6 One-Dimensional Versioning with Logical Delete

Use

You can use this function to logically (not physically) delete versions. The system creates a new version and selects it as deleted. This means that the result itself is not deleted. The validity of the result categories for which the system uses this versioning schema is dependent on a technical key that consists of the internal ID and the timestamp. The result statuses *active* or *deleted* are additional properties of the version. In contrast to one-dimensional versioning [page 1382], this function is used to store more than result aggregates. External clients can also access it.

When you save results, you have the following options:

- If the result does not yet exist, the system creates a result header that contains the internal ID. The system creates a version using the timestamp, and the version is valid from this timestamp on. The system sets the result status to *active* and saves the results data and a link to the version.

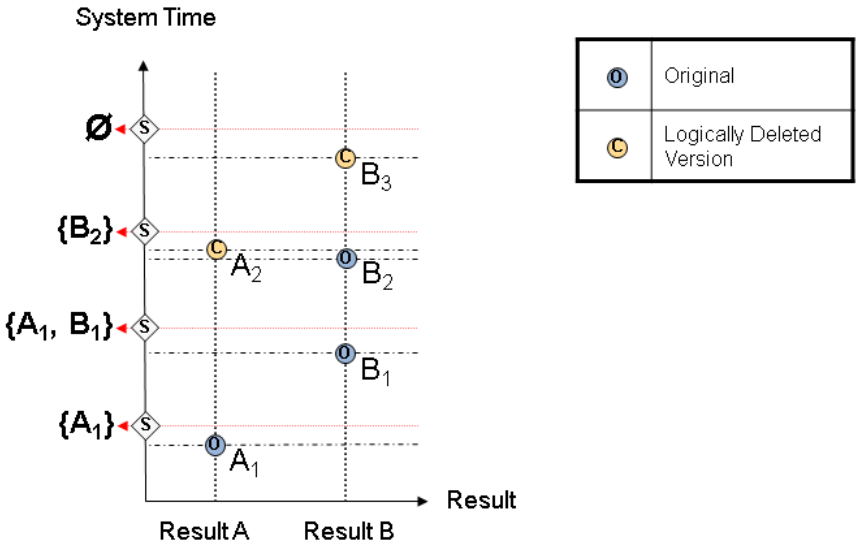
- If the result already exists, the system creates and saves a new version and sets the result status to *active*. The status of the version that was active until then is changed as its validity ends with the timestamp of the new version.
- If you logically delete a result, the system creates a new version of the same result and sets the result status to *logically deleted*. The status of the version that was active until then is changed as its validity ends with the timestamp of the new version.

> **Caution**
>
> You can logically delete a version only if the last valid version has the status *active*.

When you run a query for data, you have the following options:

- If the last valid result version is selected as *active*, the system loads this version.
- If the last valid result version is *logically deleted*, the system does not load any version.

The following figure shows the import of results using versioning schema *one-dimensional versioning with logical delete* based on status and on the time of storage and query.



One-Dimensional Versioning with Logical Delete

Activities

You can assign the versioning schema in Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area* . Select a *Results Data Area* and, in the dialog structure, double-click on *Result Types*.

5.2.1.7 Two-Dimensional Versioning with Logical Delete

Use

You can use this function to logically (not physically) delete versions. The system creates a new version and selects it as deleted. This means that the result itself is not deleted. The validity of the result categories for which the system uses this versioning schema is dependent on a technical key that consists of the internal ID, and the dimensions *key date* and *timestamp*. This function is comparable to two-dimensional versioning [page 1383]. The difference here is that the results have to be able to be logically deleted for a particular key date.

To enable the system to find the correct version in a single query, multiple versions of the same result are created.

The system follows this procedure when creating and deleting versions:

- If there is no logically deleted version, the procedure is exactly the same as for two-dimensional versioning.
- Logical delete without and with data for an earlier key date: If the key date and timestamp selected for a version that is to be deleted delivers a result version, the logically deleted version uses the data of this result version.
  For more information, see Example [page 1390].
- Logical delete with data for an earlier key date and a later key date: If you logically delete a version for a particular key date, this logically deleted version is the last valid version until the key date for which an active version exists. The creation of this deleted version triggers the system to create deleted versions for all key dates between this deleted version and the next valid version as well.
  For more information, see Example [page 1391].
- Creation of active version with data for a later key date: If you create an active version for a key date, this active version is the last valid version until the next valid version exists for a later key date. The creation of this valid version triggers the system to create active versions for all key dates between this newly created version and the next valid version as well.
  For more information, see Example [page 1392].

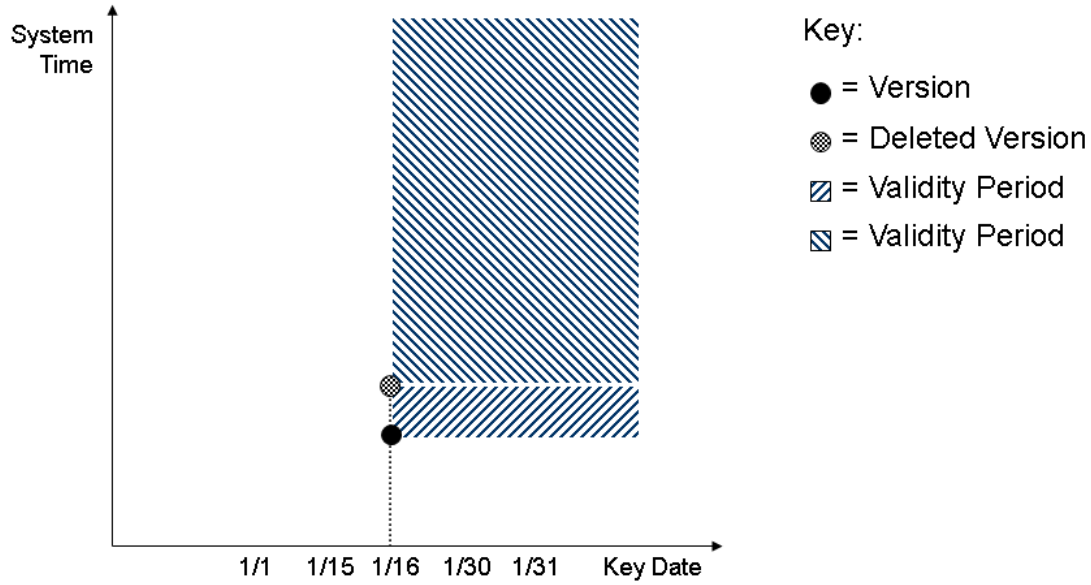
> ⚠ Caution
>
> You can logically delete a version for a particular key date only if the last version that refers to this key date is active.

Activities

You can assign the versioning schema in Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*. Select a results data area and, in the dialog structure, double-click on *Result Types*.

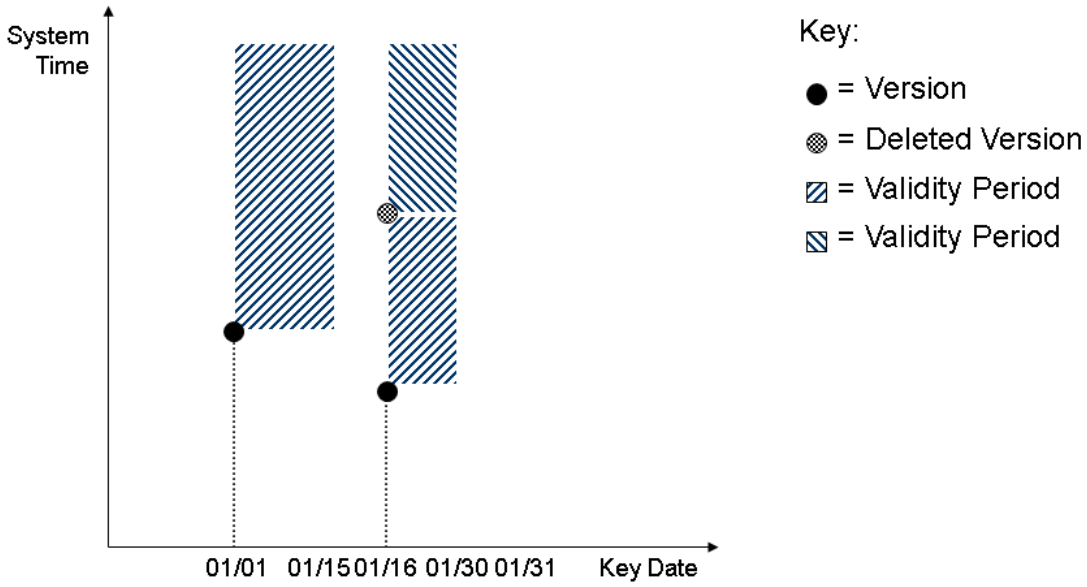
5.2.1.7.1 Logical Delete Without and With Data for an Earlier Key Date

You delete a version for key date 01/16. No data exists for a version with an earlier key date. The system uses this function only to update the validity of the existing version. The logically deleted version does not contain any data as there is no version that has an earlier key date.



Logical Delete Without Data for an Earlier Key Date

You delete a version for key date 01/16. However, a version exists that contains data for key date 01/01. The logically deleted version for key date 01/16 sets the status of the last active version on key date 01/16 to deleted. It contains the data of the last valid version, which is the version for key date 01/01 in this case. If you run a query for 01/16, the system displays the data of the last valid version, which is the version for 01/01 in this case.

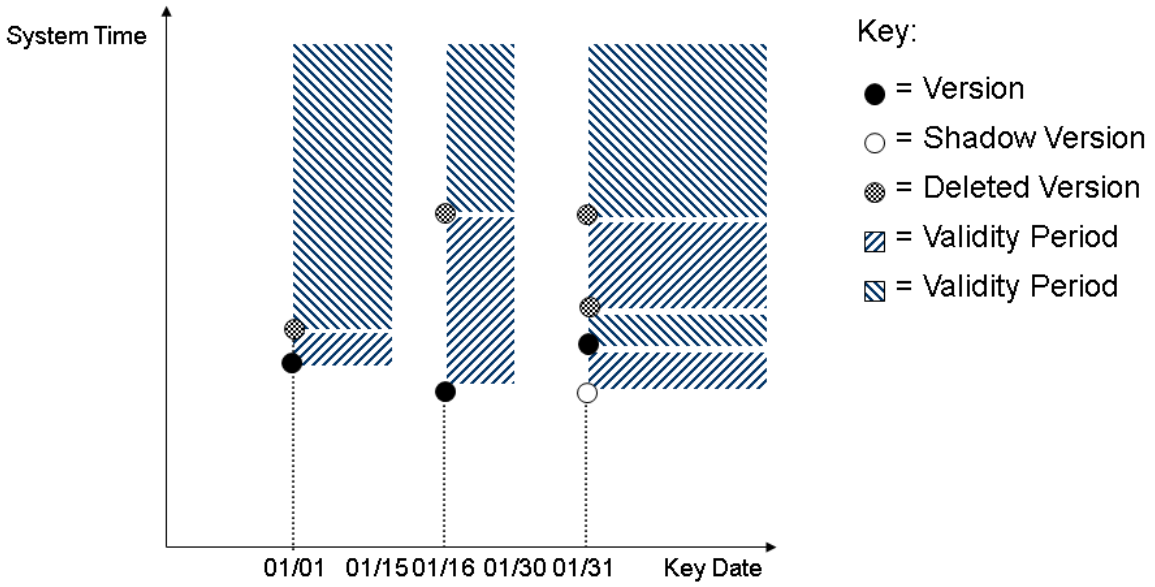


Logical Delete With Data for an Earlier Key Date

5.2.1.7.2 Logical Delete with Data for an Earlier Key Date and a Later Key Date

You delete the version for key date 01/16. The creation of this logically deleted version triggers the simultaneous creation of a new logically deleted version for key date 01/31. This is because the logically deleted version previously deleted for key date 01/31 contained the data of the valid version for key date 01/16. As this version has now been logically deleted, the system has to change all logically deleted versions that have key dates between the newly created deleted version and the next valid version. The two newly created deleted versions do not contain any data as the version that was logically deleted on 01/01 does not contain any data.

The creation of a shadow version on 01/31 is necessary so that the system can supply the correct version in a single query. Creating a version for key date 01/31 ensures that the system supplies a currently valid version for a query that uses 01/31 as the key date, but that uses a system time that is before the creation of the version on 01/31. Before the version for 01/31 was created, the version for 01/16 was valid.

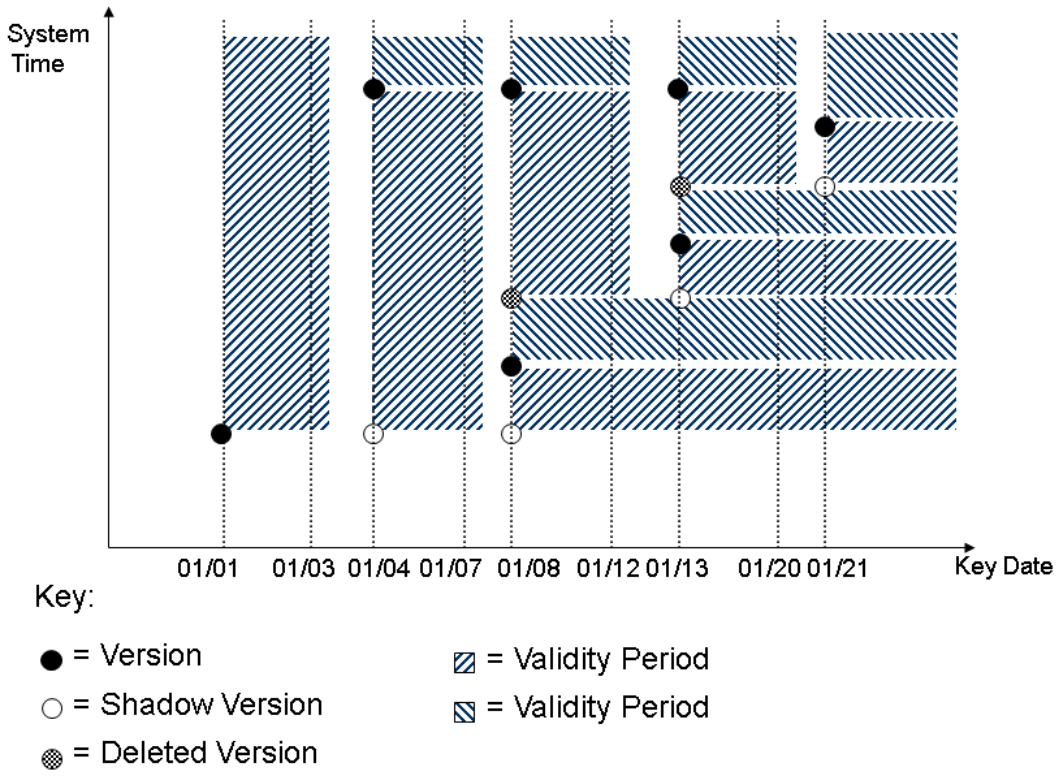


Logical Delete with Data for an Earlier Key Date and a Later Key Date

5.2.1.7.3 Creation of Active Version with Data for a Later Key Date

You create a new version for key date 01/04. However, versions exist for later key dates. The creation of the version for key date 01/04 triggers the system to create active versions for all key dates between 01/04 and the next valid version (in this case 01/21). The system supplies all newly created versions with the data of the version on 01/04.

The creation of a shadow version, for example on 01/13, is necessary so that the system can supply the correct version by running only one query. Creating a version for key date 01/13 ensures that the system supplies a currently valid version for a query that uses 01/13 as a key date, but that uses a system time that is before the creation of the version on 01/13. Before the new version for 01/13 was created, the logically deleted version for 01/08 was valid.



Creation of Active Version with Data for a Later Key Date

5.2.2 Persistence

A persistence schema controls the data model for the storage of results in the RDL. The result category assigned to a result type defines which combination of versioning and persistence schemas the system supports. The following persistence schemas are available:

- *'00' Standard RDL Data Storage*: Database model with segmentation service, header, version, and data table.
- *'01' Specific to Result Category*
- *'02' One Table, Direct Key*: The data is stored in one table in fully denormalized form. The semantic key of the result is directly used as the primary key for the table.
- *'03' One Table, Technical Key*: The data is stored in one table in fully denormalized form. The technical GUID is used as the primary key for the table.
- *'04' One Table with Shadow Version, Technical Key*: The data is stored in one table in fully denormalized form. The technical GUID is used as the primary key for the table. Each result version leads to two technical versions in the database table. The second version is the shadow version that is used for the high-performance version access.

Note

Non-standard persistence schemas are supported for selected result categories and versioning schemas. However, all of the RDL functions are not yet available for non-standard persistence schemas. Note the restrictions under Result Types and Result Categories [page 1367].

Note

The result category "HFSPD" (Posting Document) has a persistence schema implementation that is result category-specific. The system supports the persistence schemas '02', '03' and '04' for selected result categories, and the versioning schemas '2' Two-Dimensional Versioning and '4' One-Dimensional Versioning. You can find the supported combinations of versioning schemas and persistence schemas for result categories in the system table /BA1/THM_RSCT_VS.

The following table shows a selection of result categories and which versioning and persistence schemas can be assigned to them.

Result Category	Abbreviation of Result Category	Versioning Schema	Persistence Schema
Flow results	HF	1	
Calculation base of financial position	HFBAS	2	
Imported subledger documents	HFISD	1	
Posting document	HFSPD	7	1
Accrual results	HKACR	5	
Valuation results for accounting	HKAVR	5	
Cash flow result	HKCFR	2	3
Cash flow result	HKCFR	2	4
Main run and ABS preliminary run	HKCRE	3	
Main run and ABS preliminary run	HKCRE	3	3
Expected exposure	HKEET	2	
Expected exposure	HKEET	3	
Expected exposure	HKFLG	2	
Generic key date results	HKFLG	3	

Result Category	Abbreviation of Result Category	Versioning Schema	Persistence Schema
Generic key date results	HKFLG	3	3
Generic key date results	HKFLH	3	
Funding results	HKFRT	6	
Global fair value	HKGFV	5	
Preliminary run for look-through approach	HKLTA	3	
PA net present values	HKNPV	5	
Combined preliminary run	HKPRE	3	
Combined preliminary run	HKPRE	3	3
Agreed cash flow	HKRCF	6	
Risk positions	HKRPC	2	
Risk positions	HKRPC	3	
Standard cost rates	HKSCR	6	
Totals for posting period	HPAPT	5	
Average volume for a period	HPPAV	5	
Generic key date results	HWFLG	2	
Generic key date results	HWFLG	3	
Generic key date results	HWFLG	5	
Generic key date results	HWFLG	6	

5.2.3 Archiving and Destroying Results Data

Use

You can archive results data using the Archiving Engine and destroy it as part of Information Lifecycle Management (ILM).

The number of generated archiving objects and ILM objects for the results data depends on the versioning [page 1378] used and the Persistence [page 1393]:

- For results data that uses single versioning with reversal, there is only one archiving object or one ILM object for each result type because the system always archives or destroys one result completely with all of its versions.
- For results data that uses versioning (one-dimensional versioning, two-dimensional versioning or result group versioning) with persistence schema *00*, there are two archiving objects and two ILM objects.
  One of the archiving objects or ILM objects is used for archiving or destroying result versions, while the other is used for archiving or destroying the header data that groups all versions. The header data can only be archived or destroyed if the result is to be archived completely and all versions have previously been archived or destroyed.
- For results data that uses one-dimensional versioning with persistence schema *03*, two-dimensional versioning with persistence schema *02*, *03* and *04*, and result group versioning with persistence schema *03* there is one archiving object and one ILM object. These objects are used for the archiving and destruction of data records from the RDL. The number of generated archiving objects and ILM objects for the results data depends on the versioning used.
- For results data that uses result type HFSPD, there is only one archiving object or one ILM object for three database tables.
  For more information about this result type, see Result Types and Result Categories [page 1367].

Prerequisites

- To execute archiving or data destruction, there must first be archiving objects and ILM objects in the system for each result type. You enter the data for aggregation in Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Technical Settings in Results Data Area*. In the *Archiving for Result Type* dialog, you can set the indicator for archiving. When you save, the system automatically generates archiving objects and ILM objects. You can display these objects for each result type in Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*.
  When you save, the system also generates the required Customizing settings for the ILM objects. The transaction `IRM_CUST` contains the Legal Entity condition field (`ACC_LEGENT`) for ILM objects of versions. Condition fields are not required for the generated ILM objects of header tables.
- In transaction `ILMARA` (Audit Area Processing), you need to assign an audit area to each ILM object.
- In transaction `IRMPOL` (ILM Policies), you define the retention rules for your audit area for each ILM object.

Procedure

1. Archiving analysis phase:
   The system determines the archiving status of an object to be archived. It does this by running the following checks:
   - Residence time check: An object can be archived only if the actual date is later than the posting date, or of the key date of the object plus the residence time in the system. The former is the case for flow results and imported subledger documents and the latter is the case for the other result categories. When it calculates the residence time, the system uses the system date as the start date for header data and the business key date as the start date for versions.

- Business check: The system can archive an object only if it is no longer used. The versions of an object are always checked. The system uses BAdIs to check a complete result. The following BAdIs are available for each archiving object in Customizing for *Bank Analyzer* under *Results Data Layer* > *Archiving*:
  `/BA1/AL_HW_ARCH_ODV` for one-dimensional versioning
  `/BA1/AL_HW_ARCH_ODV_PS3` for one-dimensional versioning with persistence schema *03*
  `/BA1/AL_HW_ARCH_MV` for two-dimensional versioning
  `/BA1/AL_HW_ARCH_MV_PS02` for two-dimensional versioning with persistence schema *02*
  `/BA1/AL_HW_ARCH_MV_PS34` for two-dimensional versioning with persistence schema *03* and *04*
  `/BA1/AL_HW_ARCH_SV` for single versioning with reversal
  `/BA1/AL_HW_ARCH_RGV` for result group versioning
  `/BA1/AL_HW_ARCH_RGV_PS3` for result group versioning with persistence schema *03*

  The system sets only objects that are run successfully through both checks to *can be archived*. The other objects are set to *cannot be archived* and a resubmission date is set. The resubmission date determines when a new archiving check for an object is run. In this way the amount of data to be archived can be reduced to specific time periods.
  The resubmission date is found by comparing the following two date combinations and taking the later date: Reference date + residence time + 1 and current date + resubmission period.
  You can define the resubmission period in the archiving settings in Customizing.
2. Write: The system writes the objects that can be archived from the database into the archive.
3. Delete: The system deletes the objects that were written to the archive from the database.
4. Destroy: The objects are marked as destructible and can then be destroyed with the deletion run. The objects are also deleted from the archive; they cannot be reloaded.

5.3 Interfaces

Definition

Interfaces are used to exchange information between an SAP system and other systems. The Results Data Layer (RDL) in Bank Analyzer contains different interfaces. You can use them to load data, to read data, to set data to invalid, or to change the status of result groups.

Structure

The RDL has the following interfaces:

- External Interfaces [page 1398]
- User Interfaces [page 1406]

More Information

For more information, see XI Structure Mapping [page 1735].

5.3.1 External Interfaces

SAP provides a range of programming interfaces, which you can use to integrate external applications into an SAP system.

In the context of the Results Data Layer (RDL), external results can be loaded, read, or reversed in results data areas [page 1366].

The following external interfaces are provided in the RDL:

- Remote Function Call interfaces (RFC interfaces) [page 1398] for loading and deleting generic key date results, and changing the status of result groups. RFC interfaces are divided into fixed and generated RFC interfaces.
- Application-to-Application (A2A) Interfaces [page 1401] for loading, reversing or reading results in the RDL.

5.3.1.1 RFC Interfaces

Use

A Remote Function Call (RFC) is a programming interface that you can use to access Bank Analyzer's business data and processes from within another application system. RFC interfaces are divided into generated and fixed RFC interfaces.

Features

Generated RFC interfaces

If you want to load or delete imported generic key date results that are based on result category HKFLG, you have to generate RFC interfaces. The generated RFC interfaces perform the following tasks:

- Map external posting information or external cancellation information to the generic function modules in Bank Analyzer
- Call generic function modules
- Manage errors

Generic function modules are shipped with the system and are the link between the generic RFC interfaces and the Application Programming Interface (API), the latter of which is responsible for posting and canceling generic key date results.

The generation of RFC interfaces includes the following elements:

- Function group
- Function modules
- Structures of the ABAP Dictionary

For each generic key date result type, the system generates a function group with a total of eight function modules. The generation of the RFC interfaces is cross-client. The following function modules are in the function group:

- Two function modules with a Commit Work command for posting

- Two function modules without a Commit Work command for posting
- Two function modules with a Commit Work command for cancellation
- Two function modules without a Commit Work command for cancellation

The function modules contain the characteristics and key figures of the result type for which the RFC interface is generated.

Names of the elements of the generated RFC interface

The namespace of the generated RFC interfaces begins with `/1BA/` and has a user-specific part that is determined by an RFC infix that you specify in Customizing. The names for the individual elements that are generated are determined according to the following semantics:

- RFC function group: "/1BA/IN_HK_RFC_" + "UUUUUUUUUU" -> Character length 25 = 15 + 10 custom characters that guarantee the uniqueness of the RFC function group.
- RFC function modules:
  "/1BA/IN_HK_RFC_" + "XYZ_" + "UUUUUUUUUU" -> character length 29 = 15 + 4 + 10
  "X": Type of function module with one of these values: "P" = Post and "C" = Cancel
  "Y": Commit work, or no commit work with one of these values: "C" = Commit and "N" = No Commit
  "Z": Type of results data with one of these values: "M" = Mass data and "S" = Single records
  "UUUUUUUUUU": This is the custom part that is determined by the RFC infix.
- DDIC structures:
  "/1BA/HK_STR_" + "XYZ_" + "UUUUUUUUUU" -> character length 26 = 12 + 4 + 10
  "X": ABAP Dictionary structure for type of function module with one of these values: "P" = Post and "C" = Cancel
  "Y": Role of the ABAP Dictionary structures with one of these values: "I" = Result information, "O" = Object key, "K" = Key figure and "C" = Characteristic
  "Z": Type of results data with one of these values: "M" = Mass data and "S" = Single records
  "UUUUUUUUUU": This is the custom part that is determined by the RFC infix.

The generation of RFC interfaces is based on result types [page 1367] of result category HKFLG (generic key date results) that first have to be created within a results data area. These result types can change over time due to the addition of characteristics and key figures. As a result, the RFC generator is not only required to support the generation of RFC interfaces for a result type, but also to react to changed metadata in the result types. For this reason, the RFC generator involves several services:

- Generate: All elements are created for an existing result type.
- Regenerate: All elements are regenerated for a result type after the metadata (characteristics and key figures) has been successfully changed.
- Delete: The RFC interfaces for a result type are deleted. This is done by means of Garbage Collector, which you call using transaction /GC1/START.
- Check: The status of the generated elements is checked.

IMPORTING variables of the generated RFC interfaces

/BA1/HK_STR_RESULT_HEAD: Fixed structure containing the following fields:

- KEYDATE: Key date
- SCENARIO: Scenario
- GROUP_TYPE: Result group type

You have to fill all fields in this structure.

The generated tables or structures contain a fixed part OBJ_ID that acts as an internal index. This field must be populated with a unique number for each result object (any number as long as it is unique).

- I_TAB_OBJ_KEY / I_STR_OBJ_ID: Key characteristics of the result object. This must be populated and be unique for each object.
- I_TAB_CHAR_FLAT / I_STR_CHAR_FLAT: Other characteristics of a result object. This is an optional entry field.
- I_TAB_KF_FLAT / I_STR_KF_FLAT: Key figures of a result object. This is an optional entry field. If you enter a key figure, the corresponding (if there is one) unit field (for example, currency) must also be populated.

EXPORTING variables of the generated RFC interfaces

- /BA1/HK_TAB_RETURN: Reporting table derived from structure BAPIRET2. The table contains details about error messages (in the case of bulk errors with a reference to the corresponding result object via internal index OBJ_ID).
- /BA1/HK_TAB_RETURN_HEAD: Reporting table (only for bulk messages) derived from structure BAPIRET2. The table contains messages about the general status of an upload (successfully completed, partially completed, not completed).

Fixed RFC interfaces

As well as the generated interfaces, the RDL of Bank Analyzer contains two fixed RFC interfaces. These are the function modules for changing the status of result groups that are used in the RDL for the versioning [page 1378] of results.

- /BA1/IN_HK_REVOKE_S, Single record processing - Status change of result groups via RFC
- /BA1/IN_HK_REVOKE_M, Mass processing - Status change of result groups via RFC

A result group can have the following statuses:

- Released
- Not released
- Deleted

The system returns error messages in the form of a table in the /BA1/HK_STR_RETURN format. The application transfers the messages here.

IMPORTING variables of the RFC interface /BA1/IN_HK_REVOKE_S

- I_FLG_COMMIT: Controls whether the function module runs its own Commit Work or not. If the initial value is BLANK, no Commit Work is run. If 'X' (abap_true) is entered, a Commit Work is run. All other entries work in the same way as BLANK.
- I_STR_RES_GRP: This is a structure for selecting the result group(s). All fields must be populated. Contains the fields:
- AREA: Results data area
- GROUP_TYPE: Result group type
- KEYDATE: Key date. Entry format is DDMMYYYY

IMPORTING variables of the RFC interface /BA1/IN_HK_REVOKE_M

- I_FLG_COMMIT: Controls whether the function module runs its own Commit Work or not. If the initial value is BLANK, no Commit Work is run. If 'X' (abap_true) is entered, a Commit Work is run. All other entries work in the same way as BLANK.

- I_TAB_RES_GRP_M: This is a table for selecting the result group(s). All fields must be populated. The table contains the following fields:
  - OBJ_ID: Index. Links result group selection with status
  - AREA: Results data area
  - GROUP_TYPE: Result group type
  - KEYDATE: Key date. Entry format is DDMMYYYY

The selection table entries are assigned to the status table entries using index (sequence number) OBJ_ID.

Activities

You generate RFC interfaces for result types of result category *Generic Key Date Results* in Customizing for Bank Analyzer under *Bank Analyzer* > *Results Data Layer* > *Basic Settings* > *Edit Technical Settings in Results Data Area*. On the input screen, first select the results data area and then, in the dialog menu, choose *RFC: Generation of Interfaces*. The system displays the available flat generic key date types for the selected results data area.

5.3.1.2 Application-to-Application (A2A) Interfaces

Use

The Results Data Layer (RDL) provides application-to-application (A2A) interfaces as well as RFC interfaces. These A2A interfaces are based on SAP NetWeaver Exchange Infrastructure (XI).

> **Note**
>
> For a complete overview of the available enterprise services in the Results Data Layer, see http://help.sap.com/bankingservices. Choose *Integration & Analytics Information*, open the documentation and search for the term *Financial Instruments Analytical Results Management*.

Prerequisites

To use the A2A interfaces, you need to make settings in Customizing for the XI Structure Mapping [page 1735] tool. Note that mapping rules for some Bank Analyzer target fields are shipped directly as system tables [page 1405].

The Results Data Layer provides a BAdI for changing or adding to the data when an RDL result is imported after structure mapping. You can activate the SAP standard BAdI implementations or create your own implementations. You can find the BAdI definition `/BA1/SI_HX_AFTER_MAPPING` in Customizing for *Bank Analyzer* under *Results Data Layer* > *Settings for PI Interfaces* > *BAdI: RDL Inbound After PI Structure Mapping to RDL Data Model*.

Features

The A2A interfaces are provided as asynchronous or synchronous inbound and outbound XI message interfaces (services).

Each A2A interface has a corresponding XI message category that is used as an inbound message.

With asynchronous messages, the sender is usually not informed about whether the messages were successfully or incorrectly processed, unless the system sends a confirmation message. In addition, you can use the application log for information about processing.

As well as generating simple logs, the application log also saves the affected XI messages if an error occurs. The IDs of the documents concerned could then, for example, be exported in order to be sent again.

Activities

You can set up structure mapping for A2A interfaces in Customizing for Bank Analyzer under *Bank Analyzer* > *Infrastructure* > *Settings for XI Services* > *XI Structure Mapping*.

You can make additional Customizing settings for the derivation strategy for XI structure mapping. You do this in Customizing for Bank Analyzer under *Infrastructure* > *Settings for XI Services* > *Derivation Strategy for XI Structure Mapping*.

You can make specific settings for using the XI services in Customizing for Bank Analyzer under *Results Data Layer* > *Settings for XI Interfaces*.

Example

For more information, see Example of an XML Message [page 1402].

5.3.1.2.1 Example of an XML Message

```
<?xml version="1.0" encoding="UTF-8"?>
<FinancialInstrumentLedgerDocumentNotification>
<MessageHeader>
<ID>HP_MSG_001</ID>
<!--
Creation date of the XML document, not to be confound with the posting date
-->
```

<CreationDateTime>2005-12-20</CreationDateTime>

</MessageHeader>

<FinancialInstrumentLedgerDocument>

<ID schemeAgencyID="HPSL_001">DOC_HP_010</ID> <SetOfBooksCode>IAS</SetOfBooksCode>
<GroupID>HP</GroupID>

<CompanyID>SLE0</CompanyID>

<PostingDate>2005-12-15</PostingDate>

<!--

To be filled for YEC postings only

-->

<YearEndClosingFiscalYearValue />

<Item>

<ID>1</ID>

<GeneralLedgerAccountID>ACC_01</GeneralLedgerAccountID>

<GeneralLedgerMovementTypeCode>01</GeneralLedgerMovementTypeCode>

<!--

1 = debit, 2 = credit

-->

<DebitCreditCode>1</DebitCreditCode>

<FinancialInstrument>

<ID schemeAgencyID="BA">FTRAN</ID>

<!--

S = standardized, N = non-standardized

-->

<CategoryCode>0</CategoryCode>

<AccountID>SACC</AccountID>

</FinancialInstrument>

<!--

Replace "&" by "&" in keyfigure name

-->

<PostingKeyFigure>

<FinancialInstrumentPostingKeyFigureCode>&55LCPPO</FinancialInstrumentPostingKeyFigureCode>

```
<FinancialInstrumentPostingKeyFigureValue>
<Amount currencyCode="EUR">51</Amount>
</FinancialInstrumentPostingKeyFigureValue>
</PostingKeyFigure>
<!--
You can add as many properties as you want, compare to customizing
-->
<Property>
<ID schemeAgencyID="0816">&55AMNOM</ID>
<Value>
<AmountSpecification>
<Amount currencyCode="EUR">99000</Amount>
</AmountSpecification>
</Value>
</Property>
</Item>
<Item>
<ID>2</ID>
<GeneralLedgerAccountID>ACC_02</GeneralLedgerAccountID>
<GeneralLedgerMovementTypeCode>01</GeneralLedgerMovementTypeCode>
<!--
1 = debit, 2 = credit
-->
<DebitCreditCode>2</DebitCreditCode>
<FinancialInstrument>
<ID schemeAgencyID="BA">FINST</ID>
<!--
S = standardized, N = non-standardized
-->
<CategoryCode>1</CategoryCode>
<AccountID>SPACC</AccountID>
</FinancialInstrument>
```

<!--

Replace "&" by "&" in keyfigure name

-->

<PostingKeyFigure>

<FinancialInstrumentPostingKeyFigureCode>&55LCPPO</FinancialInstrumentPostingKeyFigureCode>

<FinancialInstrumentPostingKeyFigureValue>

<Amount currencyCode="EUR">-51</Amount>

</FinancialInstrumentPostingKeyFigureValue>

</PostingKeyFigure>

<!--

You can add as many properties as you want, compare to customizing

-->

<!--

Property> <ID schemeAgencyID="0815">&55QUANT</ID> <Value> <QuantitySpecification> <Quantity unitCode="MHZ">-20000</Quantity> </QuantitySpecification> </Value> </Property

-->

</Item>

</FinancialInstrumentLedgerDocument>

</FinancialInstrumentLedgerDocumentNotification>

5.3.1.2.2 Mapping Rules for A2A Services

To use the A2A interfaces, you need to make settings in Customizing for the XI Structure Mapping [page 1735] tool. Note that mapping rules for the Bank Analyzer fields of the result type are already delivered in system tables. These predefined rules are valid for all result types and require no modification (even though you could theoretically overwrite them by means of mapping rules in Customizing).

You enter the required settings in Customizing for *Bank Analyzer* under *Infrastructure* > *Settings for Enterprise Services* > *PI Structure Mapping* > *Edit PI Structure Mapping*.

5.3.2 User Interface

Definition

A user interface is used to exchange information between two programs, or applications. At the same time, it is the visible representation of a software program seen by users on their screens. For communication with the Results Data Layer (RDL), the user interfaces are predefined in the system. They can be configured to suit the customers' needs.

For more information, see User Interfaces.

Use

You use a user interface to start and manage transactions. There are transactions for user interfaces in the Results Data Layer (RDL) in the following areas:

- Reverse Subledger Documents [page 1406]
- Displaying Results Data [page 1407]
- Preaggregation [page 1408]

5.3.2.1 Reverse Subledger Documents

Context

If you have transported incorrect data into your system, you can reverse this data directly from the system without using Enterprise Services.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Results Data Layer* > *Import of Subledger Documents* > *Reverse Subledger Documents*.
2. Enter the required data and choose *Execute*.

Next Steps

Application-to-Application (A2A) Interfaces [page 1401]

5.3.2.2 Displaying Results Data

Use

You can use this transaction to display results data in an SAP List Viewer list (ALV list).

Prerequisites

In Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Results Data Area*, you have defined a results data area and a result group type.

In Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*, you have defined a result view and a filter.

To enter results data in an SAP List Viewer list, data must have been generated.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Results Data Layer* > *Display Results Data*.
2. Select a results data area and a result view.
3. For one-dimensional versioning, select a result status.
4. For two-dimensional versioning, define the *Key Date To*. For result group versioning, define the *Key Date From*, *Key Date To*, *Result Group Type*, *Result Group*, and *Status*. To get intraday results, select the *All Versions in Range* checkbox.
5. Enter the system time, the scenario, the filter, and the package size.
6. Save your entries in case you want to continue processing a variant of the results data at a later point in time.

You can make further restrictions at characteristic level (depending on the result view).

Result

The system generates an SAP List Viewer list that displays the results in view format. Hierarchical data structures are then shown in a denormalized view. The list supports the usual options for navigating when displaying details for referenced objects (for example, master data) within the Bank Analyzer environment.

By double-clicking a line of the ALV list, you can navigate to a detail view of the result, which provides a hierarchical representation of the result.

> **Note**
>
> If this is a flow result, you can display the related business transaction from the list. If this is an aggregation accrual result, you can display the individual accrual results from the list. For more information, see Aggregating Accrual Results [page 308].

5.3.2.3 Preaggregation

Use

You can use preaggregation in the Results Data Layer (RDL) to optimize the performance of aggregation queries.

Features

The preaggregation function creates aggregated results for an aggregation level and a filter, and saves them permanently on the database. If the application queries an aggregate value, saved, preaggregated data and newer data from the underlying result view is merged to provide up-to-date aggregated results (delta aggregation). You can run the preaggregation in parallel. Schedule preaggregation to run at the end of the business process chain at regular intervals.

> **Note**
>
> SAP HANA allows you to directly aggregate data very quickly. Therefore, when you use SAP HANA we recommend that you schedule preaggregation for generic aggregation levels (initial level category) at longer intervals and that you do not use preaggregation for aggregation levels of the level categories *Balance Level* (`BALLV`) and *Horizontalization Level* (`HORIV`).

> **Note**
>
> If you use data aging, note that the system takes into account the residence time **and** the corresponding preaggregation runs when it selects RDL business transactions during the data aging run. In the job administration, you therefore need to schedule the preaggregation runs for business transactions **before** the corresponding data aging runs. You use the data aging selection ID to ensure that the same data records are selected in the preaggregation run as those selected for the subsequent data aging run.
>
> For more information about the dependencies between preaggregation and data aging, see the documentation for the data aging object RDL: Business Transaction [page 1748].

You can use the run administration function within preaggregation to delete terminated, inconsistent runs and the corresponding data to allow preaggregation to be restarted. When you choose a run, the detail display of the run data is shown. You can delete the run and the aggregated data here. In addition, you can display the aggregated data of the run.

> **Note**
>
> If you are using data aging, you can only delete preaggregation runs if there is no corresponding data aging run. If a data aging run has already been executed the preaggregate should also be moved to the cold area.

You can only make unrestricted changes to the granularity of an aggregation if no preaggregates have been created. If preaggregates already exists, you have the following options:

- You can make the granularity coarser by creating a second aggregation level on the existing level and omitting certain granularity characteristics.

- You can archive or delete the preaggregate data in all systems. You then adjust the Customizing for the aggregation levels and create new preaggregates.
- You can define new aggregation levels and create preaggregates for these. In Customizing, replace all references to the old aggregation levels with references to the new aggregation levels.

The prerequisite for the last two options is that the data to be aggregated is still available in the system.

Activities

You start preaggregation or run administration on the *SAP Easy Access* screen by choosing *Bank Analyzer* > *Results Data Layer* > *Pre-Aggregation*.

Transaction	Use	Required Entries
Run Preaggregation	Preaggregation is started.	• Results Data Area • Aggregation level • Filter • Data aging selection ID • System Time • Scenario • Key Date • Result Group • Technical settings for parallel processing
Run Administration	You can display runs.	Selection of one or more runs that are to be used in the run administration process.

If a run is not successful and you wish to repeat it, you first have to save the results data of the run as a variant. Then delete the data and recall the transaction. Enter the data saved for the variant and begin the rerun.

You cannot delete the run itself. If you want to display the rerun, select a run in run administration and, in the dialog structure, double-click on *Reruns*. The system displays all reruns.

6 Analytics (FS-BA-AN)

Use

This component contains analytical applications that call results data for Processes and Methods [page 366] from the Results Data Layer (RDL) [page 1364] and, if necessary, continue to process this data.

For example, the Historical Database gets data from the Source Data Layer (SDL) [page 100] and processes it for storage based on time series as defined under Basel II.

Features

Components Relevant for Accounting

General Ledger Connector (FS-BA-AN-GL)

If you use the subledger scenario, the General Ledger Connector [page 1411] reads the subledger documents from the RDL and transfers this results data to a connected general ledger.

Financial Statement Preparation (FS-BA-AN-FSP)

Financial Statement Preparation [page 1425] includes Balance Object Manager, Balance Processing, and Aggregated Transactions. In Balance Object Manager you create balance objects (BO) that define the processing level for processes in Balance Processing, in particular the object that is to be included in reporting. Balance Processing loads results data from the RDL and prepares the period-end processing for financial products, such as the balance sheet and income statement including notes to the financial statements.

Components Relevant for Basel II

Historical Database (FS-BA-AN-HDB)

The Historical Database [page 1478] is a time-based data store and meets the Basel II requirements for managing historical data. The system can provide the HDB with data from the Source Data Layer (SDL), RDL, or another source system.

Disclosure and Reporting (FS-BA-AN-DR)

The Disclosure and Reporting [page 1527] component provides utilities for selecting and extracting reporting data and meets Basel II requirements of the Capital Accord. The Disclosure and Reporting component supports external disclosure and internal reporting, and provides support for supervisory investigations and stress test reports.

Additional Components

Limit Manager (FS-BA-AN-LM)

Limit Manager [page 1529] provides support when determining, analyzing, and limiting counterparty/issuer risks, country risks, or Basel II-specific key figures. Banks set different maximum risk amounts in order to limit the potential harm caused by the insolvency of a business partner. Limit Manager also provides operational functions and supports both internal and external reporting.

6.1 General Ledger Connector (FS-BA-AN-GL)

Use

This component aggregates the subledger documents from the Results Data Layer (RDL) [page 1364]. It sends the results daily to a general ledger by means of the outbound service in the Service Outbound Layer [page 1414].

Integration

Relationship to Other Components and Functions

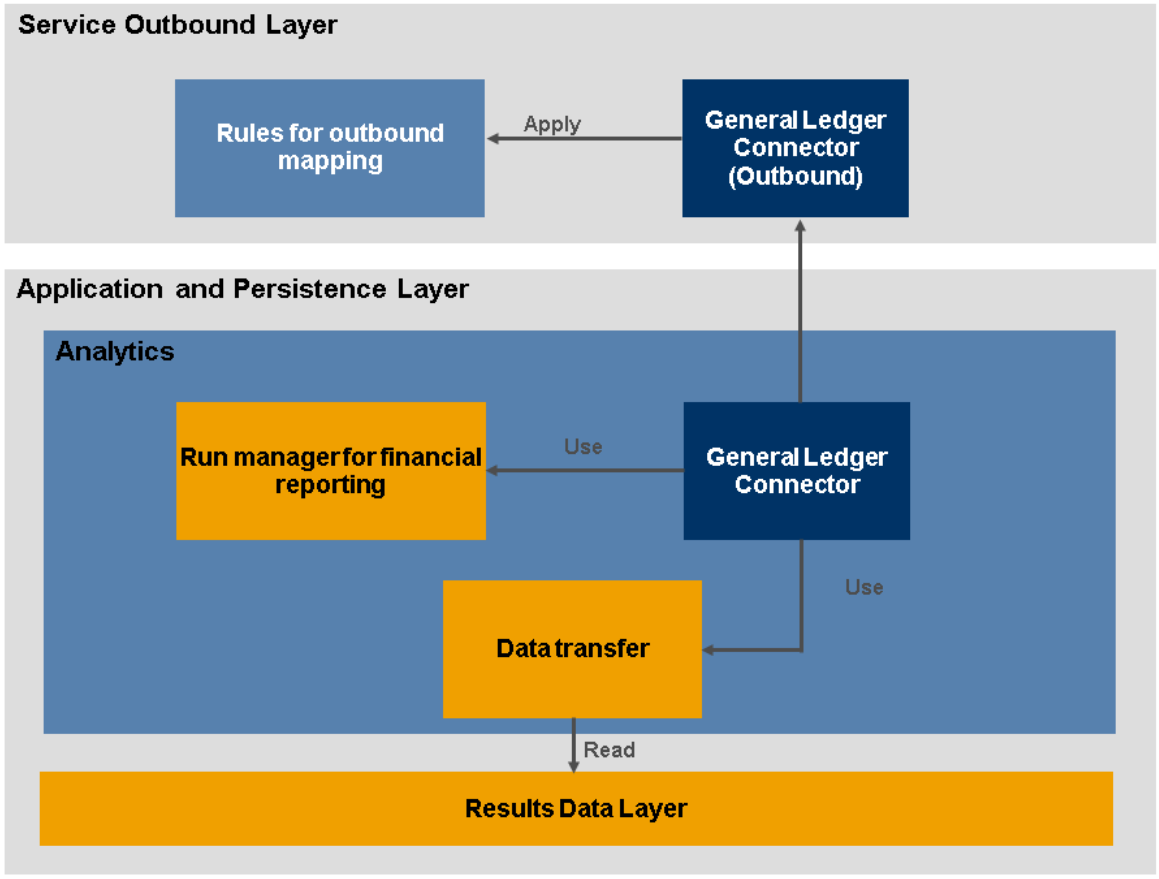
The General Ledger Connector is part of the Analytics [page 1410] component.

It requires data from the flow results and the characteristic results in the RDL.

The General Ledger Connector uses the Run Manager for Financial Reporting [page 1474].

To send data to the general ledger, the General Ledger Connector uses the Service Outbound Layer [page 1414] functions.

The following figure illustrates the relationships between the General Ledger Connector and other components and functions.



Customizing

As well as its own Customizing settings, the General Ledger Connector requires the settings for outbound mapping in Customizing for *XI Structure Mapping*.

- To view the Customizing activities that are relevant to the General Ledger Connector, see *Bank Analyzer* > *Analytics* > *General Ledger Connector*.
- To view the Customizing activities that are relevant for outbound mapping, see *Bank Analyzer* > *Infrastructure* > *Settings for XI Services* > *XI Structure Mapping*.

Features

The General Ledger Connector contains the following functions:

- Preparation of general ledger document [page 1415]
  This function uses the result data from the aggregation in the RDL to create the general ledger documents. It also includes a consistency check for the general ledger documents and storage of the general ledger documents.
- Sending of general ledger documents [page 1418]
  The General Ledger Connector enables the daily supply of general ledger documents to the general ledger.
- Resending of general ledger documents when errors have occurred [page 1418]
- Display of general ledger documents [page 1417] using the drilldown function, in which you can display all the related subledger documents in the RDL for the aggregated general ledger document.

6.1.1 General Ledger Document

Definition

Document in which postings from subledger documents are aggregated. You can display the relevant subledger documents from the Results Data Layer (RDL) [page 1364] for a general ledger document.

The General Ledger Connector [page 1411] enables the daily supply of general ledger documents to the general ledger.

Structure

A general ledger document always consists of two items: the posting itself and the technically generated offsetting entry. The total of both postings must be zero.

The system assigns a unique document number when creating the general ledger document. The general ledger document receives another unique number for the drilldown function from the general ledger to the general ledger document (that meets the requirements of the FI general ledger). This is the reference number.

The G/L account created under *Processes and Methods* is read from the RDL. The fiscal year and fiscal period are also read from the RDL.

The example below lists all entries in the general ledger document.

In the General Ledger Connector, you can also use the principle of dual currency accounting for the accounting documents. The system then saves externally provided subledger documents in the RDL. The documents have an transaction currency and an equivalent value in the functional currency. This currency accounting matches the accounting in the general ledger.

Integration

Data transfer [page 1438] delivers the data from the RDL for the general ledger document, such as the G/L account.

When the general ledger documents are created and sent, the run manager for financial reporting creates runs. You can display details about the runs and the relevant general ledger documents in the run manager for financial reporting (see Display Runs for Financial Reporting [page 1420]).

Example

This example shows a general ledger document for legal entity AP02 and accounting system S_IAS:

Posting Date	Number of G/L Doc.	Reference Number of G/L Doc.	G/L account	Credit/Debit Indicator
20031231	000000000017	00000000000000001335	9924400000	C
		00000000000000001335	9929030000	D

Amount in Transaction Currency	Transaction Currency	Amount in Functional Currency	Functional Currency	Amount in Local currency	Local Currency	Amount in Group Currency	Group Currency	Amount in Hard Currency	Hard Currency	Amount in Index-Based Currency	Index-Based Currency	Legal Entity
778,63-	USD	500,75-	EUR	500,75-	EUR	439,49-	GBP					AP02
778,63	USD	500,75	EUR	500,75	EUR	439,49	GBP					AP02

G/L Doc. Status	Time Stamp	G/L Value Type	Item No. in G/L Doc	Functional Currency

D	2006020109411900000000	A	1	EUR
D	2006020109411900000000	A	2	EUR

Chart of Accounts	Fiscal Year	Period and Fiscal Year
BK00	2004	2004000
BK00	2004	2004000

See also:

- Preparation of General Ledger Document [page 1415]
- Displaying General Ledger Documents [page 1417]
- Sending of General Ledger Documents [page 1418]

6.1.2 Service Outbound Layer

Use

The General Ledger Connector can use this function to send, using mapping rules, general ledger documents to the general ledger. At a different point in time, the data is transferred via the SAP NetWeaver Exchange Infrastructure (XI).

Prerequisites

- In Customizing, you have defined the outbound mapping under *Bank Analyzer* > *Infrastructure* > *Settings for XI Services* > *XI Structure Mapping*.
- In Customizing for the *General Ledger Connector*, you have assigned a mapping rule set to Accounting. To do this, choose Customizing *Bank Analyzer* > *Analytics* > *General Ledger Connector* > *Document Sender* > *Assign Rule Sets for Mapping to Accounting Systems*.

Features

The outbound function of the General Ledger Connector is part of the service outbound layer. This function converts the data of the General Ledger Connector according to the mapping rules that are defined in outbound mapping. It then sends the data via XI to the general ledger.

The rule set for outbound mapping is the central mapping component of the service outbound layer. This rule set defines how fields are converted.

Example

Bank Analyzer uses the *Legal Entity* field. However, the legal entity concept is not recognized outside of Bank Analyzer. For this reason, mapping rules in outbound mapping convert this field into *company code*.

See also:

- General Ledger Connector (FS-BA-AN-GL) [page 1411]

6.1.3 Preparation of General Ledger Document

Use

You use this function to create general ledger documents [page 1412] . To create general ledger documents, you extract the required data from the Results Data Layer (RDL) [page 1364] . This function checks the consistency of the general ledger documents and saves them in the persistence layer.

Integration

This function is part of the General Ledger Connector [page 1411] component. You have to execute it to create general ledger documents before you can send [page 1418] and display [page 1417] them.

Prerequisites

You have created document number ranges. For more information, see the IMG activities in Customizing under *Bank Analyzer Analytics General Ledger Connector Document Manager Basic Settings for Number Ranges*

When you prepare general ledger documents in simulation mode, use transaction SNRO and object /BA1/B8GL2 to create number ranges.

For document reference numbers, use transaction SNRO and object /BA1/B8GLR.

Note that you have to save the document number range in two steps. First, choose *Save* on the *Maintain Number Range Intervals* screen. Then choose *Save* again on the *Edit Number Range of G/L Document* initial screen.

You have defined derivations for clearing accounts in Customizing for the *General Ledger Connector* . It must be possible to determine a clearing account that is also a general ledger account for each data row returned by the Results Data Layer (RDL). Define the derivation rules flexibly so that any number of characteristics can be used and any of their values can be compared to determine the clearing account. Derivation can be time-dependent,

and you can define a default value for the derivation. For more information, see the Implementation Guide (IMG) under *Bank Analyzer* → *Analytics* → *General Ledger Connector* → *Document Preparation* → *Derivation of Clearing Accounts*

You have assigned a primary data source to the accounting system and legal entity in Customizing for the *General Ledger Connector* . For more information, see the Implementation Guide (IMG) under *Bank Analyzer* → *Analytics* → *General Ledger Connector* → *Document Preparation* → *Assign Primary Data Sources*

Features

When general ledger documents are created, the function:

Checks whether the primary data source assigned to accounting and the legal entity in Customizing is defined with the correct result view and whether it contains the fields required for aggregation.

Extracts the data from the RDL and enters it in the corresponding fields (*Accounting S ystem Legal Entity* , *Amount* , *Currency* , and a table containing characteristics and their values).

Checks the consistency of Customizing for the General Ledger Connector. If an error occurs, the system terminates the action.

Generates a general ledger document for each aggregated results row from the RDL and checks the total of the clearing accounts. The total must be zero.

Sets a status for the run.

If a document for the run contains errors, the system reverses the other documents for the run.

Saves the general ledger documents in the persistence layer.

Activities

On the *SAP Easy Access* screen, choose *Bank Analyzer* → *Analytics* → *General Ledger Connector* → *Prepare General Ledger Document.*

Specify the *Accounting Sy stem* and *Legal Entity* for which you want to generate general ledger documents.

Enter an end date.

If required, activate one of the technical options.

If you want to carry out a test run without writing data to the database or sending the general ledger documents to the general ledger, select the *Simulation Run* indicator.

You can display the general ledger documents created in the simulation run using General Ledger Connector Display [page 1417] or in the Run Manager for Financial Reporting [page 1474] .

- If you want the system to proceed without the parallel processing tool, select the *Execute Program Sequentially* indicator. We recommend you use this setting when only a small number of objects is to be processed, or when you need to find errors.

Choose *Execute*

The system executes the functions described in *Features*

6.1.4 Displaying General Ledger Documents

Use

This function displays the general ledger documents [page 1412] of a financial reporting run for a combination of legal entity and accounting system. The display is prepared according to the selection parameters entered by the user.

You can use a drilldown function to display for the selected general ledger document all relevant subledger documents from the Results Data Layer (RDL) [page 1364].

Integration

You can also call the display of the general ledger documents in the run manager for financial reporting [page 1474]. You have already called a general ledger run. Choose *Display General Ledger Documents*.

Prerequisites

You have generated general ledger documents for the specified combination of legal entity and accounting system (see Preparation of General Ledger Document [page 1415]).

Activities

Display General Ledger Documents

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *General Ledger Connector* > *Display General Ledger Connector*.
2. Specify the legal entity and the accounting system.
3. Enter at least one date or one date range.
4. Once you have entered a date range and confirmed your selection with *Enter*, you can also select from a list of characteristics. The characteristics displayed correspond to those that have been assigned to the general ledger documents in Customizing for *Bank Analyzer* under *Analytics* > *General Ledger Connector* > *Document Manager* > *Define Characteristics for General Ledger Documents*.
5. You can enter more selection criteria to further restrict the amount of general ledger documents selected.
6. Choose .

Display Subledger Documents (Drilldown Function)

1. In the *Overview of General Ledger Documents*, select the general ledger document for which you want to display subledger documents.
2. Choose  .
   The system displays all subledger documents from the RDL that have been aggregated for the selected general ledger document.

6.1.5 Sending of General Ledger Documents

Use

You can use this function to send general ledger documents to the general ledger via the SAP NetWeaver Exchange Infrastructure (XI) interface. To do so, you must first enter the appropriate accounting system and legal entity.

Integration

To send data to the general ledger, the function uses the outbound part of the Service Outbound Layer [page 1414].

In the run manager for financial reporting [page 1474], you can display the runs for sending general ledger documents.

Prerequisites

- You have generated general ledger documents for the specified combination of legal entity and accounting system (see Preparation of General Ledger Document [page 1415]).
- In Customizing for *XI Structure Mapping*, you have made the settings required for outbound mapping. To view the Customizing activities that are relevant for outbound mapping, see *Bank Analyzer* > *Infrastructure* > *Settings for XI Services* > *XI Structure Mapping*.
- In Customizing, you have assigned the accounting system a set of rules for outbound mapping under *Bank Analyzer* > *Analytics* > *General Ledger Connector* > *Document Sender* > *Assign Rule Sets for Mapping to Accounting Systems*.
- In Customizing, you have specified whether the system is to send the documents using single messages or collective messages under *Analytics* > *General Ledger Connector* > *Document Sender* > *Assign Message Dispatch Categories for G/L Documents* (optional). If you do not configure settings in this Customizing activity, the system sends single messages by default. By selecting collective messages, you can improve the performance of the General Ledger Connector.

Activities

Send General Ledger Documents to the General Ledger

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *General Ledger Connector* > *Send General Ledger Documents to General Ledger*.
2. Specify the *legal entity* and the *accounting system*.
3. If required, activate the technical options:
   - If you want to send to the general ledger only the general ledger documents from a particular run, choose *Manual Run Selection*.
   - If you want to carry out a test run without sending data to the general ledger, select the *Simulation Run* checkbox.
   - If you want the system to proceed without the parallel processing tool, select the *Execute Program Sequentially* checkbox. We recommend you use this setting when only a small number of objects is to be processed, or when you need to find errors.
4. Choose *Execute*.
   The system displays a detailed application log.
5. If errors have occurred, correct the errors and start the *Resend General Ledger Documents to the General Ledger* transaction.

Resend General Ledger Documents to the General Ledger

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *General Ledger Connector* > *Resend General Ledger Documents to General Ledger*.
2. Specify the *legal entity* and the *accounting system*.
   Usually, the legal entity and accounting system for the run that contains errors are already entered.
3. If required, activate the technical options:
   - If you want to send to the general ledger only the general ledger documents from a particular run, choose *Manual Run Selection*.
   - If you want to carry out a test run without sending data to the general ledger, select the *Simulation Run* checkbox.
   - If you want the system to proceed without the parallel processing tool, select the *Execute Program Sequentially* checkbox. We recommend you use this setting when only a small number of objects is to be processed, or when you need to find errors.
4. Choose *Execute*.
   The system displays a detailed application log.
5. If errors occur again, attempt to correct these and start the *Resend General Ledger Documents to General Ledger* transaction again.

> **Note**
>
> If you start the run again, select the *Execute Program Sequentially* checkbox to find errors.

6.1.6 Display Runs for Financial Reporting

Use

You can use this function to display runs for financial reporting for a particular legal entity and accounting system. This function displays for each run the list of existing reference runs. It displays, for a run, the current time stamp and the time stamp of the run that preceded the last run, as well as administrative information for the run.

Features

You can use this function to display runs for the following applications:

- General ledger connector [page 1411]
  General ledger connector (GLC) runs include the preparation of general ledger documents [page 1415] and the sending of general ledger documents [page 1418] to the general ledger. In the run manager, you can also display the general ledger documents [page 1412] for these runs. In the overview of the general ledger documents you can use the drilldown function; this function allows you to display, for the general ledger document, all relevant subledger documents from the RDL.
- Balance processing [page 1430] and aggregated transactions [page 1472]
  For balance processing runs, the run manager for financial reporting displays the previous run and the relevant general ledger connector and balance object runs.
  Aggregated transactions do not comprise an independent process. They are embedded in balance processing.
- Balance object processing [page 1425]
  For the current run, the run manager for financial reporting also displays the previous run.

For more information about the features of the run manager, see Run Manager for Financial Reporting [page 1474].

Activities

1. To display runs, you can start the run manager for financial reporting using one of the following methods:
   - Choose *Bank Analyzer* > *Analytics* > *General Ledger Connector* > *Start Run Manager for Financial Reporting*
   - Choose *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Tools* > *Start Run Manager for Financial Reporting*
2. If you want to display details for a particular run, and you know its run ID, then enter the *Run ID*.
3. Enter a *Legal Entity* and an *Accounting System*. These are mandatory entries.
4. If you want to display runs for a particular process, choose the appropriate *Process Type*.
   For the aggregated transactions, choose the process type *Balance Processing*.
5. You can further restrict your selection by entering the *User Name* of someone who has executed the run that you want to display. If you know it, enter the *Financial Reporting ID*.
6. Enter a time period for the runs.
7. Choose *Execute*.

6.1.7 Handling of Value-Added Tax

Use

The system can process the VAT information provided in the accounting processes and create the corresponding VAT documents in the General Ledger Connector.

In some countries, the state claims VAT for certain bank income such as interest or fees. You can also use this function for other reasons if you knowingly want to take VAT into account, for example, to reduce the input tax.

Prerequisites

To handle the VAT, the following prerequisites must be met:

- The feeder systems and the general ledger can calculate and process the VAT. The tax code, transaction key, and tax items in particular have the same format as in the General Ledger Connector.
- You have made the required entries for VAT in Customizing for *Financial Services* under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Basic SettingsBusiness Transaction Types and Item Types*. You have created a business transaction type and an item type to handle VAT.
- You have configured characteristic and key figure mapping from the business transaction to the internal structure in Customizing for *Financial Services* under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of External Business TransactionsEdit Characteristic Mapping for Item Fields of SDL Business Transactions*. You have also taken into account the characteristics and key figures specific to VAT, as well as the transaction currency (`/BA1/K55AMTRC`).
- You have created or extended a calculation procedure to handle VAT in Customizing for *Financial Services* under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Update of Secondary Business Transactions* > *Accounting ProcessesElementary Calculation Procedures*. You can use the templates `0P43` and `0Q41`, as well as the step categories `0ST1`, `0ST2`, `0ST3`, and `0ST4`.
- You have created a calculation procedure to handle VAT in Customizing for *Financial Services* under *Bank Analyzer* > *Processes and Methods* > *Accounting for Financial Products* > *After Generation* > *Financial Position Processes* > *Processing of Internal Business Transactions* > *Basic Settings* > *Calculation Procedures*.
- In Customizing for *Financial Services* under *Bank Analyzer* > *Results Data Layer* > *Basic Settings* > *Edit Results Data Area* you have defined the characteristic /BA1/C55VATIND and VAT-specific characteristics and key figures for the credit/debit components of a result type for cash flow results, as well as the tax base amount (in transaction currency /BA1/K55TAXBO, in functional currency /BA1/K55TAXBR, and in local currency /BA1/K55TAXBL). You need to enter the tax base amount in local currency only if the local currency is not the same as the functional currency.
- When you assign the results views, you have selected the checkbox for VAT relevance (where necessary) in Customizing for *Financial Services* under *Bank Analyzer* > *Analytics* > *General Ledger Connector* > *Document Manager* > *Define Characteristics for General Ledger Documents*.

- If required, you have configured the settings for grouping tax items in Customizing for *Financial Services* under *Bank Analyzer* > *Analytics* > *General Ledger Connector* > *Document PreparationHandling of Value-Added Tax*.
  For more information, see the documentation for these Customizing activities.
- You have defined the required rule set as relevant for VAT in Customizing for *Financial Services* under *Bank Analyzer* > *Analytics* > *General Ledger Connector* > *Document Sender* > *Assign Rule Sets for Mapping to Accounting Systems*. In this Customizing activity, you have also defined the characteristics and key figures that are relevant for VAT.
- You use the bulk service.

Integration

The following figure provides an overview of the functions for handling VAT with regard to feeder systems and the general ledger. The checkmarks indicate which component is responsible for which process.

You can reconcile data between the analytical banking application, the operational feeder system, and the general ledger. For example, you can reconcile the following:

- Business transactions from the source data layer (SDL) in comparison to the business transactions from the feeder system (individual objects)
- General ledger documents from the General Ledger Connector in comparison to the general ledger documents from the general ledger that is connected (totals)

To do so, you can use the reconciliation functions from SAP NetWeaver Business Warehouse (BW). For more information, see Reconciliation. You can use a server provided for the primary data source (GDS server). The actual configuration for reconciling VAT data is, however, not provided but depends on your Customizing settings.

Features

When external business transactions are posted, the system posts the corresponding business transactions from the source data layer (SDL), which contain a fee and the associated tax or tax only, for example. In this case, you can use the item templates `0IB2`, `0IB3`, `0IB4`, and `0IB5`. The business transaction typically contains the following characteristics and key figures relating to VAT:

- Tax country (`/BA1/C55TCNTRY`)
- Tax code (`/BA1/C55TCODE`)
- Reference tax code (`/BA1/C55TCODER`)
- Transaction key (`/BA1/C55ACCKEY`)
- Tax base amount in transaction currency (`/BA1/K55TAXBO`)
- Tax base amount in functional currency (`/BA1/K55TAXBR`)
- Tax base amount in local currency (`/BA1/K55TAXBL`)

The tax base amount in functional currency (`/BA1/K55TAXBR`) and in local currency (`/BA1/K55TAXBL`) is translated in Accounting.

The tax is calculated in a feeder system. Once the secondary business transactions have been updated, the system saves the calculated and posted information in the results data layer (RDL) in the form of aggregated flows and cash flows.

The system sends this information to the general ledger by means of VAT-specific general ledger documents. These are then included in the advance VAT return.

When VAT is handled, the General Ledger Connector process for preparing and sending documents changes.

VAT-specific information must be saved in the general ledger that is connected. To do this, the system must create the corresponding VAT-specific general ledger documents in the General Ledger Connector. A general ledger document that does not include VAT typically contains the following lines:

- G/L account: value from the aggregated RDL flow
- Clearing account: equivalent value

These general ledger documents do not allow the respective tax items (aggregated RDL flows) to be additionally included.

The system must therefore be able to compile these in a specific general ledger document. To enable this, you must configure the **grouping** settings for tax items in Customizing (see the *Prerequisites* section). Tax items are grouped by tax procedure. Depending on your Customizing settings, the system includes the base amount for which tax has been charged along with the actual tax. This may be necessary to process taxes with zero amounts. In the general ledger, it may also be necessary to group the base amount and include it in the line item.

The grouping process enables the system to create a VAT-relevant general ledger document, which contains separate lines for the grouped tax items.

To send these documents, a message dispatch category other than the one used for documents without tax information is required. When VAT is handled, you use the *collective message* message dispatch category.

This also impacts the service that the system uses to connect to the general ledger.

The **service for collective messages** (AccountingDocumentNotification) contains the following VAT-specific fields:

- Tax code
- Transaction key
- Reference tax code

The service for single messages remains unaffected by VAT handling since it is not used for this purpose.

Process

Preparing and Sending Documents

The General Ledger Connector carries out the following steps when VAT is handled:

1. The system automatically creates any general ledger documents that do not contain VAT-relevant items. It obtains the information from the aggregated RDL flows. Aggregated flows that contain additional tax information (that is, they contain additional line items) are temporarily saved in a buffer. The system determines whether a flow is relevant for tax from the following fields:
   - Tax code
     If this field is maintained, the aggregated flow refers to a tax item.
   - Reference tax code
     If this field is maintained, the aggregated flow refers to an amount that is relevant for tax, for example, the underlying fee.

   The system processes this initial step in parallel to create general ledger documents that are not relevant for tax and then sets a synchronization point.
2. If tax-relevant data has been saved in the buffer, the system carries out this additional step. It creates general ledger documents based on the data in the buffer and the settings for the grouping IMG activities (see the *Prerequisites* section). In this case, the system does not have to access the RDL data directly.

If a legal entity is marked as relevant for VAT in Customizing, the system carries out the following check when the document is prepared: It checks whether all tax-relevant characteristics and key figures are assigned to the primary data source, the RDL result view, and the defining characteristics (see *Prerequisites*).

Example

You post a fee of EUR 100 with 19% VAT. The system groups the country key (DE), base amount (EUR 100), tax code A1 (output tax), and transaction key MWS (VAT 19%).

It then sends a document to the general ledger, which contains a line with the specified information:

Tax code A1 (tax)	EUR 19
Reference tax code (base amount)	EUR 100

6.2 Financial Statement Preparation (FS-BA-AN-FSP)

This component includes the following components and functions for the preparation of balance sheets, profit and loss statements, and notes:

- Balance Object Manager [page 1425]
- Balance Processing [page 1430]
- Aggregated transactions [page 1472]
- Run manager for financial reporting [page 1474]

6.2.1 General Settings

In the following section, you will find the functions that you use for all components and processes involved in balance processing.

6.2.2 Balance Object Manager (FS-BA-AN-FSP-BO)

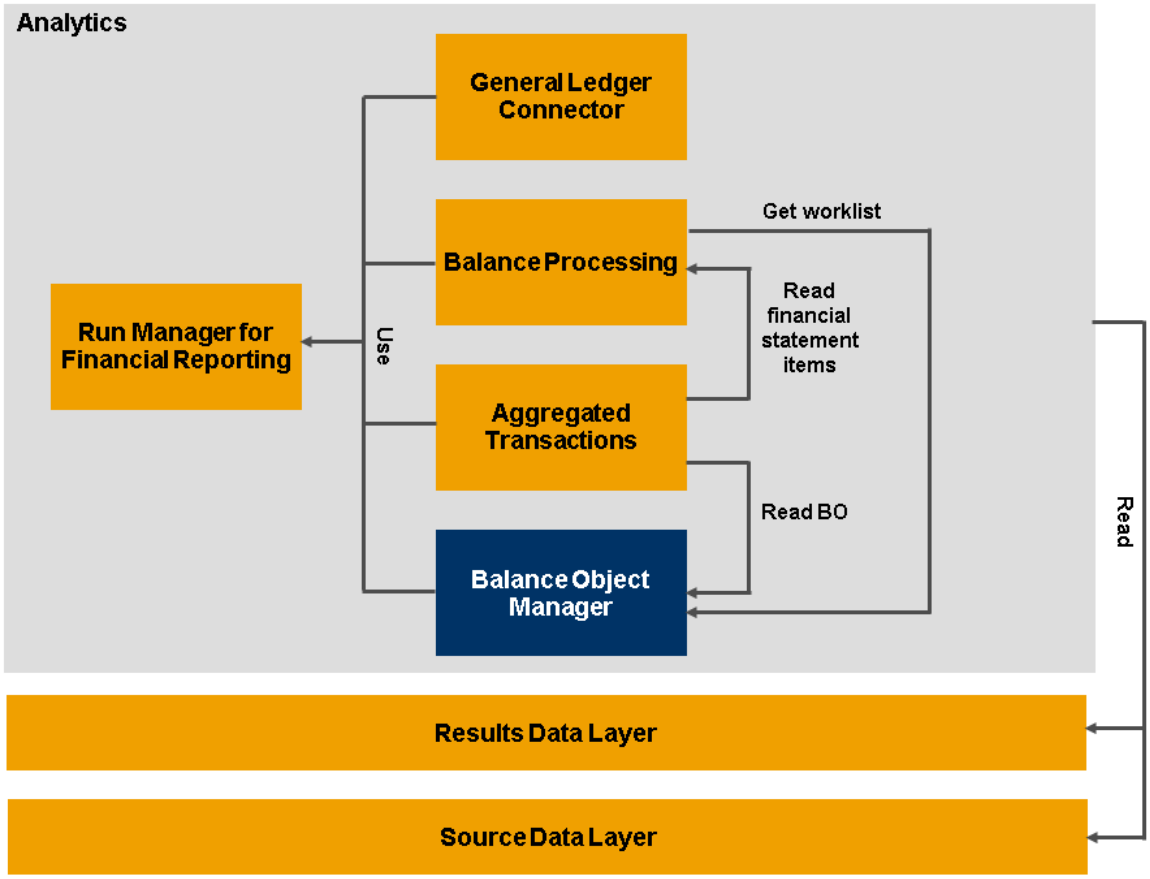
Use

This component includes all functions for editing balance objects and balance subobjects. It provides a list of balance objects and balance subobjects, which you use as a worklist for balance processing [page 1430] and aggregated transactions [page 1472].

You can use the functions of Balance Object Manager to edit the list containing the active balance objects and balance subobjects with their relevant characteristics and key figures.

Integration

Balance Object Manager is part of financial statement preparation [page 1425]. It provides the required worklist with balance objects for the Balance Processing and Aggregated Transactions components. The Balance Object Manager reads data from the Results Data Layer (RDL) [page 1364] and Source Data Layer (SDL) [page 100] to create worklists and determine data for the balance objects and balance subobjects. It also uses the run IDs from the run manager for financial reporting. The following graphic illustrates the relationships between Balance Object Manager and other components and functions.



Features

You can use Balance Object Manager to:

- Define balance objects and balance subobjects
  You define balance objects and balance subobjects in Customizing for *Balance Object Manager*. For more information, see Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Objects Manager*.
- Generate balance objects and balance subobjects
  Balance Object Manager generates balance objects (BOs) and balance subobjects (BSOs) by extracting data from the RDL.
  For more information about generating balance objects and balance subobjects, see Balance Object Creation [page 1427].
- Deactivate balance objects
  Balance Object Manager assigns a status to each balance object that indicates whether the balance object is active. Balance Object Manager automatically deactivates a balance object after one fiscal year if it is not required for the following fiscal year. It reactivates the balance object if posting is determined for the characteristic combination of the object.
  You can deactivate balance objects manually. For more information, see Deactivation of Balance Objects [page 1430].

- Store balance objects and balance subobjects
  Balance Object Manager uses the segmentation service to save balance objects and balance subobjects with their GUIDs and all defining characteristics with their corresponding values, including the active/inactive indicator. It also saves the time stamp that indicates when the BOs/BSOs were read from the RDL. The BO is saved with the first fiscal period.
- Generate and provide worklists
  Balance Object Manager generates worklists with balance objects for each type of balance processing: complete run, simulation run, and correction run. A worklist is always assigned to a legal entity, an accounting system, and a fiscal period.
  During this process, the system processes balance objects of category 5 (general financial position) first because this category contains a large amount of accounting documents.
- Store worklists
  Balance Object Manager saves the worklists for correction and simulation runs. In a correction worklist, it saves only the balance objects that have to be updated for the correction run. New balance objects generated are not saved here. They are read from the complete list of balance objects. All balance objects selected in the complete list for the simulation run are saved in the simulation list by Balance Processing.
- Determine GUID of balance subobjects
  Balance Object Manager receives the run ID from the processing of the aggregated transactions and a characteristic set with values as inbound parameters. It uses these entries to determine the relevant GUID for the balance subobject.
- Define balance objects
  Balance Object Manager receives the GUID of the balance object as an inbound parameter. It then uses the corresponding values of the balance object to determine the defining characteristics.
- Determine key dates of the balance objects
  Balance Object Manager receives the GUID of the balance object and a run ID as inbound parameters. The run ID contains the time stamp of the read operation from the RDL and the posting date. Balance Object Manager determines the defining characteristics of the balance object and their values. It also determines the balance subobjects of the balance object with their characteristics and key figures and the relevant values. If a balance subobject does not exist, Balance Object Manager generates it automatically.
- Display balance objects and balance subobjects
  You can display balance objects and balance subobjects, as well as the aggregated transactions. For more information, see Overview of Balance Objects [page 1464] and Process View of Balance Objects [page 1465].
- Archive balance objects and balance subobjects
  You can archive only inactive balance objects. A balance object is archived when all of its balance subobjects are archived. Balance Object Manager archives balance subobjects with all their properties.

6.2.2.1 Balance Object Creation

Definition

The Balance Object Manager generates balance objects (BO) and balance subobjects (BSO) [page 1432] from the flow data in the Results Data Layer (RDL). To extract characteristics with their characteristic values and the fiscal period as a granularity characteristic, the system uses data transfer [page 1438]. The Balance Object Manager generates a new BO and BSO for every combination of characteristic values. The system generates balance subobjects for every balance processing run (standard, correction, and simulation runs). A BSO is valid for the same period of time as its corresponding BO.

Customizing Settings

You have configured the Customizing activities in Customizing for Bank Analyzer under *Analytics* > *Financial Statement Preparation* > *Balance Objects Manager*. In Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Object Manager* > *Assign Generation Types to Balance Objects*, you have selected either "run" or "real time" as the generation type.

If required, you have configured the run-based creation of balance objects in Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Objects Manager* > *Assign Primary Data Sources for Worklists to Accounting Systems*. The system executes the processes for the run-based creation of balance objects in parallel jobs and based on indexes. To enable it to do this, a suitable database index must exist that uniquely identifies a single transaction (such as financial transaction ID and settlement account ID, financial instrument ID and securities position account ID, aggregated object ID).

If you do not make an entry in Customizing, the system uses the generation type "run" by default.

Generation Types of Balance Objects

You can choose the following generation types for balance objects:

- **Run** Generation Type
  The system reads the characteristic properties for all postings in a given run from the primary data sources in the Results Data Layer (RDL).
- **Real Time** Generation Type
  For accounting events, the system writes complete journal entries to the RDL as a parallel job, and transfers the balance objects straight away to the Analytics Layer.

The advantage of generating in real time is that the system can generate balance objects before balance processing and does not have to read a large volume of data. This results in improved performance.

The system's performance during the run-based creation of balance objects can be improved by creating a primary data source for a worklist. You do this in Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Objects Manager* > *Assign Primary Data Sources for Worklists to Accounting Systems*.

You can also **manually create** balance objects. For more information, see Manual Creation of Balance Objects [page 1429].

Error Handling

If errors occur when balance objects are created, the system continues processing and generates a worklist that it processes automatically using runs when next creating balance objects. To correct errors that may arise during generation in real time, select one of the following transactions depending on the situation.

- The system could not create any balance objects due to incorrect generation or missing Customizing settings.
  From the *SAP Easy Access* screen, choose *Analytics* > *Financial Statement Preparation* > *Manual Processing* > *Create Balance Objects for Posting Runs (Run)*.
- The system was able to generate one or more balance objects, but could not save. This error occurs if a balance object is locked when the system creates it. This is possible since different posting events can attempt to generate or update the same balance object at the same time. For this reason, use this transaction at regular intervals.
  On the *SAP Easy Access* screen, choose *Analytics* > *Financial Statement Preparation* > *Manual Processing* > *Postprocess Balance Object Creation (Real Time)*.

Enter the following data:

- Legal Entity
- Accounting System
- Run ID (optional)
  You can also identify the run using the posting ID (`/BA1/CR0PRRNID`).
- You can also specify that the system executes the program sequentially.

Choose  .

More Information

For more information, see Customizing for Bank Analyzer under *Analytics* > *Financial Statement Preparation* > *Balance Objects Manager* > *Assign Generation Types to Balance Objects*.

6.2.2.2 Manual Creation of Balance Objects

Use

You can use this function to manually create balance objects for a combination of legal entity and accounting system without executing balance processing. The system stores the generated balance objects in the segmentation service structure.

Prerequisites

You have made the settings for the balance object type and balance subobject type in Customizing for *Balance Object Manager*.

For more information, see Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Object Manager*.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Manual Processing* > *Create Balance Objects Manually*.
2. Enter the *Legal Entity* and the *Accounting System* for which you want to create balance objects.
3. If necessary, choose "Options" under *Technical Requirements*.
4. Choose *Execute*.
   The system creates the balance objects and displays an application log.

6.2.2.3 Deactivation of Balance Objects

Use

You can use this function to deactivate balance objects for balance processing. This is useful, for example, if you do not want to make any more postings for the balance object's combination of characteristics. Once you deactivate the balance object, it is removed from the worklist for balance processing.

> **Example**
>
> For the last fiscal period, you have generated a balance object for a bond. In the following fiscal period, the security will have expired. The system no longer calculates interest. Deactivate the balance object so that the system does not maintain it unnecessarily for the next fiscal periods.

Balance Object Manager automatically reactivates the balance object if a posting is determined for its combination of characteristics.

Prerequisites

You have made the settings for the balance object type and balance subobject type in Customizing for *Balance Object Manager*.

For more information, see Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Object Manager*.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Manual Processing* > *Deactivate Balance Objects Manually*.
2. Enter the *Legal Entity* and the *Accounting System* for which you want to deactivate a balance object.
3. Choose *Execute*.
   The system deactivates the balance object and displays an application log.

6.2.3 Balance Processing (FS-BA-AN-FSP-BP)

Use

This component carries out period-end processing for financial instruments. It uses the accounting information for the flow results, which is stored in the Results Data Layer (RDL) [page 1364].

Balance Processing provides the information for the financial statement items of financial products. The Aggregated Transactions [page 1472] component then processes this information further.

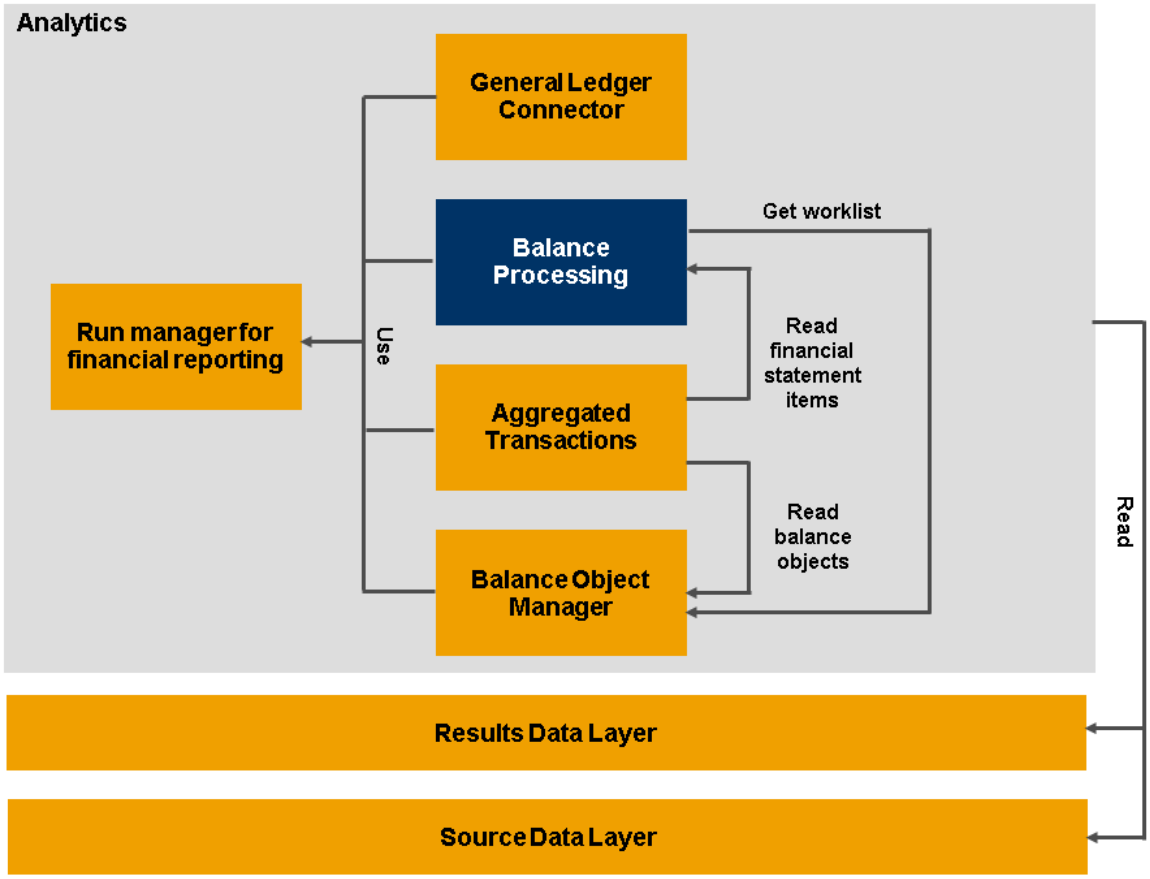
Integration

Balance Processing is part of Financial Statement Preparation [page 1425]. It requires the following components and functions:

- Balance Objects Manager [page 1425]
  The balance object is the unit used during balance processing. The Balance Object Manager provides Balance Processing with a worklist containing active balance objects.
- Run manager for financial reporting [page 1474]
  The run manager carries out balance processing runs and saves them.
- Results Data Layer [page 1364]
  The data transfer [page 1438] function provides Balance Processing with data from the RDL for flow results, characteristic results, and key date results.
- Source Data Layer [page 100]
  The SDL uses the data transfer [page 1438] function to provide the source data required for balance processing.

To process aggregated transactions, the Aggregated Transactions component reads the required information about the financial statement items of financial products from Balance Processing.

The following figure illustrates the relationships between Balance Processing and other components and functions.



6.2.3.1 Balance Object and Balance Subobject

Definition

The balance object (BO) is the principal object in the Balance Processing functions. The result data in Balance Processing is managed at balance-object level.

The financial reporting requirements are reflected in the modeling of the balance objects. You can use the balance subobjects (BSOs) to flexibly map additional requirements, for example, enter the hedged portion in the notes. Balance objects are constructed from balance subobjects.

For more information about mapping financial positions by modeling products using balance objects and balance subobjects, see Modeling of Products (Subledger Scenario) [page 705].

Structure

Balance Object Class

The balance object class (BO class) summarizes similar balance objects. You create a BO class for each accounting system. You assign a balance object category (BO category) to each BO class (for example, *Instrument* class). You can assign a BO class to one BO category only. However, you can assign more than one BO class to one BO category.

For more information about defining BO classes, see Customizing under *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Objects Manager* > *Edit Balance Object Type and Balance Subobject Type*.

In additional to the mandatory characteristics *Legal Entity* and *Accounting System*, you can refine the BO class further by entering additional defining characteristics (see the Customizing activity *Edit Characteristics and Key Figures* under *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Objects Manager*, view *Balance Object Class*).

Balance Object Category

A balance object category (BO category) is assigned to each BO class. BO categories, such as *G/L Account*, are predefined and delivered by SAP. You can assign a BO class to one BO category only. However, you can assign a BO category to multiple BO classes.

Balance Object

The balance object types (BO types) defined in Customizing are used as templates for creating balance objects. Similar BO types are grouped together in BO classes. If a new Customizing version exists for the BO type, the system regenerates all balance objects based on this BO type.

Balance Object Type

A balance object type (BO type) belongs to one BO class only.

For more information about defining balance object types, see Customizing under *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Objects Manager* > *Edit Balance Object Type and Balance Subobject Type*.

You can also refine the BO type further by entering additional defining characteristics (see the Customizing activity *Edit Characteristics and Key Figures* under *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Objects Manager*, view *Balance Object Type*).

Balance Subobject

Balance objects are divided into balance subobjects. Balance subobjects are part of the balance object. A balance object has one or more balance subobjects, which contain key figures and descriptive characteristics that are required in balance processing. In addition to the custom characteristics and key figures, the system also adds it own characteristics and key figures.

Balance Subobject Type

Characteristics and key figures are grouped in Customizing for balance subobject types, which then define the structure of the specific balance objects (see the Customizing activity *Edit Characteristics and Key Figures* under *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Objects Manager*, view *Balance Subobject Type*). The balance subobject types set up in Customizing can be used as templates for creating balance subobjects. A balance object type and a balance subobject category are assigned to each balance subobject type.

For more information about defining balance subobject types, see Customizing under *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Objects Manager* > *Edit Balance Object Type and Balance Subobject Type*.

Balance Subobject Category

The balance subobject category (BSO category) defines the purpose of a balance subobject. A BSO category is assigned to each BO type. BSO categories are predefined by SAP. The BSO category defines the properties of the BSO types and balance subobjects assigned to it. For example, the BSO category governs which of the characteristics and key figures predefined by SAP apply.

For more information about BSO categories, see Balance Subobject Category [page 1434].

6.2.3.1.1 Balance Subobject Category

Definition

Entity that defines the purpose of a balance subobject.

The balance subobject category governs which of the characteristics and key figures predefined by SAP apply. A balance subobject category is assigned to each balance subobject type. Balance subobject categories are predefined by SAP.

For more information about the BSO categories, see Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Edit Balance Object Type and Balance Subobject Type*.

Use

Figure 1 contains the balance subobject categories provided by SAP, and their properties.

BSO category	Statistical/debit - credit relevant key figures (ST/DC)	Number of balance subobjects	Main processing/ postprocessing (M/P)	Purpose	Typical key figures
MAIN	ST/DC*	1	M/P**	Control FSI determination	Total book value (statistical)
CURR	ST/DC	n	M	Financial reporting	Credit/debit-relevant key figures
MULTI	ST	n	M	Notes information (n BSOs) e.g. partial hedges	
MULTI_2	ST	n	M	Equivalent of MULTI for postprocessing	
OTHER	ST	1	M	Notes information (1 BSO) e.g. hedged portion	
OTHER_2	ST	1	M	Equivalent to OTHER for postprocessing	FairValue32; residual maturities

* If there is no balance subobject type of the category `CURR` in the balance object type.

** Read-access only in postprocessing.

MAIN (Main Part)

This balance subobject category is the main part of the balance subobjects. For each balance object type, you must create precisely one balance subobject type of category `MAIN`.

Although you can use debit/credit relevant key figures in balance subobjects of the category `MAIN`, you should do so only in exceptional cases.

Recommendation

SAP recommends that you store only statistical key figures in the main part of the balance object. You should store debit/credit relevant key figures in balance subobjects of the category `CURR`. If you do not follow this recommendation it will not be possible for an existing object to process data in more than one currency in future.

In financial reporting processes of the process category *Postprocessing*, the balance subobject category `MAIN` is used in a read-only capacity.

The following characteristics are predefined by SAP for the balance subobject type of the category `MAIN`:

- `/BA1/C55COMPL`: Force Merge

- /BA1/C55DLVTY: Delivery Type

If you choose not to follow the recommendation and do not create a balance subobject type of the category CURR for a balance object type, the following characteristics are predefined by SAP:

- Financial Position Currency (/BA1/C55AOCURR)
- Predefined foreign currency key figures
- Custom foreign currency key figures
- Other key figures used for debit/credit equations

> **Note**
>
> When using balance methods that are applied to the balance subobject type of the balance subobject category MAIN, make sure that you store all the required key figures (input/output) for the corresponding balance method. These need to be statistical key figures, as specified in the recommendation above.

CURR (Currency Leg)

The currency legs of balance objects are mapped by this balance subobject category. Each balance object type has a maximum of one balance subobject type of the category CURR.

Debit/credit relevant key figures can be stored in balance subobjects of the category CURR. We therefore recommend that you use this balance subobject category for all balance subobjects that have debit/credit key figures in the transaction currency. This includes simple transactions, meaning transactions with key figures in only one transaction currency (such as stock) as well as multiple currency transactions, such as forward exchange transactions.

Balance subobjects of the balance subobject category CURR are relevant for foreign currency. The foreign currency key figures required are generated automatically in balance subobject types of this category.

> **Caution**
>
> SAP recommends that you always store debit/credit relevant key figures in balance subobjects of the balance subobject category CURR. You should not store debit/credit relevant key figures in the main part (balance subobject category MAIN). If you do not follow this recommendation it will not be possible for an existing object to process data in more than one currency in future.

The balance subobject category CURR is not used in financial reporting processes of the process category *Postprocessing*.

You haver to use the characteristic *Financial Position Currency* (/BA1/C55AOCURR) as an additional defining characteristic for a balance subobject type of the balance subobject category CURR.

The following key figures are predefined by SAP for the balance subobject type of the category CURR:

- Predefined foreign currency key figures
- Custom foreign currency key figures
- Other key figures used for debit/credit equations

MULTI (Multiple Part)

This balance subobject category is used to depict balance subobjects that occur repeatedly in the corresponding balance subobject type but that cannot be depicted by the balance subobject category CURR. In other words, these balance subobjects can contain only statistical key figures.

> **Caution**
>
> Only statistical key figures can be stored in balance subobjects of the category `MULTI`

The balance subobject category `MULTI` cannot be used in financial reporting processes of the category *Postprocessing*.

No characteristics or key figures are predefined by SAP for the balance subobject category `MULTI`.

MULTI_2 (Multiple Part)

This is the balance subobject category that corresponds to the balance subobject category `MULTI` for the process category *Postprocessing*.

> **Caution**
>
> Only statistical key figures can be stored in balance subobjects of the category `MULTI_2`

The balance subobject category `MULTI_2` cannot be used in financial reporting processes of the category *Main Processing*.

No characteristics or key figures are predefined by SAP for the balance subobject category `MULTI_2`.

OTHER (Other Part)

This balance subobject category is used to depict balance subobjects that occur once in the corresponding balance subobject type but cannot be depicted using the balance subobject category `MAIN`.

> **Caution**
>
> Only statistical key figures can be stored in balance subobjects of the category `OTHER`

The balance subobject category `OTHER` cannot be used in financial reporting processes of the category *Postprocessing*.

No characteristics or key figures are predefined by SAP for the balance subobject category `OTHER`.

OTHER_2

This is the balance subobject category that corresponds to the balance subobject category `OTHER` for the process category *Postprocessing*.

> **Caution**
>
> Only statistical key figures can be stored in balance subobjects of the category `OTHER_2`

The balance subobject category `OTHER_2` cannot be used in financial reporting processes of the category *Main Processing*.

No characteristics or key figures are predefined by SAP for the balance subobject category `OTHER_2`.

6.2.3.2 Data Transfer

Use

This function enables the extraction of data from the Results Data Layer (RDL) [page 1364] and the Source Data Layer (SDL) [page 100]. Data is transferred from the RDL or from the SDL to the application; during the transfer, this data is not modified.

Integration

The following applications extract data from the RDL and from the SDL by means of the data transfer function:

- General Ledger Connector [page 1411]
- Balance Object Manager [page 1425]
- Balance Processing [page 1430]
- Aggregated Transactions [page 1472]

Features

This table lists the data that the system reads during the transfer of data from the RDL and the SDL and transfers to the different applications for additional processing or for display.

Data	Origin	Data Required by Application
Aggregated amounts	RDL	General Ledger Connector requires access to flow results.
Aggregated key figures	RDL	Balance Processing and Aggregated Transactions require access to aggregated key figures either from the key date results or from the flow results.
Characteristic set	RDL	Balance Object Manager and Aggregated Transactions require access to characteristic sets from the characteristic result of the flow result.
Characteristic values	RDL	Balance Processing and Aggregated Transactions require access to the characteristic result.

Metadata	RDL	General Ledger Connector, Balance Object Manager, Balance Processing, and Aggregated Transactions require characteristics from the RDL metadata.
SDL data	SDL	Balance Processing and Aggregated Transactions require access to the SDL.
Posting documents	RDL	General Ledger Connector, Balance Processing, and Aggregated Transactions require posting documents for the display for the drilldown function.

6.2.3.2.1 Transfer of SDL and RDL Data for Generation of BO and BSO

Use

This function enables the General Ledger Connector [page 1411] and the Balance Object Manager [page 1425] applications to access data from the Results Data Layer (RDL) [page 1364] and from the Source Data Layer (SDL) [page 100].

To access the RDL and the SDL, this function uses primary data sources and secondary data sources.

Integration

The General Ledger Connector uses this function to extract from the RDL and the SDL the data required for generating general ledger documents.

The Balance Object Manager uses this function to extract from the RDL and the SDL data for the generation of current worklists and correction worklists.

Prerequisites

- In Customizing for the General Ledger Connector, you have made the settings for the primary data source. For more information, see Customizing under *Bank Analyzer* > *Analytics* > *General Ledger Connector* > *Document Preparation* > *Assign Primary Data Sources*.
- In Customizing for *Balance Processing*, you have made the settings for the data transfer function. For more information, see Customizing under *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Data Transfer*.

6.2.3.3 Balance Methods

Use

The balance methods enable you to enrich the data at individual balance subobject level (BSO) before you start financial statement item determination [page 1451]. You would typically use the balance methods to determine information relevant for the notes section. You can also use them to derive control information for determining the financial statement items.

The data input for a balance method comprises the following objects:

The key figures and characteristics of one or more balance subobjects from a specific accounting system. Key figure values are stored as the balances of a period on a particular key date.

Cash flows, which are provided by the Service Functions.

The resulting information (output data) is assigned to the key figures and characteristics of the balance subobjects in question, which do not receive a value until the balance methods are applied.

You can use the following balance methods:

Totals Method [page 1441]

Fair Value Determination [page 1444]

Maturity Grouping [page 1445]

Pooling Method [page 1443]

You can also use a BAdI interface to define custom balance methods.

The balance methods are called up by the financial reporting process that is being used, and it is this process that provides them with the balance subobjects from the period being considered. At the start of processing, the relevant balance subobjects are filtered out using qualification criteria. To qualify for a balance method, a balance subobject must contain certain characteristics whose values meet particular criteria. You define these characteristics and criteria for each balance method.

Integration

The balance methods are part of the *Balance Processing* component and have to be scheduled in the financial reporting process before the determination of financial statement items. You can use the balance methods to enrich the data at balance subobject level. Then you can execute financial statement item determination.

Activities

To ensure that the balance methods are executed in a financial reporting process, you must have defined a step sequence in Customizing that contains these balance methods as a substep. For more information about integrating step sequences in the financial reporting process, see Balance Processing Start [page 1460].

Subledger Scenario

For more information about the settings for balance methods, see Customizing under *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Balance Methods*.

6.2.3.3.1 Totals Method

Use

The totals method is a balance method from the *Balance Processing* component. The system groups all the balance subobjects (BSOs) that are relevant for a given totals method into packages and calculates a specific total for each package. It groups the BSOs on the basis of a compound characteristic. Each BSO contributes certain key figures to the total. Based on the compound totals that are calculated, the system derives values for certain results attributes and assigns them to all the BSOs contained in the compound in question.

Integration

Totals methods that are not dependent on each other can be processed simultaneously. If a totals method, such as A, is dependent of the results of a different totals method, for instance B, it has to be scheduled in the financial reporting run after totals method B.

Prerequisites

Like the other balance methods, a totals method is called up during the financial reporting process. You must have made the following settings in Customizing for *Balance Processing*:

For more information, see Customizing under *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Balance Methods* > *With Qualification Criteria* > *Totals Method*.

- Defined the qualification criteria
- Definition of totals rules
- Definition of derivation rule(s)

Features

When processing the totals methods, the system carries out the following steps:

1. BSO selection
   The system chooses all the balance subobjects that are relevant for the totals method in question. This selection is based on general *qualification criteria*. You specify characteristics and their values, which the system uses to identify balance subobjects as relevant for the totals method in question.

2. Creation of groups
   Based on the compound characteristic, the system sorts all the relevant balance subobjects into groups, and calculates a total for each group. A group could be a portfolio, for example. In this case, the compound characteristic would be the portfolio number. All the balance subobjects with the same portfolio number belong to the same portfolio and consequently to the same group. The compound characteristic is defined in the *totals rules*.
3. Extraction of the key figures that are to be totaled
   The system selects those key figures of a balance subobject that contribute to the group total. This information is also defined in the *totals rules*.
4. Calculation of the total for each group
5. Application of derivation rules
   The system assigns values to the results attributes on the basis of the compound totals that have been calculated. In the simplest case, the results attribute is one key figure to which the total itself is assigned. Alternatively, or additionally, results characteristics can be derived. Size class determination illustrates this; its objective is to divide customers or products into size classes. Consequently, the compound characteristic is the size class to which the values "A, "B, or "C are assigned, depending on the size of the compound total.
6. Assignment of the result attribute values to the balance subobjects.
   The system assigns the results attribute values that have been calculated for a compound to all the BSOs belonging to that compound.

Example

The following table gives examples of totals methods:

Totals Method	Compound Characteristic	Objective of Creating Totals
Portfolio valuation	Portfolio number	Determination of the value of each portfolio
Compound total	Compound ID	Determination of the +/- sign of a particular total for determining the financial statement items. Depending on the value (+ or -), the underlying results data records are assigned to the asset or liability side of a particular account.
Size class determination	Customer ID	Determination of the size class for each customer

Activities

In order for the system to use this balance method within a financial reporting run, you have to define a step sequence in Customizing that contains this balance method as a substep.

For more information about integrating step sequences in the financial reporting process, see Balance Processing Start [page 1460].

6.2.3.3.2 Pooling Method

Use

The pooling method is one of the balance methods [page 1440] in the Balance Processing component.

This method is used to clear payables and receivables. It does this by placing balance objects into groups and calculating the aggregated book value of each group. Within a method variant, the key figures that will be totaled and the pooling amount can be either both relevant to statistics or both relevant to credit/debit.

Integration

Pooling methods that are not dependent on each other can be processed simultaneously. If pooling method A is dependent on the results of pooling method B, for instance, it has to be scheduled in the financial reporting process after pooling method B.

Prerequisites

As is the case for the other balance methods, a pooling method is called during the financial reporting process. You must have made the following settings in Customizing for Balance Processing:

1. Defined the qualification criteria
2. Defined the pooling rules

For more information, see Customizing for Bank Analyzer under *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Balance Methods* > *With Qualification Criteria* > *Pooling* .

Features

When you use the pooling methods, the system carries out the following steps:

1. Selection of balance objects
   The system chooses all the balance objects that are relevant for the pooling method in question. This selection is based on general qualification criteria: You specify the characteristics and their values that the system is to use to identify balance objects as relevant for the pooling method in question.
2. Creation of groups
   The system places the set of relevant balance objects into groups using the group characteristic and calculates a total for each group. A group could be a portfolio, for example. In this case, the compound characteristic would be the portfolio number. All the balance objects with the same portfolio number belong to the same portfolio and consequently to the same group. The group characteristic is defined in the pooling rules.
3. Extraction of the key figures that are to be totaled

The system selects those key figures of a balance object that contribute to the group total. This information is also defined in the pooling rules.

4. Calculation of the total for each group
5. Assignment of the result attribute values to the balance objects
   The system assigns the result attribute values of a group to the balance objects (see example).

Activities

So that the system uses this balance method within a financial reporting process, you have to define a step sequence in Customizing that contains the balance methods *SUMMETH_SU* and *SUMMETH _DI* as substeps.

For more information about integrating step sequences in the financial reporting process, see Balance Processing Start [page 1460].

Example

Note that in the example below for the pooling method the four balance objects all belong to the same group.

Balance Object	Book Value	Group Total	Control Field	Required Value (Virtual Value Only)	Pooling Amount	Check Sum
A	+10	+100	Initial	0	-10	+10
B	+20	+100	Not initial	+100	+80	-80
C	+30	+100	Initial	0	-30	+30
D	+40	+100	Initial	0	-40	+40

6.2.3.3.3 Fair Value Determination

Use

The fair value determination balance method calculates the fair value that is to be disclosed in the notes to the accounts, as required under IAS 32.

It calculates the fair value and enters it in a key figure for each balance subobject (BSO). This information is required for financial assets and payables.

The fair value is the same value that is used in accounting for the key date valuation for certain financial instruments. If balance subobjects are not processed in accounting, the fair value can be calculated if required information such as cash flows and conditions have been delivered to the SDL.

Features

The system calculates the fair value as follows:

1. It uses qualification criteria to select the relevant balance subobjects.
2. It calculates the *Fair Value Determination* key figure.
   It amends either the value of the existing *Book Value* key figure or a value calculated by the *Provide Fair Value* service module.

Activities

In order for the system to use this balance method within a financial reporting run, you have to define a step sequence in Customizing that contains this balance method as a substep.

For more information about integrating step sequences in the financial reporting process, see Balance Processing Start [page 1460].

6.2.3.3.4 Maturity Grouping

Use

This balance method is divided into two submethods:

1. **Maturity Determination**
   Results in an initial maturity grouping determined from the cash flows stored in the system or from characteristics that are to be provided.
2. **Adjustments to the Maturity**
   You cannot rule out the possibility of the total of the groups in the initial maturity grouping differing from a certain target value, such as a balance sheet value. This kind of deviation may occur if the balance sheet value and discounted cash flows exist in the system independently. This can happen if the balance sheet value was passed on or determined manually (impairment). In these cases, the *Maturity Corrections* submethod enables any difference to be distributed across the maturity groups, either proportionally or based on defined weightings.

Features

The submethods have the following tasks:

- **Maturity Determination**
  The system determines an initial maturity grouping using one of the following procedures:
  - Cash flow-based procedure (standard)
    Aggregation of cash flows within predefined maturity groups. The prerequisite for this is that the relevant cash flow for each balance subobject (BSO) being checked is stored in the Source Data Layer (SDL). It is not the task of this method to change a cash flow (for example, on the basis of specific conditions).

You can also choose a custom cash-flow-based determination. In this case, you can define which cash flows are relevant for determining the initial maturity grouping. To do so, define the required cash flow indicators in Customizing. This determination category is available with and without the use of correction steps.
If an impairment server environment has been defined for the accounting system, the system calls the Impairment Server to determine a cash flow. Based on the impairment attributes of a financial transaction or financial instrument, the Impairment Server determines the impairment calculation approach to be applied. According to the calculation approach, the Impairment Server calls the corresponding method of *BAdI: Calculation of Impairment Results*:

- If the *Expected Cash Flow* calculation approach applies, the system uses the expected cash flow for the maturity grouping instead of the contractual cash flow.
- If the *Expected Loss* calculation approach applies, the system uses the contractual cash flow for the maturity grouping.

- Characteristic-based procedure
  - In the case of some balance objects, maturity grouping occurs not on the basis of a cash flow, but rather from characteristics that are to be delivered. As customer requirements can vary immensely, this method can be used to define characteristic-based procedures for determining the maturity for one or more BSO classes.

In both cases, the resulting maturity grouping is assigned to the BSO in question.

- **Adjustments to the Maturity**
  The system successively replaces an existing maturity grouping with a corrected maturity grouping. You can make as many correction steps as required. For each step in the adjustment process, the maturity groups are overwritten. The difference between the old and new maturity groupings can be explained by considering the following two questions:
  - How large should the total across all the maturity groups be? You specify the required total (new maturity total) by entering one or more key figures, which when added together, give the new maturity total.
  - How is the difference between old and new maturity totals distributed across the maturity groups? Possible answers are:
    - Proportional distribution: The difference is distributed in such a way that the relative weightings of the individual maturity groups are maintained.
    - Weighting-based distribution: In this case, the difference is distributed on the basis of customer-defined weightings.

Activities

In order for the system to use this balance method within a financial reporting run, you have to define a step sequence in Customizing that contains this balance method as a substep.

For more information about integrating step sequences in the financial reporting process, see Balance Processing Start [page 1460].

Example

Possible sequential correction steps:

Correction Step	Key Figures Used to Determine New Maturity Total	Distribution Type for Difference Between New and Old Maturity Total	
Ensures that the maturity total is the same as the nominal value	Nominal value (without accrued interest)	Proportional distribution	
Distribution of accrued interest	Nominal value Accrued interest	Weighting-based distribution:	The shortest maturity band receives 100 % weighting, in other words, the prorated interest is allocated solely to the first maturity band

For more information, see Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Balance Methods* > *With Qualification Criteria* > *Maturity Grouping*.

The maturity grouping uses the following key figures, which are provided by SAP: `55DGRID` (Date Grid) and `&55MGRPS` (Maturity Groups). For more information, see Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Balance Methods* > *With Qualification Criteria* > *Maturity Grouping* > *Method-Specific* > *Method for Initial Maturity Grouping* > *Create Methods based on Characteristics (Optional)* > *Edit Secondary Data Source for Maturity Grouping*.

You edit the settings for the determination category in Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Balance Methods* > *With Qualification Criteria* > *Maturity Grouping* > *Method-Specific* > *Method for Initial Maturity Grouping*.

6.2.3.4 Financial Statement Items

Use

Financial statement items are the result of assigning results data to items in the balance sheet and the profit and loss statement. This assignment means that the system summarizes the key figures from the results data to form financial statement items by:

- Aggregating several key figures (such as *Book Value*, *Pro Rata Accrued Interest*, and *Special Provision*) to form one financial statement item, for example.
- Omitting characteristics that identify single transactions, such as *Account Number*.

The criteria you use to create your financial statement items depend on your business needs and reporting requirements. Financial statement items and their characteristics are generic - including the currency types described below. This means that the software coding does not determine which financial statement items and

characteristics are available. Rather it is up to you to select the financial statement items and characteristics you need during your implementation project.

You define financial statement items in Customizing. For this, you need to enter the following data:

- **Key for financial statement item**: An alphanumeric key identifying a financial statement item. For example, `A4711` or `FORD_A_K`.
- **Description**: A text describing a financial statement item. This text can be used as an alternative to the value. It can also be used in reporting.
- **Value type**: A characteristic (see Characteristics Provided for Financial Statement Items [page 1457]) used to differentiate the values generated in balance processing. For example:
  - **`Actual`** for the data generated originally.
  - Data generated during realignment has different value types, which you can define.
- **Report Type**: Specifies the type of report for which a financial statement item is to be used, such as *Balance Sheet*, *P/L Statement*, *Appendix*, *Off-balance Sheet Items*, *Currency Position*, *Equivalent Value for Currency Position* or *Miscellaneous*.
  While this type is not related to concrete reports in reporting, it is important for the determination of financial statement items.
  In the definition of the rules for Determination of Financial Statement Items [page 1451], financial statement items appear in the rule steps, where key figures are assigned to them. To enable the system to run a validation check when determining the financial statement items, you have to specify the report type to which a financial statement item belongs.

For more information, see Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Financial Statement Items* > *Edit Financial Statement Items*.

Key Figure Semantics

Key figure semantics play an important role in differentiating results data and financial statement items. The key figure semantic is the ID of a key figure, such as *book value* or *pro rata accrued interest*. The key figure is used in functional currency. The semantic value could appear as a column heading in an Excel table, for example.

From a technical perspective, the semantic value of a key figure is converted to a characteristic in the financial statement item. This is the reason why a key figure no longer has a name ("column heading) in a financial statement item, but is represented by a characteristic bearing its name. Such characteristics can then be used in reporting to drill down in a financial statement item according to the key figures from which it was created.

In contrast to the results data, key figures in financial statement items do not have a name that indicates their business content. They are non-semantic key figures. In the financial statement items, the business content is indicated by the characteristics. SAP provides the non-semantic key figure in functional currency (`/BA1/K55FSBIL`) and in transaction currency (`/BA1/K55FSOBJ`).

This is illustrated by the following example:

Example of Non-Semantic Key Figures

Value (Key Figure)	Financial Statement Item	Semantics of the value field in the results data record Characteristic resulting from converting the semantics of the value field in the financial statement item
- 100	Loans and advances customers	Book value
- 20	Loans and advances customers	Pro rata accrued interest

In the example above, the values (key figures) alone cannot be used to determine what information they contain. The content of the key figure becomes clear only when considered in conjunction with the financial statement item. The allows the financial statement item to group together values from the results data, such as *Book Value* and *Pro Rata Accrued Interest* in this example. The result of this is that the reports and analyses are very flexible: the financial statement items (with additional characteristics where necessary) select the values that are to be reported.

All key figures have a +/- sign to enable simple addition in reporting.

Currency

The values in the financial statement items can contain currency information. A currency code must always be specified for key figures. For balance subobjects (BSOs) in foreign currency, the foreign currency concept means that key figures must be specified in original currency. These key figures are called *currency key figures*.

A value can be stored with different currency types in the financial statement items. Currency types refer to the way one value can be displayed from different perspectives. For example, an account balance can be displayed in functional currency or object currency. The minimum requirement is that the value exists in functional currency.

Currency key figures, however, do not contain translated results for this value in functional currency. These are key figures in different currencies in the results data, which are allocated to the same financial statement item such as the key figures in functional currency.

The semantic value for the key figure applies to all currency types. Consequently, the key figures *Book Value in Functional Currency* and *Book Value in Transaction Currency* have the same semantic value. You have to set this relationship (for both key figures to use the same semantic value) in Customizing, in the basic settings for balance processing.

Example

Feeder Systems	
Source System 1	Source System 1
Loan I: 10 Million, 5 % Interest	Loan I: 10 Million, 5 % Interest
Loan II: 5 Million, 4 % Interest	Loan II: 5 Million, 4 % Interest

Transformation Layer
Normalizing of characteristics from the source systems

Results Data (SDL)				
ID	Balance Subobject Type	Country	Book Value	Interest Income
1	Loan	01	- 10 million	0.5 million
2	Loan	01	- 5 million	0.2 million
3	Loan	10	- 20 million	1.2 million

Determination of financial statement items
Processing completed in full, key figures derived using rules

Financial Statement Items	
Rules: BSO type = loan, book value -> financial statement item = 100 (receivables from customer) BSO type = loan, interest revenue -> financial statement item = 300 (interest revenue)	
Financial Statement Item	Balance
100	- 35 million
300	+ 1.9 million

Drilldown Financial Statement Item 100			
Financial Statement Item	ID	Country	Balance
100	1	01	- 10 million
100	2	01	- 5 million
100	3	10	- 20 million

Representation on Balance Sheet	
Rules: Filter = Financial Statement Item 100, Drilldown by Characteristic "Domestic/Foreign	
Financial Statement Item	Balance

Representation on Balance Sheet	
Loans and advances to domestic customers	- 15 million
Loans and advances to overseas customers	- 20 million

6.2.3.4.1 Determination of Financial Statement Items

Use

This function assigns the results data to the financial statement items. It is a processing step in the financial reporting process (see figure).

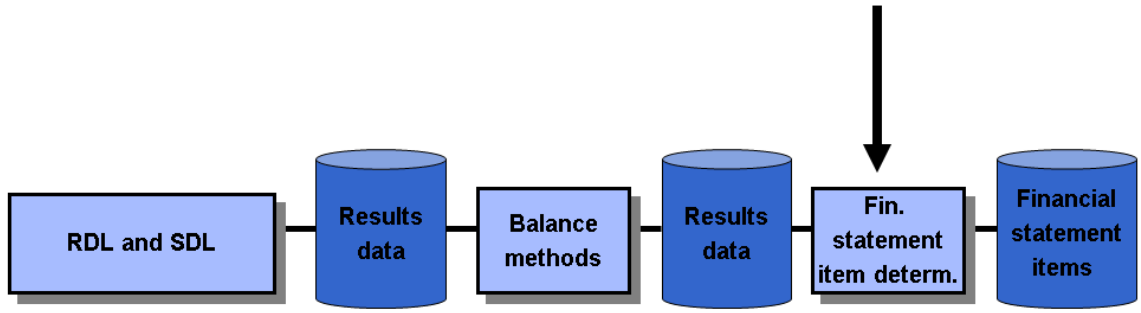


Figure: Determination of financial statement items in the financial reporting process

You make all the settings required for the *Determination of Financial Statement Items* step in Customizing.

For more information, see Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Financial Statement Items* > *Determination of Financial Statement Items*.

Features

The input data for determining financial statement items comprises the results data processed in the balance methods, and the following parameters:

- Accounting system
- Key date
- Status (test run or update run)

The result (output) of the financial reporting process following the step for determining financial statement items is that financial statement items are created. In the financial reporting process, the system displays a status message that shows when processing of the rule set began, and a log that documents the processing, and any errors.

The actual processing determination of financial statement items takes place between the input and output.

The system processes the results sequentially, in no specific order. The following processing steps take place for each results data record:

- Rule determination
- Rule application
- Check

In **rule determination**, the system filters the rules using the balance subobject type (BSO type) of the results data record. It checks the conditions with the characteristic and key figure values of the results data record. If a results data record meets the conditions with regard to both characteristics and key figures, the system applies the rules.

Rule application means that the system assigns the appropriate values of the *financial statement item* characteristic to the key figures in the results data record. For key figures in a specific currency, this assignment is made simultaneously for key figures in both functional currency and transaction currency (dependent on Customizing settings, see Financial Statement Items [page 1447]). This means that you have to define only one rule for both currencies.

Several rules can be valid for each results data record. The determining factor here is that the system checks the assignment of the individual key figures of the results data record.

If the **check** does not identify any errors in the values, assignment relevancy, or report type, the key figure is assigned to a suitable financial statement item. The financial statement item is also provided with the derived and transferred characteristics.

If errors occur during this check, the system generates a log entry for the incorrect results data record during **error handling**, and assigns all the key figures from the results data record in question, along with related characteristics, to suitable error items. Once the rules have been corrected, the incorrect balance subobjects have to be processed again.

If the system cannot find a rule in the results data record for a key figure that has to be assigned, it triggers error handling.

The log entry in error handling contains the following information:

- Identification of the balance subobjects
- Details of the error, including:
  - No rule for a results data record
  - Assignment relevancy and report type do not match
  - Rules found for the incorrect results data record

Activities

In order for the determination of financial statement items to be triggered within a financial reporting process, you have to define a step sequence in Customizing that contains the determination of financial statement items as a substep.

For more information about integrating step sequences in the financial reporting process, see Balance Processing Start [page 1460].

6.2.3.4.2 Financial Statement Item Methods

Use

You can use this function to evaluate and edit financial statement items.

Financial statement methods

- Copy existing financial statement items to a simulation position that can be in a different period
- Select a set of data records in the simulation position
- Multiply the key figures in the data records by a factor
- Generate data records according to the settings

Financial statement item methods consist of variants that group together calculation rules created from source conditions and target requirements.

Realignment is a financial statement item method. It allows financial statement items and other characteristic values (such as *profit center*) to be summarized or split to create comparative figures.

Features

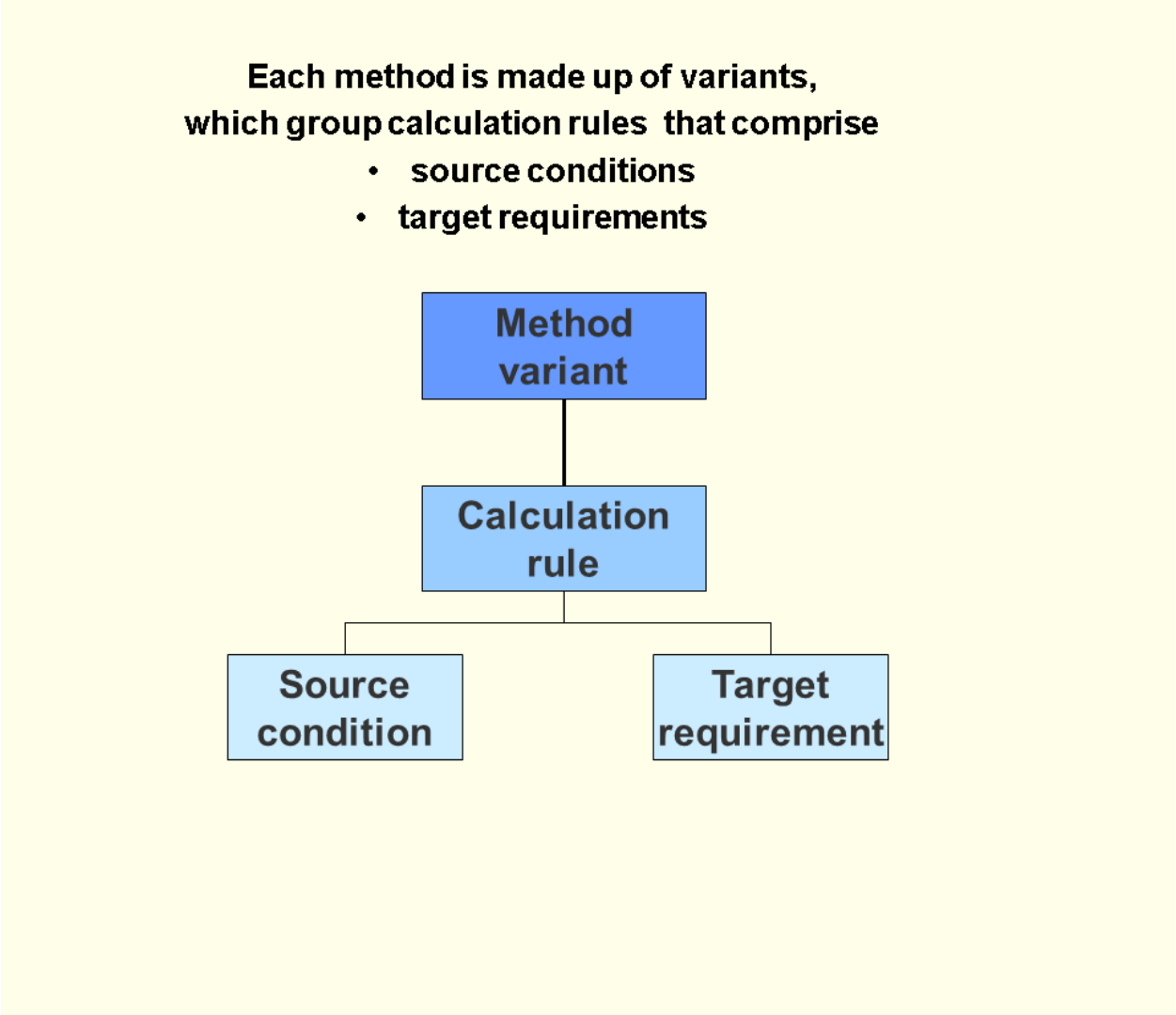
Simulation is run using the value type for simulation.

You create value types for realignment in Customizing. For more information, see Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Parameters for Financial Reporting Process* > *Edit Value Types*.

Processing Financial Statement Item Methods

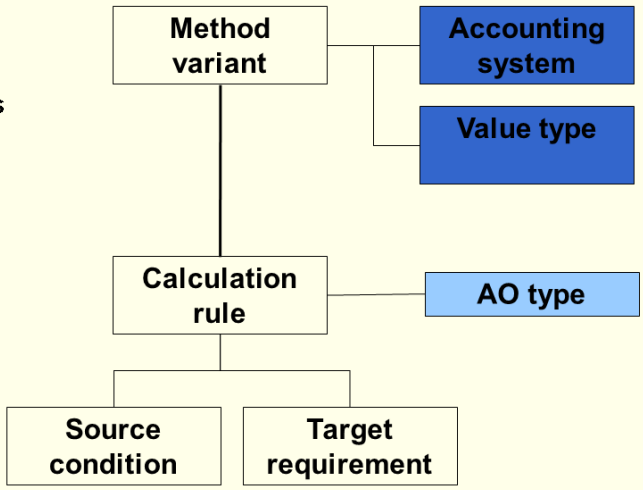
- Source conditions select the datasets that are to be processed.
- Target requirements determine what happens to the selected data.

Source conditions and target requirements can be grouped to form calculation rules. A group of calculation rules forms a method variant:



Calculation rules are valid for specific balance subobject types (BSO types). Method variants are valid for combinations of accounting system and value type:

- **Source conditions and target requirements can be defined separately and reused.**
- **Calculation rules have to be assigned consistently: The balance subobject types of the rules have to contain all the characteristics of the source conditions and target requirements.**
- **The combination of accounting system and value type must identify a method variant.**
- **Use a check report to check that the data is consistent.**



Note

A calculation rule that is valid for specific balance subobject types can contain only those source conditions and target requirements that use characteristics that are also contained in the balance subobject types in question.

The accounting system and value type of the financial reporting process in which the method is running determine which variant is used. For this reason, the combination of accounting system and value type must be unique.

The *Change Assignment* Value Type

Certain strategic managerial decisions and changes in the business environment can induce a company to change its organizational structure. There can be many reasons for changing organizations, including new market or market entry strategies, optimizing cost structures, merging computing centers, or insourcing/outsourcing.

Accounting provides functions that you can use when original conditions have changed, to compare the data from before the changes with the current, updated data within Accounting.

The *change assignment* value type changes existing data to suit the new organizational structure. It does so by simulating the financial statement items. While it is possible to change the assignment of all the characteristic value combinations, it is not always necessary to do so.

Multipliers can also be applied to the key figures. Dependent on the task, the characteristics can be retained, or individual values replaced.

In addition, any differences can be collected in a special position, so that the debit and credit sides balance for realignments.

Target Requirements in the *Change Assignment* Value Type

- **Replace the characteristics**
  - **Factor is usually 1.0, though experts can enter a different value.**
  - **All characteristics and values of the data record selected in the source are retained, except for those values that are replaced as specified**
  - **Result: Certain characteristics have new values (such as new profit center) and, if necessary, the item is also increased or decreased by a particular factor.**

- **Collect the differences**
  - **Factor specifies the delta generated by the extrapolation (difference)**
  - **The specified characteristics describe the collective item**
  - **Result: The differences generated by the factor can be collected in suitable items (such as *Other Assets*) to ensure that the debit-side and the credit-side still balance**

Activities

You create the following entities sequentially in Customizing:

- Source conditions
- Target requirements
- Calculation rules
- Variants

For more information, see Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Financial Statement Items* > *Financial Statement Item Methods*.

To ensure that the methods for financial statement items are used within a financial reporting run, you must have defined a step sequence in Customizing that contains these methods as a substep. For more information about integrating step sequences in the financial reporting process, see Balance Processing Start [page 1460].

Assignment to Secondary Data Source

You can define secondary data sources for the conditions. These allow you to include characteristics in the rule conditions that did not exist in the financial statement items. With a condition that uses the characteristic *profit center*, for example, you can add the country of the head office of the profit center at a later stage. To do this, you must be able to derive the condition from the characteristic *profit center* using the secondary data source.

6.2.3.4.3 Characteristics Provided for Financial Statement Items

We provide the following characteristics that are relevant for financial statement items.

Characteristics Provided by SAP

Description	Use
/BA1/C55REPT Report Type	Specifies for which type of reports the assigned characteristic *financial statement item* is to be used. The report type specifies whether a financial statement item is required for the balance sheet, off-balance sheet items, the profit and loss statement, the notes, or for something else. The report type is a fixed characteristic that is predefined by SAP. In the definition of the rules in the rule set, the financial statement items appear in the rule steps where key figures are assigned to them. You must specify to which report type a financial statement item belongs to enable the system to run a plausibility check for the rule set.
/BA1/C55VALT Value Type	Describes the data generated in balance processing. You specify the value type when you create a financial reporting process. Values predefined by SAP are available. You can define additional, customer-specific characteristic values for extrapolations and for realignments.

Description	Use
/BA1/C55FSI Financial Statement Item	For reporting purposes, this displays an aggregation of key figures from balance subobjects. The system aggregates the values of multiple key figures according to certain business criteria. This results in a financial statement item. Characteristics that identify single transactions, such as "account number," for example, are omitted. Examples of financial statement items include: • Items in the balance sheet • Items in the income statement • Items in the reserves You define the criteria used to create your financial statement items in accordance with your business needs and reporting requirements. You can group together the book value, pro rata accrued interest, and specific provision in a financial statement item for the financial report, for example. In the financial statement item *Loans and Advances to Customers*, you can group into one item all of the individual receivables relating to your customers.
/BA1/C55KFSEM Key Figure Semantics	The name of a key figure, for example *book value* or *pro rata accrued interest*.The name indicates the content of a key figure from a business perspective. The key figure semantic is used as a characteristic in the financial statement items. In the result data, the key figure semantic takes the form of a key figure ID in a particular currency. You can assign another key figure ID with a different currency to this key figure ID, provided the key figures both have the same business content. The key figures in financial statement items no longer have a name. They are non-semantic key figures. The semantic of the key figure is depicted in the financial statement items by the *key figure semantic* characteristic. You can then use the *key figure semantic* characteristic to break down a financial statement item into the key figures from which it was created.
/BA1/C55RUNTYP Run Type	Allows a distinction to be made between balance sheet values from an initial delivery and values from corrections and their related reversals. This information is required for drilldown purposes. It could possibly be used in certain evaluations as well.

Description	Use
/BA1/C55BPNO Sequence Number of Financial Reporting Process	Financial reporting processes for the same balance sheet are numbered sequentially. This can also be seen in the application logs for balance processing in the core Bank Analyzer system, where the financial reporting process ID is made up of the balance sheet ID and this sequential number. This number is required when navigating to the results data.

6.2.3.4.4 Overview of Financial Statement Items

Use

This function generates a list displaying the relationships between financial statement items and balance subobjects.

The function first displays the list of balance objects. When you have selected a balance object, the system displays the relationship between balance subobjects and financial statement items in detail using two display variants. Depending on the view you have chosen, the system displays more detailed information about either the balance subobject or the financial statement item. You can switch between both views as required.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Analysis* > *Financial Statement Items (Overview)*.
2. Specify the *legal entity* and the *accounting system*.
3. To specify the period of time, enter a *Fiscal Year* and *Period*.
4. If known, enter the *Accounting Run ID*.
5. If required, you can restrict the selection by entering the *BO Type* and *BSO Type*.
6. Choose *Execute*.
   The system displays a list with balance objects and detailed information, such as the GUID of the BOs.
7. Choose *FSI./BSO* or *BSO/FSI* to display the relationship between BSOs and financial statement items.

6.2.3.5 Executing Balance Processing

In this section, you will find information about the transactions that you require to execute balance processing from the *SAP Easy Access* screen under *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Processing*:

- Starting Balance Processing [page 1460]

- Closing Balance Processing [page 1461]
- Reset Balance Processing [page 1462]
- Simulate Balance Processing [page 1463]
- Data Export [page 1464]

6.2.3.5.1 Balance Processing Start

Use

You use this function to start a step in a step sequence in Balance Processing. You specify which step sequence is used by entering a run type and configuring Customizing settings for *Balance Processing*. The system automatically determines the current step number. The system executes the balance methods that you entered in Customizing for the corresponding step of the step sequence.

> **Note**
>
> For more information about configuring object models and generating objects in the Customizing system and production system, see Mechanism for Saving the Configuration [page 1466].

Prerequisites

- You have defined a step sequence that is appropriate for your scenario. This step sequence is assigned to a run type that you specify for the start of Balance Processing. To do so, use the activities in Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Step Sequences*.
- If you want to use a log variant, you must create it first in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Edit Application Log* > *Define Application Log Variant*.

Activities

You can execute the transaction directly or schedule it in the Schedule Manager.

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Start Balance Processing*.
2. First, enter the *Legal Entity* and the *Accounting System*.
3. To specify the period of time, enter a *Fiscal Year* and *Period*.
4. Choose the *Run Category* (run type).
5. Choose the *Value Type*.

6. Choose a *Log Variant*.
7. Choose *Execute*.

The system executes the balance methods that you entered in Customizing for the corresponding step of the step sequence. It generates the balance objects, balance subobjects, and financial statement items.

Example

You start Balance Processing for accounting system *IAS01*, period *10/2003*, and value type *ACTUAL*.

In Customizing, you defined a business step sequence, *IAS_VOLL*, as well as a technical step sequence, *IAS_VOLL*. You assigned both step sequences to the accounting system *IAS01* and the run type *ALL_IAS*. The assignment is valid from January 2003 to December 2003.

At the start of Balance Processing, the system finds the current step and executes the balance methods defined in the step sequence.

	Start Balance Processing	Assign Run Types to Step Sequences (Customizing)
Legal entity	LE1	
Accounting system	IAS01	IAS01
Fiscal year/period	10/2003	01/2003 - 12/2003
Run type	ALL_IAS	ALL_IAS
Value type	ACTUAL	
Step sequences		IAS_VOLL (BUS), IAS_VOLL (TECH)

6.2.3.5.2 Closing Balance Processing

Use

You can use this function to permanently close balance processing. You can close balance processing even if errors have occurred. To do so, choose the relevant status.

Prerequisites

You have carried out balance processing for the accounting system and the legal entity.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Closet Balance Processing*.
2. Enter the *Legal Entity* and the Accounting System.
3. To specify the period of time, enter the *Fiscal Year* and *Period*.
4. Choose the *Value Type*.
5. Under *Status of the Financial Reporting Process*, choose the status with which the balance processing run will be closed.
6. Choose *Execute*.
   The system closes balance processing and assigns to it the required status.

6.2.3.5.3 Reset Balance Processing

Use

If the data processed in a balance processing step contains errors, the system terminates processing. Once you have corrected the errors, you use this function to restart the balance processing step.

Prerequisites

- The system terminated a balance processing step in the financial reporting process.
- You have corrected the error.
  If you cannot correct the error, use the Close Balance Processing [page 1461] transaction.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Reset Balance Processing with Errors*.
2. Enter the *Legal Entity* and the *Accounting System*.
   The system enters automatically the values of the last balance processing.
3. Check the entries for *Fiscal Year* and *Period*.
4. Check the automatic entry for *Value Type*.
5. Enter the step number of the terminated step.
6. Enter the required technical requirements.

> **Note**
>
> If you want to search for the error, choose the option *Execute Program Sequentially*. This prompts the system to proceed without the parallel processing tool.

7. Choose *Execute*.
   The balance processing process continues from the specified step number.

6.2.3.5.4 Simulate Balance Processing

Use

You can use this function to test balance processing. This function starts a sequence of steps in Balance Processing. The system carries out balance processing without writing the data to the database.

Like in an update run for balance processing, the system uses the step sequence that you specified in Customizing for *Balance Processing*. The system determines automatically the current step number. The system executes the balance methods that you entered in Customizing for the corresponding step of the step sequence.

Prerequisites

- You have defined a step sequence that is appropriate for your scenario. This step sequence is assigned to a run type that you specify for the start of Balance Processing. To do so, use the activities in Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Step Sequences*.
- If you want to use a log variant, you must create it first in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Accounting* > *Subledger Scenario: General Settings* > *Edit Application Log* > *Define Application Log Variant*.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Simulate Balance Processing*.
2. First, enter the *Legal Entity* and the *Accounting System*.
3. To specify the period of time, enter a *Fiscal Year* and *Period*.
4. Choose the *Value Type*.
5. Choose *Execute*.

The system carries out the balance methods that you saved, in Customizing, in the relevant step in the sequence, but it does not write data to the database. The generated balance objects, balance subobjects, and financial statement items are not saved.

6.2.3.5.5 Data Export

Use

You can use this function to export your results data and financial statement items out of the IFRA solution so that you can process this information externally. This means that you can save your results data position as a text file.

Prerequisites

- You must have successfully processed and completed the reporting process for the data export.
- You have generated a *DataSource* in the Bank Analyzer core system. This *DataSource* can be used internally for the export.

Activities

1. On the *SAP Easy Access* screen, choose:
   - In the subledger scenario: *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Export Data* .
   - In the merge scenario: *Bank Analyzer* > *Analytics* > *Accounting: Merge Scenario* > *Balance Analyzer* > *Automatic Processing* > *Further Processing* > *Balance Processing* > *Export Data* .
2. Enter the *Legal Entity* and the *Accounting System*.
3. Enter the *Fiscal Year* ,*Period* and, if required, the *Number of Financial Reporting Process*.
4. Enter a *Value Type*, *Name of DataSource* and, if required, the *Package Size for Export*.
5. Choose *Execute*.
6. The system displays a dialog box. Enter the name of the file and the location to which you want to export the data.
   The system saves the data in the directory specified.

Once the data export has been completed you can access an application log with status messages.

6.2.3.5.6 Overview of Balance Objects

Use

You can use this function to display details for balance objects of a balance sheet. The overview displays only data from successfully completed financial reporting processes.

In addition to this, a detailed view provides information about the balance subobjects of a balance object. This includes information about assigned characteristics and key figures. From this overview, you can also display the

aggregated transactions. The overview includes a drilldown function that displays all postings from the RDL that are involved in an aggregated transaction.

You can use an additional detailed view of the characteristics of a balance subobject to display the financial statement items assigned to it.

Prerequisites

You have executed a financial reporting run that generated balance objects.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Analysis* > *Display Balance Objects (Overview)*.
2. Specify the *legal entity* and the *accounting system*.
3. To specify the period of time, enter a *Fiscal Year* and *Period*.
4. Choose the *Value Type*.
5. If required, you can restrict the selection of BOs by entering the BO type and the defining characteristics of the BOs that are to be selected.
6. Choose *Execute*.
   The system displays a list with balance objects and detailed information, such as the GUID of the BOs.
7. Choose *Display BO/BSO Relationships*.
   The system displays in a tree structure the BSOs for the BOs. In the detailed view on the right-hand side of the screen, you will find listed for the BSO characteristics with their values and key figures with their amounts.
8. Select a key figure, and choose *Display Financial Statement Item Assignments*.
   The system displays an overview of the assigned financial statement items.
9. Choose *Read File* to display the aggregated transactions.
10. In the overview of aggregated transactions, select an entry and choose *Display Data (Drilldown)*.
    The system displays the relevant postings from the RDL.

See also:

Process View of Balance Objects [page 1465]

6.2.3.5.7 Process View of Balance Objects

Use

You can use this function to display details for balance objects and balance subobjects of a financial reporting process.

For the characteristics and key figures, the system also displays information about the amounts from the Results Data Layer (RDL) [page 1364] that were posted to the balance objects in the financial reporting process.

The Data Transfer [page 1438] function extracts the data to be displayed from the RDL.

Prerequisites

You have executed a financial reporting run that generated balance objects.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Analysis* > *Display Balance Objects (Process View)*.
2. Specify the *legal entity* and the *accounting system*.
3. To specify the period of time, enter a *Fiscal Year* and *Period*.
4. Choose the *Value Type*.
5. If required, you can restrict the selection of BOs by entering the BO type and the defining characteristics of the BOs that are to be selected.
6. Choose *Execute*.
   The system displays a list of balance objects and detailed information, such as the GUID of the BOs and the posted amounts.

See also:

Overview of Balance Objects [page 1464]

6.2.3.5.8 Mechanism for Saving the Configuration

Balance Processing uses generated database objects for saving results data. To be able to read and save results data, the system reads configuration settings that contain information about the generated object and the balance object model as well as their relationship. This information is entered by means of IDs. These IDs contain the name of the balance object and balance subobject [page 1432] in the object model and the relationship to the generated database object. If you change or delete objects, it can lead to inconsistencies between the Customizing system and the production system. This is the case, for example, if the system was unable to delete the relevant object in the production system. Therefore, a mechanism for saving data is provided for balance processing.

SAFE Area and RAW Area

The system saves the last valid configuration for the current generation time in the **SAFE area**. After successful generation, this area contains all information about the generated objects and the object model. You cannot change the SAFE area through your own activities or Customizing transports. It can be changed only through system mechanisms and after-import methods.

Parallel to the SAFE area is the **RAW area**, which you can change through your own activities such as a generation process or Customizing transport.

If you change an object in the Customizing system, it is also changed in the RAW area and the SAFE area. However, during transport to the production system, this change may not be possible in the SAFE area because, for example, the existing data cannot be changed or deleted. This means that differences exist between the SAFE area and the RAW area. The system recognizes this error during balance processing. It terminates processing and issues an error message in the application log.

> **Note**
>
> Only the RAW area is initially filled after an upgrade. To ensure that the system can run properly, you need to activate and generate your object model. The SAFE area is initially filled in this way. You have to do this before you modify your object model in Customizing.

Error Handling

To restore a correct system configuration, perform the following steps:

1. Determine the current configuration in the production system by establishing all valid identities.
   On the *SAP Easy Access* screen, choose *Analytics* > *Financial Statement Preparation* > *Tools* > *Find Valid Generated Objects*
2. Restore the valid configuration by regenerating the IDs.
   On the *SAP Easy Access* screen, choose *Analytics* > *Financial Statement Preparation* > *Tools* > *Re-Create Generated Objects*
3. Import the Customizing transport generated by this transaction to your production system.
4. Start the activation and generation of the object model in your production system.
   Activate the characteristic structure for the object model of the financial statement items. The mechanism for saving data is not relevant for balance subobjects in the subledger scenario.

   > **Note**
   >
   > You must have activated and generated the object model to be able to change it in Customizing.
5. Import the Customizing transport created during generation to your production system.
6. Restart balance processing.

6.2.3.6 Reduction of Data Volume in the Analytics Layer

Concept

Reducing the data volume improves the performance of several processes, including those of the analytics layer (AN). The system optimizes these processes by archiving balance processing data, for example. However, this also involves processing a large quantity of data. To reduce this quantity, you should delete incorrect or obsolete data in advance. The system then does not have to include this data in the archiving process.

Incorrect or obsolete data includes data that the system no longer requires for processes or reporting, for example.

You can delete balance processing data such as financial statement items or key date results of balance subobjects, as well as data from aggregated transactions.

Integration

You should delete incorrect and obsolete data if you are planning to switch the accounting scenario, and in particular if you want to switch the scenario by coping the system.

Features

The following delete functions are available in the subledger scenario and the merge scenario:

- Deletion of Incorrect and Obsolete Data [page 1468]
- Change the Run Status to Obsolete [page 1471]
  You use this function to set the status of a balance processing run to *obsolete*. This may be necessary if you want to delete run data but the status of the data does not allow you to do so.

> **Note**
>
> Note the following information about how runs are referred to:
>
> - Technical name in both accounting scenarios: Accounting Analytics run
> - Business name in the subledger scenario: Balance processing run
> - Business name in the merge scenario: Financial reporting process
>
> The term *balance processing run* is used in general documentation.

6.2.3.6.1 Deletion of Incorrect and Obsolete Data

Use

This function deletes balance processing data and aggregated transaction data. It enables you to improve the performance of data archiving for the analytics layer (AN).

The system can delete the following:

- Data regarding financial statement items (`FSI`)
- Key date results for balance subobjects (`BSO`)
- Data regarding aggregated transactions (`AT`)

> **Note**
>
> The *Aggregated Transactions (AT)* deletion object is relevant only for the subledger scenario.

The system saves financial statement items and key date results for balance subobjects in the result database (RDB) in an RDB run, which contains multiple RDB packages with results data. Each package contains result sets, which comprise an RDB header entry, SGS characteristics, and an RDB key-figure table entry. During the deletion process, the system removes incorrect or obsolete entries from the corresponding RDB tables.

In the case of aggregated transactions, however, the system does not save any data in the RDB. It deletes the relevant data from the tables for *aggregated transaction totals* (ATT) and *aggregated transaction balances* (ATB).

> **Note**
>
> From a technical perspective, data is deleted by *deletion client*. There is one deletion client for each accounting scenario and deletion object, which is assigned to a specific deletion class and a run determination class. For example: deletion client for financial statement items in the subledger scenario (`/BA1/CL_AL_B2_DEL_FSI`, `/BA1/CL_AL_B2_RUN_SL`).

For more information, see Reduction of Data Volume in the Analytics Layer [page 1467].

Prerequisites

The option to delete obsolete data depends on the status of the balance processing run. You may therefore have to set the status of a balance processing run to *obsolete* manually. In this case, the Change the Run Status to Obsolete [page 1471] function is a prerequisite for deletion.

Customizing Prerequisites:

- **Subledger scenario**
  You have completed the Customizing activity *Use Reference Object for Selection* in Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Balance Processing* > *Deletion of Incorrect and Obsolete Data*.
  In the case of aggregated transactions, you have used the Customizing activity *Edit Deletion Configuration for Aggregated Transactions* in Customizing for *Bank Analyzer* under *Analytics* > *Financial Statement Preparation* > *Aggregated Transactions* > *Deletion of Incorrect and Obsolete Data*.
- **Merge scenario**
  You have completed the Customizing activity *Use Reference Object for Selection* in Customizing for *Bank Analyzer* under *Analytics* > *Accounting: Merge Scenario* > *Balance Analyzer* > *After Generation* > *Balance Processing* > *Deletion of Incorrect and Obsolete Data*.

> **Note**
>
> Note that the settings for the package size and block size impact the performance of the deletion report. Since the package size in this case indicates the number of entries in each package that must be deleted and the work required to generate the worklist increases with the number of packages, we recommend that you specify a high value (the maximum is 32000) depending on your system resources (in particular, the main memory). The same applies to the block size for data from aggregated transactions.

Activities

Subledger Scenario

1. On the *SAP Easy Access* screen, choose *Analytics* > *Financial Statement Preparation* > *Tools* > *Delete Incorrect and Obsolete Data*.
2. Enter the following on the selection screen:
   - Legal Entity and Accounting System
   - Balance Processing Run
     Select the balance processing run for which you want to delete objects.
   - Deletion Object
     In the "Deletion Object" field, select *Financial Statement Items*, *Balance Subobjects*, and *Aggregated Transactions*.
   - Technical Settings
     - Log Variant
     - Indicators to specify whether you want to run the program sequentially and/or in simulation mode
     - RDB Run ID
       You can enter an RDB run ID if you want to explicitly delete data for a specific RDB run. The RDB run is usually determined indirectly by means of the corresponding assignment tables in balance processing. If the assignment to an obsolete RDB run is lost (for example, because the Customizing settings have changed), you can obtain it directly from table `/BA1/R4_RUN`. The system still checks whether the RDB run belongs to a balance processing run for which deletion is permitted.
3. Choose .
4. The system deletes all objects that correspond to your selection and displays a results list.

Merge Scenario

1. On the *SAP Easy Access* screen, choose *Analytics* > *Accounting: Merge Scenario* > *Balance Analyzer* > *Tools* > *Balance Processing* > *Delete Incorrect and Obsolete Data*.
2. Enter the following on the selection screen:
   - Legal Entity and Accounting System
   - Financial reporting process
     Select the financial reporting process for which you want to delete objects.
   - Deletion Object
     In the "Deletion Object" field, select *Financial Statement Items* and *Balance Subobjects*.

     > **Note**
     >
     > Note that you cannot select *Aggregated Transactions* in the merge scenario.

   - Technical Settings
     - Log Variant
     - Indicators to specify whether you want to run the program sequentially and/or in simulation mode
     - RDB Run ID
       See the information about the RDB Run ID in the *Subledger scenario* section.
3. Choose .
4. The system deletes all objects that correspond to your selection and displays a results list.

6.2.3.6.2 Change the Run Status to Obsolete

Use

This function sets the status of a balance processing run to *obsolete*. You can use it to delete obsolete data of a balance processing run, for example, data regarding financial statement items. Only data with a specific status can be deleted.

Before you can delete data with the following statuses, you must set the balance processing run to obsolete:

- Subledger scenario:
  - Successfully completed (`FINI`)
  - Valid (`VALI`)
- Merge scenario:
  - Accounting (`ACC`)
  - Balance Processing (`BP`)
  - Completed (`FINI`)

For all other statuses in the subledger and merge scenarios, the system can delete data without you having to change the status to obsolete.

For more information, see Reduction of Data Volume in the Analytics Layer [page 1467].

Activities

Subledger scenario

1. On the *SAP Easy Access* screen, choose *Analytics* > *Financial Statement Preparation* > *Tools* > *Set the Status of Balance Processing Runs to Obsolete*.
2. Enter the following on the selection screen:
   - Legal Entity and Accounting System
   - Balance Processing Run
     Select the balance processing run that you want to set to *obsolete*.
   - Simulation mode
3. Choose  .

Merge scenario

1. On the *SAP Easy Access* screen, choose *Analytics* > *Accounting: Merge Scenario* > *Balance Analyzer* > *Tools* > *Balance Processing* > *Set the Status of Balance Processing Runs to Obsolete*.
2. Enter the following on the selection screen:
   - Legal Entity and Accounting System
   - Financial reporting process
     Select the financial reporting process that you want to set to *obsolete*.
   - Simulation mode
3. Choose  .

6.2.4 Aggregated Transactions (FS-BA-AN-FSP-AT)

Use

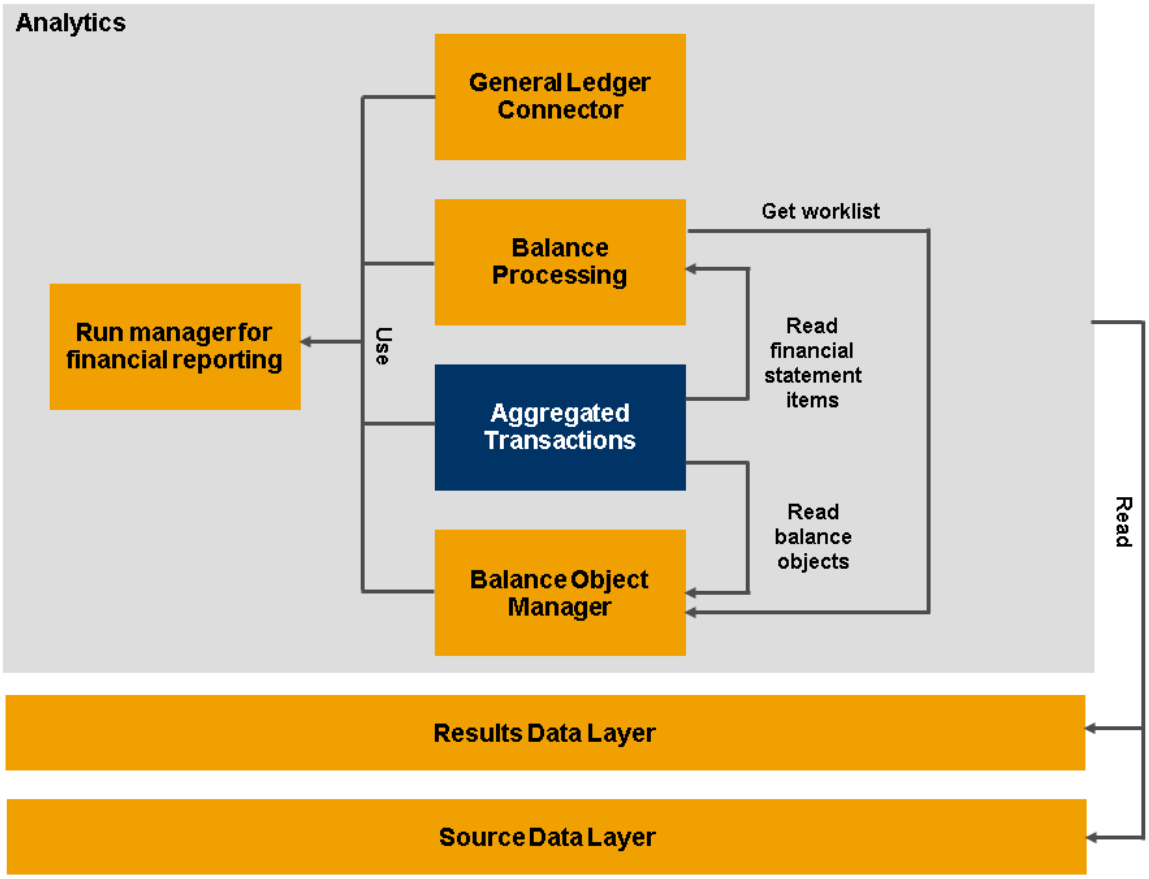
This component combines accounting documents from the flow results in the Results Data Layer (RDL) [page 1364] with the relevant financial statement items from Balance Processing and supplies the results, by means of the central extraction service, to Business Intelligence (BI). The information from the aggregated transactions (AT) in BI provides the basis for reporting and consolidation.

Integration

This component is part of financial statement preparation and is, like Balance Processing, period-end processing. It requires the following components and functions:

- Balance Objects Manager [page 1425]
  Calls the balance subobjects that belong to the aggregated transactions
- Balance Processing [page 1430]
  Calls the balance subobjects that belong to the financial statement items
- Run manager for financial reporting [page 1474]
  Creates runs for the aggregated transactions
- Results Data Layer (RDL) [page 1364]
  Supplies data from the flow results, characteristic results, and a list of available characteristics from the metadata framework by means of the data transfer [page 1438] function; this is done to limit the Customizing of aggregated transactions
- Source Data Layer (SDL) [page 100]
  Uses the data transfer function to supply source data for financial reporting
- BI extractor from the infrastructure [page 1637]
  Transfers information about aggregated transactions to BI

The following figure illustrates the relationships between the aggregated transactions and other components and functions.



Features

The processing of aggregated transactions creates the following two data sets:

- Balances of aggregated transactions
  These are the balances at the beginning of a fiscal year.
- Totals of aggregated transactions
  These are the flows from the beginning of the fiscal year until the currently processed period.

The component uses subledger documents from the Results Data Layer to create both data sets. Characteristics from the Source Data Layer can also be added to the data sets. Using the extraction tools, the system sends the aggregated transactions from the infrastructure to the OLAP reporting system.

Constraints

The aggregated transactions component is used only for financial reporting purposes.

6.2.4.1 Displaying Aggregated Transactions

Use

This function generates a list of documents from the balances of aggregated transactions (AT balances) and from the totals of aggregated transactions (AT sums). The list is prepared according to the selection parameters entered by the user.

You can use a drilldown function to display for an aggregated transaction all subledger documents from the Results Data Layer (RDL) [page 1364]. In the subledger document, you can use the financial transaction ID to display detailed information for the financial transaction. You can also call information about the settlement account.

Integration

The data transfer [page 1438] function reads data from the RDL, such as the subledger documents for an aggregated transaction.

You can use the balance object overview to display the aggregated transactions (see Overview of Balance Objects [page 1464]).

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Analysis* > *Display Aggregated Transactions*.
2. Specify the *legal entity* and the *accounting system*.
3. Enter the *financial reporting ID*.
4. If required, enter the type of aggregated transactions to restrict the display.
5. Choose *Execute*.
   The system displays a list of the aggregated transactions.
6. Select an aggregated transaction and choose *Display Data (Drilldown)*.
   The system displays for the aggregated transaction all subledger documents from the RDL.

6.2.5 Run Manager for Financial Reporting

Use

This function automatically creates runs for a combination of legal entity and accounting system for the following applications:

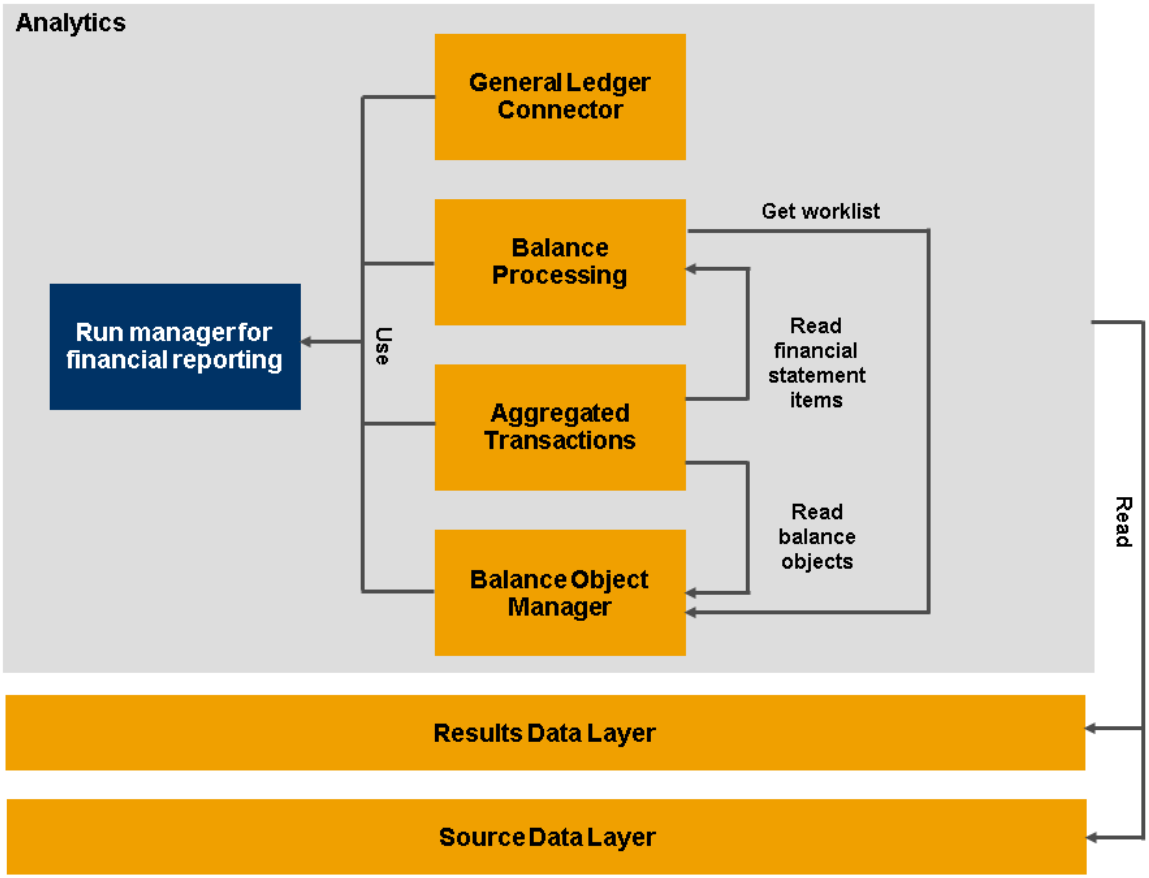
- General Ledger Connector [page 1411]

- Balance Object Manager [page 1425]
- Balance Processing [page 1430]
- Aggregated Transactions [page 1472]

It manages the valid time stamp, which is used by the applications to access the Results Data Layer (RDL) [page 1364] and the Source Data Layer (SDL) [page 100].

Integration

The functions of the run manager are integrated in the General Ledger Connector, Balance Object Manager, Balance Processing, and Aggregated Transactions applications. The following figure illustrates the relationship between the run manager and the applications listed above:



Features

For the following process types, the run manager for financial reporting automatically creates runs for a combination of legal entity and accounting system: General Ledger Connector, Balance Object Manager, Balance

Processing with aggregated transactions. Every run has a process type. The runs for aggregated transactions have the process type "balance processing."

The run manager assigns to every run a new run ID.

A run of the run manager contains multiple time references. These have one granularity, such as, year period, time, and time stamp; one use, such as, SDL or RDL; and one indicator of whether or not it is a time interval. The run manager assigns a time stamp for the creation of a run.

The run manager saves runs together with all of their attributes.

You can display the runs and their properties in the run manager (see Display Runs for Financial Reporting [page 1420]).

General Ledger Connector Runs

General Ledger Connector runs include the preparation of general ledger documents [page 1415] and the sending of general ledger documents [page 1418] to the general ledger. The run manager creates a General Ledger Connector run with a time reference of two time stamps. The system requires the time reference in order to read data from the RDL. The higher time stamp of the time reference is the time stamp for creation of the General Ledger Connector run; the lower time stamp is the higher time stamp of the previous General Ledger Connector run.

Balance Object Manager Runs

The run manager creates a balance object manager run with a time reference of two time stamps. The system requires the time reference in order to read data from the RDL and from the SDL. The higher time stamp of the time reference is the time stamp for creation of the Balance Object Manager run. The lower time stamp is the higher time stamp of the time reference, for reading RDL or SDL data, from the previous run.

Balance Processing Runs

The run manager creates a Balance Processing run with three time references:

- Technical time stamp for reading data from the RDL
- Technical time stamp for reading data from the SDL
- Posting date for reading data from the RDL
  The posting date is the last day of the fiscal period.

The Balance Processing run has a Balance Object Manager run. This reference run ensures that the Balance Processing run uses a current balance object worklist.

If the General Ledger Connector is used, the system assigns a General Ledger Connector run to the Balance Processing run. This run ensures that the balance processing data in BI is not more current than the general ledger data.

The higher time stamp of the time reference for reading RDL data is the RDL time stamp of the General Ledger Connector run. It is important that the run uses the technical time stamp for reading data from the RDL; this way, the run can also calculate postings made in fiscal period 1 but processed in fiscal period 2.

If the General Ledger Connector is not used, the higher time stamp of the time reference for reading RDL data is used as the time stamp for creation of a Balance Processing run.

The higher time stamp of the time reference for reading data from the SDL is used as the time stamp of the creation of the Balance Processing run.

A correction run for Balance Processing has a previous Balance Processing run. If the last full run for Balance Processing already has a correction run, the system uses this correction run as the reference run for the current

correction run. The lower time stamp of the time reference for reading RDL data is the RDL time stamp of the previous Balance Processing run or correction run.

Aggregated Transaction Runs

The process type for Aggregated Transaction runs is "balance processing".

The run manager creates an Aggregated Transaction run with three time references:

- Technical time stamp for reading data from the RDL
- Technical time stamp for reading data from the SDL
- Fiscal period for reading data from the RDL

The Aggregated Transaction run has a Balance Processing run. This reference run ensures the existence of financial statement items. If the last full run for Balance Processing has a correction run, the system uses this correction run as the reference run for the Aggregated Transaction run.

The technical time stamp of the time reference for reading RDL data has the higher value of the time stamp of the Balance Processing run or the correction run.

The technical time stamp of the time reference for reading SDL data has the higher value of the time stamp of the Balance Processing run or the correction run.

The run manager calls the first and last day of the fiscal period. The system requires this data in order to read RDL data.

6.2.5.1 Display Runs for Financial Reporting

Use

You can use this function to display runs for financial reporting for a particular legal entity and accounting system. This function displays for each run the list of existing reference runs. It displays, for a run, the current time stamp and the time stamp of the run that preceded the last run, as well as administrative information for the run.

Features

You can use this function to display runs for the following applications:

- General ledger connector [page 1411]
  General ledger connector (GLC) runs include the preparation of general ledger documents [page 1415] and the sending of general ledger documents [page 1418] to the general ledger. In the run manager, you can also display the general ledger documents [page 1412] for these runs. In the overview of the general ledger documents you can use the drilldown function; this function allows you to display, for the general ledger document, all relevant subledger documents from the RDL.
- Balance processing [page 1430] and aggregated transactions [page 1472]
  For balance processing runs, the run manager for financial reporting displays the previous run and the relevant general ledger connector and balance object runs.
  Aggregated transactions do not comprise an independent process. They are embedded in balance processing.

- Balance object processing [page 1425]
  For the current run, the run manager for financial reporting also displays the previous run.

For more information about the features of the run manager, see Run Manager for Financial Reporting [page 1474].

Activities

1. To display runs, you can start the run manager for financial reporting using one of the following methods:
   - Choose *Bank Analyzer* > *Analytics* > *General Ledger Connector* > *Start Run Manager for Financial Reporting*
   - Choose *Bank Analyzer* > *Analytics* > *Financial Statement Preparation* > *Tools* > *Start Run Manager for Financial Reporting*
2. If you want to display details for a particular run, and you know its run ID, then enter the *Run ID*.
3. Enter a *Legal Entity* and an *Accounting System*. These are mandatory entries.
4. If you want to display runs for a particular process, choose the appropriate *Process Type*.
   For the aggregated transactions, choose the process type *Balance Processing*.
5. You can further restrict your selection by entering the *User Name* of someone who has executed the run that you want to display. If you know it, enter the *Financial Reporting ID*.
6. Enter a time period for the runs.
7. Choose *Execute*.

6.3 Historical Database (FS-BA-AN-HDB)

Use

The Historical Database (HDB) is used to store the results of calculations (for example, recovery rates), and is a central store for the data from the various source systems. The HDB provides the basic information required by the reporting processes and supervisory review processes in Bank Analyzer.

The Basel II regulatory requirements apply to both the data upon which banks'in-house models are based, and to the parameters that are used by these models. Particular attention must be paid to the historization of the parameters used in the calibration and validation processes. The HDB hence stores data in a time-based way. This means that it is a central memory for information related to default data, and is optimized for time-based evaluations. The HDB provides a stable infrastructure, which ensures that histories can be created, even for long time series.

Since banks' in-house models, and the related data requirements, vary from one institution to the next, the HDB has an open architecture. You can therefore adjust the database to meet your individual requirements to ensure that data is retained for all necessary parameters. The HDB offers a large number of enhancements to enable future Basel II requirements to be met.

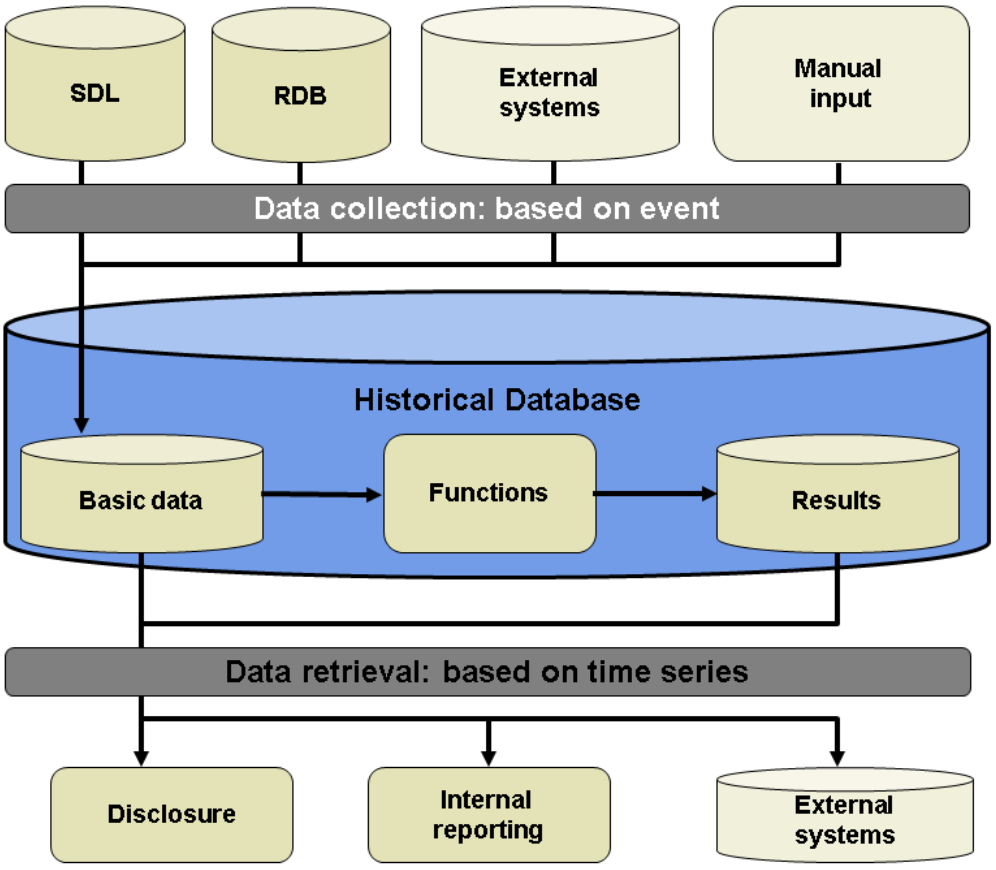
Implementation Considerations

The functions provided in the HDB aim to support banks'internal business processes, which may have to be adjusted in view of the Basel II capital guidelines, and to meet requirements regarding the retention of historical data.

The processes primarily involve bank's own estimations of certain parameters that are used to calculate the capital requirement, such as the process of creating the bank's internal rating for borrowers, for example.

Integration

The HDB is part of Bank Analyzer. It contains the data that is needed to meet the Basel II requirements. The data includes the transaction data that was transferred from the Source Data Layer (SDL) [page 100], the calculation results that were stored in the Result Database (RDB) [page 1823], and external data.

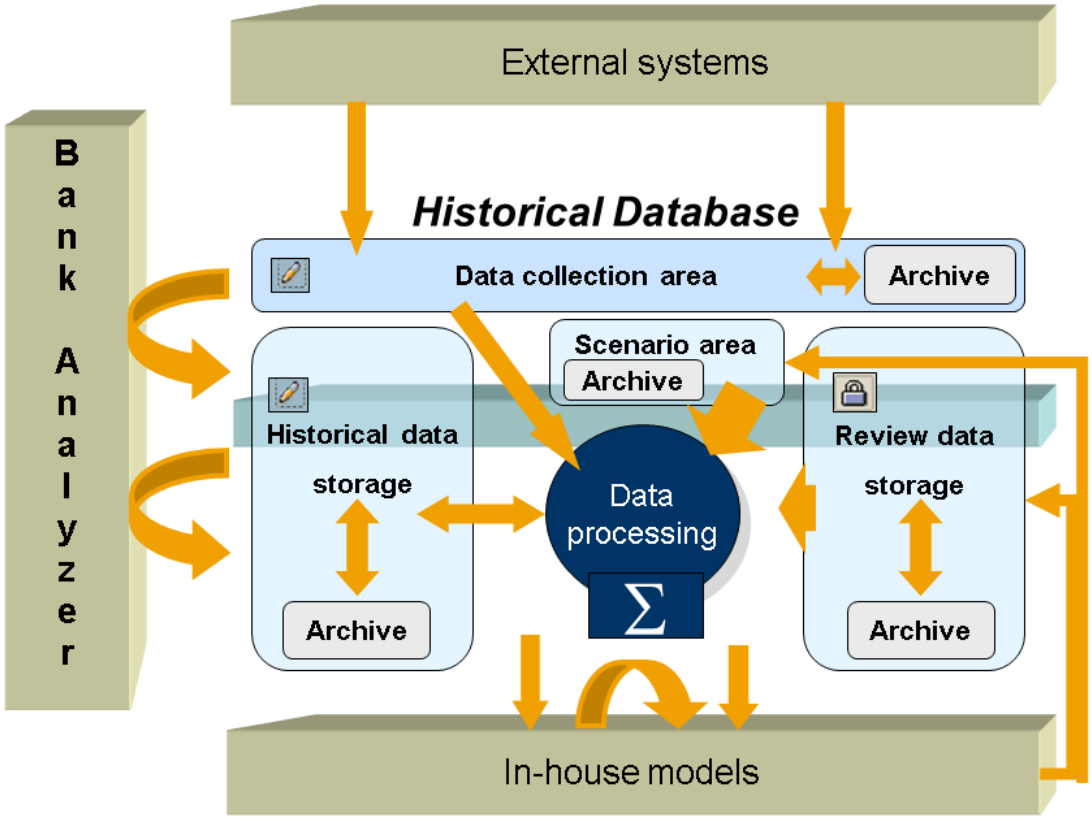


You can use ad hoc reporting [page 1517] to display the data stored in the HDB.

Features

The Historical Database contains the following functions, which are designed to help you meet the requirements of the Basel II Accord:

- Data collection, for example storage of the results of the calculation process
- Data analysis, for example the calculation of default rates
- Data retrieval, for example the provision of data to the Disclosure and Reporting component
- Generation of historical data, and the temporary storage of data in the following data areas:
  - Data processing storage: This is populated with external data by means of a BAPI.
  - Historical data storage: This is populated with data either by means of a historization run for data layers, or a historization run for the bank's in-house models.
  - Review data storage: This is populated with data from the bank's in-house models by means of a BAPI. The review data storage area includes a log function, which contains data such as input values for example, or information such as how model estimations have been produced. Therefore, it is not possible to change the data stored in the review data storage, even if, for example, incorrect input data was used for a rating. In such a case, you would have to recalculate the rating by using the correct input data. The original data remains available so that it can be reviewed at a future point in time.
  - Scenario storage: This is populated with data from stress tests [page 426] during the stress run for supplying models with data [page 1505].



6.3.1 Generation of Historical Data (Historization)

Definition

The generation of historical data, or historization, is the process in which data is stored in the historical data storage of the Historical Database (HDB) to support compliance with Basel II requirements. The historical data storage is used to create time series for data from the Source Data Layer (SDL) [page 100], the results from the calculation function, data from the review data storage or external data. It is possible to change data that is stored in the historical data storage. The historical data storage is populated with data by the run for generating historical data for data layers, the run for generating historical data for in-house models, and by the results from the runs for the various calculation functions [page 1487].

The system contains the following functions for storing data in the historical data storage:

- Starting the Historization Run for Data Layers [page 1481]
- Starting the Historization Run for Your Bank's In-House Models [page 1484]
- Uploading of Files [page 1486]

Example

Historical data can be generated for the following:

- Data at transaction level
- Data at business partner level
- Data at sector level
- Data at country level
- Results of the calculation process

6.3.1.1 Starting the Historization Run for Data Layers

Prerequisites

You have made the necessary settings to configure the data selection process in Customizing under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection*.

You have made the necessary settings to configure the data layers in Customizing under *Bank Analyzer* > *Analytics* > *Historical Database* > *Set Up Data Layers*.

> **Note**
>
> For more information, see the Customizing documentation for the relevant activities.

Context

To support you in meeting Basel II requirements, the historization run for data layers writes historical data to the Historical Database (HDB). The data layers in the HDB are used to store snapshots of data from particular points in time. This data is then used in other processes, such as reporting, or calculation functions, and can be exported to other systems. This data is then used in other processes, such as reporting and in various calculations, and can be exported to other systems.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Generate Historical Data (Historization)* > *Start Historization Run for Data Layers*.

   The *Start Historization Run for Data Layers* screen appears.
2. Make the following entries in the *ID of HDB Data Layer* area of the screen:
   - Enter the *ID of Layer* to identify the data layer. By entering the layer ID, you specify the target for the data.
   - If required, enter the ID of the data target. You need to specify the ID of the data target only if you want to use distributed data targets. For more information, see Customizing for the Historical Database under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *Edit Distributed Data Targets*.
   - If you want to start a *delta run* that is to consider only a subset of the relevant characteristic values in order to supplement the results already stored, for example, then set the appropriate checkbox. The checkbox is set by default.
   - The *Use Key Date as Base Date* checkbox is set by default. Deselect the checkbox if you want to use a specific date for the data records that are to be stored in the Result Database (RDB).
   - Enter the name of the key date of access to the data source. Today's date is used as the default value for this field. For more information about the key date, see Key Date Queries and Delta Queries [page 267].
   - In the *System Time of Worklist* field, enter the system date and time from which, and through to which, data changes are to be taken into account. If a delta query can be used with the primary data source in question, you can start the query for the time period you require by entering values for both time points. For more information about the primary data sources, see Servers for General Primary Data Sources [page 1597].
   - In the *System Time of Enrichment* field enter the time point that the system is to use to access the primary data sources of the enrichment that are not in the same data read group as the primary data sources accessed by the system time of the worklist.
3. Make the following entries in the *Data Collection* data group. Use the group ID and the selection ID to define the data source and the processing steps:
   - Group ID of the selection

- Selection ID for primary and secondary data sources
- You can use selection criteria to define which data is to be selected, for example, by specifying the business partner. We recommend that you use selection criteria, particularly for test runs.
- Once you have entered values for all the required entry fields, you can display all the relevant Customizing settings in a dialog box by choosing *Display Details*. Choose the respective tab page:
  - *Characteristics and Key Figures*
  - *Selection*
  - *Selection Options: BAdI* (selection options for worklist generation and BAdI calls)
  - *Data Target Information* (custom fields that contain additional information, which you have stored in Customizing for the data layer)

4. If required, make the following settings in the *Technical Settings* area of the screen:
   - Set the *Parallel Processing* checkbox to speed up processing. It is advisable to use the parallel processing function, particularly if you are processing a large volume of data.
   - Set the *Test Run* checkbox in order to simulate the run. The system does not write the results of the test run to the database. They are simply displayed as a list. If required, you can enter a particular layout that is to be used when the system displays the results data.
   - Enter a filter variant ID to define how messages are stored and displayed in the run administration function. You define filter variants in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages*.
5. In the *Free Selection Characteristics* data group, you can enter more characteristics from the characteristic repository. The system transfers these characteristics through to the secondary data source in the module editor. You can also use a BAdI call to define that when the system generates the worklist, the free selection characteristics are to be used in addition to those you selected by means of the selection criteria. For more information, see Customizing for the Historical Database under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *BAdIs for Selection* > *BAdI: Selection Options for Worklist Definition*.

   The following data categories are available for the free selection characteristics:
   - In the *Date 1* and *Date 2* fields, you can choose characteristics in date format.
   - In the *Time 1* and *Time 2* fields, you can choose characteristics in time format.
   - In the *Field 1* and *Field 2* fields, you can choose any characteristics.

   Then choose Enter to display the names of the characteristics that you have selected.

   Example

   Unlike the selection criteria, which you add by using the *Selection Criteria* pushbutton, you can derive the data for the free selection characteristics dynamically within the variants.
6. Choose *Execute*.

Results

If it was not a test run, the system updates the selected data to the specified data layer and records a log, which enables you to track the status of the run and carry out postprocessing if necessary. You can also display the log in run administration [page 1513].

You can use the results generated by the run for reporting or other analyses. For example, you can display the results in ad hoc reporting in the Historical Database [page 1677] or in the generic ad hoc reporting [page 1649].

6.3.1.2 Starting the Historization Run for Your Bank's In-House Models

Use

Runs for generating historical data (historization runs) for your bank's own in-house models write model-specific data from the review data storage of the Historical Database (HDB) to the historical data storage.

Unlike the historization run for data layers, in this run the target for the data is not a data layer, but the in-house model you configured in the historical data storage. This model layer does not necessarily have to have the same settings as those for the review data storage. It can contain additional fields, such as potential future risk drivers.

Prerequisites

You have made the necessary settings to configure the data selection process in Customizing under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection*.

To configure your in-house models, you have made the necessary settings in Customizing under *Bank Analyzer* > *Analytics* > *Historical Database* > *Register In-house Models*. In particular, you need to have configured a model for the historical data storage.

> **Note**
>
> For more information, see the documentation for the relevant Customizing activities.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Generate Historical Data* > *Start Historization Run for In-House Models*.
   The system displays the *Start Historization Run for In-house Models* screen.
2. Enter the following information in the *ID of In-House Model* area of the screen.
   - Enter the ID of the model to identify the in-house model. By entering the ID of the model, you specify the target for the data.
   - If required, enter the ID of the data target. You need to specify the ID of the data target only if you want to use distributed data targets. For more information, see Customizing for the *Historical Database* under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *Edit Distributed Data Targets*.

- If you want to start a *delta run* that is to consider only a subset of the relevant characteristic values in order to supplement the results already stored, for example, then set the appropriate checkbox. The checkbox is set by default.
- The *Use Key Date as Base Date* checkbox is set by default. Deselect the checkbox if you want to use a specific date for the data records that are to be stored in the Result Database (RDB).
- In the *Key Date* field specify the name of the key date of access to the data source. Today's date is used as the default value for this field. For more information about the key date, see Key Date Queries and Delta Queries [page 267].
- In the *System Time of Worklist* field, enter the system date and time from which, and through to which, data changes are to be taken into account. If a delta query can be used with the primary data source in question, you can start the query for the time period you require by entering values for both time points. For more information about the primary data sources, see Servers for General Primary Data Sources [page 1597].
- In the *System Time of Enrichment* field enter the time point that the system is to use to access the primary data sources of the enrichment that are not in the same data read group as the primary data sources accessed by the *system time of the worklist*.

3. Make the following settings in the *Data Collection* area of the screen. Use the group ID and the selection ID to define the data source and the processing steps:
   - Group ID of the selection
   - Selection ID for primary and secondary data sources
   - You can use *Selection Criteria* to define which data is to be selected, for example, by specifying the business partner. We recommend that you use selection criteria, particularly for test runs.
   - Once you have entered values for all the required entry fields, by choosing *Display Details*, you can display all the relevant Customizing settings in a dialog box. Choose the respective tab page:
     - *Characteristics and Key Figures*
     - *Selection*
     - *Selection Options: BAdI* (selection options for worklist generation and BAdI calls)
     - *Data Target Information* (custom fields that contain additional information, which you have stored in Customizing for the data layer).
4. If required, make the following settings in the *Technical Settings* area of the screen:
   - Set the *Parallel Processing* checkbox to speed up processing. It is advisable to use the parallel processing function, particularly if you are processing a large volume of data.
   - Set the *Test Run* checkbox in order to simulate the run. The system does not write the results of the test run to the database. They are simply displayed as a list. If required, you can enter a particular layout that is to be used when the system displays the results data.
   - Enter a filter variant ID to define how messages are stored and displayed in the run administration function. You define filter variants in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages*.
5. In the *Free Selection Characteristics* data group, you can enter more characteristics from the characteristic repository. The system transfers these characteristics through to the secondary data source in the module editor. You can also use a BAdI call to define that when the system generates the worklist, the free selection characteristics are to be used in addition to those you selected by means of the *Selection Criteria* pushbutton.
   For more information, see Customizing for the Historical Database under *Bank Analyzer* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *BAdIs for Selection* > *BAdI: Selection Options for Worklist Definition*.
   The following data categories are available for the free selection characteristics:
   - In the *Date 1* and *Date 2* fields, you can choose characteristics in date format.

- In the *Time 1* and *Time 2* fields, you can choose characteristics in time format.
- In the *Field 1* and *Field 2* fields, you can choose any characteristics.
  Then choose *Enter* to display the names of the characteristics that you have selected.

> **Recommendation**
>
> Unlike the selection criteria, which you add by using the *Selection Criteria* pushbutton, you can derive the data for the free selection characteristics dynamically within the variants.

6. Choose  .

Result

If it was not a test run, then the system updates the selected data to the historical data storage of the HDB, and records a log, which enables you to track the status of the run and carry out postprocessing if necessary. You can also display the log in run administration [page 1513].

You can use the results generated by the run for reporting or other analyses. For example, you can display the results in ad hoc reporting in the Historical Database [page 1516] or in the generic ad hoc reporting [page 1649].

6.3.1.3 Uploading of Files

Use

You can use this function to import files from other systems into the Historical Database in Bank Analyzer. You can import files in text format or XML format.

> **Note**
>
> The uploading of files is a front-end service that cannot be used for background processing in batch mode.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Generate Historical Data* > *Upload Files*.
   The *GUI of File Upload* screen appears.
2. Make the following entries in the *Data Target, Key Date, and Time Stamp* data group:
   - Enter the *ID of the external system*.
   - The *Use Key Date as Base Date* checkbox is set by default. Deselect the checkbox if you want to enter your own date for the data records that are to be stored in the Result Database (RDB).

- In the *Key Date* field specify the name of the key date of access to the data source. Today's date is used as the default value for this field. For more information about the key date, see the document Key Date Queries and Delta Queries [page 267].
- Once you have entered values for all the required entry fields, by choosing *Display Details,* you can display all the relevant Customizing settings in a dialog box. Choose the respective tab page:
  - *Characteristics and Key Figures*
  - *Data Target Information* (custom fields that contain additional information, which you have stored in Customizing for the data layer)
- Choose the *Edit Field List* button to enter the following settings. To define the field list, you need to have already entered the settings required in the *Settings for Upload* data group.
  - Text file: Define the order of the fields in which the data in the text file is to be stored. Use tabs to separate the fields of the text file, and make sure that the text file does not contain a header row.
  - XML file: In the field list, define which fields are to be populated with data from the XML file during the upload.
- In the *Settings for Upload* data group, enter the name of the file that you want to import into the Historical Database. Then choose `Enter` so that the data for the *directory*, *file type* and *character format* is displayed automatically.

3. If required, make the following settings in the *Technical Settings* area of the screen:
   To simulate the upload of the file, set the *Test Run* checkbox.The file is not imported into the Historical Database, but is displayed on the screen as a list. If required, you can enter a particular layout that is to be used when the system displays the results data.

6.3.2 Calculation Functions

In addition to the historization run for data layers [page 1481], the system also contains special forms of historization runs for particular calculations. The historization runs for calculation functions are used to determine:

- Default rates
- Default figures (for example, the recovery rate)
- Migration matrices [page 1496]

The results of the runs for the calculation functions are stored in the historical data storage part of the Historical Database, and are then available for use in the Disclosure and Reporting [page 1527] component.

6.3.2.1 Default Rates

Definition

The system can calculate historical default rates to suppport compliance with Basel II requirements. From these historical default rates, the system is then able to calculate the average value. It can calculate both the arithmetic mean of the default rates, and the weighted average.

Use

There are two methods for determining default rates. The methods are described in the following documents:

- Determining Default Rates [page 1488]
- Determining Average Default Rates [page 1491]

Structure

Central to the determination of default rates is the definition of a time period over which data is to be analyzed. This time period is referred to as the analysis period. It is defined by the start date and the end date of the period.

For a time-based analysis, such as the run for determining historical default rates, for example, you need to specify the start date and the end date of the time period (analysis period) over which data is to be analyzed. In addition, the end date for outflows represents a threshold date, beyond which a contract must still be valid if it is to be included in the analysis.

The runs for determining historical default rates and average default rates differ in the level of detail of their input data. Historical default rates are calculated on the basis of financial transactions and business partners, whereas average default rates are calculated on the basis of default rates that have already been determined. Hence you need to specify the end date for outflows for the calculation of historical default rates only. You define which formulas the system is to use by means of the various modules in Customizing for Module Editor.

Example

In Customizing for the *Historical Database* under *Bank Analyzer* > *Analytics* > *Historical Database* > *Configure Calculation Functions* > *BAdI for Analysis Period*, you have set up a BAdI implementation for the analysis period. The analysis period is defined by the following:

- Start of the period: January 1, 2005
- End of the period: December 31, 2005
- End date for outflows: March 31, 2005

In the run for determining default rates, the system includes only those transactions that have defaulted between the start date of the analysis period and the end date of the end date of the analysis period, and for which the contract relationship is still valid after the end date for outflows.

6.3.2.1.1 Determining Default Rates

Use

When the system calculates default rates, it uses historical default rates, and stores the results in the historical data storage area of the Historical Database (HDB).

Historical default rates are calculated after the analysis period has passed. The default rates show which percentage of existing objects (such as financial transactions, and business partners) actually defaulted in the past analysis period.

The system uses the modules from the Module Editor and the formula below, which you have to define, to calculate historical default rates:

$$DR_{i,j} = \frac{\text{Number of defaulted entities of a class in the analysis period}}{\text{Total number of entities in a class at the start of the analysis period}} \cdot 100$$

Dr i,j: actual default rate

Index i: rating segment, for example

Index j: class of the rating segment, for example

Prerequisites

The run for determining historical default rates is a special version of the historization run for data layers. In addition to the prerequisites for the historization run for data layers [page 1481], you need to ensure that the following requirements are met:

- In the General Calculation and Valuation Methods in Bank Analyzer, you need to have already applied the Determination of Default method. For more information, see Determination of Default [page 411].
- In Customizing for the *Historical Database*, you have made settings in the activities under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *Settings for Aggregation Process*.
- If you want to use your own BAdI implementation for selecting data, you have made Customizing settings for the *Historical Database* in the activities under *Bank Analyzer* > *Analytics* > *Historical Database* > *Configure Calculation Functions* > *BAdI for Analysis Period*.

> **Note**
>
> For more information, see the documentation for the relevant Customizing activities.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Calculation Functions* > *Determine Default Rates*.
   The system displays the *Determine Default Rates* screen.
2. Make the following entries in the *ID of HDB Data Layer* area of the screen:
   - Enter the *ID of Layer* to identify the data layer. By entering the ID of the layer, you specify the data layer that the system is to use to store the historical default figures.
   - If required, in the *ID of Data Target* field, specify the ID of the data target. You need to specify the ID of the data target only if you want to use distributed data targets. For more information, see Customizing for the

Historical Database under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *Edit Distributed Data Targets*.

- In the *Key Date* field specify the name of the key date of access to the data source. Today's date is used as the default value for this field. For more information about the key date, see Key Date Queries and Delta Queries [page 267].
- In the *System Time of Worklist* field, enter the system date and time from which, and through to which, data changes are to be taken into account. If a delta query can be used with the primary data source in question, you can start the query for the time period you require by entering values for both time points. For more information about the primary data sources, see Servers for General Primary Data Sources [page 1597].
- In the *System Time of Enrichment* field enter the time point that the system is to use to access the primary data sources of the enrichment that are not in the same data read group as the primary data sources accessed by the *system time of the worklist*.

3. Make the following settings in the *Data Collection* data group. Use the group ID and the selection ID to define the data source and the processing steps:
   - Group ID of the selection
   - Selection ID for primary and secondary data sources
   - You can use *Selection Criteria* to define which data is to be selected, for example, by specifying the business partner. We recommend that you use selection criteria, particularly for test runs.
   - Once you have entered values for all the required entry fields, by choosing *Display Details*, you can display the relevant Customizing settings in a dialog box. Choose the respective tab page:
     - *Characteristics and Key Figures*
     - *Selection*
     - *Selection Options: BAdI* (selection options for worklist generation and BAdI calls)
     - *Data Target Information* (custom fields that contain additional information, which you have stored in Customizing for the data layer)
4. Make the following settings in the *Analysis Period* area of the screen. By specifying the start of the period and the end of the period, you define the analysis period.
   - Start of Period
   - End of Period
   - End Date for Outflows
   - Selection Category (BAdI ID) If you enter a BAdI implementation that you set up in Customizing, then you can use this to refine the set of data selected for the worklist.
5. If required, make the following settings in the *Technical Settings* area of the screen:
   - Set the *Parallel Processing* checkbox to speed up processing. It is advisable to use the parallel processing function, particularly if you are processing a large volume of data.
   - Set the *Test Run* indicator to simulate the determination of the default event. The system does not write the results of the test run to the database. They are simply displayed as a list. If required, you can enter a particular layout that is to be used when the system displays the results data.
   - Enter a filter variant ID to define how messages are stored and displayed in the run administration function. You define filter variants in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages*.
6. In the *Free Selection Characteristics* data group, you can enter more characteristics from the characteristic repository. The system transfers these characteristics through to the secondary data source in the module editor. You can also use a BAdI call to define that when the system generates the worklist, the free selection characteristics are to be used in addition to those you selected by means of the *Selection Criteria* pushbutton.

For more information see Customizing for the *Historical Database* under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *BAdIs for Selection* > *BAdI: Selection Options for Worklist Definition*.

The following data categories are available for the free selection characteristics:

- In the *Date 1* and *Date 2* fields, you can choose characteristics in date format.
- In the *Time 1* and *Time 2* fields, you can choose characteristics in time format.
- In the *Field 1* and *Field 2* fields, you can choose any characteristics.

Then choose *Enter* to display the names of the characteristics that you have selected.

> **Recommendation**
>
> Unlike the selection criteria, which you add by using the *Selection Criteria* pushbutton, you can derive the data for the free selection characteristics dynamically within the variants.

7. Choose .

Result

If it was not a test run, then the system updates the selected data to the specified data layer and records a log, which enables you to track the status of the run and carry out postprocessing if necessary. You can also display the log in run administration [page 1513].

You can use the results generated by the run for reporting or other analyses. For example, you can display the results in ad hoc reporting in the Historical Database [page 1516] or in the generic ad hoc reporting [page 1649].

6.3.2.1.2 Determining Average Default Rates

Use

In the run for determining average default rates, the system calculates the average value from the historical default rates for a time period you specify, and then stores the results in the historical data storage area of the Historical Database.

Average default rates can be calculated as follows:

- By calculating the arithmetic mean of the historical default rates
- By calculating the weighted average of the historical default rates

The system uses various formulas to calculate these values, based on the Customizing settings for Module Editor and the aggregation function.

The system uses the following formula to calculate the arithmetical mean of the historical default rates over a particular period of time, and for each rating segment and class. You must define the formula appropriately:

$$\varnothing DR_{i,j} = \frac{\sum_{k=1}^{N} DR_{i,j,k}}{N}$$

- $DR_{i,j}$: average daily default rate

Index i: rating segment, for example

Index j: class of the rating segment, for example

Index k: year of the default rate

N: number of default rates analyzed

The system uses the following formula to calculate the weighted average of the historical default rates over a time period of several years, and for each rating segment and class. You must define the formula appropriately:

$$\varnothing DR_{i,j} = \frac{\sum_{k=1}^{N_{i,j}} a_k DR_{i,j,k}}{\sum_{k=1}^{N_{i,j}} a_k}$$

- $DR_{i,j}$: weighted average default rate

Index i: rating segment, for example

Index j: class of the rating segment, for example

a_k: weighting factor

Constraint: $a_k \geq$ zero

Index k: year of the default rate

N: number of default rates analyzed

You can define the weighting factors used here yourself.

Procedure

First the system calculates the default rates, and then it calculates the average default rates. The procedures are similar. Therefore, you should proceed as described in the documentation about the determination of default rates [page 1488].

6.3.2.2 Determining Default Figures

Use

In the run for determining default figures, the system calculates various figures for defaulted transactions, and stores the results in the historical data storage area of the Historical Database.

Examples of such figures are:

- Quotients from estimates for the probability of default and actual default rates
- Actual recovery rates
- Actual or average drawing rates

Prerequisites

The calculation of default figures is a special version of the historization run for data layers. In addition to the prerequisites for the historization run for data layers [page 1481], you need to ensure that the following requirements are met:

In Customizing for the *Historical Database*, you have made settings in the activities under *Bank Analyzer* > *Analytics* > *Historical Database* > *Calculation Functions*.

> **Note**
>
> For more information, see the documentation for the relevant Customizing activities.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Calculation Functions* > *Determine Default Figures*.
   The system displays the *Determination of Default Figures* screen.
2. Make the following entries in the *ID of HDB Data Layer* area of the screen:
   - Enter the *ID of Layer* to identify the data layer. By entering the ID of the layer, you specify the data layer that the system is to use to store the default figures.
   - If required, in the *ID of Data Target* field, specify the ID of the data target. You need to specify the ID of the data target only if you want to use distributed data targets. For more information, see Customizing for the

Historical Database under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *Edit Distributed Data Targets*.

- In the *Key Date* field specify the name of the key date of access to the data source. Today's date is used as the default value for this field. For more information about the key date, see Key Date Queries and Delta Queries [page 267].
- In the *System Time of Worklist* field, enter the system date and time from which, and through to which, data changes are to be taken into account. If a delta query can be used with the primary data source in question, you can start the query for the time period you require by entering values for both time points. For more information about the primary data sources, see Servers for General Primary Data Sources [page 1597].
- In the *System Time of Enrichment* field enter the time point that the system is to use to access the primary data sources of the enrichment that are not in the same data read group as the primary data sources accessed by the *system time of the worklist*.

3. Make the following settings in the *Data Collection* data group. Use the group ID and the selection ID to define the data source and the processing steps:
   - Group ID of the selection
   - Selection ID for primary and secondary data sources
   - You can use *Selection Criteria* to define which data is to be selected, for example, by specifying the business partner. We recommend that you use selection criteria, particularly for test runs.
   - Once you have entered values for all the required entry fields, by choosing *Display Details*, you can display all the relevant Customizing settings in a dialog box. Choose the respective tab page:
     - Characteristics and Key Figures
     - Selection
     - Selection Options: BAdI (selection options for worklist generation and BAdI calls)
     - Data Target Information (custom fields that contain additional information, which you have stored in Customizing for the data layer)
4. If required, make the following settings in the *Technical Settings* area of the screen:
   - Set the *Parallel Processing* checkbox to speed up processing. It is advisable to use the parallel processing function, particularly if you are processing a large volume of data.
   - Set the *Test Run* indicator in order to simulate the run for the calculation functions. The system does not write the results of the test run to the database. They are simply displayed as a list. If required, you can enter a particular layout that is to be used when the system displays the results data.
   - Enter a filter variant ID to define how messages are stored and displayed in the run administration function. You define filter variants in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages*.
5. In the *Free Selection Characteristics* data group, you can enter more characteristics from the characteristic repository. The system transfers these characteristics through to the secondary data source in the module editor. You can also use a BAdI call to define that when the system generates the worklist, the free selection characteristics are to be used in addition to those you selected by means of the *Selection Criteria* pushbutton.
   For more information see Customizing for the *Historical Database* under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *BAdIs for Selection* > *BAdI: Selection Options for Worklist Definition*.
   The following data categories are available for the free selection characteristics:
   - In the *Date 1* and *Date 2* fields, you can choose characteristics in date format.
   - In the *Time 1* and *Time 2* fields, you can choose characteristics in time format.
   - In the *Field 1* and *Field 2* fields, you can choose any characteristics.

Then choose *Enter* to display the names of the characteristics that you have selected.

> **Note**
>
> Unlike the selection criteria, which you add by using the *Selection Criteria* pushbutton, you can derive the data for the free selection characteristics dynamically within the variants.

6. Choose .

Result

If it was not a test run, then the system updates the selected data to the historical data storage of the HDB, and records a log, which enables you to follow the status of the run and carry out postprocessing if necessary. You can also display the log in run administration [page 1513].

You can use the results generated by the run for reporting or other analyses. For example, you can display the results in ad hoc reporting in the Historical Database [page 1516] or in the generic ad hoc reporting [page 1649].

Examples:

Calculation of Actual Recovery Rates

The cash flow from the recovery of the defaulted facility is imported from your front-end system, and stored at single-transaction level. In Customizing for the *Historical Database*, you have already configured the aggregation function and the modules for the Module Editor so that the system carries out the following steps in the run for determining default figures:

- Sorting of cash flows by the type of payment
- Determination of the recovery period in months
- Storage of the estimated values where the recovery is not completely settled
- Use of the SAP price calculator to determine the NPV
- Calculation of the recovery rate as a quotient of the NPV of the recovery cash flow and the drawing on the key date.

Calculation of Actual Drawing Rates

During the determination of default events, it was found that an object is in default. You have made settings for the modules of the Module Editor in Customizing for the Historical Database so that the system reads the following values from the source systems (SDL or HDB) for determining actual drawing rates:

- Drawing on the facility at the time of default (from the SDL)
- Drawing on the facility one year before the time of default (from the SDL)
- Free line of the facility one year before the time of default (from the HDB)

Then the following formula is used in the Module Editor to calculate the drawing rate:

$$ZQ = \frac{LA_t}{FL_{t-1} + LA_{t-1}}$$

DR : Actual drawing rate

D_t: Utilization at the time of default

D_{t-1}: Utilization one year before the default

FL_{t-1}: Free line one year before the default

6.3.2.3 Calculation of Migration Matrices

Use

In this report, the system calculates various types of matrix (migration matrix, portfolio matrix, preliminary versions of these matrices). You can display the result matrix as a table in the test run or, in the update run, you can store it in an HDB data layer in the historical data storage.

For more information about the calculation of migration matrices, see the *Program Documentation*.

Prerequisites

In Customizing for Bank Analyzer you have entered the settings required under *Analytics* > *Historical Database* > *Configure Calculation Functions* > *Edit Acquisition Matrix and Migration Matrix*.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Calculation Functions* > *Calculate Migration Matrices*.
   The *Migration Matrices* screen appears.
2. Make the following entries in the *HDB Data Layer* and *Matrix ID* data group:
   - Enter the *ID of Layer* to identify the data layer. By entering the ID of the layer, you specify the data layer in which the system is to store the results.
   - Enter the matrix ID and the result matrix.
   - Enter the evaluation currency.
   - Enter the name of the key date of access to the data source. Today's date is used as the default value for this field. For more information about the key date, see Key Date Queries and Delta Queries [page 267].
   - In the *System Time of Worklist* field, enter the system date and time from which, and through to which, data changes are to be taken into account. If a delta query can be used with the primary data source in question, you can start the query for the time period you require by entering values for both time points. For more information about the primary data sources, see Servers of the General Primary Data Sources [page 1597].

- In the *System Time of Enrichment* field enter the time point that the system is to use to access the primary data sources of the enrichment that are not in the same data read group as the primary data sources accessed by the *system time of the worklist*.

3. In the *Analysis Period* data group, specify the analysis period required.
4. Make the following entries in the *Data Collection* data group. Use the *Group ID of Selection* and the *Selection ID* fields to define the data source and the processing steps:
   - Group ID of the selection
   - Selection ID for primary and secondary data sources
   - You can use *Selection Criteria* to define which data is to be selected, for example, by specifying the business partner. We recommend that you use selection criteria, particularly for test runs.
   - Once you have entered values for all the required entry fields, you can display all the relevant Customizing settings in a dialog box by choosing *Display Details*. Choose the respective tab page:
     - *Characteristics and Key Figures*
     - *Selection*
     - *Selection Options: BAdI* (selection options for worklist generation and BAdI calls)
     - *Data Target Information* (custom fields that contain additional information, which you have stored in Customizing for the data layer)
5. If required, make the following settings in the *Technical Settings* area of the screen:
   - Set the *Parallel Processing* checkbox to speed up processing. It is advisable to use the parallel processing function, particularly if you are processing a large volume of data.
   - Set the *Test Run* checkbox in order to simulate the run. The system does not write the results of the test run to the database. They are simply displayed as a list. If required, you can enter a particular layout that is to be used when the system displays the results data.
   - Enter a filter variant ID to define how messages are stored and displayed in the run administration function. You define filter variants in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages*.
6. In the *Free Selection Characteristics* data group, you can enter more characteristics from the characteristic repository. The system transfers these characteristics through to the secondary data source in the module editor. You can also use a BAdI call to define that when the system generates the worklist, the free selection characteristics are to be used in addition to those you selected by means of the selection criteria. For more information see Customizing for the Historical Database under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *BAdIs for Selection* > *BAdI: Selection Options for Worklist Definition*.
   The following data categories are available for the free selection characteristics:
   - In the *Date 1* and *Date 2* fields, you can choose characteristics in date format.
   - In the *Time 1* and *Time 2* fields, you can choose characteristics in time format.
   - In the *Field 1* and *Field 2* fields, you can choose any characteristics.

   Then choose **Enter** to display the names of the characteristics that you have selected.
   Unlike the selection criteria, which you add by using the *Selection Criteria* pushbutton, you can derive the data for the free selection characteristics dynamically within the variants.
7. Choose *Execute* .
   The system calculates the migration matrix required.

6.3.3 Data Retrieval

As part of data retrieval, you can import data into the Historical Database and carry out stress tests.

Data retrieval contains the following functions:

- Exporting Data to In-House Models [page 1498]
- Downloading of Files [page 1500]
- Support for Stress Tests in the Historical Database [page 1505]

6.3.3.1 Exporting Data to In-House Models

Use

By means of the runs for supplying data to in-house models, the system exports data to your bank's own models. Once the data has been processed by your in-house model, a BAPI can be used to re-import the data into the HDB, where it is then stored in the review data storage. Unlike the historization run for data layers, in this run the target for the data is not a data layer, but the in-house model you configured in review data storage.

Prerequisites

You need to have already made the necessary settings in Customizing for the configuration of the data selection process under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection.*

You have made the required Customizing settings for in-house models in Customizing for the *Historical Database* under *Bank Analyzer* > *Analytics* > *Historical Database* > *Register In-House Models.*

- In particular, for the run for supplying data to in-house models, you need to have configured a model for the review data storage.
- To refine the settings that govern how data is supplied to in-house models, you can implement a BAdI on the level of the version of your in-house model.

> **Note**
>
> For more information, see the Implementation Guide (IMG) for the relevant IMG activities.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Data Retrieval* > *Start Export of Data to In-House Model.*
   The system displays the *Start Export of Data to In-House Model* screen.

2. Enter the following information in the *ID of In-House Model* data group:
   - Enter the ID of the model to identify the in-house model.
   - Specify the version of the model .
   - If required, in the *ID of Data Target* field specify the ID of the data target. You need to specify the ID of the data target only if you want to use distributed data targets. For more information see the section in the Implementation Guide (IMG) for the Historical Database under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *Edit Distributed Data Targets.*
   - If you have set up your own BAdI implementation for the in-house model version, by entering the filter ID of the BAdI in the *Filter ID for Model Version BAdI* field you can assign the BAdI implementation to the run. If you do not enter the filter ID, the system takes the assignment that you defined in Customizing for the *Historical Database* under *Register In-House Models* > *BAdI for In-House Model Version* > *Assign In-House Model Version to BAdI Filter ID.*
   - In the *Key Date* field specify the name of the key date of access to the data source. Today's date is used as the default value for this field. For more information about the key date, see the document Key Date Queries and Delta Queries [page 267] .
   - In the *System Time of Worklist* field, enter the system date and time from which, and through to which, data changes are to be taken into account. If a delta query can be used with the primary data source in question, you can start the query for the time period you require by entering values for both time points. For more information about the primary data sources, see the document Servers of the General Primary Data Sources [page 1597] .
   - In the *System Time of Enrichment* field enter the time point that the system is to use to access the primary data sources of the enrichment that are not in the same data read groupas the primary data sources accessed by the *system time of the worklist* .
3. Make the following entries in the *Data Collection* data group. Use the *Group ID of Selection* and the *Selection ID* fields to define the data source and the processing steps:
   - If required, set the *Selection ID from Model Customizing* indicator.
   - Or, you can specify the selection ID by entering it in the *Group ID of Selection* field, and by specifying the selection ID (*Selection ID* field) for the primary and secondary data sources.
   - You can use *Selection Criteria* to define which data is to be selected, for example, by specifying the business partner. We recommend that you use selection criteria, particularly for test runs.
   - Once you have entered values for the all the required entry fields, by choosing *Display Details* you can display all the relevant Customizing settings in a dialog box. Choose the respective tab page:
     - *Characteristics and Key Figures*
     - *Selection*
     - *Selection Options* : *BAdI* (selection options for worklist generation and BAdI calls)
     - *Data Target Information* (custom fields that contain additional information, which you have stored in Customizing for the data layer).
4. If required, make the following settings in the *Technical Settings* area of the screen:
   - Set the *Parallel Processing* indicator to speed up processing. It is advisable to use the parallel processing function, particularly if you are processing a large volume of data.
   - Set the *Test Run* indicator in order to simulate the run. The system does not export the results of the test run to the in-house model. They are simply displayed as a list. If required, you can enter a particular layout that is to be used when the system displays the results data.

- Enter a filter variant ID to define how messages are stored and displayed in the run administration function. You define filter variants in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages.*

5. In the *Free Selection Characteristics* data group, you can enter more characteristics from the characteristic repository. The system transfers these characteristics through to the secondary data source in the Module Editor. You can also use a BAdI call to define that when the system generates the worklist, the free selection characteristics are to be used in addition to those you selected by means of the *Selection Criteria* pushbutton. For more information see the section in the Implementation Guide (IMG) for the Historical Database under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *BAdIs for Selection* > *BAdI: Selection Options for Worklist Definition.*
   The following data categories are available for the free selection characteristics:
   - In the *Date 1* and *Date 2* fields you can choose characteristics in date format.
   - In the *Time 1* and *Time 2* fields you can choose characteristics in time format.
   - In the *Field 1* and *Field 2* fields, you can choose any characteristics.
     Then choose **Enter** to display the names of the characteristics that you have selected.

   > **Recommendation**
   >
   > Unlike the selection criteria, which you add by using the *Selection Criteria* pushbutton, you can derive the data for the free selection characteristics dynamically within the variants.

6. Choose with the quick info *Execute* .

Result

If it was not a test run, the system exports the selected data to the specified in-house model, and records a log, which enables you to track the status of the run and carry out postprocessing if necessary. You can also display the log in run administration [page 1513] .

6.3.3.2 Downloading of Files

Use

You can use this function to provide other systems with files from the Historical Database in Bank Analyzer. You can export files in text format or in XML format.

> **Note**
>
> The downloading of files is a front-end service that cannot be used for background processing in batch mode.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Data Retrieval* > *Download Files*.
   The *GUI of File Download* screen appears.
2. Make the following entries in the *Data Collection* data group:
   - Use the *Group ID of Selection* and the *Selection ID* fields to define the primary and secondary data sources and the processing steps.
   - If you want to export data from stress tests [page 1505], specify the ID of the *composite scenario*. The system uses this ID when it reads the data.
   - In the *Key Date* field specify the name of the key date of access to the data source. Today's date is used as the default value for this field. For more information about the key date, see the document Key Date Queries and Delta Queries [page 267].
   - In the *System Time of Worklist* field, enter the system date and time from which, and through to which, data changes are to be taken into account. If a delta query can be used with the primary data source in question, you can start the query for the time period you require by entering values for both time points. For more information about the primary data sources, see the document Servers of the General Primary Data Sources [page 1597].
   - In the *System Time of Enrichment* field enter the time point that the system is to use to access the primary data sources of the enrichment that are not in the same data read group as the primary data sources accessed by the *system time of the worklist*.
   - You can use *Selection Criteria* to define which data is to be selected, for example, by specifying the business partner. We recommend that you use selection criteria, particularly for test runs.
   - Once you have entered values for all the required entry fields, by choosing *Display Details*, you can display all the relevant Customizing settings in a dialog box. Choose the respective tab page:
     - *Selection*
     - *Selection Options: BAdI* (selection options for worklist generation and BAdI calls)
     - *Data Target Information* (custom fields that contain additional information, which you have stored in Customizing for the data layer)
   - You use the *Field List* pushbutton to define the fields that are written to the text file or XML file, and the order in which they appear.
3. In the *Settings for Download* data group, enter the file name under which the data downloaded from the HDB is to be stored. Then choose `Enter` so that the data for the *directory*, *file type* and *character format* is displayed automatically.
4. If required, make the following settings in the *Technical Settings* area of the screen:
   - Set the *Parallel Processing* checkbox if the data is be exported package by package. This means that if you choose text format, the data packages are grouped together in one results file. If you choose XML, the program generates a separate file for each package. The file names are numbered sequentially.
   - To simulate the download of the data selected, set the *Test Run* checkbox. The file is not exported from the Historical Database, but is displayed on the screen as a list. If required, you can enter a particular *layout* that is to be used when the system displays the results data.
   - Enter a filter variant ID to define how messages are stored and displayed in the run administration function. You define filter variants in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages*.
5. In the *Free Selection Characteristics* data group, you can enter more characteristics from the characteristic repository. The system transfers these characteristics through to the secondary data source in the module editor. You can also use a BAdI call to define that when the system generates the worklist, the free selection

characteristics are to be used in addition to those you selected by means of the *Selection Criteria* pushbutton.
For more information see Customizing for the *Historical Database* under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *BAdIs for Selection* > *BAdI: Selection Options for Worklist Definition*.
The following data categories are available for the free selection characteristics:

- In the *Date 1* and *Date 2* fields, you can choose characteristics in date format.
- In the *Time 1* and *Time 2* fields, you can choose characteristics in time format.
- In the *Field 1* and *Field 2* fields, you can choose any characteristics.

Then choose `Enter` to display the names of the characteristics that you have selected.

> **Recommendation**
>
> Unlike the selection criteria, which you add by using the *Selection Criteria* pushbutton, you can derive the data for the free selection characteristics dynamically within the variants.

6. Choose *Execute*.

6.3.3.3 Stress Tests

Use

To meet the requirements of Basel II, banks that use the IRB approach must have sound stress test methods that enable them to assess their total capital adequacy. The banks are responsible for configuring the stress tests. The tests must be both meaningful and conservative.

Stress tests are also of interest to banks that use the standardized approach if, for example, migration analyses are to be carried out for external ratings. Banks that use the standardized approach are also required to assess their capital adequacy. Stress tests can be used for this purpose.

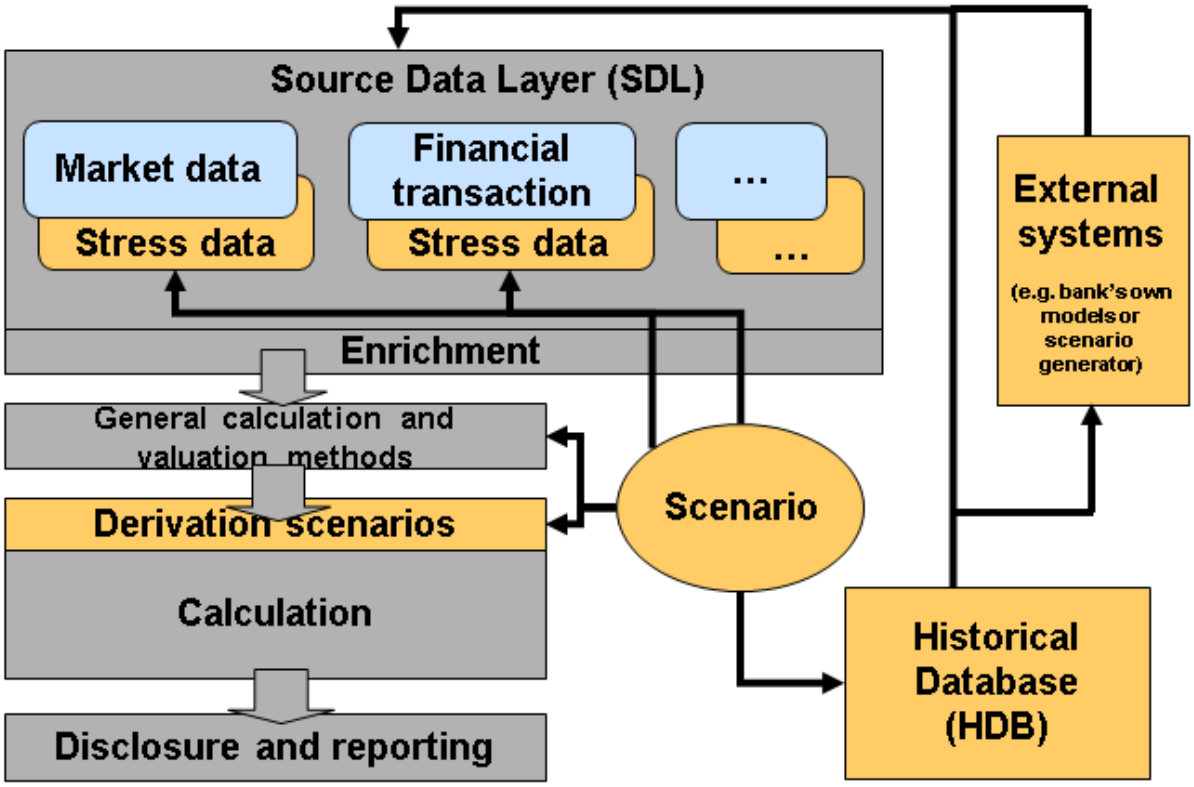
Stress tests use input data and control parameters that have been modified to reflect the (possibly hypothetical) result of a negative change in the economic conditions.

The following are examples of stress tests for Basel II:

- Modification of classic market data, such as exchange rates, interest rates, or security prices, and the effects of these on the market value of derivatives
- Modification of the PD and LGD as the result of a crisis in a particular sector or country
- Worsening of the ratings of counterparties in particular countries

Integration

The figure below shows which Bank Analyzer functions are used in stress tests.



Framework for Mapping Stress Tests

- Stress test scenarios in the Source Data Layer (SDL), for example:
  - Assumption of a worsening credit rating
  - Modified maturities
  - Modified positions
  - Modified market data

  For more information, see Scenario [page 1731] and Retrieving Stress Data [page 1734].
- Stress test scenarios in the calculation process (using derivation or using modules from the module editor), for example:
  - Country-specific rating changes

  You can carry out stress runs for the general functions and for the credit exposure calculation. For more information, see Support for Stress Tests in General Calculation and Valuation Methods [page 429] and Credit Exposure Stress Run [page 1261].
- Stress test scenarios are supported by the Historical Database (HDB), for example:
  - Modification of the input data used in banks' own models
  - Modification of position data in the Source Data Layer

  The HDB contains stress runs for banks' own models. For more information, see Support for Stress Tests in the Historical Database [page 1505].

You define which Bank Analyzer functions are involved in a stress test by creating a stress test scenario (see Stress Test Scenarios [page 428]). Here you can also combine multiple scenarios. By specifying whether factors are to be stressed implicitly or explicitly, for example, you define which scenario is depicted in a stress run.

6.3.3.3.1 Stress Test Scenarios

Definition

Different scenarios can be used for stress tests. These can be categorized as follows:

- Market data scenarios
  These are based on the use of modified classic or generic market data.
- Class determination scenarios
  These are based on the use of primary object data that has been modified.
- Modification of the input data used in banks' own models

When market data scenarios and class determination scenarios are used, the calculation process uses modified input data, which is stored along with the original data in the Source Data Layer (SDL). For more information, see Scenario [page 1731].

Use

To be able to map scenarios that reflect crises in particular regions or sectors, the scenarios do not always have to include all transactions and business partners during a calculation run. Instead, they can be limited to particular groups or individual entities. The following application areas are available for scenarios:

- **Local**
  Scenarios that are applied locally use data that is manipulated locally. This means that attributes are changed at *single transaction* or *business partner* level.
- **Global**
  Scenarios that are used globally apply modified values to the entire portfolio.
- **Subportfolio**
  Scenarios that are used for subportfolios apply modified values to a group of single transactions.

Integration

Bank Analyzer provides an infrastructure that enables you to combine stress test scenarios You define the combination of scenarios in scenario administration by assigning multiple single or composite scenarios to a composite scenario.

You define composite scenarios in Customizing for *Bank Analyzer* under *Infrastructure* > *General Scenario Administration* > *Edit Composite Scenarios*. You can still change these settings after your system has gone live. You do this on the *SAP Easy Access* screen under *Bank Analyzer* > *Infrastructure* > *General Scenario Administration*.

A composite scenario can contain only one scenario from each data category for scenarios, for example a market data scenario and a business partner scenario.

6.3.3.3.2 Support for Stress Tests in the Historical Database

To support compliance with the requirements of Basel II, banks using the IRB approach must have sound methods for stress testing. To help you to meet these requirements, the Historical Database (HDB) contains the following functions:

- Supply of in-house models with stress data from Bank Analyzer and storage of stress data in the scenario storage area of the HDB. For more information, see Stress Run for Supplying Models with Data [page 1505].
- Generation of stress data that is stored in the Source Data Layer (SDL) and used as the input data for other stress tests. For more information, see Generating Scenario Data in the Source Data Layer [page 1506].
- Import of stress data generated by banks' own scenario generators. A BAPI or the Data Transfer Workbench is used to import the data, and the imported data is then stored in the scenario storage area of the HDB.

6.3.3.3.2.1 Stress Run for Supplying Models with Data

Use

You can use this function to carry out stress tests [page 426] based on real data stored in the Source Data Layer (SDL) or on real data from the review data storage area of the Historical Database (HDB). You can then use these results as input data for your in-house models.

Prerequisites

- If you want to base your stress tests on real data from the review data storage area of the Historical Database, then the review data storage must contain data.
- In Customizing for *Bank Analyzer* under *Infrastructure* > *General Scenario Administration*, you have created scenarios and assigned them to a composite scenario.
- In Customizing for *Bank Analyzer* under *Analytics* > *Historical Database* > *Register In-House Models* > *Derivation Tool Settings for Scenarios*, you have entered the relevant settings for derivation.
  You can still change the rule entries after your system has gone live. You do this on the *SAP Easy Access* screen under *Bank Analyzer* > *Analytics* > *Historical Database* > *Data Retrieval* > *Support for Stress Tests* > *Edit Rule Entry for Derivation Step*.
- The same prerequisites apply for the stress run for supplying in-house models with data as for the update runs. For more information, see Exporting Data to In-House Models [page 1498].

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Data Retrieval* > *Support for Stress Tests* > *Start Stress Run for Export of Data to In-House Model*.

2. To start a stress test to provide data for your in-house models, proceed in the same way as for an update run.
3. In the *ID of In-House Model* section, enter the ID of the composite scenarios [page 428].
4. Choose *Execute*.

The results of the stress run for supplying in-house models with data are stored in the scenario data storage area in HDB. You can also export this stress data to your own statistics tools.

6.3.3.3.2.2 Generating Scenario Data in the Source Data Layer

Use

You can use this function to prepare real data stored in the Source Data Layer (SDL) for stress tests [page 426]. The results are stored in the SDL as stress data, and can be used in other stress tests, such as those in the calculation function.

Prerequisites

- You have created scenarios and assigned them to a composite scenario in Customizing for *Bank Analyzer* under *Infrastructure* > *General Scenario Administration*.
- You have configured a BAdI implementation in Customizing for *Bank Analyzer* under *Analytics* > *Historical Database* > *Register Bank-Internal Models* > *BAdI for SDL Scenarios*. This BAdI is used to read data from the SDL and then to write the stressed data to the SDL. The system contains a sample implementation to help you.
- If required, you have made the relevant derivation settings in Customizing for *Bank Analyzer* under *Analytics* > *Historical Database* > *Register Bank-Internal Models* > *Derivation Tool Settings for Scenarios*.
  You can still change the rule entries after your system has gone live. You do this on the *SAP Easy Access* screen under *Bank Analyzer* > *Analytics* > *Historical Database* > *Data Retrieval* > *Support for Stress Tests* > *Edit Rule Entry for Derivation Step*.
- The same prerequisites apply here as for the update run for supplying in-house models with data. For more information, see the document Exporting Data to In-House Models [page 1498].

Activities

1. On the *SAP Easy Access* screen choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Data Retrieval* > *Support for Stress Tests* > *Generate Scenario Data in the Source Data Layer*.
2. To generate scenario data in the SDL, follow the same procedure as that for starting a run for supplying in-house models with data.
3. Enter the filter ID of the BAdI for SDL scenarios.
4. Enter the IDs of the composite scenarios for reading the data and then writing the stressed data to the SDL. These are the composite scenarios to which the scenarios [page 428] that you want to use are assigned.

5. Choose  .

The results of the stress run are stored in the SDL as stress data, and can be processed further. For more information about how this data can be processed further, see:

- Support for Stress Tests in General Calculation and Valuation Methods [page 429]
- Credit Exposure Stress Run [page 1261]

6.3.4 Changing Data Manually in the Historical Database

Use

Data in the data processing storage or in the historical data storage that was generated automatically can be changed manually. This can be used when data is incorrect, and generating the data again automatically would be too cumbersome, or if the relevant data is no longer available in the feeder systems.

However, the proper procedure is to correct errors in the feeder systems first, and then re-import the data into the Historical Database (HDB), rather than changing data in the HDB manually.

You can also create and delete data manually in the data processing storage and in the historical data storage.

To support compliance with the requirements of Basel II, you **cannot** change data stored in the review data storage.

Prerequisites

Changing Data Manually in the Historical Data Storage

- Before you can change data manually in the historical data storage, you need to configure at least one data layer for the historical data storage. You do this in Customizing for the *Historical Database* under *Bank Analyzer* > *Analytics* > *Historical Database* > *Set Up Data Layers*. Alternatively, you need to configure at least one model for the historical data storage. You do this in Customizing under *Bank Analyzer* > *Analytics* > *Historical Database* > *Register In-House Models*.
- For the data layers, you need to have carried out at least one historization run for data layers [page 1481], or, if you are using your own in-house models, then at least one historization run for in-house models [page 1484].

If you just want to add new data to the historical data storage, then you do not need to have already carried out any of these runs.

- You need to have released the run for manual changes in run administration [page 1513].

Changing Data Manually in the Data Processing Storage

- Before you can change data manually in the data processing storage, you need to register at least one external feeder system for the data processing storage. You do this in Customizing for the *Historical Database* under *Bank Analyzer* > *Analytics* > *Historical Database* > *Register External Systems*.
- You also need to have imported data from an external system to the data processing storage area of the Historical Database. You can import data by using a BAPI or by using the function for uploading files [page

1486]. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Generate Historical Data* > *Upload Files*.

If you just want to add new data to the data processing storage, then you do not need to have already imported data from external systems.

Activities

Selection

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Changes to the Historical Database* > *Change Data Manually*.
   The *Changes in HDB* screen appears. You have the following options:
   In the left-hand part of the screen, the system displays an overview of all the RDB runs. These are sorted by the following criteria:
   - Historization processes
   - Bank's in-house models
   - External systems

   To display the data for a particular RDB run, expand the respective node. To display the data for the HDB runs for a particular RDB run, expand the RDB run.
2. To display the run data in the right-hand part of the screen, double click the required run in the left-hand part of the screen.
   If data exists for the entry selected, the system displays a dialog box in which you can choose a time period, or enter characteristics as selection criteria (for example, you can define that data is displayed only for a particular business partner or time period).

Action

The overview list contains the following pushbuttons:

Pushbutton	Description
Expand All Nodes *Collapse All Nodes*	Expands or collapses the nodes in the left-hand part of the screen (*hierarchy*)
Update Date/Node	Updates the display of the runs in the left-hand part of the screen (*hierarchy*)
Display Description/Hide Description	Displays or hides the description of the nodes in the left-hand part of the screen (*hierarchy*)
Change Data	Displays the data that can be changed for the run selected
Changes: History	Displays the changes that have already been made
Messages	Displays the logs of the selected run in the bottom right-hand part of the screen

The system displays the maximum number of hits in the upper window in the right side of the screen. The default value for the maximum number of hits is 500. To change the maximum number of hits, click the number that is displayed and overwrite it with the number you require.

In the overview list in the lower window in the right part of the screen, you can access the following functions for HDB runs:

Pushbutton	Description
Multiple Selection Mode *Single Selection Mode*	Enables you to delete multiple rows at the same time: • Choose the *Multiple Selection Mode* pushbutton. You can now select and delete multiple rows at the same time. • Choose the *Single Selection Mode* pushbutton to return to the original view, which contains the other pushbuttons.
Change	Enables you to change the data for a row: 1. Select the row in which you want to change data. 2. Choose the pushbutton. 3. The system displays a dialog box in which you change the data as required. It is not possible to change the data in certain key fields, such as *Financial Transaction ID*, *Base Date of the HDB Run*. 4. Choose *Save*. The changes are written to the database.
Change Column	Enables you to change the data in a column: 1. Select the column in which you want to change data. 2. Choose the pushbutton. 3. The system displays a dialog box in which you can enter the new value for all the records selected. 4. Choose *Save*. The changes are written to the database.
Create	1. Choose the pushbutton. 2. In the dialog box that is then displayed, enter the required data. 3. Choose *Save*. The new data record is written to the database.

Pushbutton	Description
Copy	1. Select the row that you want to copy. 2. Choose the pushbutton. 3. The system displays a dialog box in which you enter the data as required. The values of the row that you want to copy are displayed here as default values. This means that you need only to change the values as required, rather than entering new data. 4. Choose *Save*. The new data record is written to the database.
Delete	1. Select the row that you want to delete. 2. Choose the pushbutton. 3. In the dialog box that is then displayed, confirm that the row is to be deleted The data record selected is deleted from the database.
SDL	1. Select the row for which you want to display the transaction data that is stored in the Source Data Layer (SDL). 2. Choose the pushbutton. The system displays the transaction data from the SDL in a new window.

Result

You have changed manually the data stored in the Historical Database, and stored the changes on the database.

If you have set up a release procedure (principle of dual control) then the changes do not take effect immediately. They are released as part of the workflow. Until the changed data is released, the system continues to use the old data record in all applications. This data record cannot be changed again until the first change has been released.

You define the release procedure in the following Customizing activities for the *Historical Database*:

Bank Analyzer > *Analytics* > *Historical Database* > *Set Up Data Layers* > *Release Procedures for Data Layers*

Bank Analyzer > *Analytics* > *Historical Database* > *Register In-house Models* > *Release Procedures for Models*

Bank Analyzer > *Analytics* > *Historical Database* > *Register External Systems* > *Release Procedures for External Systems*

6.3.5 Run Administration

Definition

Run administration includes the following functions:

- Execute or create run

- Display an overview of runs
- Display Application Log
- Edit run
- Manage run
- Replace run
- Select run for archiving
- Delete run
- Log of the deletion function

> **Note**
>
> The above functions are not all available for each application. For more information, see the application-specific documentation.

Use

The following table lists the runs available for each application:

Application	Run
General Calculation and Valuation Methods [page 376]	Account Pooling [page 383] or Account Pooling (CVPM) Facility Distribution [page 399] or Facility Distribution (CVPM) Determination of the Free Line for Facilities [page 395] or Determination of the Free Line (CVPM) Collateral Distribution [page 404] or Collateral Distribution (CVPM) Determination of Default [page 410] Stress Runs [page 429]: • Stress run for account pooling or stress run for account pooling (CVPM) • Stress run for facility distribution or stress run for facility distribution (CVPM) • Stress run for determination of the free line or stress run for determination of the free line (CVPM) • Stress run for collateral distribution or stress run for collateral distribution (CVPM) • Stress run for determination of default

Application	Run
Credit Risk [page 1191]	Credit Exposure Run [page 1238] or Credit Exposure Run (CVPM) Country Risk Run [page 1336] (prototype for country risk) **Stress run:** Credit Exposure Stress Run [page 1261] or Credit Exposure Stress Run (CVPM)
Historical Database [page 1478]	**Version management:** Run for generating historical data for data layers [page 1481] Run for generating historical data for the bank's own models [page 1484] Uploading of Files [page 1486] **Calculation functions:** Determining Default Rates [page 1488] Determining Average Default Rates [page 1491] Determining Default Figures [page 1493] Calculation of Migration Matrices [page 1496] **Data retrieval:** Run for supplying data to models [page 1498] Downloading of Files [page 1500] **Stress tests:** Stress Run for Supplying Models with Data [page 1505] Generating Scenario Data in the Source Data Layer [page 1506]
Generic BI Data Extraction [page 1638]	Testing the BI Extractor [page 1639] BI extraction run **Note** Extraction runs are created and executed in SAP NetWeaver Business Intelligence (BI). The system displays information about extraction runs in run administration of Bank Analyzer.
Regulatory Reporting Interface	Data Extraction Runs
Limit Manager [page 1529]	Limit Utilization Run [page 1551]

Application	Run
Fair Value Effectiveness Test for Hedging Relationships [page 1036]	Effectiveness Test Run
Cash Flow Hedge Analysis [page 1107]	Valuation Run [page 1128] Subvaluation Run Aggregation Runs [page 1129]
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169] Portfolio Item Runs [page 1172] • Period-end run for portfolio items (subledger scenario) • Period-start run for portfolio items (subledger scenario) • Portfolio item run (merge scenario)

> **Note**
>
> For some of the Bank Analyzer components, you can use the Schedule Manager to schedule and control jobs. If you use multiple applications, you can define the sequence in which the runs are to be carried out. For more information, see Schedule Manager [page 437].

More Information

Status Overview for Run Administration [page 1248]

6.3.5.1 Special Features of Run Administration in the HDB

Use

Run administration manages information about runs. This information can include, for example, the date of the run, the version of the run based on a key date, the user, the time and date the user started the run, the name of the user who made the latest changes to the run, and the time and date the user made those changes, the run category, and the current status of the run.

Prerequisites

You have started at least one of the following HDB runs:

- Historization Run for Data Layers [page 1481]

- Historization Run for In-house Models [page 1484]
- Determining Default Rates [page 1488]
- Determining Average Default Rates [page 1491]
- Determining Default Figures [page 1493]
- Exporting Data to In-House Models [page 1498]
- Stress Run for Supplying Models with Data [page 1505]
- Generating Scenario Data in the Source Data Layer [page 1506]

Features

Run administration for the Historical Database contains an overview of the HDB runs [page 1514] and enables you to display the HDB runs that have been flagged for deletion [page 1515].

6.3.5.2 Overview of HDB Runs

Use

This report enables you to use the SAP List Viewer (ALV) to display the runs you have carried out. In the detail log for a run, you can change the status of the run manually. For example, you can complete a run, flag a run for deletion, or set the status of a run to *Completed Successfully*.

Prerequisites

You have already started at least one run.

Features

To run the report, on the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Run Administration* > *HDB Run Administration*.

The system displays the *Run Administration: Overview of HDB* screen.

Selection

Use the following input fields to select the run you require:

- External ID of Run
- Key Date
- Run Category
- Version Run

- System Status

Output

The system displays an overview of the selected runs. In this overview, you can navigate to the detailed view by double-clicking a run. Here you have the following options:

- Choose *Display RDB Run* to display more information about the run.
- Choose *Application Log* to navigate to the error log for the run.
- Choose *Filter Log* to restrict the number of messages that are displayed in the application log. This is particularly useful if the log contains a large number of messages and you want to display only a certain subset of them.
- Select the *Flag Run as Obsolete* pushbutton in order to set the status of a run to *Obsolete*. Under certain circumstances you have only the option of branching to the error log.
  - You can use the *"Run Is Obsolete" Flag Reset* pushbutton to reset the status change.
  - You can use the *Delete* pushbutton to set the deletion flag, and then delete the run from the database in the *Delete HDB Runs Flagged for Deletion* function. You can also clear the deletion flag.
- Choose *Permit Manual Changes*, to be able to change the data for a run manually.
  - You can now either use the *Change* pushbutton to directly make the required changes to the selected run or use the Change Data Manually in the Historical Database [page 1507] function.
  - You can use the *Prevent Manual Changes* pushbutton to prevent manual changes to a run.
- Choose the *Time Stamp* pushbutton to display the time stamp collections used.
- By double clicking on a row you can branch to the detailed view for the selected run. In the detail screen for the selected run, you can see which selection parameters and control parameters were entered for the run. You can also find information about the aggregation process and the BAdI implementations that were used.

6.3.5.3 Deletion of Flagged HDB Runs

Use

You use the deletion report to delete individual runs, or to delete all the runs that you have flagged for deletion. You can also carry out a test run for the report. The deletion of each run is recorded in a log.

Note that once you have deleted a run, it is deleted permanently and cannot be recovered.

Prerequisites

The runs that you want to delete have to have the system status *Flagged for Deletion*. You can set and clear the deletion flag on the Overview of HDB Runs [page 1514] screen.

Procedure

1. You can display an overview of the HDB runs that are to be deleted on the *SAP Easy Access* screen under *Bank Analyzer* > *Analytics* > *Historical Database* > *Run Administration* > *Delete HDB Runs Flagged for Deletion*.

The system displays the *Run Administration: Deletion of HDB Runs* screen..

2. Make the following settings in the *Selection Criteria of Run* data group:
   - External ID of Run
   - Key Date
   - Run Category
   - Version of Run
3. Choose one of the following *Control Parameters*:
   - Set the *Test Run* checkbox to display all the runs that can be deleted.
   - Set the *Delete Immediately* checkbox to delete all the runs that are selected straight away.
4. Choose  .

Result

- If you set the *Test Run* checkbox in the control parameters, then the system displays an overview of the runs that can be deleted. You cannot delete any of these runs from the list.
- If you did not set any of the checkboxes in the control parameters, then the system displays an overview of the runs that can be deleted. From within this list, choose the pushbutton *Delete* to delete the runs as required.
- If you set the *Direct Deletion* checkbox in the control parameters, then the system deletes the runs that meet the selection criteria.

> **Note**
>
> Instead of deleting runs, you can archive them by using the archiving function. For more information, see Archiving of HDB Data [page 1520].

6.3.6 Ad Hoc Reporting

Definition

You use ad hoc reporting to display the results generated by the runs for the Historical Database. The system accesses all the data storage areas of the HDB:

- Data processing storage
- Historical data storage
- Review data storage
- Scenario data storage

The system uses a primary data source from the category *Worklist* to create ad hoc reports.

Structure

Ad hoc reporting contains the following functions:

- Display of HDB Data [page 1517]

- Analysis of Data Sources [page 1518]

6.3.6.1 Display of HDB Data in Ad Hoc Reporting

Use

You use this function to display data from the Historical Database (HDB) in ad hoc reporting.

Prerequisites

You have made the necessary Customizing settings for ad hoc reporting in Customizing for the Historical Database under *Bank Analyzer* > *Analytics* > *Historical Database* > *Ad Hoc Reporting*. For more information, see the documentation for the relevant Customizing activities.

Activities

1. On the *SAP Easy Access* screen choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Ad Hoc Reporting* > *Display HDB Data*.
2. Make the following entries in the *Data Collection* data group:
   - Use the combination of group ID and selection ID to define the data source.
   - If you want to use ad hoc reporting for stress data, enter the ID of the relevant composite scenario.
   - Enter the name of the key date of access to the data source. Today's date is used as the default value for this field. For more information about the key date, see Key Date Queries and Delta Queries [page 267].
   - In the *System Time of Worklist* field, enter the system date and time from which, and through to which, data changes are to be taken into account. If a delta query can be used with the primary data source in question, you can start the query for the time period you require by entering values for both time points. For more information about the primary data sources, see Servers of the General Primary Data Sources [page 1597].
   - In the *System Time of Enrichment* field enter the time point that the system is to use to access the primary data sources of the enrichment that are not in the same data read group as the primary data sources accessed by the *system time of the worklist*.
   - You can use *Selection Criteria* to define which data is to be selected, for example, by specifying the business partner.
   - Once you have entered values for all the required entry fields, you can display all the relevant Customizing settings in a dialog box by choosing *Display Details*. Choose the respective tab page:
     - *Selection*
     - *Selection Options: BAdI* (selection options for worklist generation and BAdI calls)
     - *Data Target Information* (custom fields that contain additional information, which you have stored in Customizing for the data layer)

3. If required, make the following settings in the *Technical Settings* area of the screen:
   - Set the *Parallel Processing* checkbox to speed up processing. It is advisable to use the parallel processing function, particularly if you are processing a large volume of data.
   - By specifying the layout, you can define the maximum number of hits, and how the selected data is displayed.
   - Enter a filter variant ID to define how messages are stored and displayed in the run administration function. You define filter variants in Customizing under *Bank Analyzer* > *Basic Settings* > *Settings for Storing Messages* > *Edit Filter Settings for Messages* .
4. In the *Free Selection Characteristics* data group, you can enter more characteristics from the characteristic repository. The system transfers these characteristics through to the secondary data source in the module editor. You can also use a BAdI call to define that when the system generates the worklist, the free selection characteristics are to be used in addition to those you selected by means of the *Selection Criteria* pushbutton.
   For more information see Customizing for the Historical Database under *Bank Analyzer* > *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *BAdIs for Selection* > *BAdI: Selection Options for Worklist Definition* .
   The following data categories are available for the free selection characteristics:
   - In the *Date 1* and *Date 2* fields, you can choose characteristics in date format.
   - In the *Time 1* and *Time 2* fields, you can choose characteristics in time format.
   - In the *Field 1* and *Field 2* fields, you can choose any characteristics.

   Then choose Enter to display the names of the characteristics that you have selected.

   > **Note**
   >
   > Unlike the selection criteria, which you add by using the *Selection Criteria* pushbutton, you can derive the data for the free selection characteristics dynamically within the variants.
5. Choose *Execute* .

Result

The system displays an overview of the selected HDB results. By clicking you can display more information. For example, if you click the financial transaction ID you can display the master data for the transaction. Choose *Display Log* to display the log of the HDB reporting run.

6.3.6.2 Analysis of Data Sources

You can use this function to analyze and display the data for primary objects of the category *Worklist* . When you do so, you can use the following applications of the Module Editor:

- HDB0 Historization Processes
- HDB1 Reporting in HDB
- HDB2 Bank's In-house Models
- HDB3 Calculation Functions
- SDSGeneric Secondary Data Source

Prerequisites

You have made the necessary Customizing settings for ad hoc reporting in Customizing for the *Historical Database* by choosing *Bank Analyzer* > *Analytics* > *Historical Database* > *Ad Hoc Reporting.* For more information, see the Implementation Guide (IMG) for the relevant IMG activities.

In particular, you need to have already registered the primary data sources of the category *Worklist* in the Customizing for ad hoc reporting in the Historical Database.

Activities

1. On the *SAP Easy Access* choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Ad Hoc Reporting* > *Analyze Data Sources.*
2. The system displays the *Display and Analysis of Data* screen.
3. Make the following entries in the *Data Collection* data group:
   - Enter the ID of the worklist whose data you want to analyze.
   - Specify the data source category (for example, general primary data source).
   - If you want to use ad hoc reporting for stress data, enter the ID of the relevant composite scenario.
   - In the *Key Date* field specify the name of the key date of access to the data source. Today's date is used as the default value for this field. For more information about the key date, see the document Key Date Queries and Delta Queries.
   - In the *System Date/Time* field enter the system date and time from which, and through to which, data changes are to be taken into account. If a delta query can be used with the primary data source in question, you can start the query for the time period you require by entering values for both time points. For more information about the primary data sources, see the document Servers of the Historical Database.
   - You can use *Selection Criteria* to define which data is to be selected, for example, by specifying the business partner. We recommend that you use selection criteria, particularly for test runs.
4. If required, in the data group *Secondary Data Source* specify the Module Editor application. Within the application chosen, you can choose an environment, or choose a module from the environment.
5. If required, make the following settings in the *Descriptive Statistics* data group:
   - If you want to create descriptive statistics for characteristics or key figures, set the *Descriptive Statistics* indicator, and then choose Enter .
   - Specify the evaluation currency .
     If you have specified the evaluation currency, you have to enter the market data area and the exchange rate category as well, as it may be necessary to translate currencies.
   - If you use descriptive statistics, you must choose the *Define Granularity Fields* pushbutton, and define the granularity characteristics. You can choose from the characteristics in the characteristic repository of Bank Analyzer. Granularity characteristics govern by which characteristics data is aggregated.
6. If required, make the following settings in the *Technical Settings* data group:
   - Set the *Parallel Processing* indicator to speed up processing. It is advisable to use the parallel processing function, particularly if you are processing a large volume of data.
   - By specifying the layout, you can define how the selected data is displayed.
7. Choose  with the quick info *Execute* .

Result

The system displays an overview of the data records it read. By clicking the financial transaction ID you can display the master data for the transaction. You can use the  *Display Log* pushbutton to display the error log.

6.3.7 Archiving and Destroying Historical Database Data

You can archive Historical Database (HDB) data using the following archiving objects, and destroy it as part of SAP Information Lifecycle Management (ILM):

- `BA1_R4_064` *(RDB: Archiving of HDB - Scenario Data Storage)*
- `BA1_R4_065` *(RDB: Archiving HDB - Review Data Storage)*
- `BA1_R4_066` *(RDB: Archiving HDB - Historical Data Storage)*
- `BA1_R4_067` *(RDB: Archiving HDB - Data Processing Storage)*

> **Note**
>
> Note that you cannot archive individual HDB runs. The archiving function always archives complete RDB runs, each of which usually consists of multiple HDB runs.

You can execute the following write, delete, and reload programs for these ILM-enabled archiving objects in archiving administration (transaction SARA):

Archiving Object	Programs

`BA1_R4_064`	`/BA1/R6_ARCHIVING_WRITE_SCEN`
	`/BA1/R6_ARCHIVING_DELETE_SCEN`
	`/BA1/R6_ARCHIVING_RELOAD_SCEN`

`BA1_R4_065`	`/BA1/R6_ARCHIVING_WRITE_REV`
	`/BA1/R6_ARCHIVING_RELOAD_REV`
	`/BA1/R6_ARCHIVING_DELETE_REV`

`BA1_R4_066`	`/BA1/R6_ARCHIVING_WRITE_HIST`
	`/BA1/R6_ARCHIVING_DELETE_HIST`
	`/BA1/R6_ARCHIVING_RELOAD_HIST`

`BA1_R4_067`	`/BA1/R6_ARCHIVING_WRITE_COL`
	`/BA1/R6_ARCHIVING_DELETE_COL`
	`/BA1/R6_ARCHIVING_RELOAD_COL`

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction `SARA`), enter the name of the archiving object, and choose *Database Tables*.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.
   The *Archive Administration: Initial Screen* appears.
2. Choose the required object name. For example, `BA1_R4_064` *(RDB: Archiving of HDB - Scenario Data Storage)*.
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* fields to select the data to be archived (for example, HDB scenario data storage).
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.
    The system displays a list of data (for example, HDB scenario data storage). In this list, you can select individual items (for example, HDB scenario data storage items) that you want to archive.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

6.3.7.1 Archiving Control

Use

This function is supplementary to the archiving report for the Historical Database (HDB). You can use this function to manage the relevant RDB runs, to flag them for archiving, and to monitor their status, for example.

Features

HDB data is archived for entire RDB runs. An RDB run contains multiple separate HDB runs for the individual data areas (historical data storage, review data storage, data processing storage and scenario data storage). You define how many HDB runs an RDB run contains by specifying the RDB run validity in months. You do this in the Customizing for the Historical Database under *Bank Analyzer* > *Historical Database*:

- *Set Up Data Layers* > *Configure Data Layers*
- *Register In-House Models* > *Configure Models*
- *Register External Systems* > *Register External Systems*

> **Note**
>
> For performance reasons, we recommend that you use as large a time period as possible for the validity of the runs in months (for example, 24 months). However, using the smallest possible time period (for example, 3 months) gives greater flexibility. Therefore, you need to find a balance between the need for greater flexibility and for optimal system performance.

You can also change the validity of an RDB run in archiving control. However, make sure that you choose a time period that at least contains the times of the HDB runs that the RDB run already contains. You can also create RDB runs manually in archiving control.

If you have already started archiving for a particular time period, and then want to start an HDB run within this time period, the system creates a new version for the RDB run.

Activities

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Historical Database* > *Archiving* > *Control of Archiving*:

The *Control of Archiving* screen appears. You have the following options:

- In the left-hand part of the screen, the system displays an overview of all the RDB runs. The runs are in chronological order. The RDB runs are also sorted by the following criteria:
  - Data layers (historical data storage)
  - Your bank's own in-house models (historical data storage)
  - Your bank's own in-house models (review data storage)
  - Your bank's own in-house models (scenario data storage)
  - External systems (data processing storage)

  To display the data for a particular RDB run, expand the respective node. To display the HDB runs for a particular RDB run, expand the RDB run. In the RDB run, you can monitor the status of the run by means of the following symbols:

Status Display

Status Display	Description
Not Archived	The RDB run has not yet been archived. You can flag the run for archiving. To do so, select the run and choose the pushbutton *Flag for Archiving/Deregister*. By choosing the pushbutton again, you can reset the flag.
Flagged for Archiving	The RDB run has been flagged for archiving.
Archived	The RDB run has been archived. This means that archive files were generated. You can now delete the run from the database.

Deleted	The RDB run has been deleted from the database. You can read that data in the archive, or reload the data if required.
Reloaded	The RDB run has been reloaded. This means that the database contains the data once again. **Note** You can delete the data for reloaded runs again.
Write Protected	The RDB run is write-protected. This status can be set for the following reasons: • The RDB run is being archived. • The archiving of the RDB run was terminated due to an error. Restart archiving.

- In the right-hand side of the screen, choose the *RDB Data* tab page to display information about the RDB run. If required, you can change the validity dates of the run. On the *Run Data* tab page, you can display information about the HDB run.
- Depending on which level you have selected in the left-hand part of the screen, the system displays the following pushbuttons:

Pushbuttons

Pushbutton	Description	Comment
Expand All Nodes *Collapse All Nodes*	Expands or collapses the nodes in the left-hand part of the screen	General display
RDB Run: Update Nodes	Updates the display of the status of the RDB runs	General display
Display Description/Hide Description	Displays or hides the description in the left-hand part of the screen	General display
Select or Deregister for Archiving	Sets an RDB run to the status *Selected for Archiving*. If you choose this pushbutton for an RDB run that is already selected for archiving, this resets the selection. In other words, the run is deregistered for archiving.	Display of the RDB run
Create New Run	Creates a new RDB run	Display of the RDB run
Change Validity Dates	Switches to change mode so that you can change the validity dates for an RDB run	Display of the RDB run

Complete Run - Delta Run	Flags an HDB run as a complete run (the run contains a version for all relevant characteristic values) or as a delta run (the run contains only a subset of all the relevant characteristic values)	Display of the HDB run
Display Time Gaps	Within a data layer, for example, this function displays the time periods that are not covered by an RDB run	Display of the RDB run

For information about the archiving process, see the document Archiving of HDB Data [page 1520].

6.3.8 Tools

The following tools are available for the Historical Database (HDB).

- Schedule Manager [page 437]
- Deletion report *Initialize Data Processing Storage*
- Deletion repot *Delete Scenario Data*

For more information about the deletion reports, choose *Program Documentation*.

> ⚠ Caution
>
> The deletion reports do not contain a function for checking whether the data you want to delete is still being used. Before you start any of these reports, make sure that the data you want to delete is not being used anywhere else.

6.3.8.1 Schedule Manager

Definition

A tool in Bank Analyzer that you can use to execute and monitor complex business processes, such as period-end closing. When you schedule a run, you always have to specify the program with which the system calls the run. There is a separate program for each run category in an application.

The table below shows for which runs and run categories the Schedule Manager can be used, and their corresponding programs.

Application	Run Category	Program
Account Pooling	Account Pooling [page 383]	/BA1/DE_GM_MAIN_COMP

Application	Run Category	Program
Account Pooling	Stress Run for Account Pooling [page 429]	/BA1/DE_GM_MAIN_COMP_SCEN
Determination of the Free Line for Facilities	Determination of the Free Line for Facilities [page 395]	/BA1/DE_GM_MAIN_FL
Determination of the Free Line for Facilities	Stress Run for Determination of the Free Line for Facilities [page 429]	/BA1/DE_GM_MAIN_FL_SCEN
Distribution of Facilities	Distribution of Facilities [page 399]	/BA1/DE_GM_MAIN_RLV
Distribution of Facilities	Stress Run for Facility Distribution [page 429]	/BA1/DE_GM_MAIN_RLV_SCEN
Collateral Distribution	Collateral Distribution [page 404]	/BA1/DE_GM_MAIN_SV
Collateral Distribution	Stress Run for Collateral Distribution [page 429]	/BA1/DE_GM_MAIN_SV_SCEN
Determination of Default	Determination of Default [page 411]	/BA1/RA_DFLT4280_RUN
Determination of Default	Stress Run for Determination of Default [page 429]	/BA1/RA_DFLT4280_RUN_SCENARIO
Fair Value Effectiveness Test [page 1036]	Effectiveness Test Run [page 1039]	/BA1/R8_FVHE_MAIN_RUN
Fair Value Effectiveness Test	Generation of Time Series for Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MAIN_RUN
Fair Value Effectiveness Test	Modifying a Time Series for the Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MODIFY
Cash Flow Hedge Analysis [page 1107]	Subvaluation Run [page 1128]	/BA1/R5_CFHA_BTC_RUN
Cash Flow Hedge Analysis	Aggregation Run [page 1129]	/BA1/R5_START_BATCH_AGGR
Cash Flow Hedge Analysis	Joint call of Valuation Run [page 1111] and Aggregation Run [page 1123]	/BA1/R5_EXECUTE_CFHA
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169]	/BA1/R9_VPM_BTC_ID_RUN
Portfolio Fair Value Hedge	Portfolio Item Run (Merge Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_RUN
Portfolio Fair Value Hedge	Period-End Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PE_RUN
Portfolio Fair Value Hedge	Period-Start Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PS_RUN

Application	Run Category	Program
Portfolio Fair Value Hedge	Define End of Valuation Period	/BA1HM/RAH_M_PFVH_SET_VP_END
Portfolio Fair Value Hedge	Adjust Initial Portfolio Fair Value Hedges for Next Valuation Period	/BA1HM/RAH_M_PFVH_ADJUST
Portfolio Fair Value Hedge	(Re-) Designation of Initial Portfolio Fair Value Hedges	/BA1HM/RAH_M_PFVH_DESIG
Credit Exposure [page 1192]	Credit Exposure Run [page 1238]	/BA1/R2_CRA_MAIN_CRE
Credit Exposure	Credit Exposure Stress Run [page 1261]	/BA1/R2_CRA_MAIN_CRE_SCEN
Credit Exposure	Credit Exposure Run for Revolving Retail	/BA1/RR2_CRE_REVRET
Credit Exposure	Approximate Stress Run [page 1274]	/BA1/RR2_CRE_SCENARIO
Credit Exposure	Aggregation Run for Ad Hoc Calculation [page 1276]	/BA1/RR2_SCEN_AGGR
Country Risk [page 1326]	Country Risk Run [page 1336]	/BA1/R2_CRE_MAIN
Historical Database [page 1478]	Historization Run for Data Layers [page 1481]	/BA1/R6_HIST_RUN
Historical Database	Historization Run for Bank's Models [page 1484]	/BA1/R6_HIST_RUN_MODEL
Historical Database	Determine Default Rates [page 1488]	/BA1/R6_CF_DEFAULT_4210
Historical Database	Determine Average Default Rates [page 1491]	/BA1/R6_CF_DEFAULT_4220
Historical Database	Determine Default Figures [page 1493]	/BA1/R6_CF_DEFAULT_KEYFIGURES
Historical Database	Exporting Data to In-House Models [page 1498]	/BA1/R6_MOD_RUN
Historical Database	Calculation of Migration Matrices [page 1496]	/BA1/R6_CF_MIGRATIONS
Historical Database	Stress Run for Supplying Models with Data [page 1505]	/BA1/R6_MOD_RUN_SCENARIO
Historical Database	Stress Run for Supplying Data [page 1506]	/BA1/R6_FDB_RUN_SCENARIO
Regulatory Reporting Interface	Processing Framework Run Copying the Start ID	/BA1/S1_COPY_STARTID

Application	Run Category	Program
Regulatory Reporting Interface	Processing Framework Run Creating Runs	/BA1/S1_CREATE_RUN
Regulatory Reporting Interface	Processing Framework Run Release Run	/BA1/S1_RELEASE_RUN
Regulatory Reporting Interface	Processing Framework Run Creating a Worklist	/BA1/S1_CREATE_WRKL
Regulatory Reporting Interface	Processing Framework Run Processing a Run	/BA1/S1_PROCESS
Regulatory Reporting Interface	Data Extraction Runs	/BA1/S3_EXTRACT_RUN
Limit Manager [page 1529]	Limit Utilization Run [page 1551]	/BA1/R3_UTILIZATION_RUN_DPF

More Information

Schedule Manager

6.4 Disclosure and Reporting (FS-BA-AN-DR)

Use

The Disclosure and Reporting component helps you meet the disclosure and reporting requirements under Basel II.

To simplify the external disclosure process, the SAP Netweaver Business Intelligence content (BI) contains reports that you can use and adapt as required for various portfolios. The reports provided by SAP meet the quantitive requirements of Basel II for external disclosure.

You can use this component to create various reports on the basis of the same set of underlying data.

- External disclosure
  For example, you can use sorting criteria to define in what order credit items or investments and securitizations are listed or report loss-relevant data and your risk mitigation techniques.
- Internal reporting
  Internal reporting contains, for example, a set of risk reports, and flexible evaluations that can be used to analyze data from the level of institutions down to single transactions. The evaluations are based on selection criteria chosen by you, and an extendable data model.

- Support for supervisory investigations
  For example, a break-down of the information shown in internal risk reports
- Support for stress test reporting
  For example, a list of risk reports containing stress data that was determined by using scenarios [page 1731].

External reporting provides Basel II data, which was previously available for risk management only. This enables banks to report the risk situation of the bank internally, and to offer their investors greater transparency. Banks can turn the Basel regulations to their strategic advantage.

Integration

- Settings for the selection of data in Bank Analyzer
  For reporting purposes, you can use data from the Source Data Layer (SDL), the Historical Database (HDB), and results data from the Credit Exposure runs, for example. Data stored in feeder systems can be imported into Bank Analyzer by means of the SDL. You can also use transparent tables for the underlying data.
  You define the selection of date to be selected in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Generic BI Data Extraction* > *General Settings for Data Selection* > *Selection Settings for Data Extraction*.
- DataSources for the Extraction of Data to BI
  You define the DataSources for data extraction in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Generic BI Data Extraction* > *Edit Generic BI Data Extraction*.
  For more information, see Generic BI Data Extraction [page 1638].
- Drill-through from BI to Bank Analyzer
  You can assign receiver reports from Bank Analyzer to source reports and queries in Reporting BI. This drill-through function enables you to navigate from one cell of a source report in the Reporting BI system to a target report in Bank Analyzer.
  Drill-through is based on the report-report interface (RRI) of the Reporting BI system. For BI queries, receiver reports are defined as the navigation target in the source data in SDL. For more information about report-report interfaces see SAP Library under *SAP NetWeaver* > *SAP NetWeaver by Key Capability* > *Business Intelligence* > *BI Suite: Business Explorer* > *Query Design: BEx Query Designer* > *Functions on the BI Server* > *Editing Sender-Receiver Assignments to the RRI in the BI System*.

Features

In run administration [page 430], you can display the extraction runs triggered in BI.

- Set the *Use Run Admin* indicator in the *Data Processing* dialog structure in Customizing for *Bank Analyzer* by choosing *Infrastructure* > *Extraction and Reporting Services* > *Generic BI Data Extraction* > *Generic BI Data Extraction*.
- If you start an extraction run for a DataSource (by choosing *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Generic Data Extraction* > *Test BI Extractor* on the *SAP Easy Access* screen) that is defined from the point of view of characteristics and key figures, then deactivate the *Test Run* indicator in the selections.

The run administration shows you the status and the application log for extraction runs with the Data Processing Framework. You can delete logs of runs in run administration. You can delete the extraction run itself in BI. Test runs are not listed in run administration.

SAP provides queries for the following areas:

- General Queries
  For example, Report 1: *Distribution of Credit Items*
- Queries for Equity Holdings
- Queries for Securitizations
- Queries for Operational Risk
- Queries for Commercial Real Estate
- Queries for Migration Matrices

6.5 Limit Manager (FS-BA-AN-LM)

Use

Risk controlling regulations and business considerations mean the system has to provide functions for measuring, limiting, and analyzing default risks.

Banks set different maximum risk amounts in order to limit the potential harm caused by the insolvency of a business partner.

This function helps you to limit default risks by setting up and monitoring limits online. You can use the reporting functions for internal and external purposes.

Integration

Limit Manager is the part of Bank Analyzer in which the attributable amounts, calculated from Credit Exposure for example, can be applied to customer-defined limits. You can display the results of the limit utilization runs using the SAP List Viewer (ALV) or SAP NetWeaver Business Intelligence (BI).

For more information, see Architecture of Limit Manager [page 1530].

Features

You use Limit Manager to manage these risks by defining limits and monitoring them continuously to ensure that these limits are observed. Limits can be managed flexibly, since the limit characteristics that are available can be combined in any way. You can also define free characteristics for each limit type in addition to the limit characteristics. Free characteristics enable you to select more detailed data when the limit data is displayed.

Limit Manager enables you to define different levels for the limitation of default risks. The limit area is the highest level of segmentation into logically separate segments. There are different limit types for each limit area. You

assign defined limit characteristics, such as an organizational unit, a business partner, or a currency, to the limit types. Within a limit, you define specific limit amounts that are related to the characteristic values of a limit type.

You can create a limit [page 1532] for each combination of limit characteristics and limit characteristic values. A limit is a maximum amount for limit utilizations that is defined for certain values of the limit characteristics of a limit type.

6.5.1 Architecture of Limit Manager

Limit Manager is integrated in the Bank Analyzer architecture, which comprises the core Bank Analyzer system and SAP NetWeaver Business Intelligence (BI). Bank Analyzer contains different applications and the Source Data Layer (SDL). The SDL provides functions and services for managing bank data. SAP BI technology is used to manage data and to create reports, for example.

For more information about the structure of Bank Analyzer, see Bank Analyzer (FS-BA) [page 8].

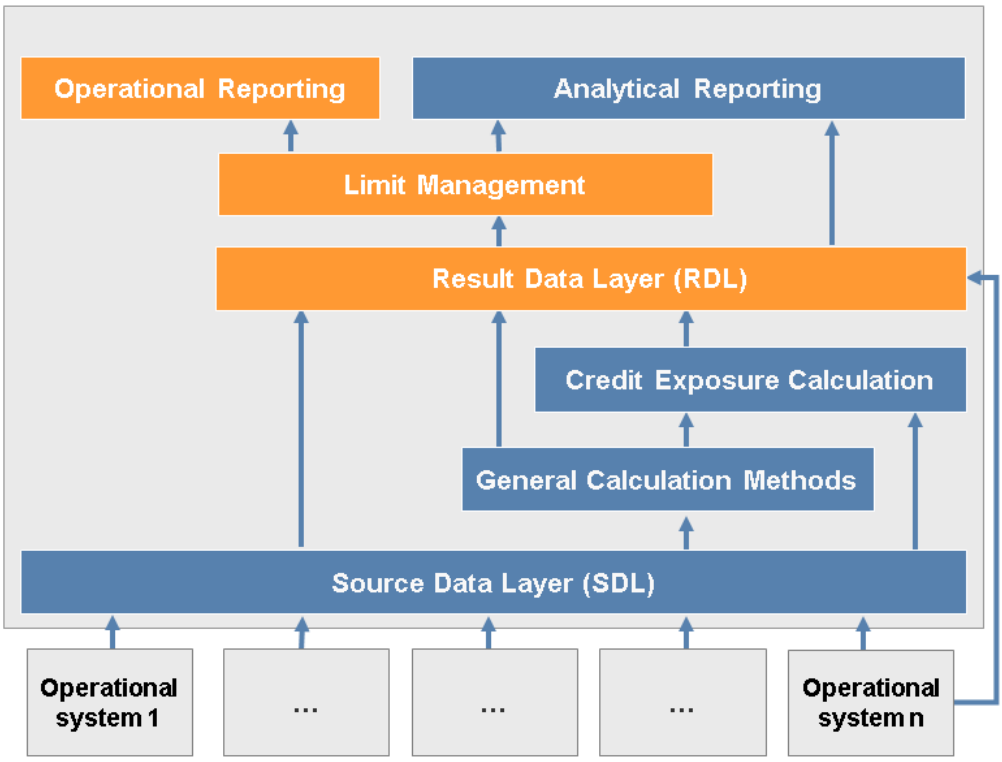
Data Flows

Various source systems supply Limit Manager with data. Rather than communicating directly with Limit Manager or with the generic methods that are upstream, the data from these source systems is first stored in the SDL. From here, Limit Manager can then access master and flow data supplied by one of the source systems.

Limit Manager can also use attributable amounts calculated by other applications, such as Credit Exposure. It accesses this data using the Results Data Layer (RDL), since the results data is stored in a corresponding RDL.

The system stores results data calculated by Limit Manager in tables within Limit Manager. You can use the SAP List Viewer (ALV) or SAP BI to display results data in lists either as accumulative figures or at single record level.

The figure below illustrates how data flows through Limit Manager:



Data Flows in Limit Manager

Prerequisites for Implementing the Data Model in the System

You have made the Customizing settings for Bank Analyzer. The following section provides a brief overview of the main Customizing settings for Limit Manager.

- You have classified characteristics and key figures by purpose (data selection, reporting, controlling, and internal). You have included these characteristics in the Limit Manager repository and decided whether they are direct or derived characteristics.
- You have defined the following limit levels:
  - Limit area
  - Limit type
  - Determination method
- You have configured the module editor in such a way that it can be used to create the secondary data source by selecting and adding data from the worklist of the primary data source.

For more information about Customizing for Limit Manager, see Customizing for *Bank Analyzer* under *Analytics* > *Limit Manager*.

6.5.2 Limits

Definition

A limit is a maximum amount for limit utilizations that is defined for certain values of the limit characteristics of a limit type.

Use

The limit is used as an amount against which checks are made to determine whether the limit has been exceeded. Each limit has a validity period.

In the limit item category, you can specify whether a limit belongs to one of the following categories:

- Master limit
- Interim limit
- Limit transfer

  > **Note**
  >
  > Since a limit transfer always involves two limits, limit transfers can be created only from the overview of limits or from the overview of limit utilizations. For more information, see Creating Limit Transfers [page 1543].

- Risk commitment period

You can check limits as follows:

- **Release procedure**
  - Principle of dual control
  - Principle of triple control
  - Principle of quadruple control

  For more information about Customizing for the release tool in Limit Manager, see Customizing for *Bank Analyzer* under *Analytics* > *Limit Manager* > *Basic Settings in Limit Manager* > *Release Procedure in Limit Manager*.
- **Review procedure**
  Review recipients and review dates define the review process.

Structure

A limit for a particular value of the limit characteristics is comprised of the following values:

- Validity start date
- Validity end date
- Percentage critical limit utilization that triggers a warning function

- Limit currency
- Review date and review recipient
  You can enter this data manually. Alternatively, you can use the derivation tool to derive this data. For more information about Customizing the derivation tool in Limit Manager, see Customizing for *Bank Analyzer* under *Analytics* > *Limit Manager* > *Basic Settings in Limit Manager* > *Derivation of Review of Limits in Limit Manager*.
- User-defined text for the limit name (optional)
- Limit check
  If you do not select this checkbox, the system does not check the limit against the current limit utilizations. Overdrafts are not included in reporting or in the single transaction check.
- Limit lock
  The system freezes all the transactions that were allocated to the limit at the time the lock was set. The system displays new business that is allocated to this limit as an overdraft. If the value of an existing transaction is changed, the system also checks whether this transaction is to be allocated to the limit.
- Lock information
  This information specifies which user has locked or unlocked the limit and when.
- Individual items
  For more information about limit items, see Creating Limit Items [page 1537].

6.5.2.1 Editing Limits

Use

For each combination of limit characteristics defined in a limit type, you can create one limit per combination of characteristic values. In this general report for the collective processing of limits, you can create new limits or change and display existing ones.

Prerequisites

You have defined at least one limit type in the system settings.

You create limit types in Customizing for *Bank Analyzer* under *Analytics* > *Limit Manager* > *Limit Levels in Limit Manager* > *Edit Limit Types*.

Procedure

Selection

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Management* > *Edit Limits*.
   The *Limit Management: Maintain Limits* screen appears.

2. Select the required processing mode:
   - Display
   - Maintain
3. In the *General Selection Criteria* data group, enter the following information (required entries):
   - Limit area
   - Limit type
   - Key date for selecting and aggregating limit item amounts for a limit. The default date is today's date. You can overwrite this date if required.
4. In the *Selection Criteria for Limit Characteristics* data group, specify how the limit is to be selected. You can use the following selection criteria:
   - Display filter

     You define display filters in Customizing for *Bank Analyzer* under *Analytics* > *Limit Manager* > *Characteristics in Limit Manager* > *Edit Display Filter*.
   - Direct characteristics
     Direct characteristics in Limit Manager are the characteristics that are supplied from the Source Data Layer (SDL) or from the characteristic repository in Bank Analyzer.
   - Derived characteristics
     Derived characteristics in Limit Manager are the characteristics that are derived from the direct characteristics.
     You can make the settings for the derivation of characteristics in Customizing for *Bank Analyzer* under *Analytics* > *Limit Manager* > *Characteristics in Limit Manager* > *Characteristic Derivation*.
5. If required, enter a layout in the *Output Control* data group. You define layouts in the *Overview of Limits* by choosing the pushbutton. Then choose .
6. Choose .
   The system displays a hierarchical overview list of all the existing limits that meet your selection criteria.
   The item line contains the limits and the header contains the related limit types.

Interaction

You can use various functions in this overview. In each case, select one of the limits and choose the appropriate pushbutton.

Interaction	Use
	You can refresh the overview of limits.
	You can create a new limit. For more information, see Creating Limits [page 1535].
	You can copy an existing limit and then customize it.
	You can change an existing limit. You change limits in the same way as you create limits.
	You can display an existing limit. You display limits in the same way as you create limits.

Interaction	Use
	You can create a limit transfer. For more information, see Creating Limit Transfers [page 1543].
	You can lock a limit. The limit is locked only when you confirm this when the release procedure is activated.
	You can remove an existing limit lock. The limit is unlocked only when you confirm this when the release procedure is activated.
	You can check whether limit utilizations exist for the selected limit. The system calls the transaction for displaying an overview of limit utilizations [page 1561].
	You can display change documents for the limits, provided the change documents function has been activated in Customizing. The system calls the transaction for displaying change documents for limits [page 1544].
	You can create a note for the limit.
	You can display the business partner master data.
[Customizing] *Limit Type Customizing*	You can display the Customizing settings for the limit type.
	You can display the details about a limit.
	You can apply the review process to the selected limits.

> **Note**
>
> If you chose the *Display* mode in the selection options, you cannot carry out any of the actions listed above.

6.5.2.1.1 Creating Limits

Prerequisites

You have called the overview of limits on the *Limit Management: Maintain Limits* screen.

You access this overview from the *SAP Easy Access* screen under *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Management* > *Edit Limits*.

Context

For each combination of limit characteristics defined in a limit type, you can create one limit per combination of the characteristic values. Transactions and reservations can then be attributed to this limit.

Procedure

1. Choose  .
2. A dialog box appears. Enter the limit type for which you want to enter the limit. Choose  .
3. A dialog box appears. Enter the required values for the characteristics that are assigned to the selected limit type. Choose  .

   A new screen appears.
4. Enter a valid-from and valid-to date for the limit.
5. If required, enter a percentage for the critical limit utilization. This entry is used only if the early warning control type *Percentage Barrier* has been set for the underlying type.
6. Enter the currency in which the system is to update totals records for the limit, and to display the limit and limit utilization amounts in reporting. The currency of the limit type is displayed as the default value.
7. If you have activated the review process in Customizing, you can enter a review date and a review recipient.

   You can also use the derivation tool to fill these input fields automatically. To do so, choose  . You can use the input fields for the review process only if you have activated the review process for the limit type in Customizing. For more information about Customizing for the review process in Limit Manager, see Customizing for *Bank Analyzer* under *Analytics* > *Limit Manager* > *Limit Levels in Limit Manager* > *Edit Limit Types*.
8. If required, you can enter your own text for the limit name.
9. Select the *Check Limit* checkbox if you want the system to check the limit against the current limit utilizations. Overdrafts are included in reporting or in the single transaction check.
10. Select the *Limit Lock* checkbox if you want to lock a limit. If a limit is locked, new transactions that are to be allocated to this limit violate the limit. You can use a report for limit monitoring to identify limit violations. If a transaction in the limit utilization run has been flagged as *Not New Business*, it does not cause a lock violation. The *Not New Business* checkbox can be selected in the data enrichment step in the module editor and set at runtime, and applied to the single utilization. In the Customizing activity *Edit Limit Types*, you can select the *Lock Violation for Utilization Increases* checkbox. In the Customizing activity *Edit Limit Types*, you can define the *Tolerance for Attributable Amounts in %*. The system uses this tolerance when it determines whether locks have been violated.
11. In the *Item Information* screen area, enter the individual items for the limit. For more information, see Creating Limit Items [page 1537].

> **Note**
>
> Since a limit transfer always involves two limits, limit transfers can be created only from the overview of limits or from the overview of limit utilizations. For more information, see Creating Limit Transfers [page 1543].

12. Enter your own characteristics.

    These characteristics can be optional or required entry fields. You define whether a characteristic is an optional or required entry field in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Segmentation Service* > *Define Context* for LIMIT_MAIN (Edit Limits) and LIMIT_READ (Read Limits). Create your own characteristic profile under *Define Characteristic Profile*. You specify in the characteristic profile which of your characteristics are optional fields and which are required fields. You assign the characteristic profile to the context in the Customizing activity *Assign Context to Characteristic Profile*.

13. Choose  .

> **Note**
>
> If you have activated the release procedure, any new limits are presented for release. Only once the items have been released are their values relevant for Limit Manager. The system sets the release indicator.
>
> Choose  to display the release history or lock behavior of the limit item.

> **Note**
>
> To delete a limit, choose  . The limit is flagged for deletion.

14. Choose the *Lock Information* pushbutton to display when a lock was set or deleted and by which user.
15. Choose  to check the limit transfer.

Results

The new limit has been created and the system returns to the overview of limits If the release procedure has been activated, a workflow is triggered that informs the relevant user that there is a new limit item that is to be released. Any limit items that have not yet been released in the release process are ignored in Limit Manager; such items are not assigned to a limit and do not affect its values. Limit transfer items and interim limit items are not subject to the release process.

6.5.2.1.2 Creating Limit Items

Prerequisites

You have called the overview of limits on the *Limit Management: Maintain Limits* screen.

You access this overview from the *SAP Easy Access* screen under *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Management* > *Edit Limits*.

Context

You create one or more limit items for each limit. This enables you to assign a specific limit to a predefined limit characteristic value.

Depending on the limit item category, a limit item can contain the following information:

- Release status (automatically entered by the system)
- Limit item category
- Valid-from date for the limit item
- Valid-to date for the limit item
- Plus/minus sign (automatically entered by the system)
- Amount of the limit item
- Absolute barrier
- Currency of the limit item
- Maximum risk commitment period for risk commitment period items
- User-defined text for a limit name (optional)

Procedure

1. Select the limit to which you want to add items and choose the *Change Limit* pushbutton.

   A new screen appears.

   > **Note**
   >
   > For more information about creating a new limit including limit items, see Creating Limits [page 1535].

2. Select a limit item category in the *Item Information* screen area.

   > **Note**
   >
   > If you select the limit item category *Risk Commitment Period*, leave the amount and currency fields empty. The validity periods of the limit items must not overlap.

3. Enter a valid-from and valid-to date for the limit item. The validity date of the limit item can differ from the validity date of the limit. For more information, see Example of a Limit History [page 1539].
4. Enter an amount for the limit item.
5. If required, enter an amount for the absolute barrier for the limit item. This entry is used only if the early warning control type *Absolute Barrier* has been set for the underlying limit type.
6. Enter the currency for the limit item. The currency of the limit item can differ from the currency of the limit.
7. If required, enter your own text to provide details on the limit item.
8. Enter the maximum risk commitment period in days only for the limit item category *Risk Commitment Period*.
9. Save your entries.

6.5.2.1.3 Example of a Limit History

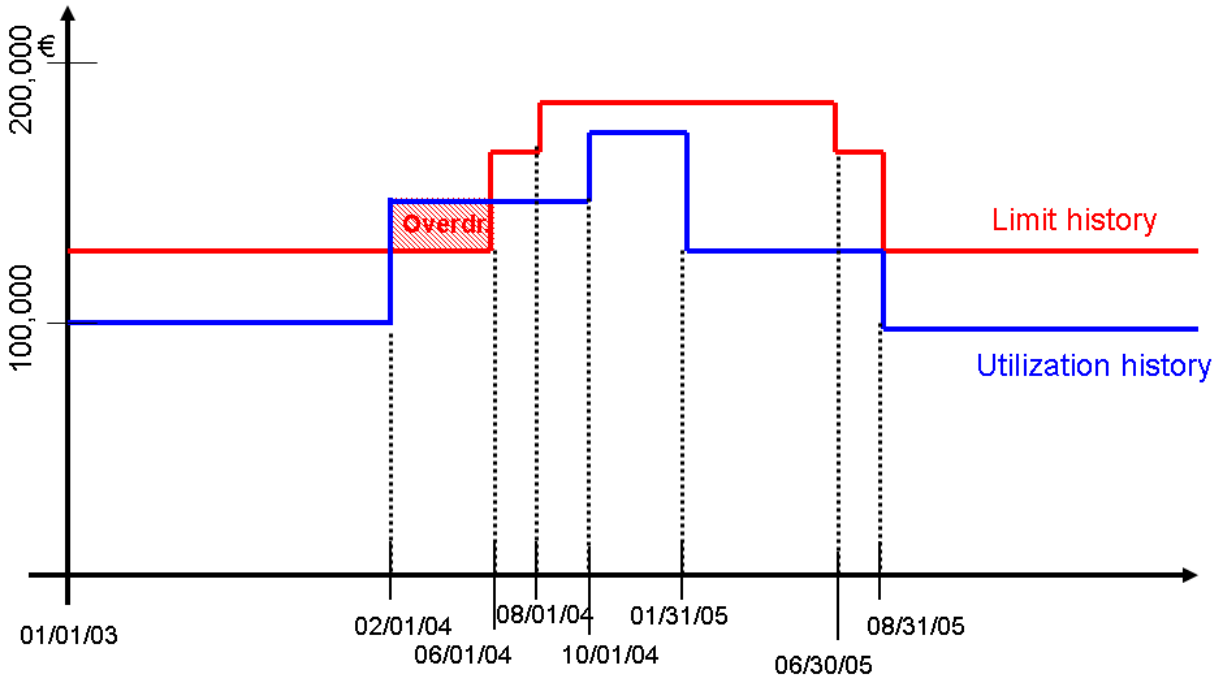
A limit comprises multiple limit items. The development of the limit over time is referred to as the limit history. This example shows a limit history together with the development of the limit utilizations. It is based on the limit items and transaction data.

The limit history comprises the following limit items:

Item	Limit Category	Validity	Amount in EUR
1	Master limit	January 1, 2003 - December 31, 2007	130,000
2	Master limit	June 1, 2004 - August 31, 2005	40,000
3	Limit transfer (transferred from another limit)	August 1, 2004 - June 30, 2005	20,000

The limit utilization history comprises the following transactions:

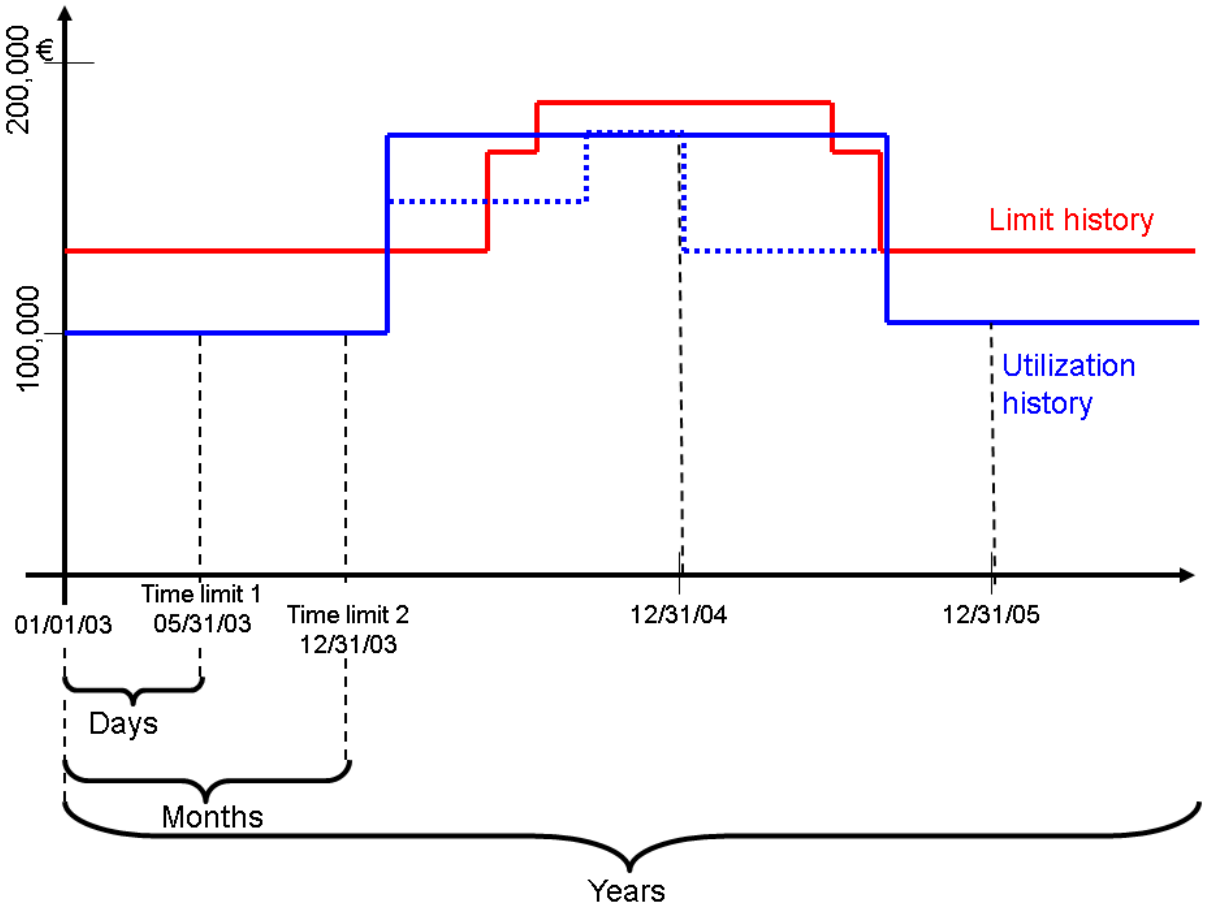
Transaction	Validity	Amount in EUR
1	January 1, 2003 - December 31, 2005	100,000
2	February 1, 2004 - January 31, 2005	50,000
3	October 1, 2004 - August 31, 2005	30,000



Assume you have made the following entries for the underlying limit type in Customizing for *Bank Analyzer* under *Analytics* > *Limit Manager* > *Limit Levels in Limit Manager* > *Edit Limit Types*:

Field	Details	Description
First Time Limit for Totals Record	180	
Second Time Limit for Totals Record	365	
Minimum Interval Before First Time Limit	1	Totals record updated on a daily basis
Minimum Interval Between Time Limits	30	Totals record updated on a monthly basis
Minimum Interval After Second Time Limit	365	Totals record updated on a yearly basis

These entries specify that daily totals records are stored for the first 180 days after the business key date, and after this monthly totals records are stored. From one year after the business key date annual totals records are stored. The annual totals records express the maximum limit utilization that occurs in the course of a particular year. This means that the risk never falls below the minimum amount.



6.5.2.2 Processing Limit Transfers Collectively

Use

You use this program in Limit Manager to obtain an overview of the existing limit transfers, and to manage the limit transfers.

Prerequisites

You have created limit transfers. You create limit transfers using the following transactions:

- Edit Limits [page 1533]
- Display an Overview of Limit Utilizations [page 1561]

For more information about creating limit transfers, see Creating Limit Transfers [page 1543].

Procedure

Selection

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Management* > *Collective Processing of Limit Transfers*.
   The *Limit Transfers: Overview/Collective Processing* screen appears.
2. In the *Limit Transfer Data* data group, enter the selection criteria required. You can restrict the selection by specifying a limit area, limit type, or valid-to date for the limit transfers.
3. In the *User Data* data group, you can restrict the selection further by specifying the user who entered the limit transfer or the user who last changed the transfer. This enables users to display only the limit transfers created by them, for example.
4. If required, enter a layout in the *Output Control* data group. You define layouts in the *Overview of Limits Transfers* by choosing the *Change Layout* pushbutton. Then choose *Save Layout*.
5. Choose *Execute*.
   The system displays a hierarchical overview list of all the existing limit transfers that meet your selection criteria.
   The item line contains the limits involved in the limit transfer together with their characteristics, amounts, and currencies. The header displays the limit area, limit type, name of the limit transfer, information about validity, and administration data.

Action

You can use various functions in this overview. In each case, select one of the limit transfers and choose the appropriate pushbutton.

Action	Use
Refresh	You can refresh the overview of limit transfers.
Copy Limit Transfer	You can copy an existing limit transfer and then change it as required.
Change Limit Transfer	You can change an existing limit transfer. You change limits in the same way as you create limit transfers [page 1543].
Display Limit Transfer	You can display an existing limit transfer. You display limit transfers in the same way as you create them.
Set Deletion Flag	You can flag a limit transfer for deletion. **⚠ Caution** Note that this deletion indicator cannot be removed.
Note Maintenance	You can create a note for the limit transfer.

6.5.2.3 Creating Limit Transfers

Prerequisites

You have called the *Edit Limit Transfers* screen.

You can access this screen using one of the following functions:

- Edit limits [page 1533]
- Display an overview of limit utilizations [page 1561]

Then choose:

1. The two limits for which a limit transfer is to be carried out

   > ⚠ Caution
   >
   > A limit transfer can be carried out only between two limits that are both assigned to the same limit type.

2. *Limit Transfer* with the quick info *Create New Limit Transfer*

The *Edit Limit Transfers* screen appears.

The system displays automatically the limit characteristics that are relevant for the limits in question.

Procedure

Carry out the following steps:

1. Enter a valid-from date and a valid-to date for the limit transfer.
2. Enter an amount for the limit transfer.
   You can display the current items of the relevant limit by choosing *Limit History* at the bottom of the screen.
3. Enter a currency for the limit transfer.
4. Select a radio button to choose the limit to which the defined amount is to be added.
5. Choose *Save*.

Result

The limit transfer has been created. The amount has been added or subtracted accordingly and the system returns to the *Overview of Limits* or *Overview of Limit Utilizations*.

The limit transfer is displayed in the relevant limits as an additional limit item with the appropriate +/- sign.

6.5.2.4 Displaying Change Documents for Limits

Prerequisites

You have activated the use of change documents for limits in Customizing for *Bank Analyzer* under *Analytics* > *Limit Manager* > *Basic Settings in Limit Manager* > *Edit Configuration*.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Management* > *Display Change Documents for Limits*.
2. Enter the required selection criteria:
   - Limit area
   - Limit type
   - Date as of which the changes are to be displayed
   - Name of the user whose changes are to be displayed
3. Choose  .

Results

The system displays the required list of changes made to the limits (provided that appropriate data is available).

6.5.3 Limit Reservations

You can use the reservation function to reserve the free part of a limit for a particular period. Reservations are included in the limit utilization run.

They are given an external reservation ID that can be stored in the transaction data. The limit utilization run assigns the relevant business transactions to the reservation. There are two types of limit reservation: reservations for single transactions and reservations for multiple transactions. If any part of the reservation is left over, this is taken into account in the utilization run.

The user enters the attributable amounts directly in the reservation.

6.5.3.1 Creating a Limit Reservation

Use

You use this procedure to create new limit reservations.

Prerequisites

Before you create limit reservations, you need to have created the relevant determination methods and limit types in Customizing. Limits and limit utilizations also have to be available in the system.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Reservations* > *Create a Reservation*.
2. Create a reservation ID, description, and, if required, a search term.
3. Enter the valid-to date for the reservation. The system sets the current date as the valid-from date.
4. If the reservation refers to several financial transactions and if it is to be reduced gradually by these during the utilization run, select the *Multi-Transaction Reservation* checkbox.
5. If you want to use the review function, enter a review date and a review recipient.
6. On the *Reservation Data* tab page, enter the (direct) reservation characteristics.
7. On the *Attributable Amounts* tab page, enter the attributable amounts of the reservation for the individual determination methods.
8. Carry out a preliminary check of the reservation, if required. Choose  .
9. Save your entries.

You can enter all direct limit characteristics and attributable amounts for all available determination methods for a limit reservation.

When you save the reservation, the system derives characteristics for the limits in question and calculates attributable amounts for limit utilizations with status 2. The system calculates attributable amounts for limit types that have only status 1 in accordance with Customizing settings during the next utilization run.

Choose  to simulate the calculation of attributable amounts for the reservation. This enables you to identify any overdrafts. The system returns the results in a dialog box. You can also display the individual utilizations of this check in this dialog box. Choose  .

> **Note**
>
> You assign a reservation to one or more financial transactions in the financial transaction. To do this, enter the external reservation ID of the Limit Manager reservation in the Limit Manager-specific characteristic */BA1/C61RESEID*. The system automatically calculates attributable amounts for valid reservations during every utilization run.

6.5.3.2 Changing a Limit Reservation

Use

You use this function to change existing limit reservations.

Prerequisites

Limit reservations already exist in the system.

Procedure

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Reservations* > *Change a Reservation*.

You can select existing reservations using the external reservation ID and then make any necessary changes.

The table below shows the various actions available in change mode.

Action	Use
Other Reservation	You can display another reservation.
Display <-> Change	You can switch to display or change mode. The check function is not available in display mode.
Check	You can check changes made to the reservation.
Edit Note	You can create a note for the reservation.

> **Note**
>
> You cannot change reservation characteristics once you have saved it for the first time.

6.5.3.3 Displaying a Limit Reservation

Use

You use this function to display existing limit reservations.

Prerequisites

Limit reservations already exist in the system.

Procedure

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Reservations* > *Display a Reservation*.

You can select existing reservations using the external reservation ID and display the details required.

The table below shows the various actions available in display mode.

Action	Use
Other Reservation	You can display another reservation.
Display <-> Change	You can switch to display or change mode. The check function is not available in display mode.
Edit Note	You can display the notes belonging to the reservation.

6.5.3.4 Collective Processing of Limit Reservations

Use

You use this function to display an overview of existing limit reservations and to create new reservations or change existing ones.

Prerequisites

Limit reservations already exist in the system.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Reservations* > *Collective Processing of Reservations*.
   The *Collective Processing of Reservations* screen appears.

2. You can restrict the amount of reservations to be selected in the *Reservation Data* data group by entering the following information:
   - Reservation number
   - External reservation ID
   - Search term for reservation
   - Validity interval for reservation end
   - If you set the *Deleted Reservations* checkbox, the system also displays reservations that have been marked for deletion.
3. You can also restrict the amount of reservations to be selected in the *User Data* data group. To do so, enter the user who created the reservation or who last changed it.
4. Choose *Execute*.

Select a reservation from the overview list and choose a function from the available pushbuttons.

You can display a list of various reservations. You can perform the following actions for entries in this list.

Action	Use
Create	You can create new reservations.
Change	You can switch to change mode and process the reservation selected.
Display	You can switch to display mode and display the reservation selected.
Copy	You can copy a reservation.
Notes	You can create a note for the reservation.
Review	You can trigger a review workflow for the reservation.

6.5.3.5 Displaying Change Documents for Limit Reservations

Use

You use this function to display change documents for limit reservations.

Prerequisites

You have activated logging of changes for reservations in Customizing (*Bank Analyzer* > *Analytics* > *Limit Manager* > *Basic Settings in Limit Manager* > *Edit Configuration*).

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Reservations* > *Display Change Documents for Reservations*.
   The *Reservations: Display All Change Documents* screen appears.
2. You can restrict the amount of reservations to be selected in the *Reservation Data* data group by entering the following information:
   - Reservation number
   - External reservation ID
   - Search term for reservation
3. You can also restrict the amount of reservations to be selected in the *User Data* data group. You can specify the person who created the reservation and the date on which it was created or the last person to change the reservation and the date on which it was last changed.
4. Choose *Execute*.

The system displays a list of all change documents for the reservations selected.

6.5.3.6 Displaying Online Limit Check Logs

Use

If you want to monitor the results of the online limit check, and more specifically identify limit violations, you can display the logs for the online limit check. You can also use this report to delete logs that are no longer required.

You can use the enterprise services of the *Credit Portfolio Limit Reservation* business object for the online limit check. For more information, see SAP Help Portal at *http://help.sap.com/* > *SOA* > *SAP for Industries* > *SAP for Banking* > *Enterprise Services for banking services from SAP*.

Prerequisites

- You are using limit reservations to allocate a limit for a transaction that is not yet available in the system and have run an online limit check.
- You have specified that logs are created for reservations for the online limit check in Customizing for *Bank Analyzer* > *Analytics* > *Limit Manager* > *Reservations in Limit Manager* > *Enter Basic Settings for Reservations*.

Procedure

Display Logs

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Reservations* > *Display Online Limit Check Logs*.
2. You can restrict the number of logs that are selected as follows:
   - You can enter the following in the *Selection by Reservation Data* group box:
     - External reservation ID
     - Last changed by
     - Result of online check for a reservation
     - If you select the *Only for Saved Reservations* checkbox, logs for unsaved reservations are excluded.
   - In the *Selection by Time Stamp* group box, you can restrict the period in which the logs were created.
   - In the *Selection by Limit Characteristics* group box, you can enter the limit characteristics you defined so that only the logs for reservations with these characteristic values are selected.
3. Choose .

A list of all the logs for the online limit check appears. Select an entry and choose one of the available pushbuttons. With *Reservation* you can navigate to display mode and display the corresponding reservation.

Adjust the Layout of the Log List

1. Perform the first two steps under *Display Logs*. After you have entered your selection criteria, enter the name of the layout that you want to use for the log list in the *Control* group box. You can use this function to influence the list column structure and sort criteria, for example.
2. Choose .

A list of all the logs for the online limit check appears with your selected layout.

Delete Logs

1. Perform the first two steps under *Display Logs*. After you have entered your selection criteria, select the *Delete Logs Immediately* checkbox in the *Control* group box.
2. Choose .
3. Confirm the deletion of the selected logs (in a dialog box).

6.5.4 Limit Utilizations

The attributable amount of a transaction (single record) is the risk amount that was determined for a single transaction and allocated to all the relevant limits using a particular determination method.

By means of limit characteristics and their values, the attributable amounts are added together to form the limit utilizations (totals record).

The limit utilization for a limit is, therefore, the total of the attributable amounts of all the transactions that were allocated to the limit on the basis of their characteristic values. This amount is determined in the limit utilization run and compared to the relevant limit amount in reporting.

6.5.4.1 Limit Utilization Run

Use

You use the limit utilization run to generate limit utilizations in order to analyze the utilization of limits from the current transaction position. Consequently, it adds up the attributable amounts of all the transactions that were allocated to the limit on the basis of their characteristic values.

You can use the SAP parallel processing tool for the limit utilization run.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Utilizations* > *Limit Utilization Run*.
   The *Limit Utilization Run* screen appears.
2. In the *Utilizations* data group, enter the following parameters for the run:
   - Limit area
   - Determination method
   - Base date of utilization run: The business key date for which the evaluation is to take place.
   - Time of worklist selection: The business key date for which the worklist is to be read (for example, results and key figures from the RDL).
   - Selection date for data enrichment: The business key date for which the enrichment is to be read (for example, business data from the SDL).
3. In the *Data Source* group box, enter the following parameters, which enable you to clearly define the selection for the run:
   - Selection group ID: The selection group ID, together with the selection ID, defines a selection.
   - Selection ID: Uniquely identifies the relevant primary and secondary data sources, which you have defined for the limit utilization run in Customizing for *Bank Analyzer* > *Analytics* > *Limit Manager* > *General Settings for Data Selection* > *Selection in Limit Manager* > *Edit Basic Settings for Data Sources*.
     Based on these selection IDs, the system reads the attributable amounts determined in the run and updates the relevant limit during the limit utilization run.
     Sel. Criteria: This is where you can enter selection criteria in the general primary data source that you have defined under the selection ID.
4. In the *Technical Settings* data group, select the following optional checkboxes:
   - Parallel processing for single records
     If you select this checkbox, the system updates the single records in parallel processes.
   - Parallel processing for totals records
     If you select this checkbox, the system updates the totals records in parallel processes.
5. Choose *Execute*.

Result

The system displays a log [page 1554] for the limit utilization run. You can display the results in the overview of limit utilizations [page 1561].

Each limit utilization run produces two result records: status 1 and status 2. Directly after the run, the results for status 1 and status 2 are identical. However, if you create a reservation, for example, for the same key date as that

of the run, the reservation amount is included in the result with status 2. The reservation does not affect status 1 until the run is carried out again.

Depending on the Customizing settings for the limit type, the system can create limits automatically during a limit utilization run; it does so if there are no suitable limits for the characteristic combination of a transaction. You make the relevant settings in Customizing for *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Levels in Limit Manager* > *Edit Limit Types*. If the option for automatic creation of limits has not been activated in Customizing, the utilization run ignores the characteristic combinations concerned.

6.5.4.2 Postprocessing Limit Utilization Runs

Use

Using the postprocessing function, you can correct and update any erroneous transactions that already exist in the data pool but were either not analyzed at all, or analyzed incorrectly in the limit utilization run and, hence, not updated correctly. You can also use the postprocessing function to re-post transactions that were analyzed correctly in the limit utilization run but that are to be changed.

When Limit Manager postprocesses limit utilization runs, it does the following:

- It accesses the results data for default risks that have changed (for example, from the RDL) as per the set of data specified.
- It assigns this results data to the utilization run selected as limit utilizations.

The postprocessing function updates status 1 and status 2 in the result records, which are displayed in the overview of limit utilizations [page 1561].

Prerequisites

The system already contains at least one limit utilization run with a single transaction you want to change or update to this run.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Utilizations* > *Postprocess Limit Utilization Runs*.
   The *Postprocessing of a Limit Utilization Run* screen appears.
2. In the *Identification of the utilization run that is to be corrected* data group, enter the selection criteria required to identify uniquely the limit utilization run that you want to correct.
3. In the *Data Collection* data group enter information about the data that is to be selected:
   - Selection group ID: The selection group ID, together with the selection ID, defines a selection.

- Selection ID: Uniquely identifies the relevant primary and secondary data sources, which you have defined for the limit utilization run in Customizing for *Bank Analyzer* > *Analytics* > *Limit Manager* > *General Settings for Data Selection* > *Selection in Limit Manager* > *Edit Basic Settings for Data Sources*.
  Based on these selection IDs, the system reads the attributable amounts determined in the run and updates the relevant limit during postprocessing.
  Sel. Criteria: This is where you can enter selection criteria in the general primary data source that you have defined under the selection ID. Here you can specify the data that has changed since the last utilization run and that you want to postprocess.

4. In the *Technical Settings* data group, select the following optional checkboxes:
   - Set the *Test Run* checkbox if you want to test the postprocessing first, before any data is written to the database tables in Limit Manager.
   - Set the *Display Results* checkbox if you want the system to display the changes made to the data in a dialog box after the data from the primary data source has been processed.
5. Then choose *Execute* to start the postprocessing run.

Result

The system displays a log [page 1554] of the postprocessing of the limit utilization run. You can display the corrected results in the overview of limit utilizations [page 1561].

6.5.4.3 Postprocessing Reservations for Limit Utilization Runs

Use

When reservations are changed, the system automatically updates limit utilizations. This means that postprocessing is usually not necessary. To enable market data changes (for example, exchange rates) to be taken into account during the day, reservations can be postprocessed.

When Limit Manager postprocesses reservations for limit utilization runs, it does the following:

1. Limit Manager selects the specified reservations and updates attributable amounts to be translated.
2. It assigns these updated amounts to the selected utilization run as limit utilizations.

The postprocessing function updates status 1 and status 2 in the result records, which are displayed in the overview of limit utilizations.

Prerequisites

There is already at least one utilization run in the system for which you want to update the reservations.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Utilizations* > *Postprocess Reservations for a Limit Utilization Run*.
2. In the *Identification of the Utilization Run That Is to Be Corrected* data group, enter the selection criteria required to identify uniquely the limit utilization run that you want to correct.
3. In the *Reservations* data group, enter the following information about the data that is to be selected:
   - External reservation ID, search term or reservation end date
   - Deleted reservations: If you select this checkbox, the system also selects deleted reservations when data is selected.
   - Selections: You can define selections for characteristics of the reservations that you now want to postprocess.
4. In the *Technical Settings* data group, select the following optional checkboxes:
   - Set the *Test Run* checkbox if you want to test the postprocessing without writing any data to the database tables in Limit Manager.
   - Set the *Display Results* checkbox if you want the system to display the changes made to the data in a dialog box after the data from the primary data source has been processed.
5. Choose to start postprocessing.

Result

The system displays a log of the postprocessing of the limit utilization run. You can display the corrected results in the overview of limit utilizations.

More Information

Displaying Logs [page 1554]

Displaying an Overview of Limit Utilizations [page 1561]

6.5.4.4 Displaying Logs

Prerequisites

The system already contains limit utilization or postprocessing runs.

Context

You use this function to display and manage the logs for the limit utilization runs and postprocessing carried out in the system. The system displays any errors that occurred during a run or during postprocessing. You can also create statistics for the results.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Utilizations* > *Display Logs*.

   The *Display Application Log* screen appears.
2. In the *Run Type* data group, specify whether you want to display a log for a limit utilization run or for postprocessing.
3. Specify the limit area so that the limit utilization or postprocessing run can be identified.
4. Enter the following dates:
   - *Date of Utilization Run* The business key date used as a basis to determine the attributable amounts. The default value is today's date.
   - *Technical Start Date*
   - *Technical Start Time*: If you enter the precise system time you can select multiple runs that were carried out on the same day.
5. Choose *Execute*.

Results

The system displays the logs that meet your selection criteria.

6.5.4.5 Run Administration

Definition

Run administration includes the following functions:

- Execute or create run
- Display an overview of runs
- Display Application Log
- Edit run
- Manage run
- Replace run

- Select run for archiving
- Delete run
- Log of the deletion function

> **Note**
>
> The above functions are not all available for each application. For more information, see the application-specific documentation.

Use

The following table lists the runs available for each application:

Application	Run
General Calculation and Valuation Methods [page 376]	Account Pooling [page 383] or Account Pooling (CVPM) Facility Distribution [page 399] or Facility Distribution (CVPM) Determination of the Free Line for Facilities [page 395] or Determination of the Free Line (CVPM) Collateral Distribution [page 404] or Collateral Distribution (CVPM) Determination of Default [page 410] Stress Runs [page 429]: • Stress run for account pooling or stress run for account pooling (CVPM) • Stress run for facility distribution or stress run for facility distribution (CVPM) • Stress run for determination of the free line or stress run for determination of the free line (CVPM) • Stress run for collateral distribution or stress run for collateral distribution (CVPM) • Stress run for determination of default
Credit Risk [page 1191]	Credit Exposure Run [page 1238] or Credit Exposure Run (CVPM) Country Risk Run [page 1336] (prototype for country risk) **Stress run:** Credit Exposure Stress Run [page 1261] or Credit Exposure Stress Run (CVPM)

Application	Run
Historical Database [page 1478]	**Version management:** Run for generating historical data for data layers [page 1481] Run for generating historical data for the bank's own models [page 1484] Uploading of Files [page 1486] **Calculation functions:** Determining Default Rates [page 1488] Determining Average Default Rates [page 1491] Determining Default Figures [page 1493] Calculation of Migration Matrices [page 1496] **Data retrieval:** Run for supplying data to models [page 1498] Downloading of Files [page 1500] **Stress tests:** Stress Run for Supplying Models with Data [page 1505] Generating Scenario Data in the Source Data Layer [page 1506]
Generic BI Data Extraction [page 1638]	Testing the BI Extractor [page 1639] BI extraction run **Note** Extraction runs are created and executed in SAP NetWeaver Business Intelligence (BI). The system displays information about extraction runs in run administration of Bank Analyzer.
Regulatory Reporting Interface	Data Extraction Runs
Limit Manager [page 1529]	Limit Utilization Run [page 1551]
Fair Value Effectiveness Test for Hedging Relationships [page 1036]	Effectiveness Test Run
Cash Flow Hedge Analysis [page 1107]	Valuation Run [page 1128] Subvaluation Run Aggregation Runs [page 1129]

Application	Run
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169] Portfolio Item Runs [page 1172] • Period-end run for portfolio items (subledger scenario) • Period-start run for portfolio items (subledger scenario) • Portfolio item run (merge scenario)

> **Note**
>
> For some of the Bank Analyzer components, you can use the Schedule Manager to schedule and control jobs. If you use multiple applications, you can define the sequence in which the runs are to be carried out. For more information, see Schedule Manager [page 437].

More Information

Status Overview for Run Administration [page 1248]

6.5.4.5.1 Features of Run Administration in Limit Manager

Use

You use the run administration function in Limit Manager to obtain an overview of all the limit utilization runs and postprocessing runs that have been carried out.

Features

Run administration in Limit Manager enables you to check the logs of limit utilization runs and postprocessing runs. You can also navigate from a particular run to the overview of limit utilizations, and you can display information about the technical time stamp of a run.

6.5.4.5.2 Run Administration in Limit Manager

Use

This report enables you to use the SAP List Viewer (ALV) to display the runs you have carried out. In the detail log for a run, you can change the status of the run manually. For example, you can complete a run, flag a run for deletion, or set the status of a run to "Completed Successfully".

Prerequisites

You have already started at least one run.

Procedure

1. You call the report on the *SAP Easy Access* screen under *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Utilizations* > *Run Administration in Limit Manager* .
   The *Run Administration: Overview* screen appears.
2. Enter the required selection criteria in the *Sel. Criteria of Run* data group.
   - The *Client of Run Administration* field has been defaulted to *61 Limit Manager* and cannot be changed.
   - The *External ID of Run* field contains the key that was used to define the runs.
   - The *Key Date* field contains the key date of the utilization run.
   - In the *Run Category* field you define which category of limit utilization run the system is to select:
     - RDL for the limit utilization run
     - CRE for the prototype functions (country risk)
   - In the *Version of Run* field you can specify the version of the run for a key date.
   - The *System Status* defines which runs of a particular status are to be selected. A run can have one of the following statuses:
     - Canx: run canceled
     - End: run finished
     - Crtd: run created
     - Del.: deletion flag
     - Rvke: deletion flag revoked
3. If required, in the *Layout* field enter a layout that has already been defined.
4. Choose .
   The system displays an overview of the selected limit utilization or postprocessing runs.

Action

You can use various functions in this overview.

- The selection criteria chosen previously are displayed at the top of the screen.
- A list of the limit utilization and postprocessing runs that meet the selection criteria appears at the bottom of the screen.

To display a particular run, double-click it.

Output

You can use the following functions in the detail screen.

Choose  with the quick info *Change Run* to change the status of a run directly.

> **Note**
>
> You use the  function to set the deletion flag and then in the*Delete Limit Runs Flagged for Deletion* function to delete the run from the database. To reset the deletion flag choose  .
>
> Choose  to switch to change mode.
>
> Choose *Messages* to display more information about the run and the messages for the run.
>
> Choose *Utilizations* to display an overview of the utilizations of a run.
>
> Choose  to display the time stamp collections.

6.5.5 Limit Monitoring

You can use the limit monitoring function to display an overview of limit utilizations.

You can also display a list of data in order to monitor limits for the business partner hierarchy.

You can use the BI data extraction function to extract results data from Limit Manager in the Bank Analyzer core system to SAP NetWeaver Business Intelligence (BI). In BI, you use the extracted data for reporting. For more information, see Generic BI Data Extraction [page 1638].

You can use generic ad hoc reporting to display the data from Limit Manager. The Data Processing Framework is used to select the data. You can display different data records, depending on the application used in the Data Processing Framework. For more information, see Generic Ad Hoc Reporting [page 1649].

Limit Manager can use general primary data sources as servers. At runtime, the data source servers analyze the selection criteria of the general primary data sources and return the data records that they determine to the calling application by means of the general primary data source.

The table below contains the servers that can be used in Limit Manager.

Technical Name	Server Description	Application	Single Record	Worklist	Package	Correction Server
LMLIMITS	Limit Manager Limits	Limit Manager	X	X	X	-

LMUTLSNGL	Limit Manager Attributable Amounts	Limit Manager	X	X	X	-
LMUTLSUMS	Limit Manager Utilizations	Limit Manager	X	X	X	-

Single record - single record selection, worklist - worklist or multiple selection, package - server supports packaging of worklists, correction server - supports the data flow management of the correction server

For more information about primary data sources and servers, see Servers for General Primary Data Sources [page 1597].

6.5.5.1 Displaying an Overview of Limit Utilizations

Use

This report enables you to monitor existing limits and their utilizations.

Prerequisites

Limit utilizations have already been calculated in the system in the limit utilization run.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Monitoring* > *Display an Overview of Limit Utilizations*.
   You can also call the overview of limit utilizations from the overview of limits. For more information, see Editing Limits [page 1533].
   The *Overview of Limit Utilizations* screen appears.
2. In the *General Selection Criteria* data group, enter the limit area and the base date (required entries). The default value for this date is today's date. You can restrict the amount of data selected further by specifying the limit type and the determination method.
3. In the *Settings for Limits* data group, select the *Unused Limits* checkbox to display all the limits for which there are no utilizations. If you select this checkbox, you have to enter a key date for unused limits. The default value for this date is today's date. If you have defined a display filter in Customizing for *Bank Analyzer* > *Analytics* > *Limit Manager* > *Characteristics in Limit Manager* > *Edit Display Filter*, you can choose this here.
   On the *Direct Characteristics* and *Derived Characteristics* tab pages, you can restrict the selection to particular characteristic values.

4. If required, enter an individual layout (totals records, single records), a display currency, and a rounding factor in the *Output Control* data group.
5. If required, enter the following selection criteria in the *Exception Reporting Control* data group:
   - Exceeded limits only
   - Exceeded limits and identical characteristic combinations
6. If required, enter the exact system date and time in the *Selection by Time of Utilization Runs* data group. For example, if four runs were started on the same date, you can identify each run uniquely by specifying the start time.
7. Choose  .
   The system displays an overview list of all the existing limit utilizations at totals record level that meet your selection criteria.
   The first line contains the limit utilizations in status 1.
   The second line contains the limit utilizations in status 2.
   A red traffic light is shown if the limit has been exceeded. If the critical limit utilization has been exceeded, the system displays a yellow traffic light.

> **Note**
>
> The results of the limit utilization runs are stored in status 1. If a limit utilization changes as a result of a reservation, it is displayed in status 2 only. By contrast, limit utilization runs or postprocessing runs are displayed with status 1 and status 2.

Interaction

You can use various functions in the overview of limit utilizations. In each case, select one of the limit utilizations and choose the appropriate pushbutton:

Interaction	Use
	You can refresh the overview of limit utilizations.
	You can change an existing limit. You change limits in the same way as you create limits [page 1535].
	You can create a limit transfer. For more information, see Creating Limit Transfers [page 1543].
	You can display the limit utilizations at single record level. From this display, you can navigate to: • Details about the master data for the transaction or reservation • The detail log for attributable amount determination
	You can display the business partner master data.
[Customizing] *Limit Type Customizing*	You can display the Customizing settings for the limit type.

Interaction	Use
	You can display the totals records to determine whether a lock exists for a limit. If the limit is defined as locked with respect to financial objects, new transactions can trigger a lock violation. This is displayed in the *Lock Viol.* (lock violation) field. Choose this totals item to branch to the list of objects that caused a lock violation.

Single Record Level

If you navigate to single record level, you can then navigate to other screens:

Interaction	Use
	You can navigate to the master data of a financial transaction or to the master data stored in the Results Data Layer (RDL). This function is based on the Customizing settings for the determination methods. You can branch to the master data that is defined as characteristics in a determination method. If you want to branch to the results data in the RDL, you need to add Bank Analyzer characteristic /BA1/CR3CEID (UOID CRE) in the Customizing settings for the characteristic assignment of a determination method.
	You can branch to the overview of results objects in Credit Exposure. When you select a results row and call online detail reporting (*Edit* > *Goto Online Detail Reporting*), you can access generic detail reporting in order to display information about how the default risks were calculated.

> **Note**
>
> The display contains the fields from the Limit Manager database tables. The *Object Number* field therefore contains a UOID for the business transaction. You can display additional characteristics from the determination method in order to display the transaction number. We recommend that you use your own custom layouts when you call this function so that the system displays all the relevant characteristics straight away.
>
> Online detail reporting enables you to check the calculation of the credit exposure by analyzing a smaller set of data online. For more information, see Online Detail Reporting for Credit Exposure [page 1255].

Detail Overview for Lock Violation

After you navigate to the detail overview for lock violations, the system compares the current run that was used to call the report with the run that was valid before the limit was set. The following information is available:

- Log for a limit run with message texts. All the objects that caused a lock violation are marked red.
- All the current objects and all the objects that caused a limit lock violation are displayed in the left-hand window.
- All the objects that existed in the run before the lock was set are displayed in the right-hand window.

6.5.5.2 Limit Utilizations: Business Partner Hierarchies

Use

You use this function to depict the structures of the corporate groups of your business partners for whom you want to monitor the limits and utilizations for each entity in the corporate group. This is particularly relevant for those limit types for which you want the system to consider the relationships between business partners when it calculates the utilizations of the limits

Integration

Unlike the function for displaying an overview of limit utilizations [page 1561], in this function you can reflect the structure of the corporate group to which the business partner belongs in a tree structure. This enables you to see how utilizations were assigned to particular limits based on the relationships between the parts of the structure.

Prerequisites

In order to use this function you must have already created limit types in Customizing for Limit Manager that reflect the relationships between the business partners. You also must have created limits and utilizations for these limit types.

In addition, you have created relationships for the business partners in question in the system.

Features

This report selects the business partners that are directly or indirectly related to the business partner you entered (in terms of the relationship category specified). These business partners are then displayed in a tree structure.

For each business partner, you can display the limit valid on the key date, and the current utilization at totals record level by double clicking the business partner or by choosing *Display Utilizations*.

If you choose *Single Utilizations* in the lower screen area you can drill down to the individual utilizations that make up the totals record.

You can also navigate to the master data of the business partner.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Monitoring* > *Limit Utilizations: Business Partner Hierarchies*.

The *Limit Utilizations for Business Partner Hierarchies* screen appears.

2. To start the report, enter the relevant values in the required fields, and specify the business partner.
   You can restrict the selection of data to that for a particular limit type. If you enter more than one limit type, after you have started the report (by choosing *Display Utilizations*) the system displays the limits and utilizations for the business partner in question for all limit types that meet the selection criteria.
   Enter the following data:
   - Limit area
   - Limit type (optional)
   - Determination method (this is optional; it is relevant only if you do not specify the limit type)
   - Status of the utilization
   - Business partner (the system uses the business partner you enter here to find the relationship and create the tree structure)
   - Base date of determination (the key date for which the utilizations were calculated)
   - Relationship category
3. If you want to display limits that have not been utilized by the base date specified, select the *Unused Limits* checkbox. If you do so, you must enter the key date for which these limits are valid.
4. Choose  .
   The system displays a list of business partner hierarchies. The system displays the business partner hierarchies in the upper screen area. If you double click a business partner in the upper screen area, the relevant utilizations are then displayed in the lower screen area.

6.5.6 Archiving of Limit Data

A large amount of data is saved in the system when Limit Manager is in productive use. To prevent evaluations being slowed down by too much data, you can use the to delete the following data from the system:

- `/BA1/R3_LM` (*Archive Limits*)
- `/BA1/R3_UT` (*Archive Limit Utilizations*)

When data is archived, it is removed from the system. You can display your archived data by calling the Archive Information System. Displaying this archived data can seriously impair system performance, however.

> **Note**
>
> In addition to archiving data, you can delete data permanently using one of the following reorganization functions in Customizing under *Bank Analyzer* > *Analytics* > *Limit Manager* > *Reorganization in Limit Manager*:
>
> - Reorganize Limit Types
> - Reorganize Utilizations

6.5.7 Tools

Use

Limit Manager contains the following tools:

- Schedule Manager: You can start complex business processes and monitor them in Schedule Manager. If you have the appropriate authorization you can define task plans. You can create composite tasks (for example, Credit Exposure run, Limit Manager run) in the structure tree (task plan) and schedule them in the daily overview.
- Derivation of Review Dates for Limits: You can let the system derive the review data for a selection of limits.
- Review of Limits: This tool checks whether limits are subject to a review process, and triggers this review process.
- Review of Reservations: This tool checks whether reservations are subject to a review process, and triggers this review process.
- Display Logs: This tool is used to display the logs from limit utilization runs and postprocessing runs.
- Comparison of Reservations and Financial Transactions - Reorganization: You can use this tool to check which financial transactions have reduced reserved amounts in a limit utilization run. You can also delete data that is no longer required from the database.
- Check GUID of a Determination Method: The system uses technical GUIDs when it processes financial transactions. Since these cannot be understood by the user, you can use this tool to display the financial transaction data for a GUID.

Activities

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Tools*.

6.5.7.1 Schedule Manager

Definition

A tool in Bank Analyzer that you can use to execute and monitor complex business processes, such as period-end closing. When you schedule a run, you always have to specify the program with which the system calls the run. There is a separate program for each run category in an application.

The table below shows for which runs and run categories the Schedule Manager can be used, and their corresponding programs.

Application	Run Category	Program
Account Pooling	Account Pooling [page 383]	/BA1/DE_GM_MAIN_COMP

Application	Run Category	Program
Account Pooling	Stress Run for Account Pooling [page 429]	/BA1/DE_GM_MAIN_COMP_SCEN
Determination of the Free Line for Facilities	Determination of the Free Line for Facilities [page 395]	/BA1/DE_GM_MAIN_FL
Determination of the Free Line for Facilities	Stress Run for Determination of the Free Line for Facilities [page 429]	/BA1/DE_GM_MAIN_FL_SCEN
Distribution of Facilities	Distribution of Facilities [page 399]	/BA1/DE_GM_MAIN_RLV
Distribution of Facilities	Stress Run for Facility Distribution [page 429]	/BA1/DE_GM_MAIN_RLV_SCEN
Collateral Distribution	Collateral Distribution [page 404]	/BA1/DE_GM_MAIN_SV
Collateral Distribution	Stress Run for Collateral Distribution [page 429]	/BA1/DE_GM_MAIN_SV_SCEN
Determination of Default	Determination of Default [page 411]	/BA1/RA_DFLT4280_RUN
Determination of Default	Stress Run for Determination of Default [page 429]	/BA1/RA_DFLT4280_RUN_SCENARIO
Fair Value Effectiveness Test [page 1036]	Effectiveness Test Run [page 1039]	/BA1/R8_FVHE_MAIN_RUN
Fair Value Effectiveness Test	Generation of Time Series for Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MAIN_RUN
Fair Value Effectiveness Test	Modifying a Time Series for the Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MODIFY
Cash Flow Hedge Analysis [page 1107]	Subvaluation Run [page 1128]	/BA1/R5_CFHA_BTC_RUN
Cash Flow Hedge Analysis	Aggregation Run [page 1129]	/BA1/R5_START_BATCH_AGGR
Cash Flow Hedge Analysis	Joint call of Valuation Run [page 1111] and Aggregation Run [page 1123]	/BA1/R5_EXECUTE_CFHA
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169]	/BA1/R9_VPM_BTC_ID_RUN
Portfolio Fair Value Hedge	Portfolio Item Run (Merge Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_RUN
Portfolio Fair Value Hedge	Period-End Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PE_RUN
Portfolio Fair Value Hedge	Period-Start Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PS_RUN

Application	Run Category	Program
Portfolio Fair Value Hedge	Define End of Valuation Period	/BA1HM/RAH_M_PFVH_SET_VP_END
Portfolio Fair Value Hedge	Adjust Initial Portfolio Fair Value Hedges for Next Valuation Period	/BA1HM/RAH_M_PFVH_ADJUST
Portfolio Fair Value Hedge	(Re-) Designation of Initial Portfolio Fair Value Hedges	/BA1HM/RAH_M_PFVH_DESIG
Credit Exposure [page 1192]	Credit Exposure Run [page 1238]	/BA1/R2_CRA_MAIN_CRE
Credit Exposure	Credit Exposure Stress Run [page 1261]	/BA1/R2_CRA_MAIN_CRE_SCEN
Credit Exposure	Credit Exposure Run for Revolving Retail	/BA1/RR2_CRE_REVRET
Credit Exposure	Approximate Stress Run [page 1274]	/BA1/RR2_CRE_SCENARIO
Credit Exposure	Aggregation Run for Ad Hoc Calculation [page 1276]	/BA1/RR2_SCEN_AGGR
Country Risk [page 1326]	Country Risk Run [page 1336]	/BA1/R2_CRE_MAIN
Historical Database [page 1478]	Historization Run for Data Layers [page 1481]	/BA1/R6_HIST_RUN
Historical Database	Historization Run for Bank's Models [page 1484]	/BA1/R6_HIST_RUN_MODEL
Historical Database	Determine Default Rates [page 1488]	/BA1/R6_CF_DEFAULT_4210
Historical Database	Determine Average Default Rates [page 1491]	/BA1/R6_CF_DEFAULT_4220
Historical Database	Determine Default Figures [page 1493]	/BA1/R6_CF_DEFAULT_KEYFIGURES
Historical Database	Exporting Data to In-House Models [page 1498]	/BA1/R6_MOD_RUN
Historical Database	Calculation of Migration Matrices [page 1496]	/BA1/R6_CF_MIGRATIONS
Historical Database	Stress Run for Supplying Models with Data [page 1505]	/BA1/R6_MOD_RUN_SCENARIO
Historical Database	Stress Run for Supplying Data [page 1506]	/BA1/R6_FDB_RUN_SCENARIO
Regulatory Reporting Interface	Processing Framework Run Copying the Start ID	/BA1/S1_COPY_STARTID

Application	Run Category	Program
Regulatory Reporting Interface	Processing Framework Run Creating Runs	/BA1/S1_CREATE_RUN
Regulatory Reporting Interface	Processing Framework Run Release Run	/BA1/S1_RELEASE_RUN
Regulatory Reporting Interface	Processing Framework Run Creating a Worklist	/BA1/S1_CREATE_WRKL
Regulatory Reporting Interface	Processing Framework Run Processing a Run	/BA1/S1_PROCESS
Regulatory Reporting Interface	Data Extraction Runs	/BA1/S3_EXTRACT_RUN
Limit Manager [page 1529]	Limit Utilization Run [page 1551]	/BA1/R3_UTILIZATION_RUN_DPF

More Information

Schedule Manager

6.5.7.2 Derivation of Review Data for Limits

Use

You can use this function to derive review data for the selection of limits. The review data comprises the person who is to review the limit (review recipient) and the date on which the limit is to be reviewed (review date). You can start a test run first to check the changes made to the data before the changed data is written to the database.

Integration

The review function is part of the processing of limits, and can be activated for individual limit types. For more information about Customizing for the review process in Limit Manager, see Customizing for *Bank Analyzer* under *Analytics* > *Limit Manager* > *Limit Levels in Limit Manager* > *Edit Limit Types*.

Prerequisites

The system can derive review data only if you created derivation rules in the derivation tool for Limit Manager. For more information about the Customizing of the derivation tool, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Basic Settings in Limit Manager* > *Derivation of Review of Limits in Limit Manager*.

Features

This function derives the review recipient and the review data from the derivation tool, and saves this data in the limits you selected.

Activities

Selection

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Tools* > *Derivation of Review Dates for Limits*.
   The *Derivation of Review Data for Limits* screen appears.
2. In the *General Selection Criteria* data group, enter the following information:
   - Limit area
   - Limit type
   - Review date
3. In the *Technical Settings* data group, enter the following information:
   - Set the *Test Run* checkbox
   - Save the application log
   - Filter variant ID
4. In the *Selection Criteria for Limit Characteristics* data group, enter the selection criteria for limits. You can use the following selection criteria:
   - Display filter

     You define the display filter in Customizing for *Bank Analyzer* under *Analytics* > *Limit Manager* > *Characteristics in the Limit Manager* > *Edit Display Filter*.
   - Direct characteristics
     Direct characteristics in Limit Manager are the characteristics that are supplied from the Source Data Layer (SDL) or from the characteristic repository in Bank Analyzer.
   - Derived characteristics
     In Limit Manager, derived characteristics are the characteristics that are derived from the direct characteristics.
     You can make the settings for deriving the characteristics in Customizing for *Bank Analyzer* under *Analytics* > *Limit Manager* > *Characteristics in the Limit Manager* > *Characteristic Derivation*.
5. Choose *Execute*.
   The system displays an overview list of the limits that were selected and the information about whether the review data has changed.

Action

You can use various functions in this overview. In each case, select one of the limits and choose the appropriate pushbutton.

Action	Use
Log	You can display the application log for this run.
Display Limit	You can display the master data of the limit.

6.5.7.3 Review of Limits

Use

You use this function to send selected limits for review. You can start a test run first to check the limits before these are sent for review.

Integration

The review function is part of the processing of limits, and can be activated for individual limit types. For more information about Customizing for the review process in Limit Manager, see Customizing for *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Levels in Limit Manager* > *Edit Limit Types*.

Prerequisites

You have to have activated the review process for a limit type.

Features

This process triggers a review process for the selected limits.

Activities

Selection

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Tools* > *Review of Limits*.

The *Review of Limits* screen appears.

2. In the *Selection Criteria for Limits* data group, enter the following information:
   - Limit area
   - Limit type
   - Review date
3. In the *Technical Settings* data group, enter the following information:
   - Set the *Test Run* checkbox
   - Save the application log
   - Filter variant ID
4. Choose *Execute*.

The system displays an overview list of the limits that have been selected for review. If you started a test run, the system displays the limits affected. If the run was an update run, the workflow was triggered.

Choose *Limits* to display the master data for a limit.

6.5.7.4 Review of Reservations

Use

You use this function to send selected reservations for review. You can start a test run first to check the reservations before these are sent for review.

Integration

The review function is part of the processing of reservations, and can be activated in Customizing. For more information about Customizing for the review process in Limit Manager, see Customizing for *Bank Analyzer* > *Analytics* > *Limit Manager* > *Reservations in Limit Manager* > *Enter Basic Settings for Reservations*.

Prerequisites

You have to have activated the review process for reservations.

Features

This process triggers a review process for the selected reservations.

Activities

Selection

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Tools* > *Review of Reservations*.
   The *Review of Reservations* screen appears.
2. In the *Reservation Data* data group, enter the following information:
   - External reservation ID
   - Search term
   - Reservation end
   - Review date
3. In the *User Data* data group, enter the following information:
   - The name of the user who created the reservation
   - The name of the user who last changed the reservation
4. In the *Technical Settings* data group, enter the following information:
   - Set the *Test Run* checkbox
   - Save the application log
   - Filter variant ID
5. Choose *Execute*.

The system displays an overview list of the reservations that were selected for review. If you started a test run, the system displays the reservations affected. If the run was an update run, the workflow was triggered.

In the overview list you can navigate to the master data of a reservation by choosing *Reservation*.

6.5.7.5 Comparing Reservations with Fin. Transactions: Reorganization

Use

In the limit utilization run you can use a custom characteristic to assign financial transactions to a reservation. You can use single reservation or a multiple reservation.

The system stores these assignments in an auxiliary table. You can use this table to check the assignments the system made during a limit run. You can also reorganize this data.

Integration

This function is part of the function for processing limit reservations.

Prerequisites

You can compare and attribute business transactions to a reservation only if the following conditions are met:

- The business transaction contains the characteristic */BA1/C61RESEID* and this characteristic is assigned an external reservation ID.
- Limit Manager contains a valid reservation for this external reservation ID. In this instance, you differentiate between a single attribution and a multiple reservation. If you set the *Multiple Reservation* checkbox in the reservation itself, the amount reserved is offset against the attributable amount of the financial transaction for the determination method in question. In this case, the reservation is not removed completely when the reservation is cleared automatically by the financial transactions during the limit utilization run. For more information, see Limit Reservations [page 1544].

Features

You use this function to check how the transaction data was attributed to the reservation, and to reorganize the data of the amounts attributed to limits during the limit utilization runs.

Activities

Selection

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Tools* > *Comparison of Reservations and Financial Transactions - Reorganization*.
   The *Reservations: Compare with Financial Transactions* screen appears.
2. In the *Selection Parameters* data group, enter the following data:
   - Limit area
   - Determination methods
   - External reservation ID
3. Use the *Delete Data* checkbox to define whether the selected data is to be deleted from the auxiliary table in order to reorganize data.
4. Choose *Execute*.

The system displays an overview of the financial transactions that were attributed to limits.

Action

You can use various functions in this overview. Select one of the limit reservations and choose the appropriate pushbutton:

Action	Use
Master Data for Reservation	You can display the master data of the reservation.

Master Data for Transaction	You can navigate to the master data of the financial transaction. This function is based on the Customizing settings for the determination methods. You can branch to the master data that is defined as characteristics in a determination method.

6.5.7.6 Checking of the GUID of a Determination Method

Use

You use this application to check the object number GUID stored in Limit Manager against the combination of characteristic values of the determination method. You use this program if you want to display technical GUIDs instead of transaction IDs in the limit utilization run.

Integration

The GUID, which is stored in the object number in Limit Manager, belongs to a particular combination of characteristic values for a determination method.

Activities

Selection

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Tools* > *Check GUID of Determination Method*.
   The *Check the GUID of the Determination Methods in Limit Manager* screen appears.
2. Enter the following information:
   - Limit area
   - Determination method
   - Object number
3. Choose *Execute*.

The characteristic values are displayed in a dialog box, and you can navigate to the transaction data.

You can also enter the run GUID of a utilization generation run. This enables you to navigate to different applications, such as to the results calculation in Credit Exposure.

6.5.8 Prototype Functions for Country Risk

The programs, analyses, and other applications in this section are available as part of a project solution for country risk. This project solution has not been released for the market.

6.5.8.1 Limit Utilization Run (Country Risk)

Context

The limit utilization run generates limit utilizations in order to analyze the utilization of limits from the current transaction position, in other words, it adds up the attributable amounts of all the transactions that were allocated to the limit on the basis of their characteristic values.

You can use the SAP parallel processing tool for the limit utilization run.

> **Recommendation**
>
> Start the limit utilization run in the background using Schedule Manager [page 437]. In Schedule Manager, you can schedule the country risk run to start first, followed immediately by the relevant limit utilization run. This ensures that the attributable amounts determined in the country risk run are allocated directly to the appropriate limit.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Prototype Functions (Country Risk)* > *Limit Utilization Run (Country Risk)*.

   The *Limit Utilization Run* screen appears.
2. In the *Utilizations* data group, enter the following parameters for the run:
   - Limit area
   - Determination method
   - Base date of utilization run: The business key date for which the evaluation is to take place.
3. In the *Data Source* group box, enter the following parameters, which enable run management to identify a run uniquely.

   > **Note**
   >
   > At present, country risk (CR) is the only caller available. The unique identification for a country risk run enables the attributable amounts determined in the run to be updated to the relevant limit during the limit utilization run.

The *Selection ID of Data Source* uniquely identifies the relevant primary and secondary data sources, which you have defined for the limit utilization run in Customizing for *Bank Analyzer* > *Analytics* > *Limit Manager* > *General Settings for Data Selection* > *Selection in Limit Manager*.

4. Choose *Execute*.

Results

The system displays a log [page 1554] for the limit utilization run. You can display the results in the overview of limit utilizations [page 1578].

Each limit utilization run produces two result records: status 1 and status 2. Directly after the run, the results for status 1 and status 2 are identical. However, if you create a reservation, for example, for the same key date as that of the run, the reservation amount is included in the result with status 2. The reservation does not affect status 1 until the run is carried out again.

Depending on the Customizing settings for the limit type, the system can create limits automatically during a limit utilization run; it does so if there are no suitable limits for the characteristic combination of a transaction. You make the relevant settings in Customizing for *Bank Analyzer* > *Analytics* > *Limit Manager* > *Limit Levels in Limit Manager* > *Edit Limit Types*. If the option for automatic creation of limits has not been activated in Customizing, the utilization run ignores the characteristic combinations concerned.

6.5.8.2 Postprocessing Limit Utilization Runs for Country Risk

Prerequisites

The system already contains at least one limit utilization run with a single transaction you want to change or update to this run.

Context

Using the postprocessing function, you can correct and update any erroneous transactions that already exist in the data pool but were either not analyzed at all, or analyzed incorrectly in the limit utilization run and, hence, not updated correctly. You can also use the postprocessing function to re-post transactions that were analyzed correctly in the limit utilization run but that are to be changed.

When Limit Manager postprocesses limit utilization runs, it does the following:

1. Starts the country risk postprocessing function [page 1346]. The results are stored in the Result Database (RDB).

2. Accesses these results and assigns them to the selected limit utilization run as limit utilizations.

The postprocessing function updates status 1 and status 2 in the result records, which are displayed in the overview of limit utilizations [page 1578].

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Prototype Functions (Country Risk)* > *Postprocess Limit Utilization Runs (Country Risk)*.

   The *Postprocessing of a Limit Utilization Run* screen appears.
2. In the *Identification of the Utilization Run That Is to Be Corrected* data group, enter the selection criteria required to identify uniquely the limit utilization run that you want to correct.
3. In the *Include Changes That Were Made Up To* data group, enter the exact system date/time up to which changes to transaction objects are to be included when the limit utilization is determined.
4. In the *New Transaction Objects That Are to Be Valued* data group, enter the *object category* and *object number* for the individual transaction objects that you want to change.
5. Choose [Test Run] *Test Postprocessing* to test postprocessing.

   Then choose [Update Run] *Postprocess* to carry out postprocessing.

Results

The system displays a log [page 1554] of the postprocessing of the limit utilization run. You can display the corrected results in the overview of limit utilizations [page 1578].

6.5.8.3 Displaying an Overview of Limit Utilizations (Country Risk)

Use

This report enables you to monitor existing limits and their utilizations.

Prerequisites

Limit utilizations have already been calculated in the system in the limit utilization run.

Procedure

Selection

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Analytics* > *Limit Manager* > *Prototype Functions (Country Risk)* > *Display an Overview of Limit Utilizations*.
   You can also call the overview of limit utilizations from the overview of limits. For more information, see Editing Limits [page 1533].
   The *Overview of Limit Utilizations* screen appears.
2. In the *General Selection Criteria* data group, enter the limit area and the base date (required entries). The default value for this date is today's date. You can restrict the amount of data selected further by specifying the limit type and the determination method.
3. In the *Settings for Limits* data group, select the *Unused Limits* checkbox to display all the limits for which there are no utilizations. If you select this checkbox, you have to enter a key date for unused limits. The default value for this date is today's date. If you have defined a display filter in Customizing for *Bank Analyzer* > *Analytics* > *Limit Manager* > *Characteristics in Limit Manager* > *Edit Display Filter*, you can choose this here.
   On the *Direct Characteristics* and *Derived Characteristics* tab pages, you can restrict the selection to particular characteristic values.
4. If required, enter an individual layout (totals records, single records, grouping level 1, grouping level 2), a display currency, or a rounding factor in the *Output Control* data group.
5. If required, enter the following selection criteria in the *Exception Reporting Control* data group:
   - Exceeded limits only
   - Exceeded limits and identical characteristic combinations
6. If required, enter the exact system date and time in the *Selection by Time of Utilization Runs* data group. For example, if four runs were started on the same date, you can identify each run uniquely by specifying the start time.
7. Choose *Execute*.
   The system displays an overview list of all the existing limit utilizations at totals record level that meet your selection criteria.
   The first line contains the limit utilizations in status 1. The second line contains the limit utilizations in status 2. A red traffic light is shown if the limit has been exceeded. If the critical limit utilization has been exceeded, the system displays a yellow traffic light.

> **Note**
>
> The results of the limit utilization runs are stored in status 1. If a limit utilization changes as a result of a reservation, it is displayed in status 2 only. By contrast, limit utilization runs or postprocessing runs are displayed with status 1 and status 2.

Action

You can use various functions in the overview of limit utilizations. In each case, select one of the limit utilizations and choose the appropriate pushbutton.

Action	Use
Refresh	You can refresh the overview of limit utilizations.

Action	Use
Change Limit	You can change an existing limit. You change limits in the same way as you create limits [page 1535]. When you then return to the overview of limit utilizations, the list is automatically updated with the changes you made in the maintenance screen.
Limit Transfer	You can create a limit transfer. For more information, see Creating Limit Transfers [page 1543].
Individual Utilizations	You can display the limit utilizations at single record level. From this display, you can navigate to: • Details about the master data for the transaction or reservation • The detail log for attributable amount determination
Grouping by Limit Characteristics	You can group the totals records on the basis of any limit characteristics, such as a limit for the company code on the basis of the ratings of the business partners assigned to the transactions.
Single Records with Attributable Amounts	You can group the totals records on the basis of single transactions with attributable amounts for each determination procedure.
Business Partner Master Data	You can display the business partner master data.
Limit Type Customizing	You can display the Customizing settings for limit types.

7 Infrastructure (FS-BA-IF)

You can use this component to call functions that provide central services to the various Bank Analyzer components.

Infrastructure contains the following functions:

- Data Load Layer (FS-BA-DL) [page 1581]
- Communication and Worklist Services [page 1593]
- Calculation and Valuation Process Management [page 1622]
- Extraction and Reporting Services [page 1637]
- Outbound Connections Using Third-Party Web Services [page 1715]
- Correction Services [page 1719]
- General Scenario Management [page 1731]
- Settings for XI Services [page 1735]
- Portfolios [page 1740]
- Data Aging in Bank Analyzer [page 1742]

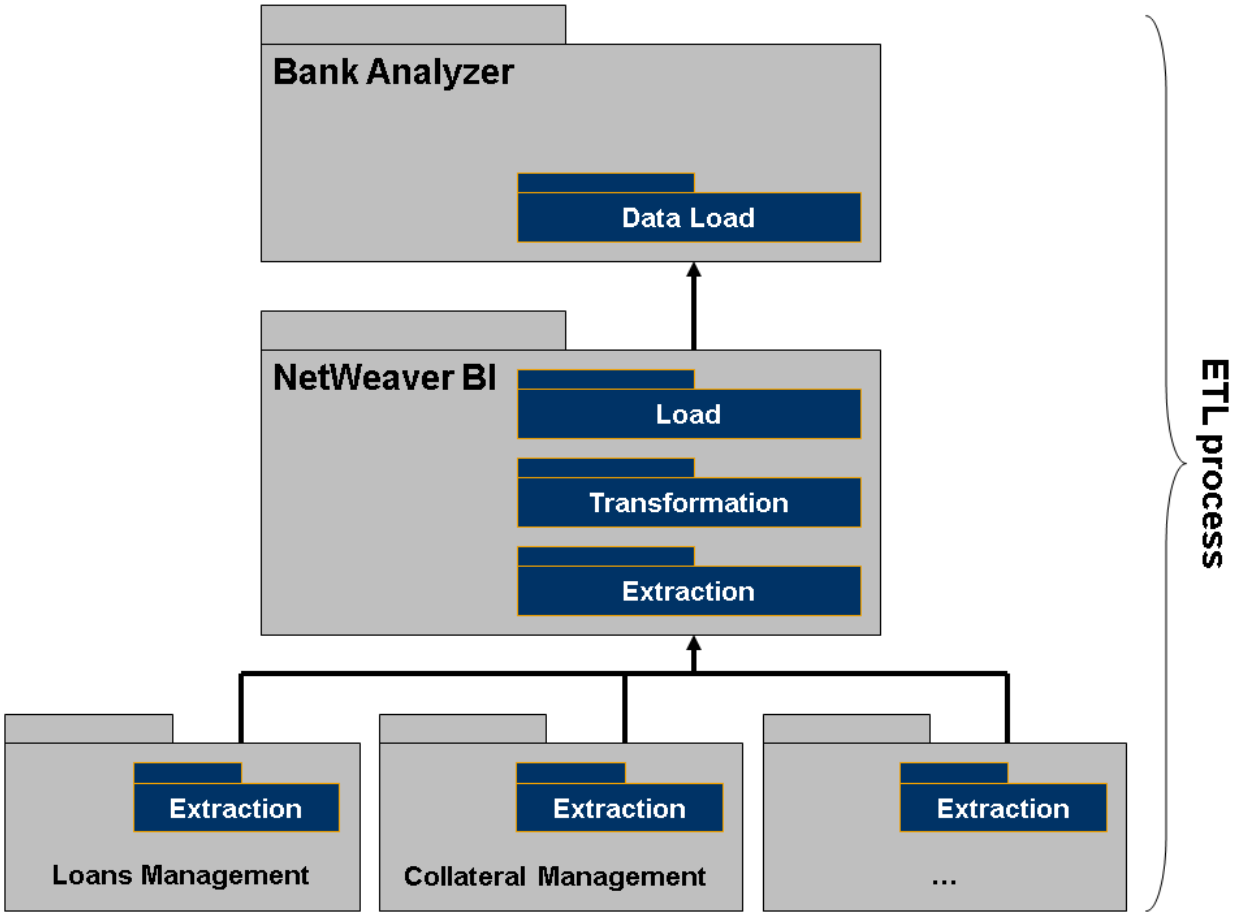
7.1 Data Load Layer (FS-BA-DL)

Use

This component contains the functions for importing source data and results data from SAP NetWeaver Business Intelligence (BI) to the specific interfaces in the Source Data Layer (SDL) or Results Data Layer (RDL) in Bank Analyzer. This is part of the common extraction, transformation and loading process (ETL process) that you can use to transfer data from your own source systems to Bank Analyzer.

Integration

The following diagram shows the components that are part of the ETL process:



1. Extraction

The system extracts the data from the operative systems (full load or delta load) and saves the extracted data in SAP NetWeaver BI. The data is stored in DataStore objects, which have the same structure as the data from the feeder system.

2. Transformation

In SAP NetWeaver BI, the system transforms the extracted operational data into an analytical format, and saves this as the result of the **transformation process**. The analytical format is largely the same as the format used in the inbound interfaces for the Source Data Layer and Results Data Layer.

3. Loading

The system loads the transformation results from SAP NetWeaver BI as InfoProviders into Bank Analyzer.

Features

The load process

- The Data Load Layer connects the transformed data within SAP NetWeaver BI and the storage locations in Bank Analyzer, and reads the data from the InfoProviders in SAP NetWeaver BI. It calls the relevant interfaces in the Source Data Layer and Results Data Layer.
- Since the volume of data may be large, the data load process can be run as a parallel job [page 1592].
- Custom key figures and characteristics can be transformed flexibly during the data load process if appropriate Customizing settings are made.

Process control

- Process control is part of the Data Load Layer and is also integrated in the SAP NetWeaver BI technology. This ensures that the complete ETL process is subject to a standard process control and monitoring.
- The function is integrated into BI technology, which contains the new process chain category, FS Data Load Function, which can be used in the definition of a BI process chain. The process is scheduled and monitored in BI.
- The status of the process is written back to BI.

Tracking of changes

- Each object that was changed during the transformation process in BI is included in the loading process. The changes are handled as change pointers in the Change Notification Service (CNS). This tool collects all the changes made to an object (in this case the Bank Analyzer primary object) in order to make the all the changes at once.
  The change indicators, which are created in BI and stored in Bank Analyzer, are the starting point for the loading process. The loading process updates in Bank Analyzer all the objects that were changed in NetWeaver BI (the update BAPIs are called for the SDL objects, or the APIs are called for RDL data).
  Package Change Pointers: For the DLL export object types FT_S and BT_H, you can use change pointers for packages instead of change pointers for individual objects. To do so, set the *Package Change Pointers* indicator in the DLL process variant. The DataStore object from which the package change pointers are to be written must be write-optimized or it must be an (advanced) DataStore object and *Activate Data* must be deactivated in the modeling properties.
- A log is created of all the primary objects that were changed.

Constraints

- The Data Load Layer does not contain data checks The system sends data that has been transformed and mapped directly to the inbound interface of the Bank Analyzer system.
- Each load process can supply the last version of an object only. It is not possible to process more than one version for each business day.

7.1.1 Data Load Process

Use

This process enables the import of source data and results data from SAP Business Information Warehouse (BW) to the specific interfaces in the Source Data Layer (SDL) or Results Data Layer (RDL) in Bank Analyzer.

Prerequisites

You have a BW system in which you can store transformation results in InfoProviders (DataStore objects). You define the relevant system landscape in Customizing by choosing *Bank Analyzer* > *Basic Settings* > *Global Settings* > *Define System Landscape for Bank Analyzer* >.

> **Note**
>
> If you store the transformation results in a tool BW that runs on the same system as the Bank Analyzer system, you can leave the *Staging BI* field (BW system from which the Data Load Layer reads) blank in the system settings.
>
> If you store transformation results in a different system to the Bank Analyzer system, then you have to make an entry in the *Staging BI* field.

Process

Process Step/Component	What do I have to take into account for this process step?
Transformation/SAP NetWeaver BW	You have to include a function module that is dependent upon an export object category [page 1588] as an end routine in the transformation rules. This ensures that the system generates a change pointer with the status *Created* (C) for the transformed objects in BW using the Change Notification Service and stores it in the Bank Analyzer tables. To avoid multiple change pointers being generated for each business object, we recommend that you create exactly one end routine for each export object category. Generating multiple change pointers per business object only makes sense if data for an object is delivered from multiple source systems at different times, and cannot be delivered from one source system.

Process Step/Component	What do I have to take into account for this process step?
Data Load Process/Bank Analyzer: The system gets object data, including the relevant change pointers that were generated during the defined process chain, for each client from the BW.	You define a process chain that contains the DLL process category *FS Data Load Function* by choosing *Bank Analyzer* > *Infrastructure* > *Data Load* > *Data Load Processes* > *Define Loading Processes in BI*. For the variant, you store the export object category, the relevant change pointers with their status, and the data that is dependent on the export object category [page 1588] (for example, template and template category). To improve performance, pay attention to the sequence in which the values of the key fields are written to the change pointers when using object categories with a generic key [page 1587]. The system performance is also better the larger the packages are. If you use a package size of over 1000, the Bank Analyzer system uses the configured size only to read data into the BW system. As soon as the BW data has been read, the Data Load Layer divides the packages into subpackages each with 1000 objects for calling the interfaces in the SDL or RDL in order to lighten the load on the lock server. The average number of active SAP locks is then lower than if the package size is over 1000.

Process Step/Component	What do I have to take into account for this process step?
Mapping/Bank Analyzer The system reads the data and transfers it to the structures of the inbound interfaces of Bank Analyzer.	• You define mapping between the field names in Bank Analyzer and the corresponding InfoObjects in BW, and also between parameters of the inbound interface in Bank Analyzer and InfoProviders in BW, in Customizing under *Bank Analyzer* > *Infrastructure* > *Data Load* > *Default Mapping*. • You define object-specific mapping of field names and parameters in Customizing under *Bank Analyzer* > *Infrastructure* > *Data Load* > *Special Mapping for Parameters*. • When the system maps **key figures**, it uses the assignments of InfoObjects to key figures that you have defined in Customizing for Bank Analyzer under *Basic Settings* > *Settings for Metadata* > *Key Figures* > *Key Figure Monitor*. The system maps key figures belonging to the *quantity* and *amount* categories only if the unit has been specified. The system maps key figures of the *percentage* category only if the percentage is not zero. • When the system maps **key figure categories**, it uses the assignments of key figure categories to key figures that you have defined in Customizing for Bank Analyzer under *Source Data Layer* > *Primary Objects* > *Master Data* > *Settings for Attributes* > *Key Figures* > *Edit Key Figures Categories*, and then carries out the mapping in the same way as for the key figures. • When the system maps **characteristics**, it uses the assignments of InfoObjects to characteristics that you have defined in Customizing for Bank Analyzer under *Basic Settings* > *Settings for Metadata* > *Characteristics* > *Characteristic Monitor*. The system maps all defining characteristics. For descriptive characteristics, the system does not map the empty ones.
Results/Bank Analyzer	The system saves all data that it has loaded in the Bank Analyzer system.

Result

The load process is complete. You can display the data that has been loaded.

You can also display the results of the load process for single objects in the process chain display by choosing *Goto* > *Log View*.

The system changes the status of the change pointers from C (created) to P (processed), or E (error). To display an overview of all change pointers that are in the current client, on the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Data Load Layer* > *Change Pointer* > *Display Change Pointers*. To help improve performance, you can delete any entries that you no longer need (those with status P, for example) by choosing *Bank Analyzer* > *Infrastructure* > *Data Load Layer* > *Change Pointer* > *Delete Change Pointers*.

7.1.1.1 Performance Improvement for Object Categories with a Generic Key

When you use object categories that have a generic key, the values of the key fields have to be written to the change pointers in an appropriate sequence. An inappropriate sequence can cause considerable problems with the main memory.

You have defined characteristics A, B, C, D, and E as keys for an object category. There are 100, 000 objects for this object category. All 100,000 objects have the same values in key fields (defining characteristics) A, B, C, and D, but in key field E, the values differ.

In the end routine, the generic key fields are transferred to a function module in the form of a table containing the fields *Characteristic Name* and *Characteristic Value* for the creation of change pointers. This can be done in sequence A, B, C, D, E, as in the following table:

Row	Name of Characteristic	Characteristic Value
1	A	11111
2	B	2222
3	C	237467898
4	D	23467342
5	E	2347283

Exactly this sequence causes problems in the main memory because field E is the only field that separates the 100,000 objects from each other, and field E is in the last line of this table. To prevent problems with the main memory when reading and sorting change pointers, the separating field has to be moved further up in the table.

The problems with the main memory are avoided if the key fields are written to the change pointers in sequence E, A, B, C, D, instead of in sequence A, B, C, D, E. Here, separating field E is in the first line, as shown in the following table:

Row	Name of Characteristic	Characteristic Value
1	E	2347283
2	A	11111
3	B	2222

4	C	237467898
5	D	23467342

7.1.1.2 Advanced Error Handling

7.1.2 Export Object Categories

Definition

Export object categories define which data you load into the Bank Analyzer system by means of the Data Load Layer.

Structure

The system currently supports the following export object categories:

Export Object Category	Function Modules to Generate Change Pointers	Selection Criteria	Key Fields
ACC: Account	/BA1/ L3_API_ACC_CNS_INSERT	Account category, template, template version	Account type, external account number, scenario, date
BABP: Business partner for Bank Analyzer	/BA1/ L3_API_BABP_CNS_INSERT_M	None	Business partner number, date
BT: Business transactions	/BA1/ L3_API_BT_CNS_INSERT, /BA1/ L3_API_BT_CNS_INSERT_M	Business transaction class, source system	Business transaction ID, source system, scenario, date

Export Object Category	Function Modules to Generate Change Pointers	Selection Criteria	Key Fields
BT_H: Business transactions	/BA1/L3_API_BT_H_CNS_INSERT_M /BA1/L3_API_BT_H_CNS_INSERT_P (for package change pointers)	Source system	Business transaction ID, source system, scenario, date
BT_S: Business transactions	/BA1/L3_API_BT_S_CNS_INSERT, /BA1/L3_API_BT_S_CNS_INSERT_M	Source system	Business transaction ID, source system, scenario, date
BTR: Reversal business transactions	/BA1/L3_API_BTR_CNS_INSERT, /BA1/L3_API_BTR_CNS_INSERT_M	Source system	Business transaction ID, source system
CKDR: Cancel gen. key date results	/BA1/L3_API_CKDR_CNS_INSERT	Results data area, result type	Key fields of the result type used in the RDL
CTRY: Country	/BA1/L3_API_COUNTRY_CNS_INSERT	Template, template version	External key, scenario, date
FI: Financial instrument	/BA1/L3_API_FI_CNS_INSERT, /BA1/L3_API_FI_CNS_INSERT_M	Template, template version	External contract key, scenario, date
FT: Financial transaction	/BA1/L3_API_FT_CNS_INSERT, /BA1/L3_API_FT_CNS_INSERT_M	Template, template version	External contract key, scenario, date
FT_S: Financial transaction	/BA1/L3_API_FT_S_CNS_INSERT_M /BA1/L3_API_FT_S_CNS_INSERT_P (for package change pointers)	Source system	External contract key, scenario, date
FX: Exchange Rates	/BA1/L3_API_FX_CNS_INSERT	Market data area, exchange rate type	Market data area, exchange rate type, from currency, to currency, conversion type, date

Export Object Category	Function Modules to Generate Change Pointers	Selection Criteria	Key Fields
FXV: Volatilities for exchange rates	/BA1/ L3_API_FXV_CNS_INSERT	Market data area, volatility type	Market data area, volatility type, from currency, to currency, external key option, date
GMD: Generic market data	/BA1/ L3_API_GMD_CNS_INSERT, /BA1/ L3_API_GMD_CNS_INSERT_M	Generic market data class, generic market data area	Generic market data class, generic market data area, term, unit of term, scenario, date
HDB: Historical database	/BA1/ L3_API_HDBE_CNS_INSERT, /BA1/ L3_API_HDBE_CNS_INSERT_M	Name of the memory area for external systems in the Historical Database	Name of the memory area for external systems in the Historical Database, table for characteristics
IMDB: In-house model of the Historical Database	/BA1/ L3_API_HDBI_CNS_INSERT	Name of the in-house model in the HDB, version of the in-house model in the HDB	Name of the in-house model in the HDB, version of the in-house model in the HDB, scenario, table for characteristics
HREL: Hedging relationship	/BA1/ L3_API_HREL_CNS_INSERT	Template, template version	External key, date
IRR: Interest rates	/BA1/ L3_API_IRR_CNS_INSERT	Market data area	Market data area, reference interest rate, date
IRS: Interest rate spread	/BA1/ L3_API_IRS_CNS_INSERT	Market data area, spread type	Market data area, spread type, reference interest rate, date
IRV: Volatilities for interest rates	/BA1/ L3_API_IRV_CNS_INSERT	Market data area, volatility type	Market data area, volatility type, reference interest rate, external key option, date
ISD: Imported subledger documents	/BA1/ L3_API_FISI_CNS_INSERT	Results data area, result type	Sender ID, sender key, date
PBA: Position	/BA1/ L3_API_PBA_CNS_INSERT, /BA1/ L3_API_PBA_CNS_INSERT_M	Position class, account category	Position class, account category, account, scenario, exchange rate type, external key, parent, child, node number, date

Export Object Category	Function Modules to Generate Change Pointers	Selection Criteria	Key Fields
PBAMD: Position master data	/BA1/L3_API_PBAMD_CNS_INSERT	Position class, account category	External key, date
PHAS: Physical asset	/BA1/L3_API_PHAS_CNS_INSERT, /BA1/L3_API_PHAS_CNS_INSERT_M	Template, template version	External key, scenario, date
RDATB: Results data (Smart AFI)	/BA1/L3_API_RDATB_CNS_INSERT_M, /BA1/L3_API_RDATB_CNS_INSERT_P (for package change pointers)	Results data area, result type, result view	Key fields of the result type used in the RDL
RSD: Subledger reversal document	/BA1/L3_API_FISR_CNS_INSERT	Results data area, result type	Sender ID, sender key, date
SCBA: Standard condition BA	/BA1/L3_API_SCBA_CNS_INSERT	Condition group type	Condition group, condition group type, date
SEC: Prices/rates for financial instruments	/BA1/L3_API_SEC_CNS_INSERT	Market data area, exchange, exchange rate type	Market data area, exchange, exchange rate type, ID number, accrued interest indicator, date
SEV: Volatilities for financial instruments	/BA1/L3_API_SEV_CNS_INSERT	Market data area, volatility type	Market data area, volatility type, external key underlying, exchange rate type underlying, external key option, exchange rate type option, date
WKDR: Generic key date results	/BA1/L3_API_WKDR_CNS_INSERT, /BA1/L3_API_WKDR_CNS_INSERT_M	Results data area, result type	Key fields of the result type used in the RDL

7.1.3 Parallelization of the Data Load Process

Use

You use this function to process data packages that are independent of one another in different batch processes at the same time during the data load process

Features

You can parallelize the data load process in one of the following ways:

- You can run each individual process step of the process category FS Data Load Function in parallel within a process chain. As soon as the system has generated a list of objects resulting from the selection criteria (using change pointers), the system can read and load data in parallel.
- The system can carry out several process steps of the process category FS Data Load Function (for example, various export object categories) within one process chain in parallel, as long as the loaded data are independent of each other. If the system loads data that references an object that is yet to be loaded, the system issues an error message in the inbound interfaces of Bank Analyzer.

7.1.4 Advanced Error Handling

You can use advanced error handling to simplify the monitoring of errors in DLL when you have large volumes of data.

Procedure

1. 1. Define an error variant to restrict your worklist to specific messages. To do so, on the SAP Easy Access screen for Bank Analyzer, choose *Infrastructure* > *Data Load Layer* > *Data Load Processes* > *Define Error Variants*.
2. In transaction `RSPC` , set the *Advanced Error Handling* indicator for the required process variant. This means that error messages for objects that could not be loaded are not only logged in the application log but also in the transparent table `/BA1/L1_ERR_LOG`. (Once an object has been loaded successfully, the system deletes the corresponding entry from the table again). You then enter the error variant defined in the previous step.

Delete error messages and change pointers

You can delete error messages and change pointers from table /BA1/L1_ERR_LOG. To do so, on the SAP Easy Access screen, choose *Infrastructure* > *Data Load Layer* > *Change Pointers* > *Delete Change Pointers*.

7.2 Communication and Worklist Services

Communication and worklist services include all communication and data access functions from other components to data from the Source Data Layer and the Results Data Layer.

- *Primary data sources* of the *Worklist* and *Single Record* category
- *Secondary data sources*
- Framework for *Unified Object Identifiers (UOID)*

7.2.1 Primary Data Sources

Use

Primary data sources are configurable interfaces that provide flexible access to the uniform data in the Source Data Layer and the Results Data Layer, and to the results data in Bank Analyzer applications.

> **Note**
>
> The dataset read by a primary data source is defined by the values of the selection characteristics.

Features

There are two different primary data source types [page 1593]:

- Primary data source for single record selection
- Primary data source for worklist creation (multiple selection)

The data that is required by the processes is provided by the infrastructure by means of data source categories [page 1594], object transfer structure (OTS) [page 1594], or InfoSet [page 1616]. The Bank Analyzer applications themselves can also provide results data by means of the general primary data source [page 1596]. To do this, the applications provide servers that read the data from its dataset. There is also a server for database tables and for files.

7.2.1.1 Data Source Type

Infrastructure contains the following data source types:

- Single record selection
- Worklist (multiple selection)

The differences between these data source types are detailed in the table below.

	Single Record Selection	Worklist
Use	The characteristics and key figures of only one object (for example, a primary object) are read.	You use the worklist to set up a number of hits.
Data Selection	Data is selected on the basis of one object key.	Data is selected on the basis of user-defined selection criteria.
Data Source Category	• Object transfer structure (OTS) • General primary data source	• InfoSet • General primary data source
Additional Functions		• Data can be returned as a list or using package processing. • You can use the parallelization tool to process packages. • InfoSets and some servers of the general primary data sources support the delta read process.

7.2.1.2 Data Source Category

Infrastructure contains the following data source categories:

- Object transfer structure (OTS) [page 1594]
- General primary data source [page 1596]
- InfoSet [page 1616]
- Generic primary data source [page 1617]

7.2.1.2.1 Object Transfer Structure

Use

Object transfer structures (OTSs) are used for reading characteristics and key figures from the primary objects of the Source Data Layer. They are customer-defined structures. The module editor fills these structures at runtime with the attribute values of the primary objects using the derivation tool.

Integration

Since only one version of a primary object is read in a flat structure, the data source OTS supports only the data source type [page 1593] *Single Record Selection*.

Prerequisites

You have assigned the OTS to process category 05 (*Primary Data Source*). This ensures that the OTS is available as a primary data source in Bank Analyzer. You do this in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Object Transfer Structures (OTS)* > *Assign OTS to Processes*.

Example

You can access position master data [page 152] in primary data sources of category *OTS*. These object transfer structures belong both to the technical component *position master data* and to the component *Position*. The data of the respective other components can then be accessed during the relevant OTS population. You make the relevant settings in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Object Transfer Structure (OTS)* > *Edit OTS for Primary Objects* > *Edit OTS for Position Master Data*.

7.2.1.2.1.1 Data Source for Subobjects

Use

The data source for subobjects of the category *object transfer structure* (OTS) enables primary object categories *Financial Transaction*, *Financial Instrument*, and *Physical Asset* of the master data framework to directly access subobjects from within structured master data objects.

> **Example**
>
> Using an OTS for subobjects, you can exclusively consider individual subobjects of physical assets.

Integration

To use a data source for subobjects, you have to first create an OTS for the primary object category in question. The data source for subobjects contains additional import parameters and characteristics (parent template, child template, and node number) to enable the subobject to be identified.

Activities

You can define an OTS for a complete object or for a subobject in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Object Transfer Structure (OTS)* > *Assign OTS to Processes*.

7.2.1.2.2 General Primary Data Source

Use

You use the general primary data source to read results data and files or database tables from outside of the Tool BI system.

Integration

A general primary data source differs from the primary data sources InfoSet [page 1616] and Object Transfer Structure (OTS) [page 1594] in that it is always based on a data source server, which provides access to the data in the data source. This server provides the data, which is transferred to the requesting client by means of the primary data source.

> **Note**
>
> You can use only the servers provided by SAP for a primary data source. This also means that only the data for which a server has been provided by SAP can be read.

The general primary data source supports the data source types [page 1593]*single record selection* and *worklist (multiple selection)*. You can find information about whether a data source server supports a *worklist* or a *single record*, or both data source types in the system table `/BA1/TF3_GDS_SV`.

Prerequisites

You have installed a server for the Analyzer you want to read and you have defined a general primary data source in Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.

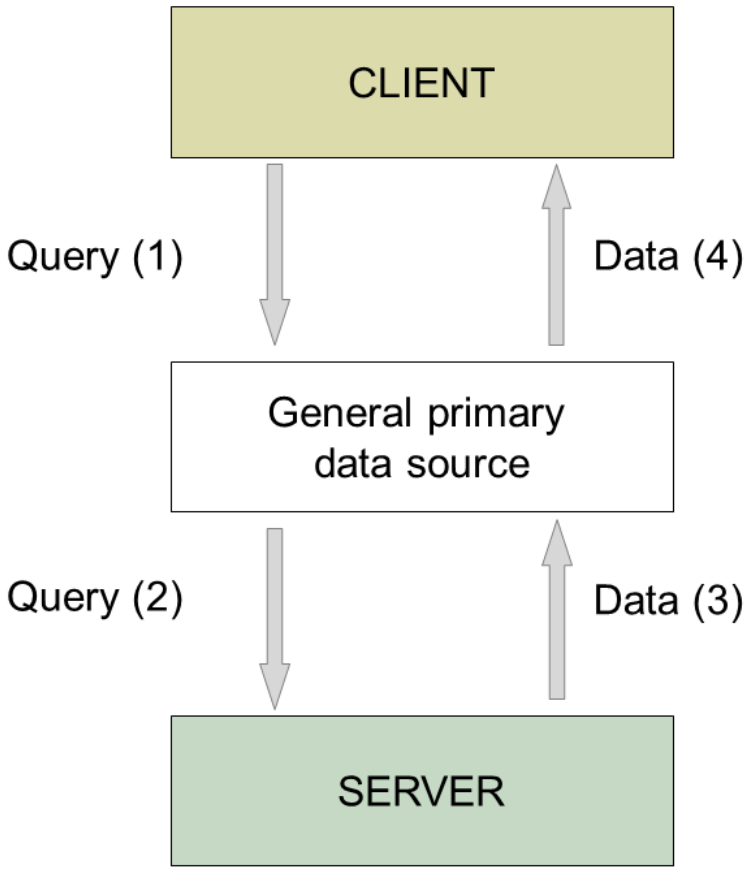
Activities

To edit a general primary data source in Customizing for *Bank Analyzer*, choose *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.

> **Caution**
>
> If you delete or add characteristics or key figures in primary data sources [page 1619], the system might regenerate the transfer structure, and, as a result, errors might occur in applications that are called by the primary data source.

7.2.1.2.2.1 Servers for General Primary Data Sources

A server is assigned to every primary data source. At runtime, the data source servers analyze the selection criteria of the general primary data sources and return the data records that they determine to the calling application by means of the general primary data source.



For an overview of the available servers and their properties (for example, single record selection), see the system table `/BA1/TF3_GDS_SV`.

Additional Information About Specific Servers

> **Note**
>
> There are no applications assigned to the following servers: *File*, *Master Data Hierarchy*, and *Table*.

Smart Accounting for Financial Instruments

For more information, see Servers for Smart Accounting for Financial Instruments [page 1602].

Key Figure Determination in Financial Statement Preparation, Results Data, and Financial Statement Items

This server delivers results data only to the Profit Analyzer. The prerequisite for this is that you have completed the accounting processes.

The server attributes have the following meanings:

- `AO_TYPE`: The balance subobject type for which you want to read data. `AO_TYPE` selects the structure and database that are to be read.
- `VALUE_TYP`: The value type of the financial reporting process for which the data is read. `VALUE_TYP` enables you to differentiate between real and simulated data. You can read key figures of only *one* currency type with *one* primary data source. Otherwise, you will not be able to aggregate.
- `CURR_TYPE`: The currency type of the key figure that you want to read.

Characteristics of the Financial Position

You can use the general data sources of this server to read all of the characteristic values in a financial position ID.

Financial Position ID

The general primary data sources of this server enable you to select the financial position IDs from the list of defining and descriptive characteristics. You can determine the financial position IDs in Customizing for *Bank Analyzer* under *Infrastructure* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.

Financial Position ID Express

> **Note**
>
> This server works in the same way as `BA_FP_MID`; however, it supports parallel reading and is therefore faster than `BA_FP_MID`.

You can use the server attribute *Excluded Financial Position Categories* to exclude one or more financial position categories. The possible financial position categories are fixed values for the domain `/BA1/B1_FP_CAT`. To exclude a financial position category, enter a fixed value, for example `E` (for valuation entity) or `R` (general relationship position, not to be valuated). To exclude several financial position categories, string the fixed values together in a series, for example `ER`.

You can use the server attribute *Consider Posting Date* to restrict the selection of financial positions in relation to the posting date of the financial position (as of SAP Note 1929129).

As of SAP Note 1929129, the delta read process for data records is also supported; in other words, the selection of financial positions that were changed or created in a given time period.

Financial Position Totals

You use the general primary data sources of this server to read the totals of the financial positions for each financial position ID for a particular period.

Business Transactions

The following server attributes are available:

- Business transaction class
- Item class
- Account category
- Template category
- Uses L-selection
- Date field for L-selection
- Selection using multiple BT classes
- Analytical service operation
- Posting date
- Aggregation view

Business Transactions as a Single Record

The following server attributes are available:

- Single record: reversal type
- Single record: type of reversal transformation

Primary Objects of Master Data Framework in Flat Storage:

For more information, see Server for Primary Objects of Master Data Framework [page 1603].

General Methods

- Account pooling
- Distribution of facilities
- Determination of the free line for facilities
- Collateral distribution

For more information, see Server for General Methods [page 1604].

Credit Exposure

For more information, see Server for Credit Exposure [page 1607].

Cash Flow Hedge Proposal

General primary data sources of this server are used for hedge management. They provide information about the hedge proposals that are generated by the user in cash flow hedge analysis reporting [page 1140]. You generate hedge proposals when you add or delete hedge transactions to cash flow hedge relationships, and save this assignment as a proposal for future cash flow hedge relationships. The server takes as attributes the name, version, and key date of the aggregation run [page 1123] that forms the basis of the hedge proposal. The version of the hedge proposal is also one of the attributes of the server because there can be several hedge proposals for every aggregation run.

Enhanced Worklist

You can use the enhanced worklist to create a worklist from data that was read from different InfoSets. The system groups multiple InfoSets in an enhanced worklist that is used by a general primary data source of the category

Worklist. For more information, see Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Index Function* > *Process Extended Worklist*.

Generic Primary Objects

General primary data sources of this server are used to access primary data sources of category object transfer structure (OTS) [page 1594]. You make this assignment in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Object Transfer Structure (OTS)* > *Assign the OTS to the Generic Primary Data Sources*. You access the primary data sources of category *OTS* by means of the *Unified Object Identifier (UOID)*.

> **Note**
>
> You can use the generic primary data sources [page 1617] to determine information about objects with business roles that are assigned to different primary object categories. (Assets that can appear as physical assets, financial transactions, positions, or position master data are an example of this.)

File

General primary data sources that are assigned a *file* server are used to read data from binary files. You can use this primary data source in test scenarios to replace or simulate other data sources that have not yet been completely implemented. The prerequisites for this are that you have created a logical file name in transaction *FILE* and have described the file structure in the *data dictionary*. You make the Customizing settings for the file structure in *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Service* > *Data Sources* > *Primary Data Sources* > *Assign Fields for Tables and Structures to Characteristics*.

> **Note**
>
> If you remove characteristics or key figures from the data source, the range of the results structure is reduced. However, this does not greatly improve performance because the system always imports the complete file.

Fair Value Hedging Relationship

General primary data sources of this server are used to deliver the results of the fair value effectiveness test [page 1039] that runs in batch mode. The system uses these primary data sources as part of the key date valuation in key figure determination in financial statement preparation to read ineffective hedging relationships and their date of ineffectiveness, and then to further process them. The prerequisite is that you have completed a batch run of the fair value effectiveness test, and that you have subsequently released the run via the run management for Hedge Accounting.

Apart from key date, the server also uses the following characteristics for the selection of the effectiveness test: *legal entity* (/BA1/C55LGENT) and *accounting system* (/BA1/C55ACCSY). In addition, the characteristic list of the server includes the characteristics *hedging relationship identifier* (/BA1/C47HEDGE) and *valid from* (/BA1/C47VALFROM). The primary data sources populate both characteristics and return them to the calling program. Characteristic /BA1/C47HEDGE is a granularity characteristic and uniquely identifies the returned data record.

Generic Market Data Classes

The system automatically generates the general primary data sources of the category *Single Record* for this server when you create generic market data classes. You create general primary data sources of the category *Worklist* for this server manually. You make manual changes to existing primary data sources (*Single Record* and *Worklist*) via the assignment to application areas.

Yield Curves, Exchange Rates, and Generic Market Data

For the general primary data sources for the *Yield Curves, Exchange Rates, and Generic Market Data* servers, you can supply key figures. You can also specify an application for the data extraction. You assign exchange rates and yield or spread curves to an application. These exchange rates, yield curves, and spread curves are then made available to other applications by means of the outbound service for data provision.

You define applications in Customizing under *Financial Services* > *Foundation* > *Market Data* > *Create Application*.

Default Data

For more information, see Servers of the Determination of Default [page 1610].

Historical Database:

- Historization processes
- Bank's in-house models
- External systems

For more information, see Servers of the Historical Database [page 1612].

Hedging Relationship Data

For more information, see Reading Data from the Hedge Accounting Workplace [page 1028].

Master Data Hierarchy

The general primary data sources of the *master data hierarchy* server is used to access master data hierarchies [page 337] for characteristics. In addition, the Bank Analyzer applications calculate the values for each external characteristic node from the predecessor node to the top node.

> **Note**
>
> Master data hierarchies of the tool BI [page 16] have only one validity period (key date query). They are additionally managed in the Bank Analyzer core system and given time stamps.

You can access data if you have registered the characteristic in the characteristic repository of Bank Analyzer, and if a master data hierarchy of other characteristic nodes exists in the Tool BI system for this characteristic. You make the Customizing settings for characteristics in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Metadata* > *Characteristics*.

> **Note**
>
> You can create master data hierarchies for local [page 62] and fixed characteristics [page 59] (as well as for custom characteristics [page 25]) if their master data was also replicated in the tool BI system and if master data hierarchies were generated from these characteristics.

Limit Manager

Limit Manager provides the following primary data sources:

- Limits and limit items
- Utilizations and limits on a totals record level
- Utilizations on a single record level

For more information, see Servers for Limit Manager [page 1614].

Table

You use general primary data sources of this server to read data from database tables of the same SAP system. You can use these data sources for test scenarios. To access this data, you first have to register the database tables and make the relevant settings in Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Assign Fields for Tables and Structures to Characteristics*.

Note the following constraints for this server:

- You cannot create a data source to read single records. However, this function is provided with the derivation tool.
- Since there is no aggregation, you always have to assign characteristics to the key fields of the table. These characteristics must then be integrated into the primary data source as granularity characteristics.
- You cannot evaluate the change history of the table data. The time stamp is therefore not used.

7.2.1.2.2.1.1 Servers for Smart Accounting for Financial Instruments

Use

For general primary data sources, you can use the following servers to read Smart Accounting for Financial Instruments (Smart AFI) data:

Server	Server Description
GL_CONN	Smart AFI: General Ledger Connection
REGISTER	Smart AFI: Register Process
RTA_PC	Smart AFI: Process Controller Objects

> **Note**
>
> These servers are only for SAP-internal use in Smart AFI. Changes must not be made to the generated primary data sources that are assigned one of these servers.

Features

These servers support the unpackaged and (for parallelization) the packaged selection of worklists.

In Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources* , you can display the attributes, characteristics and key figures of generated primary date sources.

GL_CONN

The `/BA1/RBR_GL_SEND` (Send General Ledger Documents) process uses the primary data source of this server that the system generates. This data source reads the general ledger documents from G/L document storage in the Results Data Layer (RDL).

REGISTER

The `/BA1/RBR_REGISTER` (Register) process uses the generated primary data sources of this server that read the listener entries from the tables `/BA1/BR_REG_BT`, `/BA1/BR_REG_MD` and `/BA1/BR_REG_BPC`.

The characteristics of the primary data sources depend on the attribute.

RTA_PC

Processes including `/BA1/RBR_DAY_END` (End-of-Day Processing/ Contract), `/BA1/RBR_PERIOD_END` (Period-End Processing/ Contract), `/BA1/RBR_PERIOD_START` (Period-Opening Processing/ Contract) and `/BA1/RBR_BCF` (Balance Carryforward/ Contract) use the generated primary data sources of this server. These data sources read the process controller entries from the table `/BA1/TFC_PCDSTAT`.

The characteristics of the primary data sources depend on the attribute.

More Information

for Smart Accounting for Financial Instruments

7.2.1.2.2.1.2 Server for Primary Objects of Master Data Framework

Use

The server `SDLCONFLAT` (contract server for primary objects (MDF)) is provided for the primary objects of the master data framework that are persisted in a flat database table.

This server supports the following functions:

- Selection of worklists
- Reading of data by package
- Parallel package reading
- Selection by primary object status

Features

Server-Dependent Attributes

Server-dependent attributes identify where data is stored:

- Version selection
  - Last version available on key date
  - Version dependent on system time
- Template category

Characteristics

The system maps the characteristics to fields. This mapping is stored directly in the coding and cannot be changed.

7.2.1.2.2.1.3 Servers for General Methods

Use

The following servers are available for general methods:

General Methods	Server	Data in the RDB Client
Account Pooling	Account Pooling	71
Distribution of Facilities	Distribution of Facilities	72
Determination of the Free Line for Facilities	Determination of the Free Line for Facilities	73
Collateral Distribution	Collateral Distribution	74

> **Note**
>
> For more information about the general method "Determination of Default", see Server for the Determination of Default [page 1610].

These servers support the following functions:

- Selection of single records
- Selection of worklists
  - Unpackaged

Features

> **Note**
>
> The servers for general methods do not support the correction server.

Attributes of the Servers for General Methods

The servers for general methods have the following attributes:

Server-Dependent Attributes

Server-dependent attributes identify where data is stored:

- RDB: semantic run name
  - Selection of the external run name
- RDB reference object identification
  - Determines the name of the RDB reference object identification of the respective server. The reference object category identifies the stage in the run of the general methods that is to be read (**`VSID`** for collateral distribution, **`FGID`** for the other servers).
- RDB process step
  - At the moment only **`FINAL`** is available for the process step.

General Attributes

- Read depth
  You use the read depth to determine the status that the system considers:
  - Read depth with the value "1":
    If you enter the value "1" for the read depth, the system searches for runs that have the run status *Cmtd* (*Run Completed*) only.
  - Read depth with the value "2":
    If you enter the value "2" for the read depth, the system searches for runs that have the run status *Cmtd* (*Run Completed*) first. If the system does not find any runs with the status *Cmtd*, it searches for runs that have the status *End* (*Run Completed*).
- Read history
  - Reading of the history using value "0":
    If you have entered the value "0" for this attribute, the system considers only runs whose key date is identical to the selection date. In this case, the system searches for the run based on the setting in the *Read Depth* attribute.
  - Reading of the history using value "0":
    If you have entered the value "1" for this attribute, the system also considers runs whose key date is before the selection date. The determination of the run is then based on the setting of the *Read Strategy* attribute. You control how far back in time (from the selection date) the system reads by means of the *Number of Days* attribute.
- Read strategy
  The prerequisite for this attribute is that you have entered the value "1" for the *Read History* attribute.
  The read strategy determines the priorities that the *Read Depth* and *Read History* attributes use when determining the run. In other words,
  - Does the system search for the most up-to-date run based on the read depth? (read depth is prioritized)
  - Does the system search for the highest status? (reading of the history is prioritized)

- The number of days to be read for the key date
  The prerequisite for this attribute is that you have entered the value "1" for the *Read History* attribute. In this case, the system uses the number of days that have to be read to determine – based on the selection date – the key date for which runs are considered. If you leave this attribute blank, the system considers all runs. For reasons of system performance, we recommend that you restrict the number of days you enter.

  > **Example**
  >
  > You have entered the value "2" for the *Read Depth* attribute, and the value "1" for the *Read History* attribute. In addition, you have entered October 10, 2004 as the selection date, and have entered the value "1" for the *Number of Days to Be Read* attribute.
  >
  > If the *Read Strategy* attribute contains the value "0", the system first searches on October 10, 2004 for a run with the status *Cmtd*, and then, if it does not find a run with the status *Cmtd*, it searches on the same day for a run with the status *End*. If the system still does not find a run, it repeats the same procedure for the key date October 9, 2004.
  >
  > If the *Read Strategy* attribute contains the value "1", the system first searches on October 10, 2004 for a run with the status *Cmtd*. If the system does not find a run with the status *Cmtd* for this key date, it searches for a run with the status *Cmtd* for October 9, 2004. If the system still does not find a run, it repeats the same procedure for the status *End*.

Characteristics and Key Figures

You have already defined the characteristics available for the primary data source. You defined these for the RDB packages upon which the primary data sources are based.

Explanation of the Characteristics

- Granularity indicator
  The settings for the granularity indicator are not evaluated for the server in question. When you use single record accesses, check that all of the characteristics that uniquely define the record are transferred. If this is not the case, the system issues an error message.
- Deliverable characteristics
  The settings for the deliverable characteristics are not evaluated for the server in question. This means that the system always returns all of the data saved.
- Selection characteristic
  When you use the primary data source in the module editor, then the settings you make for the selection characteristics govern whether these characteristics are required or optional input fields.

Explanation of the Key Figures

You have already defined the key figures available for the primary data source. You defined these for the RDB packages upon which the primary data sources are based.

- Selection unit
  This indicator is not evaluated for the server in question.

Support of Stress Tests

The following scenario categories are provided for the support of stress tests:

Component	Scenario Category	Description
Account Pooling	COMP	Single record
Distribution of Facilities	RLV	Single Record
Determination of the Free Line for Facilities	FLE	Single Record
Collateral Distribution	SV	Single Record
All Components	WLR	Worklist

7.2.1.2.2.1.4 Server for Credit Exposure

Use

The system reads the data from the Credit Exposure runs from RDB client 63.

This Credit Exposure server supports the following functions:

- Selection of single records
- Selection of worklists
  - Unpackaged
  - Packaged (for parallel processing)

Features

> **Note**
>
> The server for Credit Exposure does not support the correction server.

Attributes of the Server for Credit Exposure

The server for the Credit Exposure has the following attributes:

Server-Dependent Attributes

Server-dependent attributes identify where data is stored:

- Credit exposure run type
  In Credit Exposure, you use the run type to select the run that has the relevant external run ID.
- RDB reference object indicator
  You use the reference object indicator to flag the calculation levels (level 1, level 2, or level 3) that are to be read for a Credit Exposure run.

General Attributes

- Read depth
  You use the read depth to determine the status that the system considers:
  - Read depth with the value "1":
    If you enter the value "1" for the read depth, the system searches for runs that have the run status *Cmtd* (*Run Completed*) only.
  - Read depth with the value "2":
    If you enter the value "2" for the read depth, the system searches for runs that have the run status *Cmtd* (*Run Completed*) first. If the system does not find any runs with the status *Cmtd*, it searches for runs that have the status *End* (*Run Completed*).
- Read history
  - Reading of the history using value "0":
    If you have entered the value "0" for this attribute, the system considers only runs whose key date is identical to the selection date. In this case, the system searches for the run based on the setting in the *Read Depth* attribute.
  - Reading of the history using value "1":
    If you have entered the value "1" for this attribute, the system also considers runs whose key date is before the selection date. The determination of the run is then based on the setting of the *Read Strategy* attribute. You control how far back in time (from the selection date) the system reads by means of the *Number of Days* attribute.
- Read strategy
  The prerequisite for this attribute is that you have entered the value "1" for the *Read History* attribute.
  The read strategy determines the priorities that the *Read Depth* and *Read History* attributes use when determining the run.
  - Read strategy with the value "0":
    If you enter the value "0" for the read strategy, the system searches for the latest run depending on the read depth (read depth has priority).
  - Read strategy with the value "1":
    If you enter the value "1" for the read strategy, the system searches for the run with the highest status (reading of the history has priority).
- Number of days
  The prerequisite for this attribute is that you have entered the value "1" for the *Read History* attribute.
  This value determines the number of days to be read, calculated from the selection date. The system considers runs up until this key date in the past.

> **Note**
>
> If you enter the value "0" or no value for the *Number of Days* attribute, the system considers all available runs. In this case, the system searches the runs using the attribute values for *Read Strategy* and *Read Depth* for the selection date and for dates in the past.
>
> For reasons of system performance, we recommend that you restrict the number of days you enter.

> **Note**
>
> You have entered the value "2" for the *Read Depth* attribute, and the value "1" for the *Read History* attribute. In addition, you have entered October 10, 2004 as the selection date, and have entered the value "1" for the *Number of Days to Be Read* attribute.

If the *Read Strategy* attribute contains the value "0", the system first searches on October 10, 2004 for a run with the status *Cmtd*, and then, if it does not find a run with the status *Cmtd*, it searches on the same day for a run with the status *End*. If the system still does not find a run, it repeats the same procedure for the key date October 9, 2004.

If the *Read Strategy* attribute contains the value "1", the system first searches on October 10, 2004 for a run with the status *Cmtd*. If the system does not find a run with the status *Cmtd* for this key date, it searches for a run with the status *Cmtd* for October 09, 2004. If the system still does not find a run, it repeats the same procedure for the status *End*.

Characteristics and Key Figures

You have already defined the characteristics available for the primary data source. You defined these for the RDB packages upon which the primary data sources are based.

Explanation of the Characteristics

- Granularity indicator
  The settings for the granularity indicator are not evaluated for the server in question. When you use single record accesses, check that all of the characteristics that uniquely define the record are transferred. If this is not the case, the system issues an error message.
- Deliverable characteristics
  The settings for the deliverable characteristics are not evaluated for the server in question. This means that the system always returns all of the data saved.
- Selection characteristic
  When you use the primary data source in the module editor, then the settings you make for the selection characteristics govern whether these characteristics are required or optional input fields.

Explanation of the Key Figures

You have already defined the key figures available for the primary data source. You defined these for the RDB packages upon which the primary data sources are based.

- Selection unit
  This indicator is not evaluated for the server in question.

Support of Stress Tests

The following scenario categories are provided for the support of stress tests:

Component	Scenario Category	Description
Credit Exposure	CRE	Single record
All Components	WLR	Worklist

7.2.1.2.2.1.5 Server for the Determination of Default

Use

The *Determination of Default* server is used for the primary data sources that are to read the data for the determination of default.

This server supports the following functions:

- Selection of single records
- Selection of worklists
  - Unpackaged
  - Packaged (for parallel processing)

Features

Attributes of the Server for the Determination of Default

The server for the determination of default has the following attributes:

Server-Dependent Attributes

The server-dependent attributes uniquely identify where data is stored for the determination of default:

- Persistent data storage:
  - Default Data
  - Scenario data
  - No entry: If a scenario ID is transferred, the server reads the scenario data. If no scenario ID is transferred, or no scenario data is found, the server reads default data.

General Attributes

- Number of past days for which data is to be read:
  - The value 0 days means that only those runs are taken into account whose business record date matches the key date of the selection.
  - A positive number *n* means that those runs are taken into account whose business date matches the key date of the selection, plus those with business dates of *n* days before the date of the selection.
  - If no value is entered, all the runs are taken into account whose business record dates match that of the key date of the selection, or are before the key date of the selection.
- Read one run only:
  - *Yes*: The program reads only the data from the last run that was successfully completed. Any runs that were terminated, or are obsolete, are ignored.
  - No: No criteria are applied to the selection of the runs.
- Read mode
  - *D:* Read data from the database only (the program ignores archived data, data that has been deleted from the database, and archived data that was reloaded)
  - *A:* Read data from the archive only (all data that is not read in read mode D)
  - *I*: Read data from the database and from the archive (all data; the sets of data read in read modes D and A)

- No entry: The program reads data from the database only, including any archived data that was reloaded

- Read all versions:
  - *Yes*: The program reads all data records that meet the selection conditions and the other attribute values. Versioning is not activated here.
  - *No*: For each value of the granularity characteristics, the system reads only the most current data record based on the fixed characteristics Base Date (/BA1/C65DATRUN) and Technical Time (/BA1/C65JCTIME). Versioning is activated here.
- Delta read with key date
  - *Yes*: The system uses the key date in the report, in addition to both time stamps, as a selection characteristic. The key date cannot then be used to restrict the worklist.
  - *No*: The key date in the report is not used to restrict the worklist.
  - *Field Catalog for Reading Archive*: Enter the name of the field catalog if the data source reads from the archive. For performance reasons, we recommend that you support the read access with a suitable field catalog. For more information, see the documentation about data archiving.

Characteristics and Key Figures

Explanation of the Characteristics

You have already defined the characteristics available for the primary data source. You defined these for the default layers upon which the primary data sources are based.

You can also use a scenario ID (/BA1/C92SCEN). However, you cannot use this ID as a selection characteristic for worklists. You define the scenario ID when you start runs in the report, or you can assign an ID in the module editor (secondary data source). When doing so, make sure that the ID is for a composite scenario. When the system reads worklists for stress data, the worklist has to be registered as a component of the ID of the composite scenario.

The base date (/BA1/C65DATRUN) is the date of the default determination run. Although this date exists as an output characteristic for primary data sources of the category single record, you cannot use it directly as a selection characteristic. The business record date (key date) of a default determination run is automatically compared with the base date.

> ⚠ Caution
>
> In the case of a delta read, that is if the system uses two time stamps to read the worklist, the server does **not** automatically consider the base date. The server considers the base date only if you are actively using the base date as a selection characteristic and if you have set the *Delta Read with Key Date* to *No*. See Determination of Default [page 411] and Key Date Query and Delta Query [page 267].

Granularity indicator:

The default layers do not recognize business keys. Instead, business keys are defined by means of the granularity characteristics of the primary data sources, and can be used for different types of read-only access.

If versioning is switched on, the setting stored for the granularity indicator in the individual characteristics governs which data the program returns. If the primary data source key, which is defined by the granularity indicator, has a value that does not match the value of the business key, the program returns either too many or too few data records.

Explanation of the Key Figures

You have already defined the key figures available for the primary data source. You defined these for the default layers upon which the primary data sources are based. The system does not support selections using the unit of a key figure. This means that the *Selection Unit* key figure must remain blank.

7.2.1.2.2.1.6 Servers of the Historical Database

Use

The following servers are provided for the primary data sources that are to read the data for the Historical Database:

- Data layers
- Bank's in-house models
- External systems

These servers support the following functions:

- Selection of single records
- Selection of worklists
  - Unpackaged
  - Packaged (for parallel processing)

Features

Attributes of the Servers of the Historical Database

The servers of the Historical Database have the following attributes:

Server-Dependent Attributes

Server-dependent attributes identify where data is stored:

- Data layers:
  ID of data layer
- In-house models:
  - HDB areas:
    - Review data storage
    - Historical data storage
    - Scenario data storage
    - No entry: If a scenario ID is transferred, the server reads the scenario area. If no scenario ID is transferred, or no scenario data is found, the server reads data from the review data storage.
  - ID of Model
  - Model version
- External systems:
  - ID of the external system

General Attributes

- Number of past days for which data is to be read:
  - The value 0 days means that only those runs are taken into account whose business record date matches the key date of the selection.
  - A positive number *n* means that those runs are taken into account whose business date matches the key date of the selection, plus those with business dates of *n* days before the date of the selection.
  - If no value is entered, all the runs are taken into account whose business record dates match that of the key date of the selection, or are before the key date of the selection.
- Read one run only:
  - *Yes*: The program reads only the data from the last run that was successfully completed. Any runs that were terminated, or are obsolete, are ignored.
  - *No:* No criteria are applied to the selection of the runs.
- Read mode
  - *D:* Read data from the database only (the program ignores archived data, data that has been deleted from the database, and archived data that was reloaded)
  - *A:* Read data from the archive only (all data that is not read in read mode *D*)
  - *I:* Read data from the database and from the archive (all data; the sets of data read in read modes *D* and *A*)
  - No entry: The program reads data from the database only, including any archived data that was reloaded
- Read all versions:
  - *Yes*: The program reads all data records that meet the selection conditions and the other attribute values. Versioning is not activated here.
  - *No:* For each value of the granularity characteristics, the system reads only the most current data record based on the fixed characteristics *Base Date* (/BA1/C65DATRUN) and *Technical Time* (/BA1/C65JCTIME). Versioning is activated here.
- Delta read with key date (valid for all servers of the HDB except the external systems)
  - *Yes*: The system uses the key date in the report, in addition to both time stamps, as a selection characteristic. The key date cannot then be used to restrict the worklist.
  - *No*: The key date in the report is not used to restrict the worklist.
  - *Field Catalog for Reading Archive*: Enter the name of the field catalog if the data source reads from the archive. For performance reasons, we recommend that you support the read access with a suitable field catalog. For more information, see .

Characteristics and Key Figures

Explanation of the Characteristics

You have already defined the characteristics available for the primary data source. You defined these for the data layers, models, and external systems upon which the primary data sources are based.

The system contains a scenario ID (/BA1/C92SCEN) for models. However, you cannot use this ID as a selection characteristic for worklists. You define the scenario ID when you start runs in the report, or you can assign an ID in the module editor (secondary data source). When doing so, make sure that the ID is for a composite scenario. When the system reads worklists for stress data, the worklist has to be registered as a component of the ID of the composite scenario.

The base date (/BA1/C65DATRUN) is the date of the HDB run. Although this date exists as an output characteristic for primary data sources of the category *single record,* you cannot use it directly as a selection characteristic. The business record date (key date) used to read from the HDB is automatically compared with the base date for all servers except the *External Systems* server. For external systems the key date is compared with the start date of the RDB run.

> ⚠ Caution
>
> In the case of a delta read, that is if the system uses two time stamps to read the worklist, the server does not automatically consider the base date. The server considers the base date only if you are actively using the base date as a selection characteristic and if you have set the *Delta Read with Key Date* to *No*. For more information, see the documentation for the HDB runs and under Key Date Queries and Delta Queries [page 267].

Granularity indicator:

The results layers in the Historical Database do not recognize business keys. Instead, business keys are defined by means of the granularity characteristics of the primary data sources, and can be used for different types of read-only access.

If versioning is switched on, the setting stored for the granularity indicator in the individual characteristics governs which data the program returns. If the primary data source key, which is defined by the granularity indicator, has a value that does not match the value of the business key, the program returns - based on this business key - either too many or too few data records.

Since versioning is not possible for the *External Systems* server, the system ignores the granularity indicator in the characteristics.

Explanation of the Key Figures

You have already defined the key figures available for the primary data source. You defined these for the data layers, models, and external systems upon which the primary data sources are based. The system does not support selections using the unit of a key figure. This means that the Selection Unit indicator must remain blank.

7.2.1.2.2.1.7 Servers for Limit Manager

Use

The following servers are provided for the primary data sources that are to read data for Limit Manager:

- Limit Manager Limits: limits and limit items
- Limit Manager Utilizations: utilizations and limits at totals record level
- Limit Manager Attributable Amounts: utilizations at single record level

These servers support the following functions:

- Selection of single records
- Selection of worklists
  - Unpackaged
  - Packaged (for parallel processing)

The delta read process for data records is not always supported. As of SAP Note 2015679, the delta read process is supported for server `LMLIMITS`. To do this, the system reads a time stamp for the limit. This time stamp is filled only after the SAP Note has been implemented and if limits are changed or created. This means that a delta read for older limits is not possible.

Recommendation

If you are using Limit Manager data sources of the *single record* category, you should also set up the worklist using a Limit Manager data source because it is only here that you can prepare the key fields for individual accesses in a simple way.

A combination of Limit Manager worklist and Limit Manager single record can improve system performance if the worklist contains only the key fields for single record access and if the characteristics of the segmentation service are accessed during parallel processing, which can be very time-consuming.

Features

Attributes of the Limit Manager Servers

The Limit Manager servers have the following attributes:

Server-Dependent Attributes: LMLIMITS Limit Manager Limits

The server-dependent attributes identify the available characteristics and key figures for the result structure of the data sources. Limit area and limit type or determination method (if defined as an attribute) are also fixed selection options for accessing the database:

- Limit area
  - Limit area ID
- Limit type (optional)
  - Limit type ID
- Select limit items
  - Indicator: yes/no
    If you set the indicator to *Yes*, this increases the number of characteristics available for the results list. This also changes the granularity of the result records because the system also displays the items for each limit header.

Server-Dependent Attributes: LMUTLSUMS Limit Manager Utilizations

- Limit area
  - Limit area ID
- Limit type (optional)
  - Limit type ID
- Read run data
  - Indicator: yes/no
    If you set the indicator to *Yes*, this increases the number of characteristics available for the results list.
- Read limits
  - Indicator: yes/no
    If you set the indicator to *Yes*, this increases the number of characteristics available for the results list.

    ### Note

    If you include the characteristic `/BA1/C61LOCKV` (limit lock violated) in the result structure on the totals server, it can significantly impair system performance because the system has to identify the

lock violation at runtime at limit/single-record level. This also includes a comparison between current and previous single record data from different limit utilization runs.

Server-Dependent Attributes: LMUTLSNGL Limit Manager Attributable Amounts

- Limit area
  - Limit area ID
- Determination method (optional)
  - Determination method ID
- Read run data
  - Indicator: yes/no
    If you set the indicator to *Yes*, this increases the number of characteristics available for the results list.

Characteristics and Key Figures

Explanation of the Characteristics

The characteristics available in the primary data source are defined in Customizing and in the local database tables for Limit Manager. These characteristics can be the following:

- Characteristics of the limit types
- User-defined characteristics of the limit types
- Characteristics of the determination methods
- Technical characteristics (such as date or ID of limit utilization run)

> ➡ Recommendation
>
> If you are using characteristics as selection options, check the underlying tables for the segmentation services and if necessary set indexes to improve system performance. Note that the Limit Manager servers do not offer all characteristics as selection options.

Explanation of the Key Figures

The key figures available in the primary data source are predefined in the local database tables for Limit Manager. The system does not support selections using the unit of a key figure. This means that the *Selection Unit* key figure must be *initial*.

7.2.1.2.3 InfoSet

Use

InfoSets are used to read characteristics of the primary objects of the Source Data Layer. A worklist is created as a result of this reading.

Integration

InfoSets that are created in the Tool BI [page 16] system (where they are also edited) are automatically available in the system as a primary data source. Therefore, you do not need to register them.

Features

Reading an InfoSet results in a worklist, in which each data record meets the selection conditions.

Activities

You edit InfoSets in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Index Function* > *InfoSets* > *Edit InfoSets*.

> **Note**
>
> The InfoSet supports only the data source type [page 1593] *Selection of Worklist*. Since the InfoSets are based on DataStore objects, which contain characteristics only, only characteristics can be read from the InfoSets.

> **Note**
>
> Whereas you can create only one DataStore object, you can create as many InfoSets as you want. As a result, we recommend that you create and activate a separate InfoSet for each application and each processing step within a process. If you use a Tool BI system for several clients from the core Bank Analyzer system, all the InfoSets that exist in the Tool BI system are available in the corresponding input help in each of the core clients. This means that you should define the application areas in order to assign the InfoSets according to their use.

> **Note**
>
> In the case of the primary objects of the master data framework that are persisted in a flat database table, you can also use a general primary data source based on the server `SDLCONFLAT` (contract server for primary objects (MDF)) to read data.

7.2.1.2.4 Generic Primary Data Source

Use

A generic primary data source groups together primary data sources of the object transfer structure (OTS) [page 1594] category in such a way that you can use the same data source for different primary object categories.

Integration

You access the generic primary data source by means of the *unified object identifier (UOID)*, which provides a uniform standard for the identification of primary object categories in the Source Data Layer (also for subobjects of the master data framework). The UOID contains information about:

- Identifiers of the object or the subobject
- Bank Analyzer components or subcomponents

You can define which OTS is integrated in a generic primary data source in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Object Transfer Structure (OTS)* > *Assign the OTS to the Generic Primary Data Sources*.

> **Note**
>
> You can integrate an OTS for each object category (complete object or subobject).

Prerequisites

To ensure that the selected fields of the integrated object transfer structures deliver values during the filling of the OTS, make the following settings in Customizing for *Bank Analyzer*:

- To edit the OTS for primary object categories or components, choose *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Object Transfer Structure (OTS)* > *Edit OTS for Primary Objects*.
- To assign OTS to processes, choose *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Object Transfer Structure (OTS)* > *Assign OTS to Processes*.
- To configure the filling of the OTS in the module editor, choose *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Filling of OTS with Data* > *Configuration of Module Editor* > *Edit Module for Filling OTS with Data*.

Activities

To define the generic primary data source and select all of the characteristics and key figures of the integrated data sources for the worklist fields that are to be delivered, choose Customizing for *Bank Analyzer* > *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.

7.2.1.3 Application Areas for Primary Data Sources

Use

You can use application areas to make a further division of the primary data sources. Application areas enable you to restrict or simplify the selection of primary data sources using the input help (F4 help).

Prerequisites

To enable the useful restriction of the primary data sources by means of the application areas, you first have to assign the data sources to the application areas. You make this assignment in Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*.

Activities

You create or edit application areas for primary data sources in Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Application Areas for Primary Data Sources*.

7.2.1.4 Characteristics and Key Figures in the Primary Data Source

Use

Communication between the primary data sources and the server takes place exclusively by means of characteristics and key figures to ensure that you can configure and extend the settings flexibly.

Based on their tasks in the primary data sources, characteristics and key figures are classified as follows:

- Granularity characteristics
  These are characteristics that are combined and that uniquely identify a data record (they are comparable to the key fields of a database table).
- Selection characteristics
  These are characteristics that are used to restrict the data records supplied in a primary data source. In other words, the system transfers only those data records that meet the selection criteria to the client.
- Deliverable characteristics
  All deliverable characteristics of a primary data source are contained in the data transfer structure and correspond to one column each in the results data.

> **Note**
>
> Granularity characteristics or selection characteristics are transferred to the data transfer structure only if they are also flagged as *deliverable characteristics*.

- Selection using the unit of a key figure
  When you use data sources to access a key figure that is maintained in different units, you have to use the currency field or the unit field of this key figure for the selection of data.

Activities

> **Note**
>
> You can configure characteristics and key figures for general primary data sources [page 1596] only.

You can edit the characteristics and key figures of general primary data sources in Customizing for *Bank Analyzer* under *Infrastructure* > *Communication and Worklist Services* > *Data Sources* > *Primary Data Sources* > *Edit Primary Data Sources*. In the dialog structure, double-click the primary data source that you want to edit and then change to processing mode.

The system displays the complete list of characteristics from the respective repository. You can select as many as you require. If you have assigned a server, it proposes selection values which you can change.

> **Note**
>
> You should only make settings that are supported by the respective server. Because of this you should carry out a test run after you make Customizing settings.

7.2.1.5 Nesting of Data Sources

Use

You can use the nesting of data sources function to refer to another primary data source from within a primary data source of the object transfer structure (OTS) [page 1594] category.

Prerequisites

You have defined all primary data sources that are integrated in the step sequence of the module version and have made the settings required for filling.

Process

You configure nesting of data sources in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services* > *Access to Source Data* > *Filling of OTS with Data* > *Configuration of Module Editor* > *Edit Module for Filling OTS with Data*. Within the step sequence of a module version, you can use the *primary data source* step for this. Within the dialog structure, in the respective *Module*, double-click *Step Sequence*. In the screen area on the right-hand side, use the secondary mouse button to click *Step Sequence*, then choose *Insert Step* > *Primary Data Source*.After you have entered the settings for the respective data source ID and for the associated data source category [page 1594], the system fills not only the OTS but also each of the integrated primary data sources.

> **Note**
>
> Since the function module has to be called before you refer to a primary data source, you have to put the *Primary Data Source* step at the end of the sequence.

7.2.1.6 Deleting Data Packages for Primary Data Sources

Use

You use this transaction to delete data packages for primary data sources. The system displays all of the data packages that have not yet been processed and that are buffered in the cluster tables.

> **Caution**
>
> Be careful to delete only those data packages that are no longer required for processing. Generally, these are packages that were generated during a test run or that were not further processed after a program termination.

Prerequisites

You are authorized to delete your own or external data packages.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Tools* > *Reorganization* > *Primary Data Sources* > *Delete Data Packages for Primary Data Sources*.
2. Select the packages that you want to delete and choose with the quick info text *Delete Data Package*. The data packages that you have selected are deleted.

The system logs the deletion process for data packages. To display the deleted data sources, go to the *SAP Easy Access* screen and choose *Bank Analyzer* > *Tools* > *Reorganization* > *Primary Data Sources* > *Display Logs for Deleted Data Packages*.

7.3 Calculation and Valuation Process Manager (CVPM)

Use

Calculation and Valuation Process Manager (CVPM) provides a framework in which you can configure and run analytical processes. It is not confined to Bank Analyzer. The framework is part of the IFRA architecture.

CVPM provides you with a uniform environment for all valuations. CVPM Process Monitor [page 1635] allows you to track all of the Bank Analyzer processes configured in CVPM and their runs [page 1631] centrally. In addition, you can use CVPM to create and edit your own analytical processes.

SAP delivers a number of preconfigured analytical processes. These include processes for financial accounting (subledger scenario), for the fair value production process, and for data retrieval processes in Asset/Liability Management.

You can also use these processes as templates for custom processes.

Implementation Considerations

You configure an analytical process in Customizing for the *Calculation and Valuation Process Manager*.

Features

The CVPM is divided into the following areas:

- Process Configuration [page 1622]
- Process Orchestration [page 1634]
- CVPM Process Monitor [page 1635]

7.3.1 Process Configuration

Use

In Calculation and Valuation Process Manager (CVPM), you can create analytical processes and define different step sequences for a process. The analytical process defines the scope of the run [page 1631]. The step sequence defines the order in which the steps are processed. The steps contain logic.

Prerequisites

You have already created a subobject in Customizing under *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *General Settings for Custom Processes* > *Application Log: Edit Subobject for CVPM*.

Process

1. You create analytical processes in Customizing under *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *General Settings for Custom Processes* > *Edit Analytical Processes*. When the system saves the analytical process, it generates a report with the same technical name.
2. You can create multiple step sequences for an analytical process. You define these in Customizing for the *Calculation and Valuation Process Manager* under *Edit Step Sequences for Analytical Processes*.
3. There are a number of ways of running the report for a corresponding analytical process:
   - From the application menu of the relevant application of Bank Analyzer
   - Using Schedule Manager
   - Using transaction `SE38`
   - Using custom transactions that you add to your application menu

Result

You can evaluate various attributes of the individual runs, such as the status and error messages in the CVPM Process Monitor [page 1635].

7.3.1.1 Step Controller

Use

The step controller in the Calculation and Valuation Manager (CVPM) forms the basis for the steps and step sequences in an analytical process. You can save a step controller for each analytical process in Customizing for *Edit Analytical Processes* under *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *General Settings for Custom Processes*.

The step controllers in CVPM provided in the standard system include the following:

- Generic step controller (`/BA1/CL_AL_FW_STEP_CONTROL`)
  The generic step controller provides various functions, including time stamp calculation, time stamp intervals, process restarts, and repackaging. When you define the sequence of steps, the generic step controller is much more flexible than the tool-based step controller.

- Sequential step controller (`/BA1/CL_AL_FW_STEP_CONTROL_SQ`)
  The sequential step controller is a special variant of the generic step controller. With the "general" generic step controller, the steps saved in the step sequence can be processed seqentially package by package in several parallel jobs. By contrast, when the sequential step controller is used the system always processes the packages in a single job.

> **Note**
>
> We recommend that you use the generic step controller for new processes and the sequential step controller for new processes with a small worklist or no worklist at all.

- Tool-based step controller (`/BA1/CL_AL_FW_STEP_CONTROL_DPF`)

In addition to the step controllers provided by SAP, you can also define your own process-specific step controllers. This could be useful if you want to migrate existing applications to CVPM, for example.

> **Note**
>
> - You can only use process-specific step controllers for the corresponding application.
> - To use your own implementation, you require in-depth ABAP knowledge. If you want to implement your own classes in addition to those delivered with the system, you must implement the interface `/BA1/IF_AL_FW_STEP_CONTROL`.

More Information

Step Controllers in Smart Accounting for Financial Instruments [page 1624]

7.3.1.2 Step Controllers in Smart Accounting for Financial Instruments

Use

Smart Accounting for Financial Instruments (Smart AFI) uses its own step controllers for day processing and for period-end processing. You cannot use these step controllers in custom processes.

Day processing

The *Register* process step transfers different types of information (for example, business transactions and master data changes) in consecutive steps. The CVPM, therefore, provides a separate step sequence for each of the available information types. The step controller `/BA1/CL_AL_BR_REG_STEP_CONTROL` executes each of these step sequences one after the other. You cannot select them on the selection screen.

Period-end processing

A step sequence for consolidation and a step sequence for execution is available for period-end processing:

- The **consolidation step sequence** determines for which date which transactions are processed. The step controller `/BA1/CL_AL_FC_STEP_CONTROL_PS` executes this step sequence once.
- The **execution step sequence** controls the chronological execution of the relevant process steps for period-end processing. The step controller `/BA1/CL_AL_FC_STEP_CONTROL_PS` executes this step sequence once in succession for every date to be processed.

7.3.1.3 Time Stamp

The system needs time stamps to select data, for example from the Source Data Layer (SDL).

You define which time stamp the system uses by making the following Customizing settings for the *Calculation and Valuation Process Manager (CVPM)* under *General Settings for Custom Processes* > *Edit Analytical Processes*:

- Leading Process
  You can define that you want the system to use the time stamp from a leading process. For more information about leading and dependent processes, see Leading Process [page 1627].
- Process Category
  The process category is used to fine-tune the time stamp transfer. For more information, see Process Category [page 1629].
- Data Reading Groups
  The system uses the data reading groups to read different data sources using different time stamps. SAP delivers the following data reading groups:
  - Source data (SDL)
  - Results data and analytical data (RDL)
  - Market Data
  - Smart AFI: Register documents

  You can make the following settings in the data reading group:
  - On Selection Screen
    When you select this checkbox, the details of the time stamp and the *Calculation* checkbox appear on the selection screen for the corresponding report.
  - Interval
    When you select this checkbox, you can restrict the dataset to be read by specifying a time stamp interval rather than a single time stamp.
  - Calculation
    When you select this checkbox, the system calculates the time stamp from the leading process. If you also select the *On Selection Screen* checkbox, the *Calculation* checkbox appears on the selection screen for the report. This gives you the option of activating or deactivating the calculation before you run the report.

> **Note**
>
> You can only use the time stamp calculation if the process uses the *process partition* process category and the generic step controller [page 1623]. In addition, a leading process must exist.

Some examples are provided below to demonstrate how the settings you make influence the time stamp the system uses:

- If the data reading group is not contained in the process but rather the leading process, the system uses the time stamp of the leading process.
- If the data reading group is not contained in the process and the process does not have a leading process, the system cannot determine a time stamp.
- If the data reading group has been saved in the process, the system calculates the time stamp as follows:

On Selection Screen	Interval	Calculation	Comment	Time Stamp
		X	The time stamp and the *Calculation* checkbox do not appear on the selection screen for the report belonging to the process.	The system uses the (upper) time stamp of the same data reading group (for example, the *Source Data Layer (SDL)*) of the leading process (for the current run in the same process partition). If no corresponding data reading group exists in the leading process, the system uses the current system time stamp.
	X	X	The system uses a time stamp interval to select data. The time stamps do not appear on the selection screen for the process.	**Upper limit of interval:** (Upper) time stamp of the same data reading group of the leading process (for the current run in the same process partition). If no corresponding data reading group exists in the leading process, the system uses the current system time stamp of the process. **Lower limit of interval:** Upper limit of last successful run for the same process. If this does not exist, the system uses the initial time stamp 01.01.0001 (12:00).
X		X	The time stamp and the selected *Calculation* checkbox appear on the selection screen for the report belonging to the process. You can deselect the checkbox and enter your own time stamp.	The system uses the (upper) time stamp of the same data reading group of the leading process (for the current run in the same process partition). If no corresponding data reading group exists in the leading process, the system uses the current system time stamp of the process.

On Selection Screen	Interval	Calculation	Comment	Time Stamp
X	X	X	The system uses a time stamp interval to select data. **Note** The lower time stamp and the selected *Calculation* checkbox appear on the selection screen for the report belonging to the process. You can deselect the checkbox and enter your own lower time stamp. The upper time stamp does not appear on the selection screen and is therefore protected from being changed manually.	**Upper limit of interval:** (Upper) time stamp of the same data reading group of the leading process (for the current run in the same process partition). If no data reading group exists in the leading process, the system uses the current system time stamp of the process. **Lower limit of interval:** Upper limit of last successful run for the same process. If this does not exist, the system uses the initial time stamp 01.01.0001 (12:00). If you enter a lower time stamp on the selection screen of the report belonging to the process, the system uses it.
	X		The system uses a time stamp interval to select data.	**Upper limit of interval:** Current system time. **Lower limit of interval:** 01.01.0001 (12:00)
X	X		The system uses a time stamp interval to select data	**Upper limit of interval:** Entered time stamp **Lower limit of interval:** Entered time stamp
X				You enter the time stamp on the selection screen for the report belonging to the process.

7.3.1.3.1 Leading Process

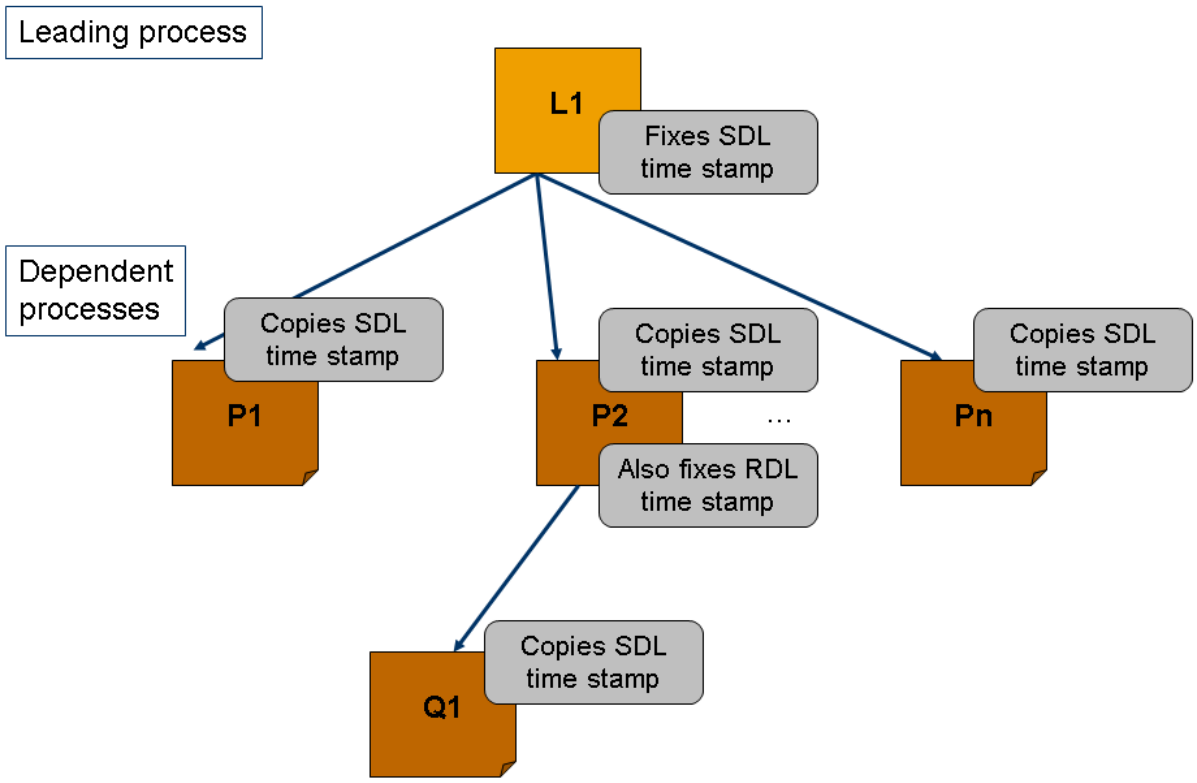
In Customizing for Calculation and Valuation Process Managers (CVPM) under *General Settings for Custom Processes* > *Edit Analytical Processes*, you can define a leading process for one or more analytical processes. When you schedule certain processes, you have to ensure that the system runs a leading process before the dependent processes. Only then can the leading process determine data reading timestamps for dependent processes. This can be necessary if the underlying data basis has changed, and this is relevant for running the dependent processes.

The following data reading groups are available:

- Source data (SDL)
- Results data and analytical data (RDL)
- Market data
- Smart AFI: Register documents

If you want the dependent processes to use the timestamp determined by the leading process for creating the worklist, you have to name the leading process in Customizing for the dependent process.

The following restriction applies to this concept: **Each process can have only one leading process.**The leading process can, however, have any number of dependent processes.



A dependent process cam also be a leading process for other processes. In the example, process `P2` plays the role of leading process for `Q1` and `Q2`, and `L1` is the leading process for `P2`.

> **Note**
>
> A model such as this, which you can configure in CVPM, defines which process determines its timestamp itself, or which leading process passes its timestamp to which dependent process. You have to have previously run the leading process at least once.
>
> Process orchestration defines the sequence in which the system runs the processes.

When they run the creation of the worklist of source data, `P1-Pn` use the timestamp of leading process `L1`, which was determined in the SDL data reading group. When `Q1` and `Q2` run the creation of the worklist of results data, they use the timestamp of leading process `P2`, which was determined in the RDL data reading group.

SAP provides the leading process *Set SDL Time Stamp* `/BA1/RB0_CLOSE_SDL`, which determines the time stamp of the data reading groups 'SDL' and 'market data' for a number of processes in *Processes and Methods*. The *Set SDL Time Stamp* process only allows monotonously increasing time stamps. Apart from this, the process does not have more complex processing logic. The process category is used to fine-tune the time stamp transfer. For more information, see Process Category [page 1629].

7.3.1.3.2 Process Category

Definition

Allows you to configure the granularity of analytical processes and therefore fine-tune the settings for transferring the time stamp.

Structure

Analytical Processes

Depending on whether an analytical process is a leading process or a dependent process, the system differentiates between the following categories:

- Standalone Process
  Standalone processes are not connected to leading processes. You cannot assign a partition to standalone processes.
  If you configure an analytical process as a standalone process, the process does not require a leading process to fix the time stamp or time stamps of the data reading group(s). All data reading groups are linked to this standalone process. You either enter the data reading group on the selection screen, or the system uses the current system time stamp. You can run this process independently.
- Process Partition
  A process in the *Process Partition* category can be both a leading process and a dependent process.
  If it is a dependent process and the data reading group is not attached to the process, the leading process manages the time stamp. Here, the leading process uses the process partition to transfer the time stamps. Consequently, *Process Partition* appears as a mandatory field on the selection screen for the report belonging to the process for both the leading and the dependent process. This allows you to schedule the leading and dependent processes in parallel with different process partitions. Note that the leading process must also belong to the *Process Partition* or *Process Partition Adapter* categories.

  > **Note**
  >
  > The data reading time stamps used by the analytical process are attached not only to the last run of the leading process, but also to the last successful run of the corresponding partition.

- Legal Entity

A process in the *Legal Entity* category can be both a leading process and a dependent process.
If it is a dependent process and the data reading group is not attached to the process, the leading process manages the time stamp. Here, the leading process uses the process partition to transfer the time stamps. Consequently, the leading process either has *Legal Entity* or *Process Partition* flagged as a mandatory field. The dependent process has *Legal Entity* as an additional mandatory field on the selection screen. This allows you to schedule the leading and dependent processes in parallel with different process partitions. Note that the leading process must also belong to the *Legal Entity* or *Process Partition* categories.

> **Note**
>
> The data reading time stamps used by the analytical process are attached not only to the last run of the leading process, but also to the last successful run of the corresponding partition or legal entity.
>
> The *legal entity* process category is used mainly by accounting scenarios.

- Source System
  A process in the *Source System* process category can be both a leading process and a dependent process.
  If it is a dependent process and the data reading group is not attached to the process, the leading process manages the time stamp. Here, the leading process uses the process partition to transfer the time stamps. Consequently, the leading process either has *Source System* or *Process Partition* flagged as a mandatory field. The dependent process has *Source System* as an additional mandatory field on the selection screen. This allows you to schedule the leading and dependent processes in parallel with different process partitions. Note that the leading process must also belong to the *Source System* or *Process Partition* process category.

  > **Note**
  >
  > The data reading time stamps used by the analytical process are attached not only to the last run of the leading process, but also to the last successful run of the corresponding partition or of the source system.
  >
  > The *Source System* process category is used mainly by accounting scenarios.

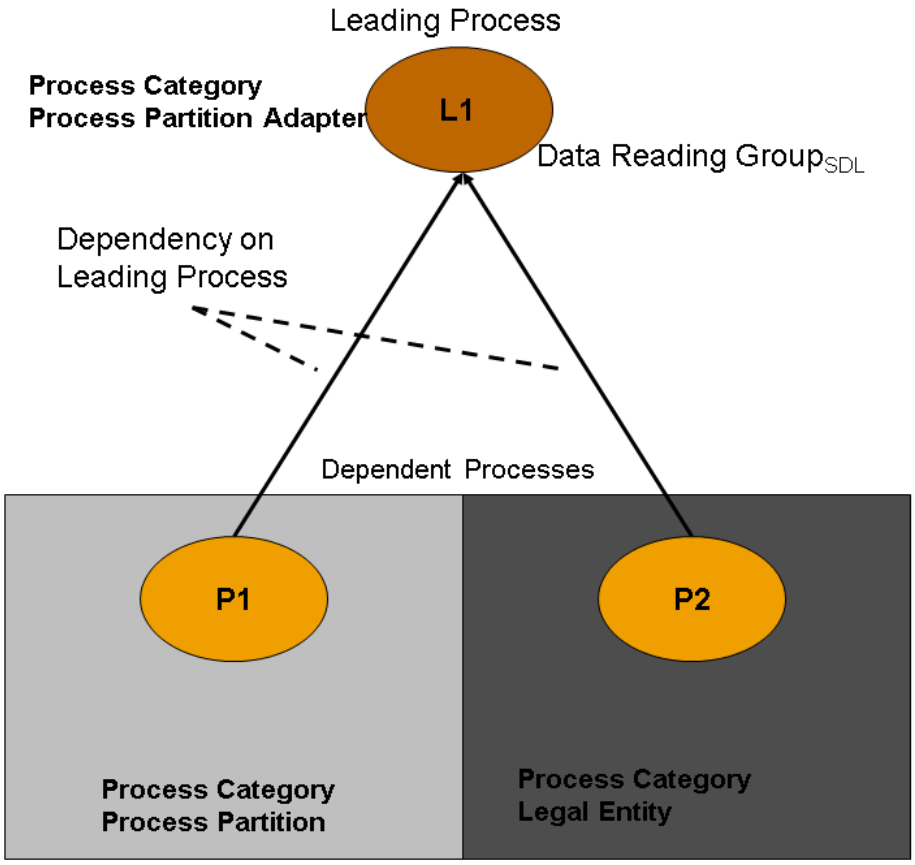
- Process Partition Adapter
  Processes in the *Process Partition Adapter* category are leading processes that manage their time stamps themselves. The processes that are dependent on this type of process must belong to the *Process Partition* category.
  The selection screen for the report belonging to the process contains *Process Partition* as an additional mandatory field and *Reference Partition* as an optional field. The selection screen for the report belonging to the dependent process contains *Process Partition* as an additional mandatory field. This allows you to schedule the leading and dependent processes in parallel with different process partitions or reference partitions.

  > **Note**
  >
  > If you enter a process partition that has already been processed in the *Reference Partition* field, the system transfers the time stamp of the reference position for the process partition entered in the *Process Partition* field. In this case, the reference partition behaves like a leading process. This is illustrated by the following example.

Example

The following example explains the relationship between reference partitions and analytical processes:

Analytical processes `P1` and `P2` are dependent processes of leading process `L1`, which determines the SDL time stamp.



Example Using Reference Partition

First the system runs process `L1` for process partition `ABC`. The optional field for the process partition adapter is empty. This means that the SDL time stamp is linked to process partition `ABC`. The subsequent run of process `P1` for process partition `ABC` uses the same time stamp.

Then the system reruns process `L1` using legal entity `XYZ` and process partition adapter `ABC`. The subsequent run of `P2` uses the time stamp of process partition `ABC`.

7.3.1.4 Run Group

When the system executes an analytical process, it generates a run and then groups together all of the runs it generates.

In Customizing for CVPM, you can make the settings for run group generation under *General Settings for Custom Processes* > *Edit Analytical Processes*. When creating run groups, you can choose between these two settings:

- Always
  If you choose *always* when creating a run group, the system creates a new run group for every run.
- Optional
  If you choose *optional* when creating a run group, you can use an indicator before you run the analytical process to determine whether the system creates a new run group.
- Automatic
  If you choose *Automatic* when creating a run group, the system uses the values of the process parameters flagged as run group separators to decide whether it has to generate a new run group. You cannot override this decision.

The system lists all of the runs and run groups for the different analytical processes in CVPM process monitor.

> **Note**
>
> If you encounter technical problems, you can restart an analytical process, in other words you can create a new run in the original run group.
>
> This only works if:
>
> - You are using a step controller delivered by SAP. Ensure that the step sequence does not contain an aggregation step.
> - When editing the analytical process for run group generation in Customizing for CVPM, you selected *optional*. If you select *always*, the system automatically creates a new run group for the new run, and you cannot control the run in the original run group.

7.3.1.5 Capability to Restart Parallel Processes

Use

You can use this function to restart incomplete, canceled, or failed processes in the Calculation and Valuation Process Manager (CVPM). You can restart processes using the following methods:

- Process restarts controlled by run groups
  In the CVPM, you can define how you want the system to control run group creation for analytical processes. The following statements apply for runs whose analytical process uses a step controller provided as part of CVPM:
  - If you choose *Always* when creating a run group, you cannot restart runs.
  - If you choose *Optional* when creating a run group, you can restart incomplete runs (run status *Incomplete)* by restarting the run with identical values to the process parameters flagged as run group separators and with the *New Run Group* checkbox deselected.
  - If you choose *Automatic* when creating a run group, you can restart incomplete runs using identical values to the process parameters flagged as run group separators.

  When you restart a run, the system decides which process step to restart in and identifies the data that has not been processed fully.

- Restarting at restart points
  In the CVPM, you can define restart points for analytical processes that use the generic or sequential step controllers provided as part of the CVPM. If, for example, data records are not processed for any reason, you can use restart points to restart runs at a particular step in the analytical process. In this case, the data records no longer have to go through the entire process.

For more information about the different step controllers, see Step Controller [page 1623].

Prerequisites

- If you want to control process restarts on the basis of run groups in your analytical process, you have to select *Optional* or *Automatic* for run group creation in your analytical process in Customizing for *Bank Analyzer* under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *General Settings for Custom Processes* > *Edit Analytical Processes*.
- If you want to use a restart point for your process, you must make at least one of the following settings in Customizing for the CVPM under *Edit Custom Step Sequences for Analytical Processes*:
  - You have defined the corresponding *Restart Point Category* for one or more steps in your process.
  - You have selected the *Restart Point* checkbox for a step with the category *Aggregation*.

Features

Restarting at restart points

You can define steps with the categories *Worklist Creation*, *Aggregation*, and *Repackaging* as restart points. You can select which data you want the system to save using the following settings in the *Restart Point Category* field:

- No Restart Point
  The system does not save any data for the synchronization point.
- Partial Restart Point
  The system only saves data records that are lost or flagged as incorrect after this step.
- Full Restart Point
  The system saves all data records for a run.

If the restart point has the step category *Worklist Creation* or *Repackaging*, the system always saves the status of the data **after** the respective step.

If the step category is *Aggregation*, you can define restart points both before and after the step. You define a restart point before aggregation by selecting the *Restart Point* checkbox for the step in category *Aggregation*. In this case, the system saves **all** the data records for the run **before** the corresponding step. You define a restart point after the aggregation step using the *Restart Point Category* field.

> **Note**
>
> Note that activating one or more restart points can affect performance.

If you have defined restart points for an analytical process in Customizing and you then start a run, the following functions are available in the CVPM process monitor:

- You can display the step sequence that was defined and the places where the process can be restarted for the run.
- You can restart a run at a restart point in the process.
  If the value in the *No. of Objects* field is greater than zero for a step defined as a restart point, the CVPM provides data for this restart point. To restart the process from this restart point, select the respective step and choose the *Restart Process* pushbutton.
- You can delete data contained in the CVPM for potential restarts.

7.3.2 Process Orchestration

Use

You use process orchestration to run multiple analytical processes in succession. You pool the analytical processes in a process group. You also define the sequence in which the system runs the analytical processes here.

Features

You can run process orchestration using Schedule Manager, for example. Here, the system carries out the following activities:

- Transfer of saved data to Schedule Manager
- Transfer of status of run to status of Schedule Manager
- Reference of Schedule Manager to application log of a run
- Generation of restart of run using Schedule Manager
- Process and transfer of variables from Schedule Manager to the run
- Distribution of the timestamp in the execution of a process chain using Schedule Manager
- Integration of a viewer that is specific to the process from the application log to Schedule Manager.

> **Note**
>
> The flow definition in Schedule Manager is used to define process groups.

> **Note**
>
> When the system sets the `Finished` status in Schedule Manager, the subsequent process is started in line with the flow definition.
>
> If the system sets the `Reporterror` status in Schedule Manager, processing in the flow definition stops. The system sends you a work flow work item, which you can use for a restart, or to proceed with processing of the subsequent process according to the flow definition.

> **Note**
>
> You can use `XBP` interface to access tools for job scheduling in process orchestration.

More Information

For more information about Schedule Manager, see the Application Help for SAP ERP Central Component under http://help.sap.com/erp. Choose *Accounting* > *Controlling (CO)* > *Controlling (CO)* > *Controlling Methods* > *Schedule Manager (CA)*.

7.3.3 CVPM Process Monitor

Use

You can analyze all of the runs for your analytical processes in CVPM Process Monitor. On the *SAP Easy Access* screen, choose *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor*.

Prerequisites

You have created and executed an analytical process in Customizing for Bank Analyzer under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *General Settings for Custom Processes* > *Edit Analytical Processes*.

Features

On the *CVPM Process Monitor* screen, you can choose to read the data from the database, the archive, or the database and the archive. You can also specify a layout that you saved previously for the list of runs in CVPM process monitor and thereby control which information the CVPM process monitor displays in the list of runs.

When you execute the CVPM process monitor, the system creates a list of all runs for the selected processes. The run display provides pushbuttons for a number of functions, including the following:

- *Application Log*
  You can display the complete application log for a selected run here.
- *Parameters*
  The technical parameters that the selected run used can be found here. You can also read data reading groups and timestamps here.

- *System Status*
  Displays the technical system status. You can also release result groups of the results data in the Results Data Layer (RDL) or reset the release here.

  > **Note**
  >
  > To use this function, the system must use result group versioning in the Results Data Layer (RDL). In addition, you must specify a results data area and result data type in Customizing under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Custom Step Sequences for Analytical Processes*.

- *Run Dependencies*
  This view shows you how the run hierarchy is structured and specifies the leading process.
- *Detail Statistics*
  In this view you can read the run times, job distribution, and memory usage, for example.
- *Log Analyzer Tool*
  The log analyzer provides you with additional functions for analyzing large application logs.
- *To Be Archived*
  The system flags all the runs selected under *Archive* for archiving in the Archiving Engine.
- *Run Group*
  Displays all the runs belonging to a run group.
- *Selections*
  With the two selection functions you can select all the runs in the list for the *Archive* field, or deactivate the selection.
- *Archiving Engine*
  Starts the Archiving Engine, where you can archive the flagged runs.
- *Information About the Restart Points*
  Here, you can display the defined step sequence and the available restart points for each run in an analytical process. Steps in the categories *Worklist Creation*, *Aggregation*, and *Repackaging* can act as restart points. You must define the restart points in Customizing before you start the run.
  If the Calculation and Valuation Process Manager (CVPM) contains data for a step that you have defined as a restart point, you can restart the process from there. In other words, the process starts again with the next process step.
  A report is also available for deleting the data being kept in CVPM for a possible restart.
  For more information on restart points, see Capability to Restart Parallel Processes [page 1632].
- Process-Specific Pushbuttons
  If you have assigned the selected process to one or more process-specific pushbuttons, these pushbuttons appear in the run display in the CVPM process monitor.
  SAP provides the following process-specific pushbuttons:
  - Detail Log
  - Display DMCPM Run Type
  - Display DMCPM Customizing
  - Display Loss Distribution
  - Display Detail Screen
  - Display RDL Results
    Displays the RDL data for the selected run. To use this function, the process must have saved data in the RDL. Result group versioning must be active in the system. In addition, you must specify the results data

area and result view in Customizing under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Custom Step Sequences for Analytical Processes*.

To create process-specific pushbuttons or assign a process to a pushbutton, go to Customzing for Bank Analyzer and choose *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *General Settings for Custom Processes* > *Define Process-Specific Pushbuttons*. You can also use this activity to create your own process-specific pushbuttons.

7.3.3.1 Archiving of CVPM Process Monitor

To control the size of the dataset, you can archive data at run level using Archiving Engine.

If you archive data by mistake, you can use Archiving Engine to reload it.

7.4 Extraction and Reporting Services

The extraction and reporting services provide functions that applications can use to access Bank Analyzer data and process it.

Whereas the communication and worklist service processes data so that is can be made available within Bank Analyzer, the extraction and reporting services encapsulate the calls of the communication and worklist services so that the data can be used externally or for reporting purposes.

Data can be accessed externally in three ways:

- The function is called synchronously by an external system (for example, by a BAPI)
- The data is stored in a file or in a database table that can be accessed by an external system
- The data is extracted to a BI system, or processed in a BI system
  - With a generic extractor module
  - With a delta queue procedure
- Generation of receiver reports for a drill-through from the BI system back to Bank Analyzer

7.4.1 Generic BI Data Extraction

You use the generic BI data extraction function to extract results data, for example from the Results Data Layer (RDL), and to extract source data from the Source Data Layer (SDL) in the Bank Analyzer core system to SAP NetWeaver Business Intelligence (BI). In BI, you use the extracted data for reporting.

For instance, you can use the generic BI data extraction function to transfer to BI the results data you need to disclose for Basel II. In BI you then create the reports required under Basel II. Due to the different data requirements of the various risk types (credit risk, equity holdings, and securitizations), you can use a number of data sources for the extraction.

You have the following options for extracting data from the Bank Analyzer core system:

- Generic BI data extraction with the Data Processing Framework (DP Framework) (see Generic BI Data Extraction with the DP Framework [page 1638])
- Generic BI data extraction with the Calculation and Valuation Process Manager (CVPM) (see Generic BI Data Extraction with the CVPM [page 1644])

7.4.1.1 Generic BI Data Extraction with the DP Framework

Use

You can extract data with generic BI data extraction from the Bank Analyzer core system to SAP NetWeaver Business Intelligence (BI) using the Data Processing Framework.

> **Note**
>
> You can set up the selection of the data that is to be extracted flexibly and can choose an extraction structure.

Prerequisites

- You have defined the selection settings in Customizing for Bank Analyzer under *Infrastructure* > *Extraction and Reporting Service* > *Generic BI Data Extraction* > *General Settings for Data Selection* > *Selection Settings for Data Extraction*.
- You have generated a DataSource with a suitable extraction structure in Customizing for Bank Analyzer under *Infrastructure* > *Extractions and Reporting Service* > *Generic BI Data Extraction* > *Edit Generic BI Data Extraction*.

Process

1. Extractor test

   > **Recommendation**
   >
   > We recommend that you also simulate extraction without having your BI system connected.
   >
   > This allows you to test whether the Bank Analyzer core system is selecting the data correctly and placing it in the extraction structure. For more information, see Test BI Extractor [page 1639].
   >
   > You can also test the selection made by the Data Processing Framwork using the Generic Ad hoc Reporting [page 1649] Bank Analyzer function.

2. Extraction
   For the actual extraction, you copy the data sources to the BI system and trigger the extraction there.

3. Run administration
   The run administration [page 430] shows you the status and the application log for extraction runs with the Data Processing Framework. You can delete logs of runs in run administration. You can delete the extraction run itself in BI. Test runs are not listed in run administration.

7.4.1.1.1 Testing the BI Extractor

Prerequisites

- You have defined the selection settings in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Generic BI Data Extraction* > *General Settings for Data Selection* > *Selection Settings for Data Extraction*.
- You have defined a DataSource with a suitable extraction structure in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Generic BI Data Extraction*.

Context

Before you extract data from the Bank Analyzer core system to SAP NetWeaver Business Intelligence (BI), you should test whether the system selects the required data, and fills the extraction structure correctly. Only by doing these tests can you ensure that the BI data extraction works as expected using your settings, and that the system will later extract the data in the required form.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Generic Data Extraction* > *Test BI Extractor*.
2. Enter the DataSource.

   The system enters the request number **TEST**, and sets the default number of data records to 100. The default number of extractor calls is 100, and the default update mode is **F** (transfer of all requested data).

   You can change these entries. You may need to do so if, for example, you want to test data extraction for more than 100 records, or if one call of the extractor is adequate for your test.
3. In the Business Date field, enter a business record date, and specify which data is to be selected by choosing the parameters for your selection criteria:
   - The DataSource is defined from the point of view of characteristics and key figures.
     If you want to display the application log for the data processing framework immediately after the extraction run, set the *Application Log* indicator

If you set the *Test Run* indicator, you can also set the *Application Log* indicator to display the log. The system does not write test runs to run administration or the application log for the data processing framework.

- The DataSource is defined from the point of view of data processing.
  Set the *Use Run Admin* indicator in the *Data Processing* dialog structure in Customizing for *Bank Analyzer* by choosing *Infrastructure* > *Extraction and Reporting Services* > *Generic BI Data Extraction* > *Generic BI Data Extraction*.

4. Choose *Start Extraction*.

Results

The system selects the data.

You can then carry out the following actions:

- To display a list of the data that was selected, choose *Display List*.
- To display the system log for the function *Test BI Extractor*, choose *Display Log*.
- To display the application log for the data processing framework, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Generic Data Extraction* > *Run Administration* > *Run Administration for BI Data Extraction*.

7.4.1.1.2 Run Administration

Definition

Run administration includes the following functions:

- Execute or create run
- Display an overview of runs
- Display Application Log
- Edit run
- Manage run
- Replace run
- Select run for archiving
- Delete run
- Log of the deletion function

> **Note**
>
> The above functions are not all available for each application. For more information, see the application-specific documentation.

Use

The following table lists the runs available for each application:

Application	Run
General Calculation and Valuation Methods [page 376]	Account Pooling [page 383] or Account Pooling (CVPM) Facility Distribution [page 399] or Facility Distribution (CVPM) Determination of the Free Line for Facilities [page 395] or Determination of the Free Line (CVPM) Collateral Distribution [page 404] or Collateral Distribution (CVPM) Determination of Default [page 410] Stress Runs [page 429]: • Stress run for account pooling or stress run for account pooling (CVPM) • Stress run for facility distribution or stress run for facility distribution (CVPM) • Stress run for determination of the free line or stress run for determination of the free line (CVPM) • Stress run for collateral distribution or stress run for collateral distribution (CVPM) • Stress run for determination of default
Credit Risk [page 1191]	Credit Exposure Run [page 1238] or Credit Exposure Run (CVPM) Country Risk Run [page 1336] (prototype for country risk) **Stress run:** Credit Exposure Stress Run [page 1261] or Credit Exposure Stress Run (CVPM)

Application	Run
Historical Database [page 1478]	**Version management:** Run for generating historical data for data layers [page 1481] Run for generating historical data for the bank's own models [page 1484] Uploading of Files [page 1486] **Calculation functions:** Determining Default Rates [page 1488] Determining Average Default Rates [page 1491] Determining Default Figures [page 1493] Calculation of Migration Matrices [page 1496] **Data retrieval:** Run for supplying data to models [page 1498] Downloading of Files [page 1500] **Stress tests:** Stress Run for Supplying Models with Data [page 1505] Generating Scenario Data in the Source Data Layer [page 1506]
Generic BI Data Extraction [page 1638]	Testing the BI Extractor [page 1639] BI extraction run **Note** Extraction runs are created and executed in SAP NetWeaver Business Intelligence (BI). The system displays information about extraction runs in run administration of Bank Analyzer.
Regulatory Reporting Interface	Data Extraction Runs
Limit Manager [page 1529]	Limit Utilization Run [page 1551]
Fair Value Effectiveness Test for Hedging Relationships [page 1036]	Effectiveness Test Run
Cash Flow Hedge Analysis [page 1107]	Valuation Run [page 1128] Subvaluation Run Aggregation Runs [page 1129]

Application	Run
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169] Portfolio Item Runs [page 1172] • Period-end run for portfolio items (subledger scenario) • Period-start run for portfolio items (subledger scenario) • Portfolio item run (merge scenario)

> **Note**
>
> For some of the Bank Analyzer components, you can use the Schedule Manager to schedule and control jobs. If you use multiple applications, you can define the sequence in which the runs are to be carried out. For more information, see Schedule Manager [page 437].

More Information

Status Overview for Run Administration [page 1248]

7.4.1.1.2.1 Special Features of Run Administration in Generic BI Extraction

Use

Run administration manages information about extraction runs that have been executed already. For example, the runs may stem from Generic BI Data Extraction with the DP Framework [page 1638] or from the Test BI Extractor [page 1639] function. The run overview gives you access to the application log for the run.

> **Note**
>
> To see your extraction runs in run administration, you first have to select the *Use Run Admin* checkbox for your DataSource in the *Data Processing* area in Customizing for *Edit Generic BI Data Extraction* under *Bank Analyzer* > *Infrastructure* > *Extractions and Reporting Service* > *Generic BI Data Extraction*.

Features

To execute the functions, go to the *SAP Easy Access* screen and choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Generic Data Extraction* > *Run Administration*

- *Run administration for BI data extraction*
- *Delete flagged runs*
- *Display deletion logs*

Delete Flagged Runs

You can delete logs of runs in run administration. You can delete the extraction run itself in BI.

1. Call up the *Run Administration for BI Data Extraction* function and enter the selection criteria.
2. Select the relevant extraction run in the run overview.
3. Choose the *Enable Deletion* pushbutton followed by *Set Deletion Flag*.
4. Save your changes.
5. Call up the *Delete Flagged Runs* function and enter the selection criteria.

> **Note**
>
> Make sure that the *Test Run* control parameter is not selected. When test run is selected, you are in display mode.

6. Select the extraction run in the run overview and choose the *Delete Selected Run* pushbutton.
7. Call up the *Display Deletion Logs* function to obtain an overview of the status of the deletion process.

7.4.1.2 Generic BI Data Extraction with the CVPM

Use

In addition to generic BI data extraction with the Data Processing Framework, you can also use the functions of the Calculation and Valuation Process Manager (CVPM) to extract data from the Bank Analyzer core system to SAP NetWeaver Business Intelligence (BI).

The following four analytical CVPM processes are provided for generic BI data extraction with the CVPM in Customizing for the CVPM under *General Settings for Custom Processes* > *Edit Analytical Processes*:

- *Generic BI extraction with key date*
  `/BA1/RR7_GEN_BI_EXTR_CVPM_KD`
- *Generic BI extraction with key date and SDL time stamp*
  `/BA1/RR7_GEN_BI_EXTR_CVPM_KDTS`
- *Generic BI extraction with delta worklist*
  `/BA1/RR7_GEN_BI_EXTR_CVPM`
- *Generic BI extraction with delta worklist and SDL time stamp*
  `/BA1/RR7_GEN_BI_EXTR_CVPM_DLTS`

Four transactions corresponding to these processes are provided in the area menu. To call up these transactions, go to the SAP Easy Access screen and choose *Financial Services* > *Bank Analyzer* > *Infrastructure* > *Extractions and Reporting Services* > *Generic BI-Data Extraction*

- *Extraction with key date*
  You can use the process belonging to this report to extract data for a key date of your choice.

- *Extraction with key date and SDL time stamp*
  You can use the process belonging to this report to extract data for a key date of your choice, taking into account the SDL time stamp.
- *Extraction with delta worklist*
  You can use the process belonging to this report to extract data from a delta worklist.
- *Extraction with delta worklist and SDL time stamp*
  You can use the process belonging to this report to extract data from a delta worklist, taking into account the SDL time stamp.

> **Note**
>
> In the application menu under *Interfaces to Other Applications*, the *BI Data Extraction* transaction is also provided. You can use the process belonging to this report to extract data for a key date of your choice, taking into account **several** scenarios. To call up this transaction, go to the SAP Easy Access screen and choose *Financial Services* > *Bank Analyzer* > *Interfaces to Other Applications* > *Data Supply for AL* > *BI-Data Extraction*.

You can use the transactions to test data extraction for the extraction procedures *delta queue*, *direct access*, *drill-through*, *full update (or generic delta)*. You do this before you start the process by selecting the *Test Run* checkbox on the selection screen for the report belonging to the process.

You trigger the actual data extraction for the *direct access*, *drill-through* and *full update (or generic delta)* procedures from BI; for the delta queue procedure, you use the aforementioned transactions.

Process

1. Create step sequence
   Create at least one step sequence for the aforementioned analytical processes in Customizing for Bank Analyzer under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Edit Custom Step Sequences for Analytical Processes*.
   SAP provides four sample classes for generic BI data extraction with the CVPM. You can specify these sample classes in the *Steps* area in the *Step Execution* field:
   - `/BA1/CL_AL_R7_GEN_BI_EXTR_PE`
     Parameter enrichment when using the *delta queue* extraction procedure
   - `/BA1/CL_AL_R7_GEN_BI_EXTR_ST`
     Data enrichment for writing data to the delta queue database table
   - `/BA1/CL_AL_R7_GEN_BI_EXTR_AC`
     Data enrichment when using the *direct access*, *drill-through*, and *full update (or generic delta)* extraction procedure from the BI system to Bank Analyzer
   - `/BA1/CL_AL_R7_GEN_BI_EXTR_DY`
     Result view (only in SAP Bank Analyzer, for example in a test run of the CVPM process)

   An example of a CVPM step sequence is provided in the *Example* section.
2. DataSource
   1. Generate a DataSource in Customizing for Bank Analyzer under *Infrastructure* > *Extraction and Reporting Service* > *Generic BI Data Extraction* > *Edit Generic BI Data Extraction Using CVPM*. The extraction structure here is defined by the configuration of the CVPM process.

2. Assign the DataSource with an analytical process and a step sequence for delta queue extraction or for *direct access* or *full update*.
3. Activate the DataSource.

3. Extractor test
   Before you extract data to BI, you should test whether the system is selecting data correctly. To do so, select the *Test Run* checkbox on the selection screen for the report belonging to the process before you start the CVPM process run.

   > **Recommendation**
   >
   > We recommend that you simulate extraction even if you do not have a BI system connected.

   > **Note**
   >
   > If you have generated the DataSources for the *direct access*, *drill through*, or *full update* extraction procedures in the *Edit Generic BI Data Extraction* Customizing activity, you can also use the *Test BI Extractor* transaction to test data extraction.

4. Extraction
   Before extraction takes place, replicate your DataSources to the BI system.
   - Delta queue procedure
     You can use the aforementioned transactions to trigger the extraction in the delta queue procedure. To do so,deselect the *Test Run* checkbox on the selection screen for the report belonging to the process.
     The processes write data to the delta queue database table
   - Direct access, drill-through, full update
     You trigger data extraction in BI. The extraction procedure then also uses the CVPM processes to extract data.
5. Run administration
   You can view the run status and the log in the CVPM process monitor. Test runs are not listed in the CVPM process monitor. To access the CVPM process monitor, go to the SAP Easy Acess screen and choose *Financial Services* > *Bank Analyzer* > *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)* > *Start CVPM Process Monitor* > .

Example

Example of a CVPM step sequence:

- First step in the step sequence
  Parameter enrichment (with class `/BA1/CL_AL_R7_GEN_BI_EXTR_PE`) is the first step in the step sequence. The class tests in an update run whether the delta queue (`Transaktion RSA7`) has already been initialized. If initialization has not yet taken place, the process is terminated. In the test run and update run, the class determines the DDIC structure assigned to the DataSource. It also transfers the key date defined on the selection screen to the process.
- Second and subsequent steps
  Parameter enrichment is followed by the *worklist creation* step and also, for example, the *data enrichment* step.
- Penultimate step

- Delta queue procedure
  Data enrichment step for writing data (only in the update run) to the delta queue database table (class `/BA1/CL_AL_R7_GEN_BI_EXTR_ST`). In the *Execution Condition* field, select *Update Run*
- Direct access
  Direct accesses from BI to the Bank Analyzer use data enrichment (class `/BA1/CL_AL_R7_GEN_BI_EXTR_AC`) as the penultimate CVPM process step. In the *Execution Condition* field, select *Update Run*

- Last step
  The last step in the step sequences is the results display (class `/BA1/CL_AL_R7_GEN_BI_EXTR_DY`). In the *Execution Condition* field, select *Test Run*

More Information

For more information about the CVPM, see Calculation and Valuation Process Manager (CVPM) [page 1622].

7.4.1.3 Data Extraction with the Open Hub Service

Use

The open hub service is part of the *SAP NetWeaver Business Intelligence* solution and can be accessed via the integrated *SAP NetWeaver Business Intelligence* system.

The open hub service allows you to distribute data from *Bank Analyzer* to other applications using existing *SAP NetWeaver* technology.

This ensures that you are able to manage the distribution of data to several systems. The open hub service is used to define the data to be distributed, the object it comes from, and the destination it is to be forwarded to.

The following Business Intelligence objects can be used as open hub data sources:

- DataSource
- InfoSource
- InfoCube
- DataStore object
- InfoObject (attributes or text)

Database tables and flat files can be used as open hub destinations. You can choose between full and delta extraction modes.

Prerequisites

A data basis must exist to perform extraction with open hub. For example, for the *SAP Basel II* solution, you first have to carry out the following processes and make the following Customizing settings within Bank Analyzer:

1. Create financial transactions in the *Data Source Layer*.
2. Execute a run for selected financial transactions in *Basel II CE*.
3. Release the run in *Basel II CE* with run administration.
4. Create a primary data source of type worklist to be selected for data extraction.
5. - Using the Data Processing Framework
   Create a group ID and selection ID so that the primary data source can be included and save these IDs in the Bank Analyzer DataSource.
   - As an alternative to the Data Processing Framework you can use a CVPM step sequence.
6. Generate the Bank Analyzer DataSource and automatic replication to the BI system.

You have to make the following Customizing settings in the *SAP BI system* (*Tool BI* of *Bank Analyzer*):

1. Create an InfoPackage for the BI DataSource.
2. Create the open hub destination with BI DataSource as the template.
3. Define the transformation with the open hub destination as the destination.
4. Create the data transfer process with the open hub destination as the destination.
5. Create a process chain.
   1. Include the InfoPackage (BI DataSource as the data source) in the process chain.
   2. Include the data transfer process in the process chain.
   3. Activate the process chain.

> **Note**
>
> You can start the process chain from a *third-party application* via API and get extracted data from a database table.

For more information, see SAP Library for SAP NetWeaver on SAP Help Portal under *http://help.sap.com* > *SAP NetWeaver* > *SAP NetWeaver 7.0 (2004s)* > *SAP NetWeaver 7.0 Library* > *SAP NetWeaver Library* > *Key Areas of SAP NetWeaver* > *Key Areas of Information Integration* > *Business Intelligence* > *Data Warehousing*.

- *Data Distribution* > *Open Hub Services*
- *Transformation*
- *Data Distribution* > *Data Transfer Process*
- *Data Warehouse Management* > *Process Chain*

Process

You can trigger execution of the process chain by means of the following events:

- Execute the process chain via job control.
- Start an external call via the *third-party application* via *RFC* (Remote Function Call) RSPC_API_CHAIN_START.

Result

You can verify that the process has been executed successfully in the BI monitor.

The open hub service presents the data as a relational database table or a flat table.

For open hub destinations of type *third-party tool*, you can obtain the data from the *third-party application* from the database table with API **RSB_API_OHS_ DEST_READ_DATA**.

7.4.2 Generic Ad Hoc Reporting

Use

Generic ad hoc reporting displays data from Bank Analyzer. The Data Processing Framework is used to select the data. Depending on which application of the Data Processing Framework you use, you can display various data records, such as historical data or limit data.

> **Recommendation**
>
> We recommend that you use generic ad hoc reporting for small volumes of data.

Prerequisites

You have defined selection settings (such as the selection ID) in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Generic Ad Hoc Reporting* > *Edit Basic Settings for Data Sources*.

> **Note**
>
> For more information about selecting data, see the documentation in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Generic Ad Hoc Reporting* > *Edit Basic Settings for Data Sources*.

Activities

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Generic Ad Hoc Reporting* > *Ad Hoc Reporting*.
2. Enter the application of the Data Processing Framework, the group ID, and the selection ID for retrieving data. You have to enter the key date. The system uses this date to select data for the key date in the worklist and to select data when using the secondary data source.

- If you do not enter a system date/time, the system enters the current system date/time in the first system date/time field.
- If you enter two system dates/times, the system starts a delta query. The use of the key date depends on how the *Worklist* primary data source was configured.

> **Note**
>
> If you have entered only a "to" system date/time, the system enters *0* in the first system date/time field and starts a delta query.

3. If required, select a layout for displaying the ALV list.
4. Enter the maximum number of data records that you want to be selected by means of a primary data source of the *Worklist* category (for example, InfoSet).
5. Choose *Execute*.

7.4.3 Source Data and Cash Flow: Other Reporting and Export Functions

Bank Analyzer contains the following reporting and export functions:

- Processing framework [page 1651]
- Processing rule [page 1661]
- Reporting in Bank Analyzer [page 1662]
- SDL data export [page 1682]
- Cash flow export [page 1698]

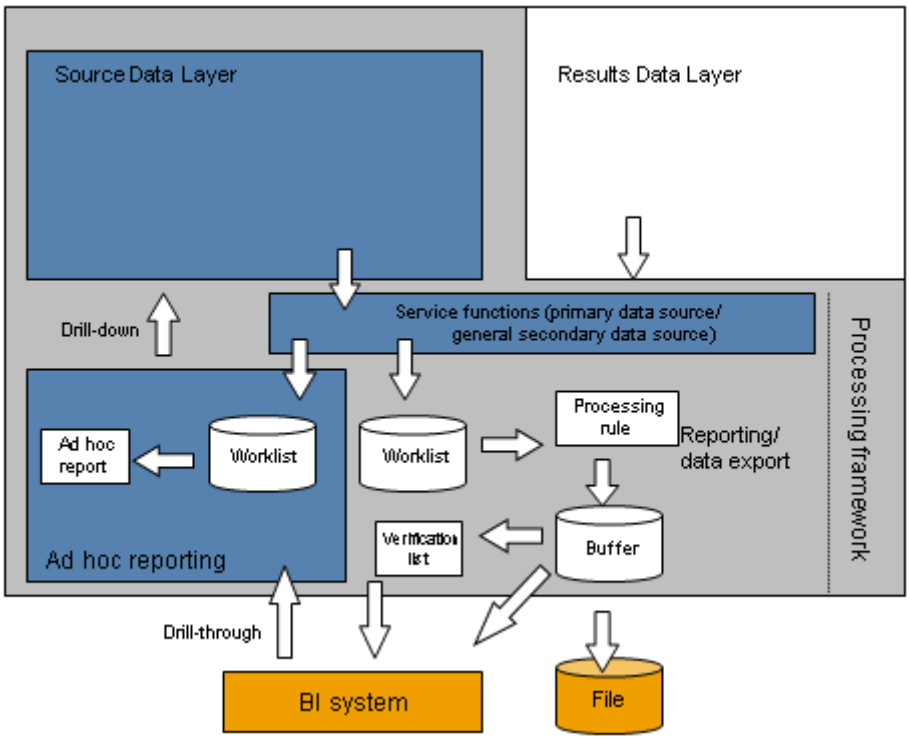
These functions allow you to extract data from across the primary objects and from Analyzer result values for reporting and for exporting purposes. Since the extraction usually involves large amounts of data, batch processing is used.

The process is as follows:

- The worklist is extracted using data sources
- The system calls a processing rule
- The extracted data is buffered for reporting and exporting purposes

In ad hoc reporting, data from all objects is listed in a report that is based on the SAP List Viewer (ALV).

The following figure provides an overview of the reporting and data export functions:



7.4.3.1 Processing Framework

Use

This function uses suitable selection conditions from data sources [page 1655] of various categories to create a worklist. Each single record in the worklist can also be modified by using processing rules [page 1661]. The overall processing results are forwarded to a buffer (temporary storage). Here, verification lists [page 1658] can be used to check whether:

Results are plausible from a business perspective

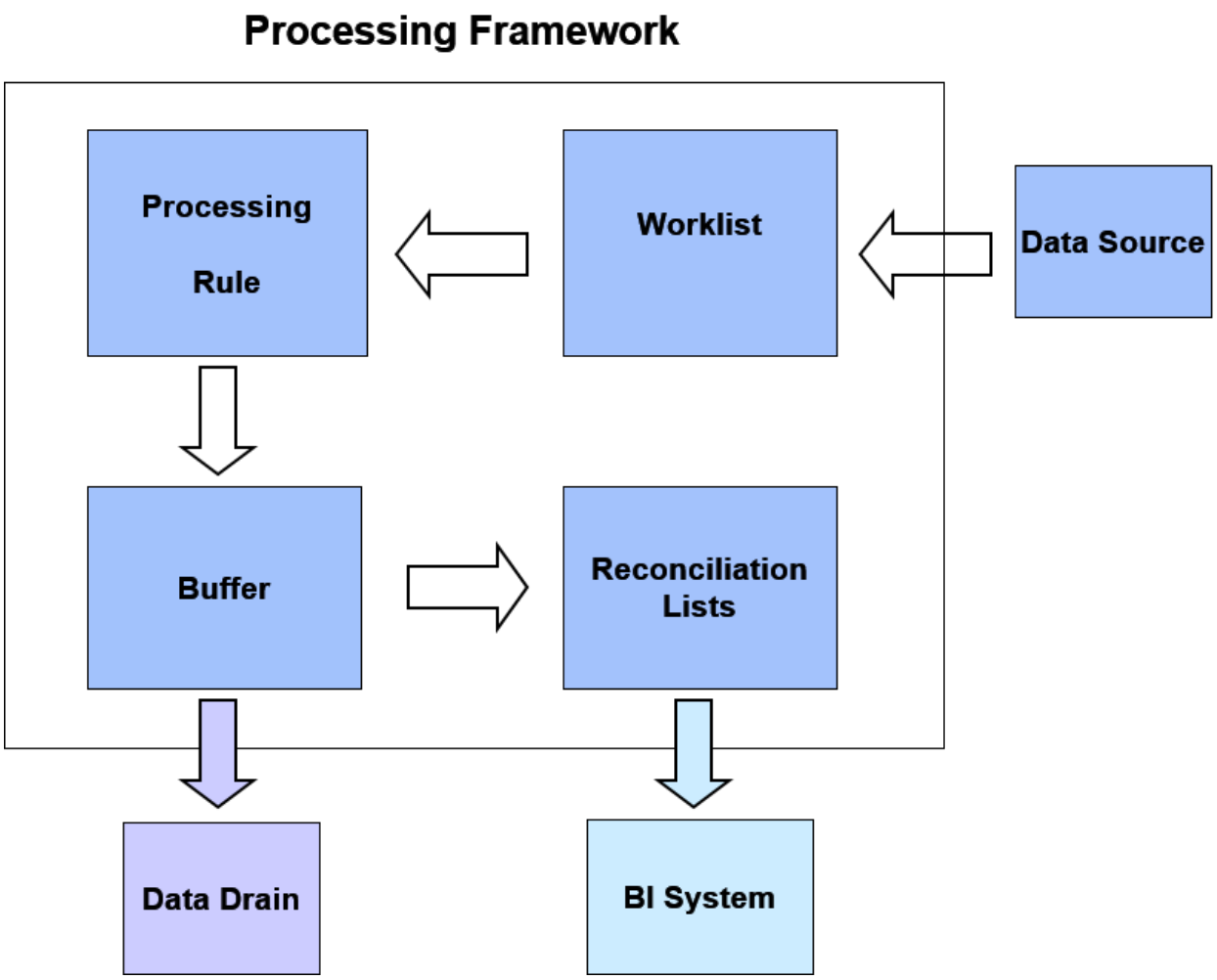
The processing results match the data that you transferred to the Source Data Layer earlier

It is then possible to update the extracted and modified data in a data drain [page 1655] (this processing step is possible for SDL data exports only).

If you require a specific report, you can generate a DataSource from the verification list and transfer it to the Reporting BI system.

The processing framework features a status management function. Here, the current processing statuses and logs of processing runs that have been completed can be displayed. The processing steps can be manually started from here. Normally, however, the processing steps are executed automatically by the Schedule Manager [page 437] as part of the processing activities at the end of the month.

The following figure illustrates the process in the processing framework:



Prerequisites

In Customizing, you have defined a field catalog for the processing framework containing all the key figures and characteristics you can use in the processing framework. You can include any of the characteristics, key figures, and complex key figures available in the Source Data Layer as fields in the catalog. The field catalog you define will represent the maximum range of characteristics, key figures, and complex key figures that you can use for SDL Reporting [page 1663] and SDL Data Export [page 1682]. In the case of cash flow exports [page 1698], the fields of the cash flows that are to be exported are supplied automatically in the processing framework.

You define the field catalog for SDL reporting in Customizing for *Bank Analyzer* under *Infrastructure* → *Extraction and Reporting Services* → *Source Data and Cash Flows: Additional and Alternative Settings* → *Reporting* →*SDL Reporting* → *Edit Field Catalog for SDL Reporting*.

You define the field catalog for SDL data exports in Customizing for *Bank Analyzer* under *Infrastructure* → *Extraction and Reporting Services* → *Source Data and Cash Flows: Additional and Alternative Settings* → *Data Export* →*SDL Data Export* → *Edit Field Catalog for SDL Data Export*.

You define the field catalog for cash flow exports in Customizing for *Bank Analyzer* under *Infrastructure* → *Extraction and Reporting Services* → *Source Data and Cash Flows: Additional and Alternative Settings* → *Data Export* → *Export of Cash Flows* → *Edit Field Catalog for Cash Flow Export*.

You make the settings for the SDL reporting processing framework in Customizing for *Bank Analyzer* under *Infrastructure* → *Extraction and Reporting Services* → *Source Data and Cash Flows: Additional and Alternative Settings* → *Reporting* →*SDL Reporting* → *Set Up Processing Framework for SDL Reporting*.

You make the settings for the SDL data export processing framework in Customizing for *Bank Analyzer* under *Infrastructure* → *Extraction and Reporting Services* → *Source Data and Cash Flows: Additional and Alternative Settings* → *Data Export* →*SDL Data Export* → *Set Up Processing Framework for SDL Data Export*.

You make the settings for the cash flow export processing framework in Customizing for *Bank Analyzer* under *Infrastructure* → *Extraction and Reporting Services* → *Source Data and Cash Flows: Additional and Alternative Settings* → *Data Export* →*Export of Cash Flows* → *Define Processing Framework for Cash Flow Export*.

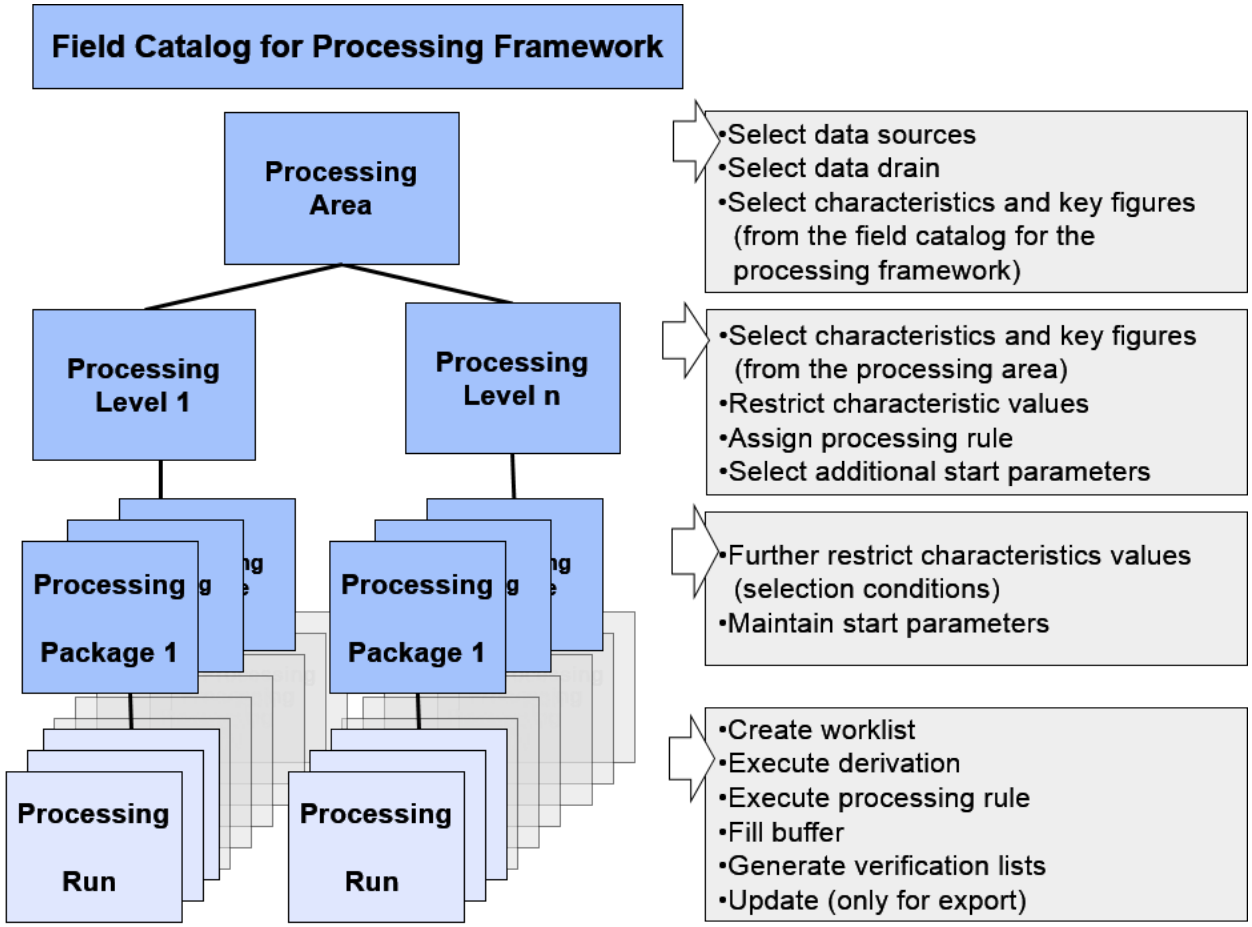
Features

The architecture of the processing framework is subdivided into:

Processing Area [page 1654]

Processing Level [page 1656]

Processing Package [page 1658]



7.4.3.1.1 Processing Area

The processing area is the highest tier in the processing framework [page 1651] architecture. The processing area establishes the link to the data sources in SDL reporting and in SDL data exports.

These data sources [page 1655] supply the data that is to be extracted as well as the data that is to be processed by a processing rule [page 1661].

In SDL data exports, the processing area also establishes the link to the data drain [page 1655] in which the extracted data can be updated.

The characteristics and key figures required for processing and selecting data are also chosen in a processing area. The characteristics and key figures contained in the field catalog that you edit in Customizing for SDL reporting and SDL data exports are available.

You can use different processing areas to group together processing procedures for reporting or for data exports in the individual processing areas. The processing procedures for a processing area use the same data sources and data drains and require the same characteristics and key figures.

You can create various processing areas and give them a business perspective in addition to the technical perspective. This enables you to distinguish the processing areas easily. Separate processing areas are used in the

SDL reporting and SDL data export applications, where it is particularly useful to distinguish processing areas according to their business matter.

In SDL reporting and SDL data export, processing areas are defined by their technical and descriptive names, data sources, data drains (in FDB data export only), and the selection of key figures and characteristics. Once a processing area has been activated, its settings cannot be changed. The system settings and the results produced using them can thus be reproduced. A processing area can contain any number of processing levels.

7.4.3.1.1.1 Data Sources

The data records that were selected in the processing framework [page 1651] on the basis of the start parameters are read from a data source. The processing framework supplies these data records to a processing rule [page 1661] as initial data. In other words, the output parameters of the data sources are the input parameters for the processing rule. The data records contain data from primary objects and Analyzer results data.

When processing areas are processed, a distinction is made between:

- Primary data sources [page 1593]
- General secondary data sources (only for SDL reporting and SDL data export)

A primary data source for selecting the worklist and several secondary data sources for reading other data can be assigned to a processing area.

7.4.3.1.1.2 Data Drains

Use

The result records from processing are stored in the processing framework [page 1651] in a buffer, from where they can be transferred to a data drain. In the processing framework, there are two data drain categories for SDL data exports: category 01 (table), and category 02 (file). Only data drains of category 02 (file) can be used for cash flow exports.

Features

Category 01 (Table)

This category allows data to be written to any table in the ABAP Dictionary. The table in question must be activated and meet the following requirements:

- Data is written only to table fields with the same technical name as defined in the field catalog of the processing framework.
- From a technical perspective, the fields must be of the same type and length.
- If a quantity field or amount field is involved, assign the appropriate unit field or currency field when you define the table. This is the field that was assigned to the key figure of the same name in the field catalog of the processing framework.

When results data is written to a table, this does not affect the data already in the table.

Category 02 (File)

This category allows data to be written to any file. In addition to the logical file name, you have to enter a structure from the ABAP dictionary, which is used when writing data to the file.

> **Note**
>
> We deliver the following predefined structure for cash flow exports: /BA1/FP_STR_CFE.

The structure in question must be activated and meet the following requirements:

- Data can only be written to files with the same technical name as defined in the field catalog of the processing framework. Fields with a different name will be ignored.
- From a technical perspective, the fields must be of the same type and length.
- If a quantity field or amount field is involved, the appropriate unit field or currency field should be assigned when you define the structure. This is the field that was assigned to the key figure of the same name in the field catalog of the processing framework.

7.4.3.1.2 Processing Level

Use

A. Processing Level for SDL Reporting and SDL Data Export

A processing level is the second tier down in the processing framework architecture. In the processing area [page 1654], data is extracted from all objects and structured in a worklist. A processing level, by contrast, enables the extracted data to be modified using a processing rule. You assign a processing rule [page 1661] to the

to the processing level and select suitable characteristics and key figures from the pool in the processing area. You select characteristic values to restrict the scope of the worklist, for example, you can restrict it to a specific business environment.

Even if you do not wish to make any modifications, you still have to use processing rules, because they transfer the extracted data to the buffer.

Processing levels restrict the total number of characteristics and key figures provided by the processing area to a subset that is relevant for a particular business matter. You can also restrict (select) the characteristics in the processing level to further reduce the data to which the processing rule refers.

> **Note**
>
> You can restrict the values for a particular characteristic in the processing level or in the processing packages [page 1658] assigned to the level. It is not possible to select values for the same characteristic in the processing level and in the processing package.

If you restrict the characteristics in the processing level to particular values, the restriction applies to all processing packages that are assigned to this processing level.

B. Processing Level for Cash Flow Export

A processing level is the second tier down in the processing framework architecture. In the processing area [page 1654], data is extracted from all objects and structured in a worklist. In cash flow exports, you can select only one

processing rule [page 1661] for generating and extracting the cash flows for financial transactions and financial instruments from the worklist on the basis of the start parameters.

> **Note**
>
> When the processing rule extracts cash flows, a separate data record is generated for each cash flow item.

In a processing level, you have to select the relevant characteristics for cash flow exports. You can also restrict (select) the characteristics in the processing level to further reduce the data to which the processing rule refers.

> **Note**
>
> You can restrict the values for a particular characteristic in the processing level or in the processing packages [page 1658] assigned to the level. It is not possible to select values for the same characteristic in the processing level and in the processing package.

If you restrict the characteristics in the processing level to particular values, the restriction applies to all processing packages that are assigned to this processing level.

Features

A. Processing Level for SDL Reporting and SDL Data Export

Processing levels are defined by selecting characteristics and key figures from the pool in the processing area to which the processing levels are assigned, and by selecting a processing rule.

In addition, you specify the following information in the processing level for each characteristic:

- Whether the selection is to be restricted to particular characteristic values. Alternatively, a restriction can be made in a processing package.
- Whether the characteristic is to be used as the start parameter [page 1660] in addition to the standard default parameters when a run is created.

You can assign any number of processing packages to a processing level.

B. Processing Level for Cash Flow Export

Processing levels are defined by selecting characteristics and key figures from the pool in the processing area to which the processing levels are assigned, and by selecting a processing rule.

In the processing level, you also define for every characteristic whether the selection is to be restricted to particular characteristics. Alternatively, a restriction can be made in a processing package.

You can assign any number of processing packages to a processing level.

7.4.3.1.3 Processing Package

Use

The lowest tier in the processing framework [page 1651] architecture. In contrast to a processing level [page 1656], where you make all of the settings relevant to your business processes, you use processing packages to actually process data. You can assign any number of processing packages to a processing level. A processing package takes the characteristics and key figures selected in a processing level, and also allows you to define restrictions for characteristics whose values you did not restrict in the processing level. The start parameters [page 1660] are recorded in the processing package.

A processing package is the unit on which processing is performed. By restricting characteristic values, you can bundle data records into packages for processing that are optimized from a business or technical perspective.

Example

You have set up a processing level to extract a specific type of financial transactions, which are supplied from the source systems at different times. You now set up a processing package for each source system. This means you can start processing the data records at a set time as soon as the source system has supplied the financial transaction. The data volume is much easier to manage and the individual data volumes can be reconciled.

Features

A processing package requires a name, as well as information on any restrictions (selection) you defined for characteristics at the corresponding processing level.

You also define for every characteristic in a processing package whether the selection is to be restricted to particular characteristics. Alternatively, a restriction can be made in a processing level.

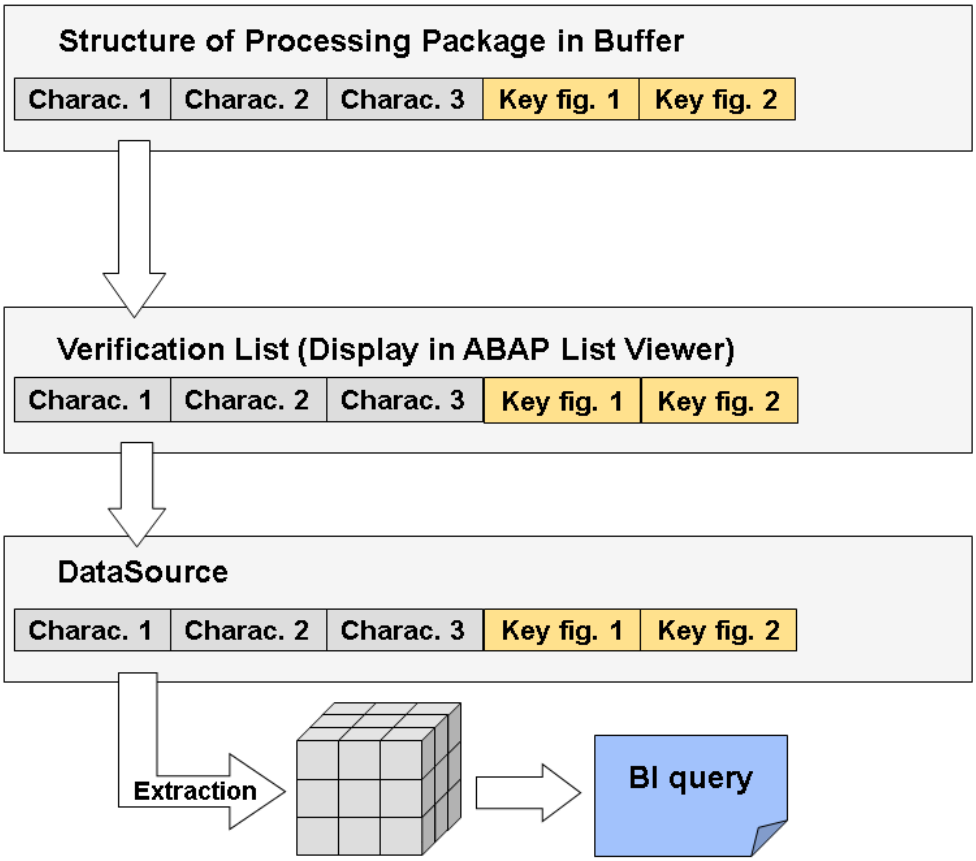
7.4.3.1.4 Verification Lists

Use

You use verification lists in the SDL reporting and SDL export processes. Verification lists show all of the buffered characteristics and key figures in a given processing package [page 1658].

Verification lists are used to verify data supplied from the buffer. A verification list always refers to the results of one processing package.

You can generate a DataSource from a verification list and use it to extract the selected data to a Reporting BI system. Other reports are available in BI, such as a comparison of current data with historical data. Verification lists are displayed using the SAP List Viewer (ALV).



Prerequisites

You have made settings for the processing framework [page 1651]. This means that you have created processing areas [page 1654], processing levels [page 1656], and processing packages [page 1658]. You have carried out processing rules [page 1661] to fill the buffer with results.

You define verification lists for SDL reporting in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *SDL Reporting* > *Verification Lists* > *Set Up Verification Lists*.

You define verification lists for SDL data exports in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *SDL Data Export* > *Verification Lists* > *Set Up Verification Lists*.

You define verification lists for cash flow exports in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *Export of Cash Flows* > *Verification Lists* > *Set Up Verification Lists*.

7.4.3.1.5 Start Parameters

Use

You can define start parameters at runtime for processing a processing package [page 1658]. These parameters are integrated in the processing framework [page 1651] by means of a start ID. A start ID must be selected for every processing step.

A processing package can have several start IDs. This is the case, for example, if a new start parameter is required on the last day of each month to ensure that all the old runs can be carried out with the old start parameters. The default start parameter for SDL reporting and SDL data exports is the *Business Record Date*. The default start parameters for cash flow exports are: *Cash Flow Generation*, *Generation Key Date*, *Time Stamp (Date/Time)*, *Market Data Area*.

Prerequisites

You have selected the start parameter as a characteristic in Customizing for the processing area [page 1654]:

- For SDL reporting under *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *SDL Reporting* > *Set Up Processing Framework for SDL Reporting*.
- For SDL data exports under *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *SDL Data Export* > *Set Up Processing Framework for SDL Data Export*.

> **Note**
>
> The system defaults the start parameters automatically for cash flow exports.

Furthermore, in Customizing for SDL reporting and SDL data exports you can define in the selection criteria for a processing level [page 1656] which additional selection characteristics you want to be supplied via the start parameters. In other words, you can use other start parameters in addition to the default parameter.

You can create a start ID for the start parameters for a processing package in two ways:

- By creating/changing a start parameter in Customizing for the processing framework: In the processing package in the “Start Parameter View”.
- By copying an existing start ID in the *Copy Start ID* transaction in the SAP Easy Access menu. This transaction creates a new run simultaneously. Only the default start parameter can be changed in this case.

7.4.3.2 Processing Rule

Use

A. Processing Rule for SDL Reporting and SDL Data Export

This function is assigned to processing levels [page 1656] and is thus part of the

processing framework [page 1651], which is used for SDL reporting and SDL data exports. You always have to assign a processing rule to a processing level. Only “calculation” processing rules are available. You can define these processing rules as modules in the module editor. Modules consist of an ordered sequence of processing steps.

A module transforms data delivered to its input interface (input structure) and provides its output interface (output structure) with the results. The processing framework calls the processing rule for single records. It transfers one record at a time from the worklist to the module’s input interface and is provided with the results via the output interface.

The processing rule is used to modify extracted data. If you do not want to modify the extracted data, it can be transferred without being changed. The “field values” step sequence is to be used for transferring data.

B. Processing Rule for Cash Flow Export

This function is assigned to processing levels [page 1656] and is thus part of the

processing framework [page 1651], which is used for SDL reporting and SDL data exports. You always have to assign a processing rule to a processing level. In a cash flow export, there is only one processing rule of the category “Cash Flow Export” available for generating and extracting the cash flow for a financial transaction or financial instrument on the basis of the start parameters specified.

A separate data record is generated for every item; then returned to the processing framework, and stored in the buffer.

Prerequisites

A. Processing Rule for SDL Reporting and SDL Data Export

- You define processing rules for SDL reporting in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *SDL Reporting* > *Configuration of Module Editor* > *Edit Modules for SDL Reporting*.
- You define processing rules for SDL data exports in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *Configuration of Module Editor* > *Edit Modules for Source Data Export*.
- You assign modules for SDL reporting to a processing level in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *SDL Reporting* > *Set Up Processing Framework for SDL Reporting*.
- You assign modules for SDL data exports to a processing level in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *Set Up Processing Framework for SDL Data Export*.

Features

A. Processing Rule for SDL Reporting and SDL Data Export

Features

A module can comprise elementary or complex processing steps. Processing steps that have been predefined in modules are called up by the processing framework. The following processing steps are provided:

- Condition: You can define a condition (if). This means that a particular step will be executed only if clearly defined conditions are met by particular input or output components.
- DO loop: This processing step provides a control structure (in addition to the table loop) for loop processing. Using the DO loop, you can create any loops independently of internal tables. Note that you can only leave the loop by using an explicit EXIT statement.
- Exit: Depending on the context, you use the EXIT statement to leave a module, a loop, or a DO loop.
- Field values: This processing step is used to assign other components (input or output) or fixed values to output components.
- Formula: A formula uses ABAP arithmetic operations to link the elements of the module environment to new values. You can use the basic arithmetic operations (+, -, *, /) as well as other functions such as "to the power of" (**), square root (SQRT), exponential function (EXP), and logarithm (LOG).
- Function module: This processing step enables you to include retrieval and calculation functions that are depicted as function modules in the structure of a calculation module. The interface for the function modules must be supplied with data by the module environment.
- Comment: You can create processing steps that consist only of comment lines.
- Message: The system can output error, warning, and information messages. You can use control parameters to define when processing, or the processing of the data record, is to be terminated and when processing is to be classed as correct.
- Internal table: The system provides processing steps for internal tables. You can use these steps to initialize internal tables (REFRESH) and append a line to an internal table (APPEND). You can also determine the number of entries in an internal table (COUNT).
- Table loop: This processing step enables you to process an internal table. You can insert other processing steps within the "Loop–End Loop" step. A WHERE condition can be used to restrict the number of rows processed.

7.4.3.3 Reporting in Bank Analyzer

Use

The purpose of reporting in Bank Analyzer is to provide functions for cross-object reporting based on data from the primary objects and associated Analyzer results.

Features

Various functions are available for different purposes:

- SDL reporting: Using the processing framework of SDL reporting, you can extract data from all primary objects and from associated Analyzer results and transfer it to a buffer. Reporting functions (verification lists) are available for the data in the buffer. You can also transfer the extracted data to the BI system for additional reporting.
- Ad hoc reporting: You can use this function to display data from the primary objects and Analyzer results "ad hoc" by means of cross-object reports in SAP List Viewer (ALV). In particular, you can use ad hoc reporting as a general access screen for the correction of primary objects. You can access the display and processing modes for the relevant primary objects from the individual characteristics and key figures of the report. Furthermore, a drill-through can be performed from queries in the Bank Analyzer Reporting BI system to ad hoc reporting.

7.4.3.3.1 SDL Reporting

Use

SDL reporting allows global extraction of the following data:

- Primary objects
- Analyzer result values

This is for reporting purposes. You can use SDL reporting to extract large volumes of data.

> **Note**
>
> You can optimize the timing of data extraction by splitting up the processing into packages to be executed simultaneously or at different times. As well as this, you can increase efficiency within processes by dividing them into several background jobs. You make parallel processing settings in Customizing for *Bank Analyzer* under *Basic Settings* > *Parallel Processing* using the application type *PA (Processing Framework Applications)*.

Features

The procedure is as follows:

1. Extract the worklist via data sources
2. The system calls a processing rule
3. Buffer the extracted data

Further processing of the buffer data depends on the reporting objective. You can use the following functions:

- Generate verification lists (lists from the SAP List Viewer)

- Generate a DataSource for the BI system to export the data from the verification list or buffer to the BI system. Data is available in the BI system for additional reporting.

Activities

Formatting the Data and Associated Settings

The processing framework features a **status management** function. Here, you can display the current processing statuses and logs of processing runs that have been completed. The processing steps can be manually started from here. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Reporting* > *SDL Reporting* > *Status Management* > *Call Status Management*.

Alternatively, you can schedule the processing steps as jobs in **Schedule Manager** so that processing runs automatically at periodic intervals.

For more information, see SDL Reporting Process [page 1665].

Generation of Verification Lists

The **results** of a processing run are placed in the buffer. The buffer data is displayed in the **verification list** using an ALV report. In addition to the standard ALV functions, specific functions are provided for SDL reporting. Data is displayed for each processing package.

Interaction options are available in the verification list, both when you specify the selection condition and in the ALV display.

On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Reporting* > *SDL Reporting* > *Processing* > *Evaluate Verification Lists*. Alternatively, verification lists can be displayed from the status management for processes which are in the verification phase.

> **Note**
>
> The structure of a verification list is either the same as the one in the buffer or it is a user-defined structure. If the latter is the case, you can remove or add buffer fields. You make the settings in Customizing for *Bank Analyzer* under:

- *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *SDL Reporting* > *Verification Lists* > *Define Output Structures for Verification Lists*
- *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *SDL Reporting* > *Verification Lists* > *Set Up Verification Lists*

After you have selected the processing area, the processing level, the processing package, and, if required, the start parameter and process, the system displays all of the verification lists that have already been set up for this combination. When you select a verification list, all the fields it contains become available for selection. You can formulate a selection condition using these fields.

You can save the selection conditions. The system displays the variants below the selected verification list.

As well as the selection conditions, you can create top or bottom reports. To do this, select a key figure and choose *Show First N* or *Show Last N*. Enter a number for *N*. If you now execute the report, the system displays only those

data records that - in addition to the other selection criteria - have the highest or lowest values for the selected key figure. The number of data records is restricted to the number input.

You cannot save the top and bottom conditions as a variant. You can only ever use one key figure for defining top and bottom conditions.

Generating a DataSource for the BI System

You can generate a DataSource for a complete processing package or for the entire buffer. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Reporting* > *SDL Reporting* > *Processing* > *Extract Processing Package* .Select a processing package (even if you want to extract the entire buffer, in other words, even if specifying the processing area would suffice). On the following screen, select whether you want the system to extract the package only, or the entire buffer. In both cases, the system generates a DataSource. You can also generate a DataSource from within the verification list.

In addition to the standard ALV functions in the verification list, the selected data can also be made available as a DataSource in the buffer structure. You can choose the name of the DataSource. The system places the prefix *1_BA_BA_PA* in front of the name you enter. The DataSource is assigned to the *0VERIFICATION-DATA* application component.

7.4.3.3.1.1 SDL Reporting Process

Use

The SDL reporting process in the processing framework [page 1651] takes the form of a structured sequence of logically related processing steps, and applies to one processing package [page 1658]. The individual processing steps can be controlled manually, using the status management [page 1676] facility, or with the Schedule Manager [page 437].

The processing steps in SDL reporting are carried out in the processing framework using the following transactions:

- Copying the Start ID
- Create New Run
- Release Run
- Worklist Creation
- Execute Processing
- Evaluate Verification List

These transactions are part of the processing framework and are called manually or from the status management facility. They can also be scheduled as jobs in the Schedule Manager to automate periodic processing.

Prerequisites

- You have created and activated a processing area, processing level, and processing package with start parameters in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* >

Source Data and Cash Flows: Additional and Alternative Settings > *Reporting* > *SDL Reporting* > *Set Up Processing Framework for SDL Reporting*. This involves, in particular, the selection of the following parameters:
    - Data source(s)
    - Processing rule
    - Characteristics and Key Figures
- You have made settings in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *SDL Reporting* > *Configuration of Module Editor* > *Edit Modules for Source Data Reporting*. In this activity, you defined the processing rule used in the processing framework.

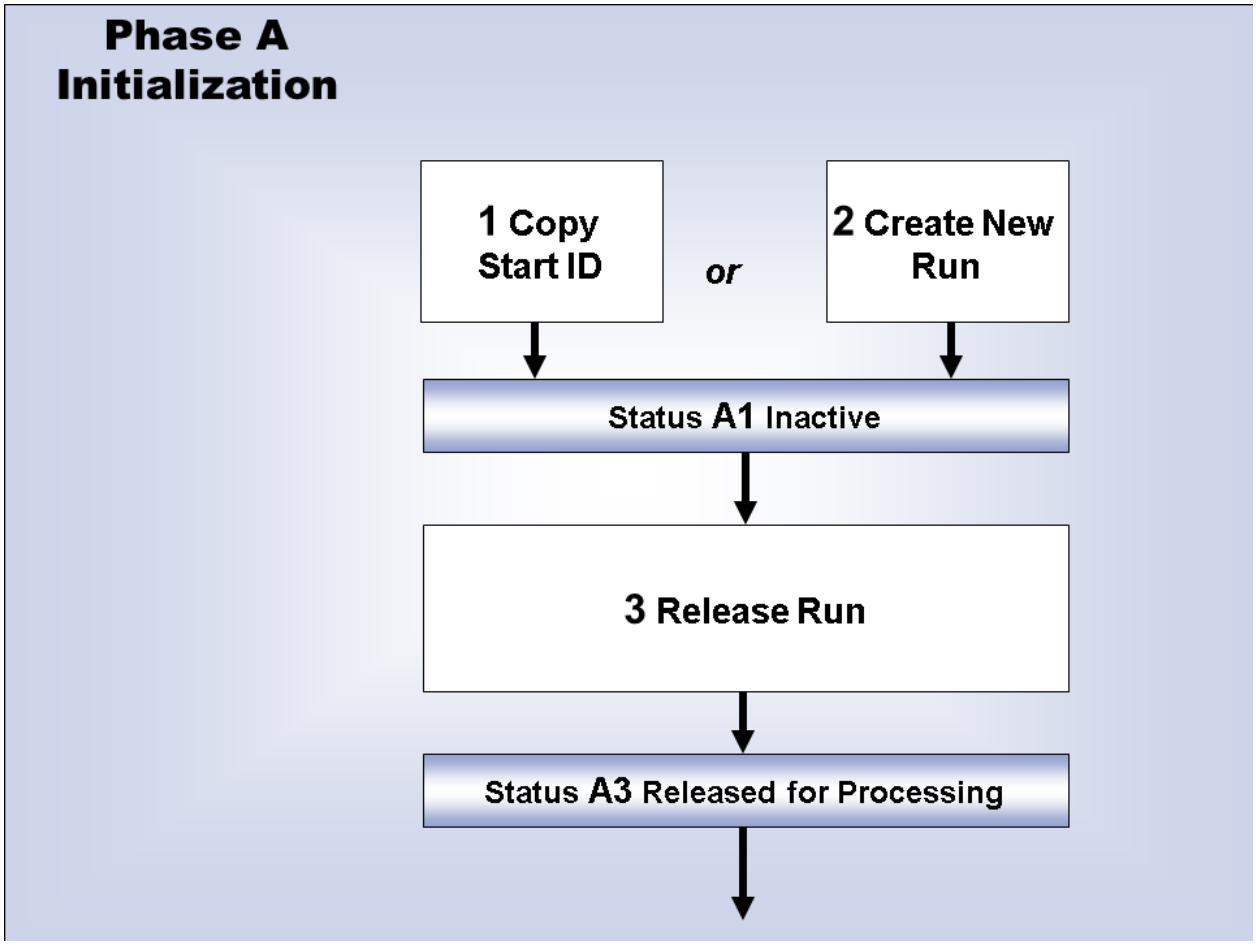
Process

Phases and statuses of a processing package in the SDL reporting process:

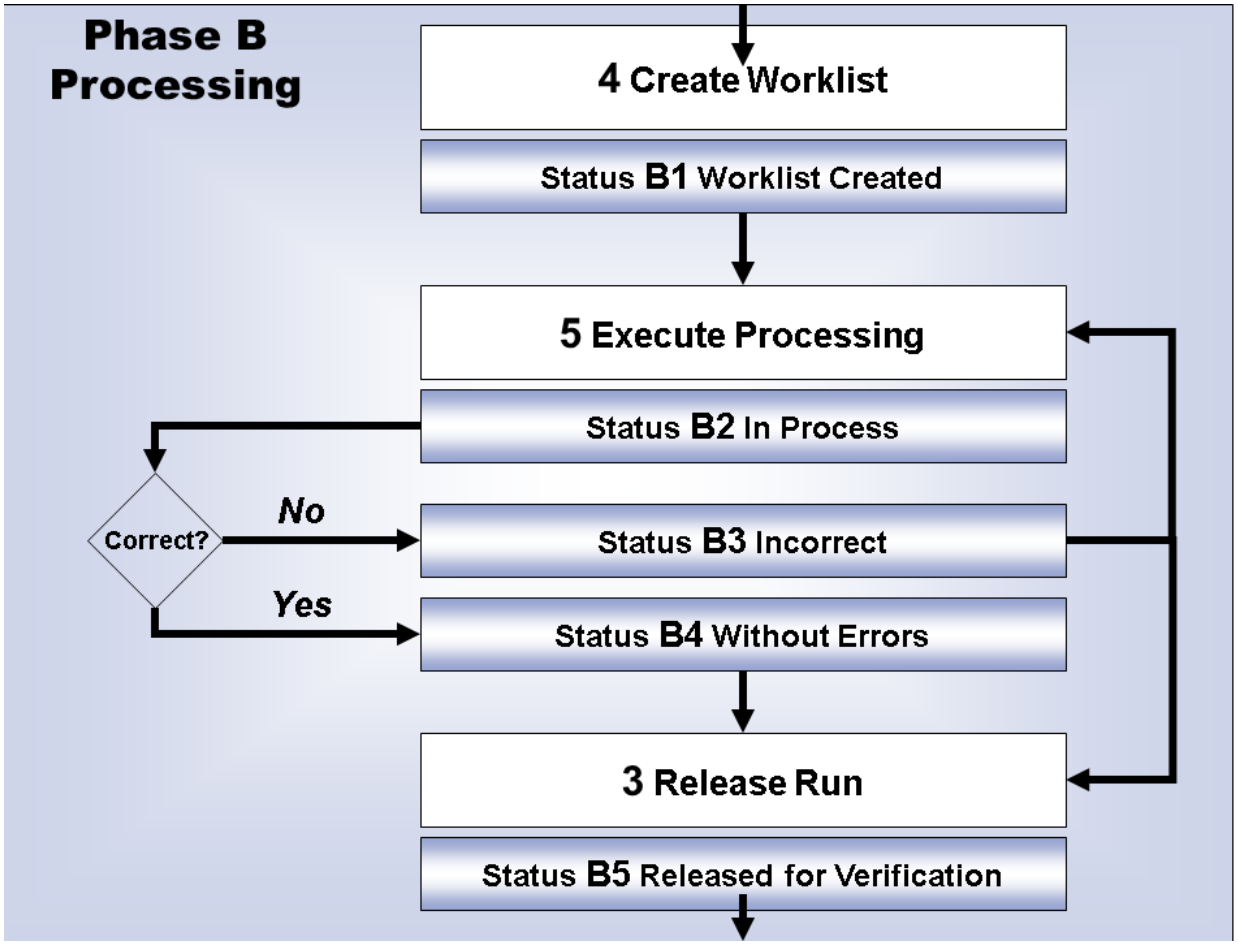
- Phase A, initialization: create and release run
- Phase B, processing: read and process the data, and save the data in the buffer
- Phase C, verification: provide data for verification

The following figures outline the processing phases and statuses:

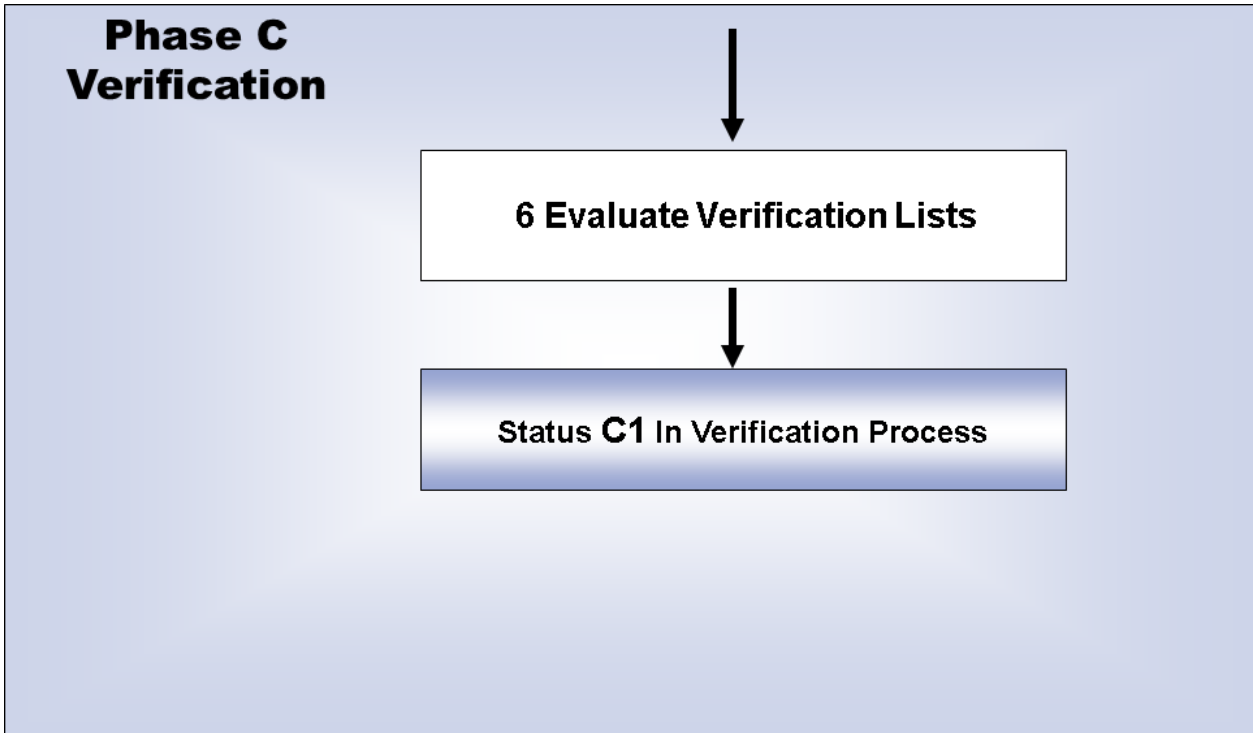
Phase A: Initialization: Processing is initialized by calling the *Copy Start ID* or *Create New Run* program followed by *Release Run*.

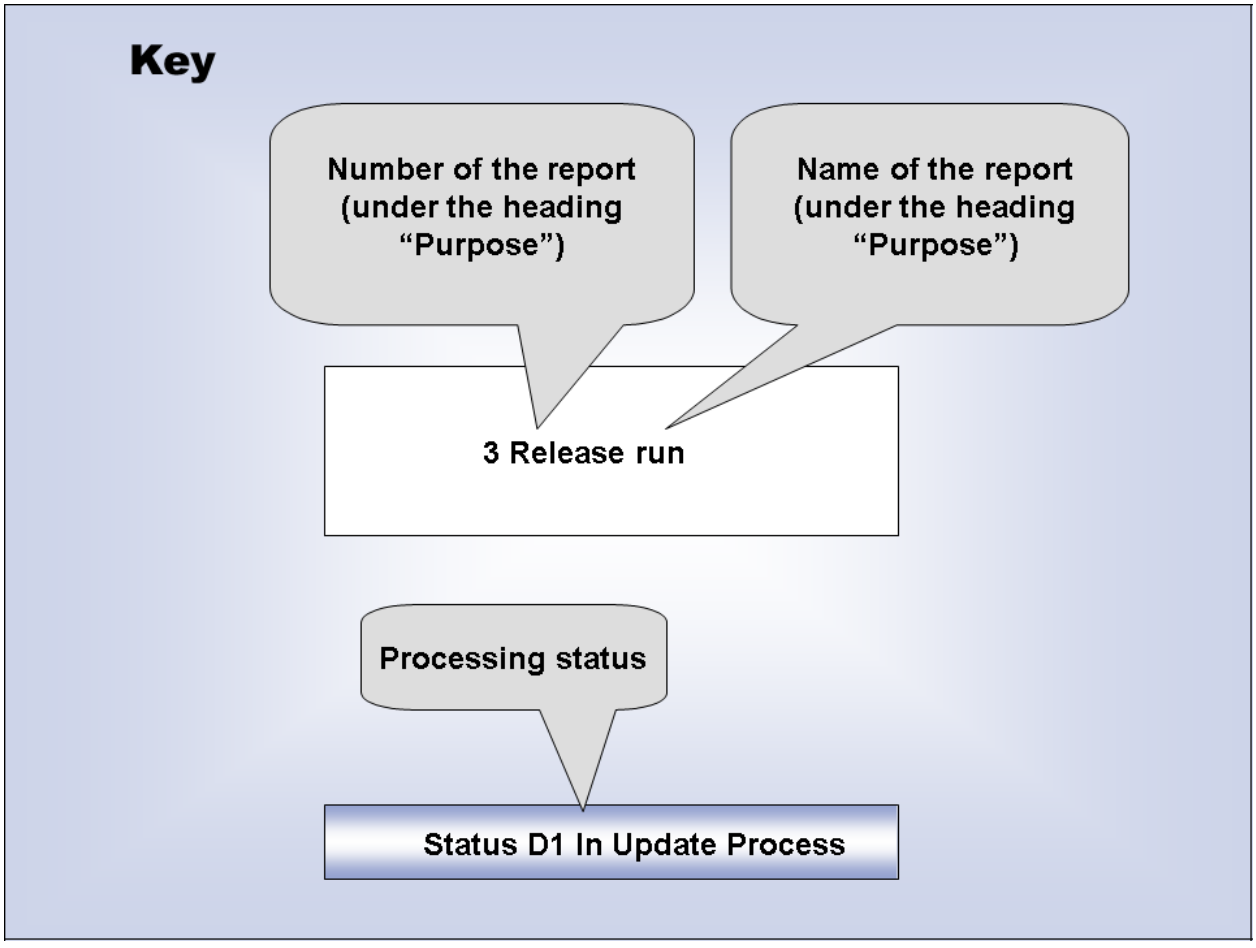


Phase B: Processing: Once the run has been released in phase A, the system creates the worklist and starts processing. Incorrect as well as correct runs can be released for verification. The processing phase can be triggered again for incorrect runs (after correction, for example in the processing rules).



Phase C: Verification: The results are checked using **verification lists**. Processing always ends with status C1 (in verification process). This happens because the results records cannot be updated. You can correct the results in the Source Data Layer, for instance, or you can correct the processing rules, and then create a new run. The data from the incorrect run remains in the temporary buffer.


Phase C
Verification
6 Evaluate Verification Lists
Status C1 In Verification Process


Key
Number of the report (under the heading "Purpose")
Name of the report (under the heading "Purpose")
3 Release run
Processing status
Status D1 In Update Process

Note that the statuses that are not generated by calling a program (B3 and B4) are set by the system. While the statuses in phases A and B can be scheduled as jobs in the Schedule Manager and thereby **automated**, verifying the results in phase C must take place **manually** using verification lists. Phase C can be automated using a workflow.

Periodic Processing

The procedure described above applies to **one** processing package. Multiple processing areas, levels, and packages are normally used. This means that the programs listed above have to be executed for each package and start ID.

Processing for a processing package can be done automatically by bundling the individual steps to form processes in the SAP Schedule Manager. The individual processes can be included in general processing activities at the end of the month. In this case, variants have to be created for the programs for each processing package and start parameter.

You can use the *Copy Start ID* program to copy existing start parameters. The business record date for the new start parameters can be set by the user. Variants for the existing start parameters can be adjusted automatically to suit the new start parameters. This enables you to use the process that has already been defined for processing a processing package on any number of dates. The *Copy Start ID* program can also be scheduled at the beginning of a new processing run for each processing package, so that a recurring, periodic processing run is set up only once and can be used in other periods.

7.4.3.3.1.1.1 Copying the Start ID

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *SDL Reporting* > *Set Up Processing Framework for SDL Reporting*.
- You have already created a start ID for the processing package.

Context

This program creates a new start ID by copying one that already exists for a processing package. It reads all the start parameters [page 1660] of the start ID that you want to copy. You can choose any name and business record date for the new start ID. This program is the first step (if you do not create a processing run directly) in the initialization phase of the SDL reporting processing procedure.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Reporting* > *SDL Reporting* > *Processing* > *Copy Start ID*.
2. Select a processing area, processing level, and processing package.
3. Select the start ID you want to copy using input help.
4. Enter a name for the new start ID you are creating.
5. Enter a business record date.
6. Choose *Execute*.

Results

The new start ID has been generated.

7.4.3.3.1.1.2 Create New Run

Prerequisites

You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *SDL Reporting* > *Set Up Processing Framework for SDL Reporting*.

Context

This transaction generates a new run for a processing package [page 1658] and a start ID with associated start parameters [page 1660]. This transaction is the first step (if you have not already called this step automatically by copying a start ID) in the initialization phase of the processing process for SDL reporting.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Reporting* > *SDL Reporting* > *Processing* > *Create New Run*.

2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Choose *Execute*.

Results

The system has created a new processing run for the start ID and associated processing package that you selected.

7.4.3.3.1.1.3 Release Run

Use

You can perform the following releases in this transaction:

- Release for editing: from status A1 to status A3 (released for editing)
- Release for verification: From status B4 to status B5 (released for verification)

You use this program for the following actions:

- Release a processing run defined in the *Create New Run* transaction
- Release for verification the results data determined in the *Execute Processing* transaction

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *SDL Reporting* > *Set Up Processing Framework for SDL Reporting*.
- In the *Create New Run* transaction, you have created a processing run for the processing package defined in Customizing.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Reporting* > *SDL Reporting* > *Processing* > *Release Run*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.

4. Select the sequential number of the run you want to release.
5. Choose one of the following phases:

· Choose *Initialization* if you want to release a newly created run.

· Choose *Processing* if you want to release the results data for verification.

7.4.3.3.1.1.4 Create Worklist

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *SDL Reporting* > *Set Up Processing Framework for SDL Reporting*.
- In the *Create New Run* transaction, you have defined a processing run for the processing package defined in Customizing, and released it in the *Release Run* transaction.

Context

This program reads the settings for the processing package and uses them to determine how the initial data for the processing rule is to be selected. The results of this selection are used to generate the worklist. This program is the first processing step in the SDL reporting process.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Reporting* > *SDL Reporting* > *Processing* > *Create Worklist*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Select the sequential number of the run for which you want to create a worklist. The system automatically sets the time stamp.
5. Choose *Execute*.

Results

The system creates the worklist.

7.4.3.3.1.1.5 Execute Processing

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *SDL Reporting* > *Set Up Processing Framework for SDL Reporting*.
- In the *Create New Run* transaction, you have defined a processing run for the processing package defined in Customizing, and released it in the *Release Run* transaction.
- In the *Create Worklist* transaction, you have generated a worklist for the run you have released.

Context

This program reads the data from the worklist, which was compiled from primary data sources [page 1593] and possibly contains additional information from secondary data sources. It then calls the processing rule. The result data produced by the processing rule is stored in the buffer. The additional information is read and the initial data processed in parallel. The block size defined previously is used here. This program is the second processing step in the SDL reporting process.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Reporting* > *SDL Reporting* > *Processing* > *Execute Processing*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Select the sequential number of the run you want to process.
5. Choose *Execute*.

Results

The system performs processing and writes the data it determines to the buffer.

7.4.3.3.1.1.6 Analyze Verification Lists

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *SDL Reporting* > *Set Up Processing Framework for SDL Reporting*.
- In the *Create New Run* transaction, you have defined a processing run for the processing package defined in Customizing, and released it in the *Release Run* transaction.
- In the *Create Worklist* transaction, you have generated a worklist for the run you have released.
- You have carried out the *Execute Processing* transaction.

Context

Verification lists are used to verify data supplied from the buffer. You can also generate a DataSource from a verification list and use it to extract data from the Source Data Layer and transfer it to the Reporting BI system.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Reporting* > *SDL Reporting* > *Processing* > *Evaluate Verification Lists*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Choose *Execute*.
5. On the *Select List* screen, expand the *Verification Lists* folder and select a verification list that you defined in Customizing.
6. Choose *Execute*.
7. The verification list is set up using the characteristics and key figures in the buffer.
8. On the application toolbar, choose *Layout Settings* and *Change Layout*.
9. Select the additional columns you need to adjust your layout.

If you want to generate a DataSource from the verification list, proceed as follows:

10. Choose *Create BI DataSource*.
11. Enter a name for the DataSource.
12. Save your entries.

Results

The system generates a DataSource from the verification list.

7.4.3.3.1.2 Status Management in the SDL Reporting Process

This function enables you to trigger processing [page 1665] for SDL reporting in the processing framework [page 1651] and control all the required processing steps at processing package [page 1658] level using the relevant status.

This means you can use the following transactions for SDL reporting in status management:

- Copy start ID
- Create new run
- Release run
- Create worklist
- Execute processing
- Analyze verification list

7.4.3.3.1.2.1 Calling Status Management

Prerequisites

You have set up and activated a processing area [page 1654], a processing level [page 1656], and a processing package [page 1658]. The data sources are filled with initial data that can be transferred to a processing rule [page 1661].

Context

You can use the status management facility to trigger the processing of a processing run and control the necessary processing phases and status.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Reporting* > *SDL Reporting* > *Status Management* > *Call Status Management*.
2. Select a processing area, processing level, and processing package.
3. Specify a *Key Date From* (the date as of which you want to display processing runs).
4. Choose *Execute*.
5. Expand the file structure. The folders represent the selected processing area, processing level, processing package, and start parameters. Runs that have already been created are displayed below the start parameters.
6. To create a new run, start the appropriate program in the context menu (right mouse click) for the processing package. The new run is displayed below the *Start Parameters* folder.
7. You can call the context menu for a processing package for any run and start the transactions from the SDL reporting process for status management. In the context menu, you can also call the log for the run.

7.4.3.3.2 Ad Hoc Reporting

Use

Ad hoc reporting provides a further reporting function that is not based on the processing framework with extraction of data in the buffer, but rather places reporting data in the SAP List Viewer on an ad hoc basis. Ad hoc reporting allows you to select data from all primary objects and Analyzer results.

> **Note**
>
> You should only use ad hoc reporting for small volumes of data. At runtime, you can enter the maximum number of data records that you want to be selected by means of a primary data source of the "Worklist" category (for example, InfoSet).

> **Note**
>
> It is possible to navigate directly from the characteristics and key figures in the ALV to the processing function for the primary object in question. You can navigate to both display and change mode. Ad hoc reporting also allows you to access the general correction screen.

> **Note**
>
> You can set up an ad hoc report as the destination for a query in the Reporting BI system (that is, you can assign the receiver reports in Bank Analyzer to a query in the Reporting BI system). You must have defined a receiver report for this ad hoc report. You define receiver reports in Customizing for Bank Analyzer under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flow: Additional and Alternative Settings* > *Reporting* > *Ad Hoc Reporting* > *Configure Receiver Reports for Drill-Through*.

Features

You can use ad hoc reporting for a global object search (InfoSet and primary data source). It expands the *Global Object Search* (InfoSet and primary data source) function because the object search does not include only the characteristics in the worklist of the primary data source; it also includes characteristics and key figures of the primary objects that can be identified using global secondary data sources. The characteristic or key figure that is to be changed is selected on the general correction screen. The system automatically displays the relevant processing transaction (such as *Change Financial Transaction*) for the appropriate primary object. This is where the actual correction is made. In accordance with the versioning concept for primary objects, a new version of the primary object concerned is created using the changed data. This means that if you want to make a correction, you do not need to know the primary object in which the characteristic or key figure is located.

Ad hoc reporting is an additional function for making general corrections. It enables the correction screen to contain characteristics and key figures from the object transfer structures (OTS) for the primary objects, in addition to the characteristics from the index tables (*InfoSets*).

You can use ad hoc reports for drill-throughs from reports in the Reporting BI system to the *core Bank Analyzer* system. A drill-through uses report-report interface technology and describes the process of jumping from one cell of a source report to a target report (in this case, an ad hoc report). The information about the selected cell of the source report is transferred to the target report specified as the jump destination, and used to select the relevant data records.

Configuring the Data Source for Ad Hoc Reporting

You identify the data to be reported by selecting any primary data source of the *Worklist* category and then importing the other characteristics and key figures by means of a general secondary data source. At runtime, specify the primary data source of the *Worklist* category that is relevant for an ad hoc report. When you configure the ad hoc report, you define which module of the general secondary data source is used to read further characteristics and key figures. You make this configuration in Customizing under *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *Ad Hoc Reporting* > *Configure Ad Hoc Reports*.

Data that has been selected for the ad hoc report is displayed using a report based on the SAP List Viewer (ALV).

Interaction

It is possible to navigate directly from the characteristics and key figures in the ALV list of the ad hoc report to the processing function for the following primary objects and to the *Position Master Data* component:

- Financial transaction
- Financial instrument
- Account
- Country
- Hedging relationship
- Physical asset
- Position
- Business partner

You can navigate to both display and processing mode.

You can navigate from the *Unified Object Identifier (UOID)* or from the key characteristics of the primary objects mentioned above without making any further settings in Customizing. You can only navigate from the other listed characteristics and key figures to the primary object if you have defined a navigation option for them in

Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *Ad Hoc Reporting* > *Specify Field Navigation Options for Ad Hoc Reporting*. This field navigation option determines which primary object the system navigates to. Furthermore, the report must also contain the UOID or the key of the primary object that it has to navigate to.

In change mode, the following options are also available for navigating to processing mode in the following Analyzers:

- *Profit Analyzer*: manual entry of line items
- *Balance Analyzer*: create correction
- *Balance Analyzer*: change correction

In addition, the *complex key figure* is displayed as a symbol that you can use to navigate to the respective detail display.

You can also use ad hoc reports as the destination of the jump for drill-throughs from reports in the Reporting BI system to the core Bank Analyzer system. To use an ad hoc report as the destination, you first have to define and generate a receiver report in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *Ad Hoc Reporting* > *Configure Receiver Reports for Drill-Through*.

7.4.3.3.2.1 Using Ad Hoc Reporting

Prerequisites

Ad hoc reporting requires the following input parameters:

- Report ID
- InfoSet
- InfoSet profile, if selection criteria have to be assigned for characteristics
- Key date
- Date up to which changes are to be taken into account
- Maximum number of result records for the InfoSet query

A worklist is filled by means of an InfoSet query, and after the data has been read, from secondary data sources. The structure of the worklist is defined by the output fields of the secondary data source. Finally, the worklist data is displayed in an ALV list.

You define the report ID in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flow: Additional and Alternative Settings* > *Reporting* > *Ad Hoc Reporting* > *Configure Ad Hoc Reports*. As well as the report ID, you have to specify the environment and the module for the secondary data source. If you want the system to check the authorization for displaying individual characteristics, you can specify an authorization characteristic profile when you configure the report.

You define the InfoSet in Tool BI [page 16]. An InfoSet comprises one or more index tables. You can define an InfoSet profile for an InfoSet in Customizing for *Bank Analyzer* under *Source Data Layer* > *Source Data Services*

Access to Source Data > *Index Function* > *InfoSets* > *Edit Additional Settings for InfoSets*. An InfoSet profile is valid for one specific InfoSet and contains all the characteristics for which selection criteria must be entered to generate a worklist.

You can navigate directly from the characteristics and key figures in the ALV list to the processing mode for the primary object in question.

To create a relationship between a characteristic or key figure and the corresponding primary object, you have to define a field navigation option in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Reporting* > *Ad Hoc Reporting* > *Specify Field Navigation Options for Ad Hoc Reporting*.

Context

You use ad hoc reporting to compile ad hoc reports on small data volumes in the form of characteristics and key figures without using the processing framework for SDL reporting. In ad hoc reporting, you can navigate from individual characteristics and key figures in the ALV list directly to the processing mode for the relevant primary object.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Reporting* > *Ad Hoc Reporting*.
2. Select a report that has been defined in Customizing.
3. Select an InfoSet. If an InfoSet profile has been assigned to the InfoSet, the system automatically copies it over. If an InfoSet profile has been copied over automatically, edit the selection criteria for the characteristics in the InfoSet profile.
4. Specify the maximum number of hits for the InfoSet query.
5. Enter a *key date*.
6. If necessary, select a *layout* for displaying the ALV list.

   > **Note**
   >
   > If you want to select a layout, you must already have executed the *Ad Hoc Reporting* procedure once, and created a layout.

7. Choose *Execute*.
8. To navigate directly to the display mode of the primary object in question, select a characteristic.
9. If you wish to navigate directly to change mode for the primary object in question when you select a characteristic, choose *Change*. Select the characteristic you want to change and make the corrections.
10. Save your changes.

7.4.3.3.3 Reporting in the BI System

Use

SDL reporting enables DataSources to be generated in order to transfer data from the buffer. Subsequent processing takes place using the standard functions of the BI system.

Features

Drill-Through to Ad Hoc Reporting

You can use ad hoc reporting for drill-throughs from reports in the Reporting BI system to the core Bank Analyzer system. Drill-throughs use report-report interface technology and describe the process of jumping from one cell of a source report to a target report (in this case, an ad hoc report). The information about the selected cell of the source report is transferred to the target report specified as the jump destination, and used to select the relevant data records.

An example of a drill-through to ad hoc reporting can be found in the area of disclosure and reporting requirements. Here, data is extracted to the Reporting BI system from different data sources in the Bank Analyzer core system (Source Data Layer (SDL), Historical Database (HDB), Credit Exposure (CE)). In the Reporting BI system the data is stored at single transaction level (ODS). During this procedure, you can partly modify the data from the single transactions (aggregation, selection). The original information beneath the single transaction level is then in the Bank Analyzer core system only. The Reporting BI system views this as an external system. The system accesses the original data of the Bank Analyzer core system by jumping from a BI report (sender report) via the report-report interface (RRI) of the Reporting BI system to the receiver reports of ad hoc reporting (receiver report) in the Bank Analyzer core system.

To enable the system to drill-through to the components of the SDL, HDB, and CE, you can select the data for ad hoc reporting by selecting any primary data source of category *Worklist* (InfoSet, for example).

Activities

Use of the Report-Report Interface

The jump destinations of a query are defined in the BI system using the report-report interface. This enables navigation from one query to an ABAP report in the core system. Possible jump destinations for a query are assigned as sender/receiver assignments in the required Reporting BI system.

In order to navigate from a query to ad hoc reporting, you have defined and generated a special receiver report in the core Bank Analyzer system. The generated report can be assigned as the jump destination for a query in the report-report interface, using sender/receiver assignments.

Ad hoc reporting cannot be called directly, since the information about the selected cell of the query must first be converted to the input parameters of the ad hoc report. The information about the selected cell of the query is made available to the accessed ABAP report. Since the report that is started in the core system is not known to the BI system, selections are prepared by the BI system in accordance with the loading process. The report-report interface transfers the DataSource fields containing the selections to the core system by RFC connection and

starts the SAP GUI. The assignment is based on inverse transfer rules from the report-report interface. For this purpose, a complete chain must exist from the DataSource of the source system to the InfoSource and the update rules, through to the data target.

The field names of characteristics and key figures used for DataSources generated in Bank Analyzer are taken from the characteristic or key figure repository.

1. If the BI source report is based on a generated data source in Bank Analyzer, the report-report interface transfers the information about the selected cell to the receiver report, using the field names. The generated receiver reports in ad hoc reporting take this into account and convert the transferred field name selections to characteristic names.
2. The system transports the transferred values to the ad hoc report input parameters for the selection and, if required, the time values for key date and time stamp are also transported.
3. The system then calls the ad hoc report. The system transports the transferred information to the selection by means of a primary data source of the category *Worklist* using characteristics with the same names.

> **Note**
>
> Any characteristics that are not contained in the respective primary data source of the *Worklist* category are ignored during the selection. Note this when you select receiver reports in ad hoc reporting as the jump destinations of a query. In this case, only receiver reports in which the selections in the ad hoc report (by means of a primary data source of the *Worklist* category) are largely identical to the characteristics and attributes of the query are suitable. If none of the selections transferred from the report-report interface matches the options for the selection of the receiver report by means of a primary data source of the *Worklist* category, the ad hoc report is started without restrictive selection.

In the case of BI source reports that are not based on a generated data source in Bank Analyzer, you can use receiver reports from ad hoc reporting as the jump destination. However, the following conditions must be met:

• The loading process is executed from the core Bank Analyzer system of the accessed receiver report. More specifically, a complete chain exists from the DataSource of the source system, to the InfoSource and the update rules, through to the data target.

• The DataSource fields match the field names of the characteristics and key figures in Bank Analyzer.

7.4.3.4 SDL Data Export

This function is based on the data held in the buffer. A buffer always refers to one processing package [page 1658]. When it is updated, the data is transferred from the buffer:

- To a data drain of category 01 (table)
- To a data drain of category 02 (file)

The data drain is assigned in the processing area [page 1654]. You make the data drain settings in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *SDL Data Export* > *Data Drains*.

The SDL data export function provides verification lists for data in the buffer. The data can also be transferred from the buffer to the Reporting BI system (so that it can be verified against values from previous periods, for example).

The data export function is implemented using the processing steps and functions of the processing framework [page 1651] for SDL data exports.

7.4.3.4.1 SDL Data Export Process

Use

The SDL data export process in the processing framework [page 1651] takes the form of a structured sequence of logically related processing steps, and applies to one processing package [page 1658]. Each processing step can be controlled manually, or by using the status management [page 1697] facility, or the Schedule Manager [page 437].

The processing steps for SDL data exports are carried out in the processing framework using the following programs:

- Copying the Start ID
- Create New Run
- Release Run
- Worklist Creation
- Execute Processing
- Analyze Verification Lists
- Update Data
- Reverse Update

These transactions are part of the processing framework and are called manually or from the status management facility. These programs can also be scheduled as jobs in the Schedule Manager to automate periodic processing to a large extent.

Prerequisites

- You have created and activated a processing area, processing level, and processing package with start parameters in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *SDL Data Export* > *Set Up Processing Framework for SDL Data Export*. In particular, this involves selecting the data source(s), data drains, processing rule, characteristics, and key figures.
- You have defined the data drains in which result records in the temporary buffer can be updated in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *SDL Data Export* > *Data Drains* > *Define Data Drains - Category 01 (Table) and Define Data Drains - Category 2 (File)*.
- You have defined the processing rules used in the processing framework in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *SDL Data Export* > *Configuration of Module Editor* > *Edit Modules for Source Data Export*.

Process

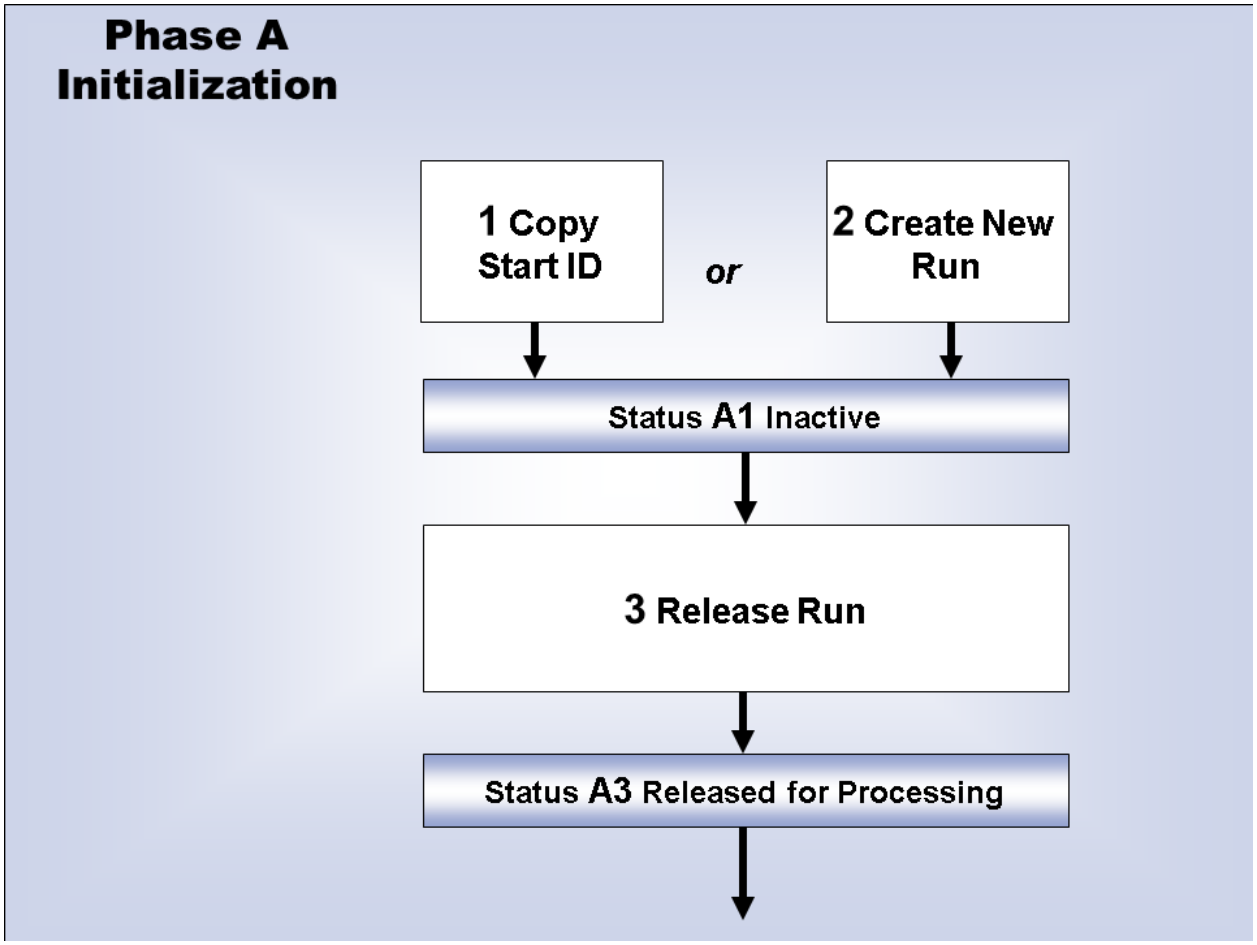
Phases and statuses of a processing package in the SDL data export process:

- **Phase A, Initialization**: Create and release run
- **Phase B, Processing**: Read and process the data, and save the data in the buffer
- **Phase C, Verification**: Provide data for verification
- **Phase D, Update**: Update data in a data drain

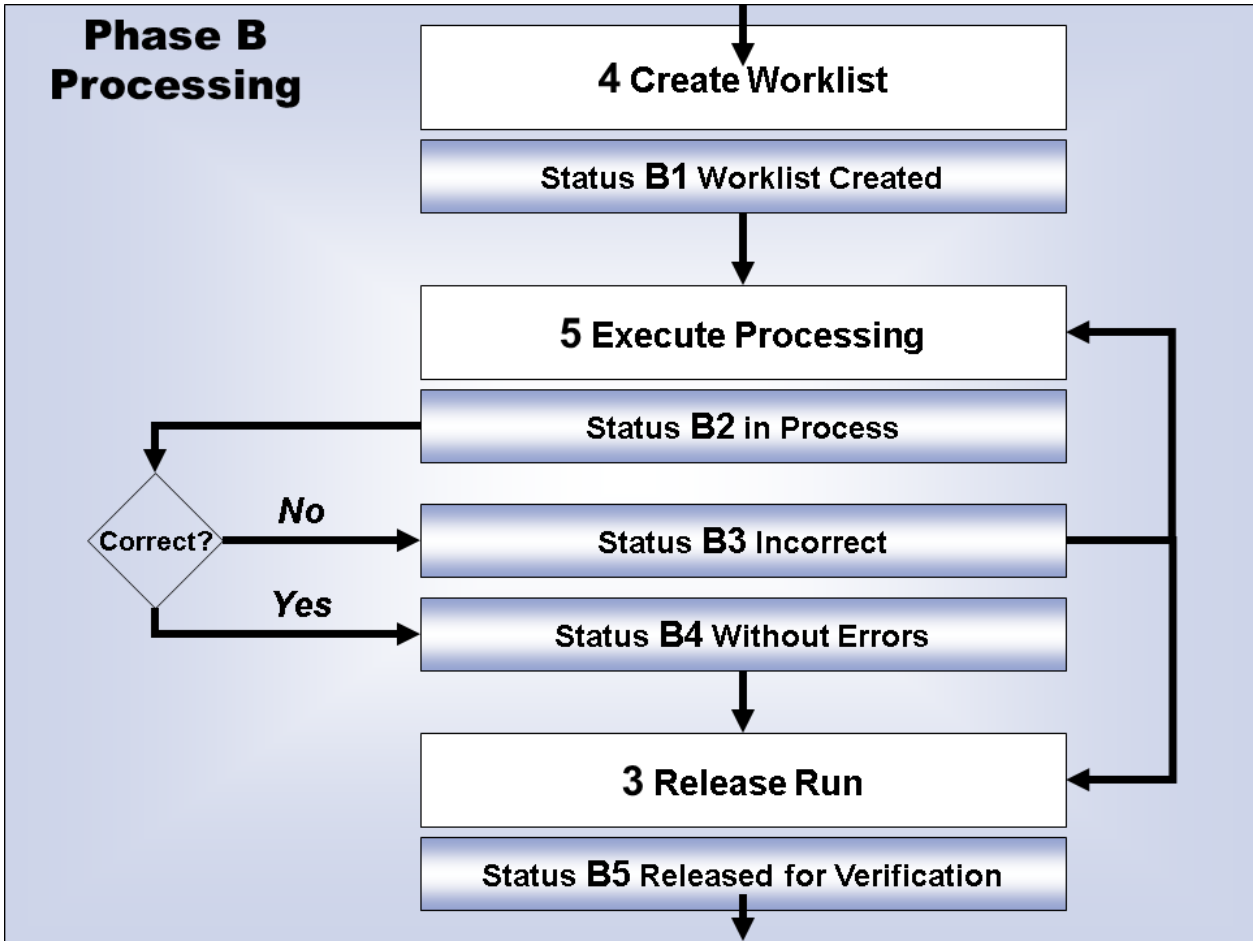
Data can be reversed after updating (phase D). If it emerges during the verification phase that data is not to be updated, you create a new run.

The following figures outline the processing phases and statuses.

Processing is initialized by calling up the *Copy Start ID* or *Create New Run* program followed by *Release Run*.

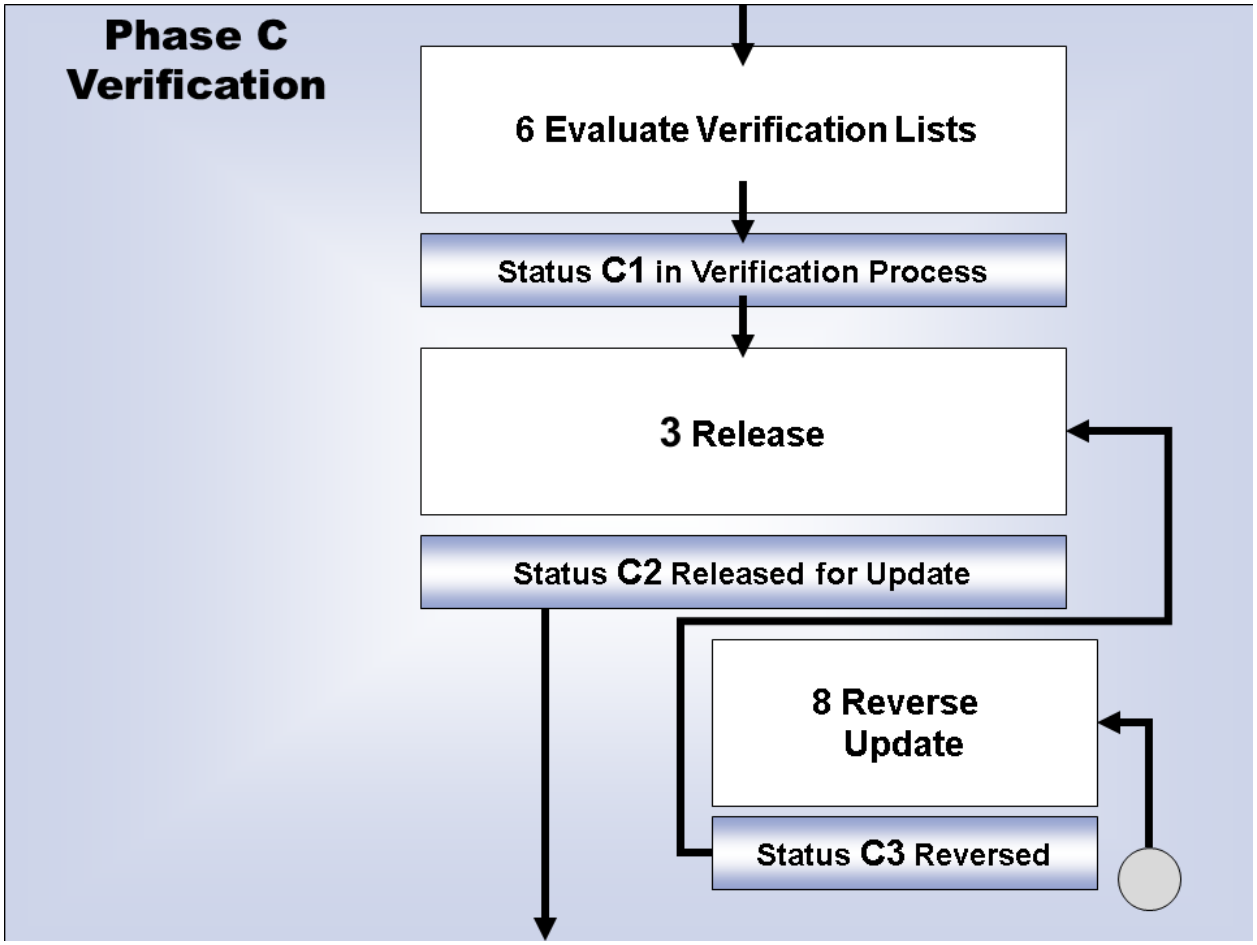


Once the run has been released in phase A, the system creates the worklist and starts processing. Incorrect as well as correct runs can be released for verification. The processing phase can be triggered again for incorrect runs (after correction, for example in the processing rules).

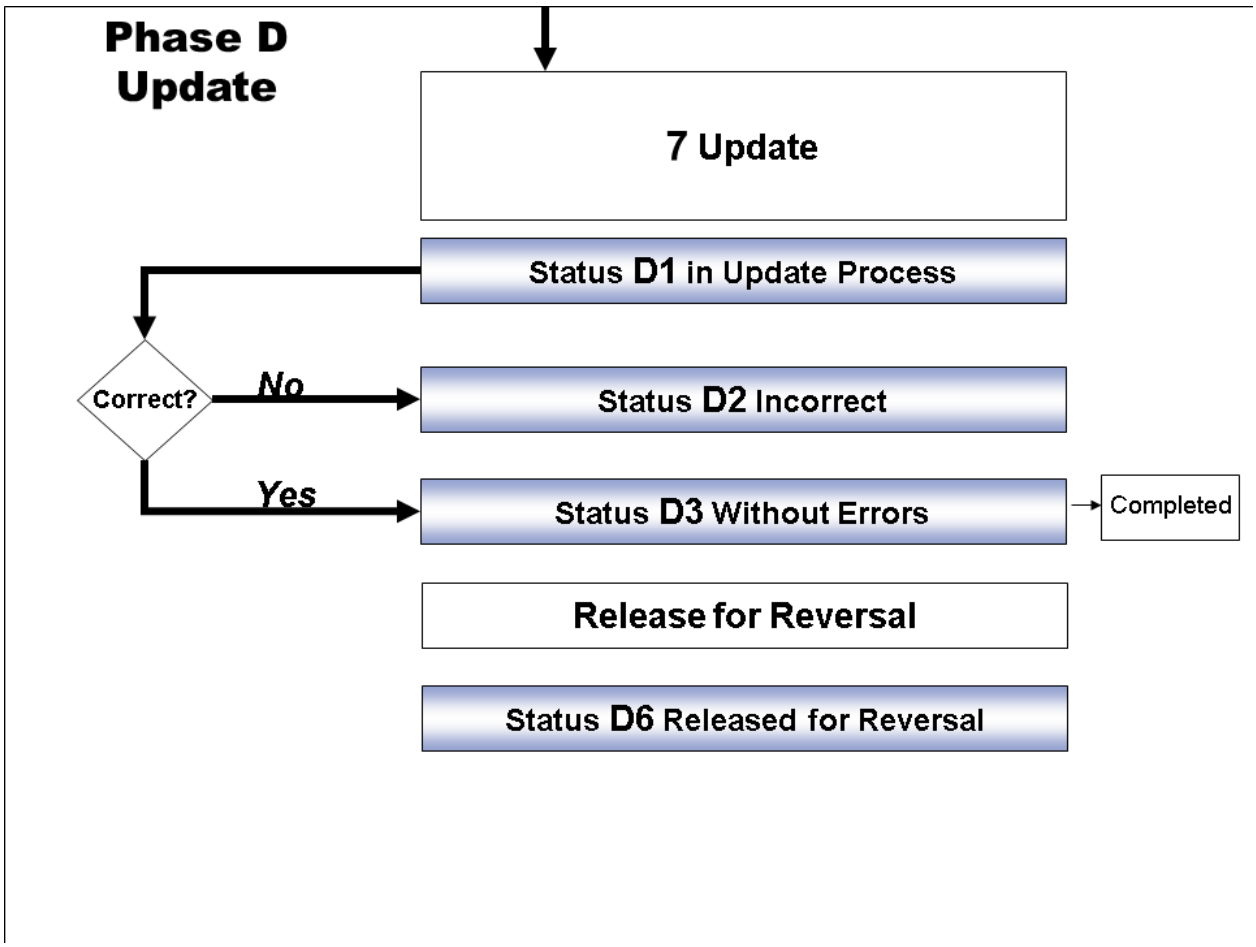


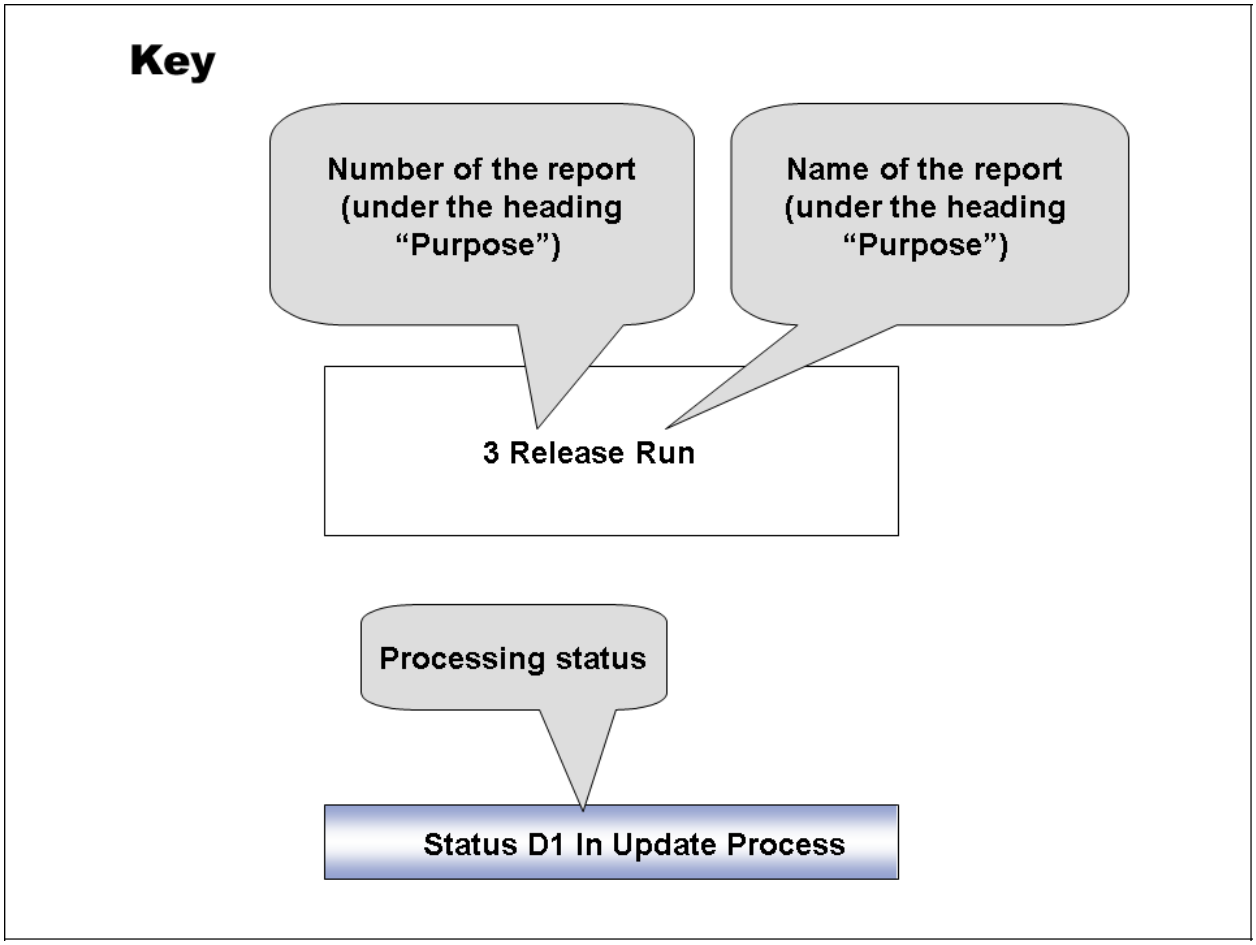
The results are checked using verification lists. After the verification phase, results can be released for updating. Updating can take place once only for each start parameter. This ensures that the results (for example, the profitability analysis) are not written several times to a data drain, which would lead to incorrect evaluations. Processing can end with the status C1 (in verification process). This status is achieved when the result data records are not to be updated because the check determined that the results are not plausible from a business perspective. You can make corrections, for example in the source data or in the processing rules, and then create a new run. The data from the incorrect run remains in the temporary buffer. If the results records have already been updated in the data drain, you can reverse the update.

> **Note**
>
> The *Reverse Update* program actually reverses the data and does not delete it. Creating reversal records has an impact on the volume of data in the data drain.



The system checks whether the update phase was successful and sets an appropriate status.


Phase D Update
7 Update
Status D1 in Update Process
Correct?
No
Status D2 Incorrect
Yes
Status D3 Without Errors
Completed
Release for Reversal
Status D6 Released for Reversal



Note that the statuses that are not generated by calling a program (B3 and B4) are set by the system. While the statuses in phases A and B can be scheduled as jobs in the Schedule Manager and thereby automated, verifying the results in phase C must take place manually using verification lists. Phase C can be automated using a workflow.

Periodic Processing

The procedure described above applies to one processing package. Multiple processing areas, levels, and packages are normally used. This means that the programs listed above have to be executed for each package and start ID.

Processing for a processing package can take place automatically by bundling the individual steps to form processes in the SAP Schedule Manager. The individual processes can be included in general processing activities at the end of the month. In this case, variants have to be created for the programs for each processing package and start parameter.

You can use the *Copy Start ID* program to copy existing start parameters. The business record date for the new start parameters can be set by the user. Variants for the existing start parameters can be adjusted automatically to suit the new start parameters. This enables you to use the process that has already been defined for processing a processing package on any number of dates. The *Copy Start ID* program can also be scheduled at the beginning of a new processing run for each processing package, so that a recurring, periodic processing run has to be set up only once and can be used in other periods.

7.4.3.4.1.1 Copying the Start ID

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *SDL Data Export* > *Set Up Processing Framework for SDL Data Export*.
- You have already created a start ID for the processing package.

Context

This transaction creates a new start ID by copying one that already exists for a processing package. It reads all the start parameters [page 1660] of the start ID that you want to copy. You can choose any name and business record date for the new start ID. This program is the first step (if you do not create a processing run directly) in the initialization phase of the SDL data export processing procedure.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *SDL Data Export* > *Processing* > *Copy Start ID*.
2. Select a processing area, processing level, and processing package.
3. Select the start ID you want to copy from the input help.
4. Enter a name for the new start ID you are creating.
5. Enter a business record date.
6. Choose *Execute*.

Results

The system generates the new start ID.

7.4.3.4.1.2 Create New Run

Prerequisites

You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *SDL Data Export* > *Set Up Processing Framework for SDL Data Export*.

Context

This transaction generates a new run for a processing package [page 1658] and a start ID with associated start parameters [page 1660]. This transaction is the first step (if you have not already called this step automatically by copying a start ID) in the initialization phase of the SDL data export processing procedure [page 1683].

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *SDL Data Export* > *Processing* > *Create New Run*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Choose *Execute*.

Results

The system creates a new processing run for the start ID and associated processing package that you selected.

7.4.3.4.1.3 Release Run

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *SDL Data Export* > *Set Up Processing Framework for SDL Data Export*.
- In the *Create New Run* transaction, you have created a processing run for the processing package defined in Customizing.

Context

The following release status changes are executed within this transaction:

- Release for editing: from status A1 to status A3 (released for editing)
- Release for verification: from status B4 to status B5 (released for verification)
- Release for update: from status B5 (released for verification) or C1 (in verification process) or C3 (reversed) to status C2 (released for update)
- Reverse release for update: from status C2 to status C1 (in verification process)
- Release for reversal: from status D3 (update has no errors) to status D6 (released for reversal)

You use this transaction to:

- Release a processing run defined in the *Create New Run* transaction
- Release for verification the data determined in the *Execute Processing* transaction

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *SDL Data Export* > *Processing* > *Release Run*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Select the sequential number of the run you want to release.
5. Choose:
   - Initialization if you want to release a newly created run
   - Processing if you want to release the results data for verification
6. Select the appropriate status for the phase.

7. Choose *Execute*.

7.4.3.4.1.4 Create Worklist

Use

The program for creating a worklist reads the settings for the processing package [page 1658] and uses them to determine how the initial data for the processing rule [page 1661] is to be selected. The results of this selection are used to generate the worklist. This program is the first processing step in the SDL data export process [page 1683].

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *SDL Data Export* > *Set Up Processing Framework for SDL Data Export*.
- In the *Create New Run* transaction, you have defined a processing run for the processing package defined in Customizing, and released it in the *Release Run* transaction.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *SDL Data Export* > *Processing* > *Create Worklist*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Select the sequential number of the run for which you want to create a worklist.
5. The system automatically sets the time stamp.
6. Choose *Execute*.

Result

The system creates the worklist.

7.4.3.4.1.5 Execute Processing

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *SDL Data Export* > *Set Up Processing Framework for SDL Data Export*.
- In the *Create New Run* transaction, you have defined a processing run for the processing package defined in Customizing, and released it in the *Release Run* transaction.
- In the *Create Worklist* transaction, you have generated a worklist for the run you have released.

Context

This program reads the data from the worklist, which was compiled from primary data sources [page 1593] and possibly contains additional information from secondary data sources. It then calls the processing rule [page 1661]. The result data produced by the processing rule is stored in the buffer. The additional information is read and the initial data processed in parallel. The block size defined previously is used here. This program is the second processing step in the SDL data export process [page 1683].

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *SDL Data Export* > *Processing* > *Execute Processing*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Select the sequential number of the run you want to process.
5. Choose *Execute*.

Results

The system performs processing and writes the data it determines to the buffer.

7.4.3.4.1.6 Analyze Verification Lists

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *SDL Data Export* > *Set Up Processing Framework for SDL Data Export*.
- In the *Create New Run* transaction, you have defined a processing run for the processing package defined in Customizing, and released it in the *Release Run* transaction.
- In the *Create Worklist* transaction, you have generated a worklist for the run you have released.
- You have carried out the *Execute Processing* transaction.

Context

You use verification lists [page 1658] to verify data that has been written to the buffer. You can also create a DataSource from a verification list and use it to extract data from the Source Data Layer and transfer it to the Reporting BI system.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *SDL Data Export* > *Processing* > *Evaluate Verification Lists*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Choose *Execute*.
5. On the *Select List* screen, expand the *Verification Lists* folder and select a verification list that you defined in Customizing.
6. Choose *Execute*.
7. The verification list is set up using the characteristics and key figures in the buffer.
8. On the application toolbar, choose *Layout Settings* and *Change Layout*.
9. Select the additional columns you need to adjust your layout.

   If you want to generate a DataSource from the verification list, proceed as follows:
10. Choose *Create BI DataSource*.
11. Enter a name for the *DataSource*.

12. Choose *Save*.

Results

The system generates a DataSource from the verification list.

7.4.3.4.1.7 Update Data

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *SDL Data Export* > *Set Up Processing Framework for SDL Data Export*.
- You have defined a data drain of category 01 (table) or category 02 (file) in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *SDL Data Export* > *Data Drains*.
- In the *Create New Run* transaction, you have defined a processing run for the processing package defined in Customizing, and released it in the *Release Run* transaction.
- In the *Create Worklist* transaction, you have generated a worklist for the run you have released.
- You have carried out the *Execute Processing* transaction.

Context

You use this transaction to update data that you have stored in a buffer during a processing run in a data drain of category 01 (table) or category 02 (file).

> **Note**
>
> You can update exactly one run per start ID. If you have created several runs for one start ID and have already updated one of the runs, you have to reverse the updated run before you can update a different run using the same start ID.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *SDL Data Export* > *Processing* > *Update*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Select the sequential number of the run you want to update.
5. Choose *Execute*.

7.4.3.4.1.8 Reverse Update

Prerequisites

You have set the status to *Released for Reversal* under *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *SDL Data Export* > *Processing* > *Release Run*.

You have updated the data records in a buffer to a data drain under *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *SDL Data Export* > *Processing* > *Update*.

Context

You use this transaction to reverse data records in a buffer when you have updated them in a data drain of category 01 (table) or 02 (file).

> **Note**
>
> You can reverse the update run, but the updated data records in the data drains are not reversed at the same time. This means that the data records still exist in the data drain and are not deleted. If you have reversed an incorrect update run and then updated it again successfully, the data drain contains the updated data records that are correct as well as those that have errors.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *SDL Data Export* > *Processing* > *Reverse Update*.
2. Select a *Processing Area*, *Processing Level*, and *Processing Package*.
3. Select a start ID using input help.
4. Select the sequential *Number of the Run* whose update you want to reverse.
5. Choose *Execute*.

7.4.3.4.2 Status Management in the SDL Data Export Process

This function enables you to trigger the processing procedure for SDL data exports in the processing framework [page 1651] and control all the required processing steps

at processing package [page 1658] level using the relevant status.

This means you can use the following transactions for the SDL data export process in status management:

- Copy start ID
- Create new run
- Release run
- Create worklist
- Execute processing
- Analyze verification lists
- Update data
- Reverse update

7.4.3.4.2.1 Calling Status Management

Prerequisites

You have set up and activated a processing area, a processing level, and a processing package. The data sources are filled with initial data that can be transferred to a processing rule.

Context

You can use the status management facility to trigger the processing of a processing run and control the necessary processing phases and status.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *SDL Data Export* > *Status Management* > *Call Status Management*.
2. Select a processing area, processing level, and processing package.
3. Specify a *Key Date From* (the date as of which you want to display processing runs).
4. Choose *Execute*.
5. Expand the file structure. The folders represent the selected processing area, processing level, processing package, and start parameters. Runs that have already been created are displayed below the start parameters.

   To create a new run, start the appropriate program in the context menu (right mouse click) for the processing package. The new run is displayed below the *Start Parameters* folder.

   You can call the context menu for a processing package for any run and start the transactions from the SDL data export process for status management. In the context menu, you can also call the log for the run.

7.4.3.5 Cash Flow Export

This function is based on the data held in the buffer. A buffer always refers to one processing package [page 1658]. An update run transfers the data in the buffer to a data drain of category 02 (file).

The data drain is assigned in the processing area [page 1654]. You make the data drain settings in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *Export of Cash Flows* > *Data Drains*.

The cash flow export function is implemented using the processing steps and functions of the processing framework [page 1651] for cash flow exports.

7.4.3.5.1 Cash Flow Export Process

Purpose

In the processing framework [page 1651], the process used to extract cash flows comprises a sequence of processing steps, each of which builds on the previous step. These steps apply to one processing package [page 1658]. Each processing step can be controlled manually, or by using the status management [page 1697] facility, or the Schedule Manager [page 437].

In the processing framework, the following programs are used to export cash flows:

- Copy Start ID
- Create New Run
- Release Run
- Create Worklist
- Execute Processing
- Analyze Verification Lists
- Update
- Reverse Update

These transactions are part of the processing framework and are called manually or from the status management facility. These programs can also be scheduled as jobs in the Schedule Manager to automate periodic processing to a large extent.

Prerequisites

- You have created and activated a processing area, processing level, and processing package with start parameters in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Data Export* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *Export of Cash Flows* > *Define Processing Framework for Cash Flow Export*. In particular, this involves selecting the data source(s), data drains, processing rule, characteristics, and key figures.
- You have defined the data drains in which the result records in the temporary buffer can be updated in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Data Export* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *Export of Cash Flows* > *Data Drains* > *Define Data Drains - Category 2 (File)*.

Process Flow

Phases and statuses of a processing package in the cash flow export process:

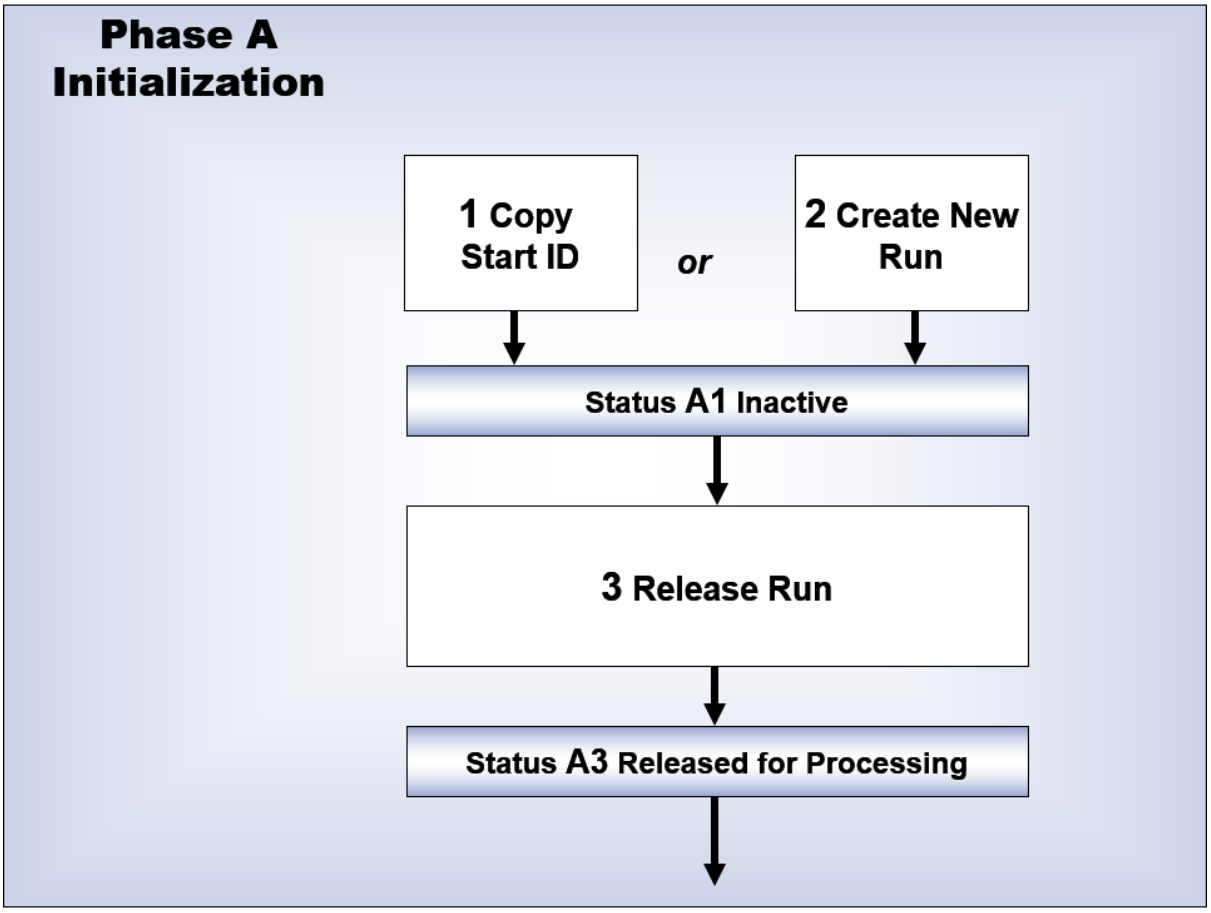
- **Phase A, Initialization:** Create and release run
- **Phase B, Processing:** Read and process the data, and save the data in the buffer
- **Phase C, Verification:** Provide data for verification
- **Phase D, Update**: Update data in a data drain

Data can be reversed after updating (phase D). If it emerges during the verification phase that data is not to be updated, you create a new run.

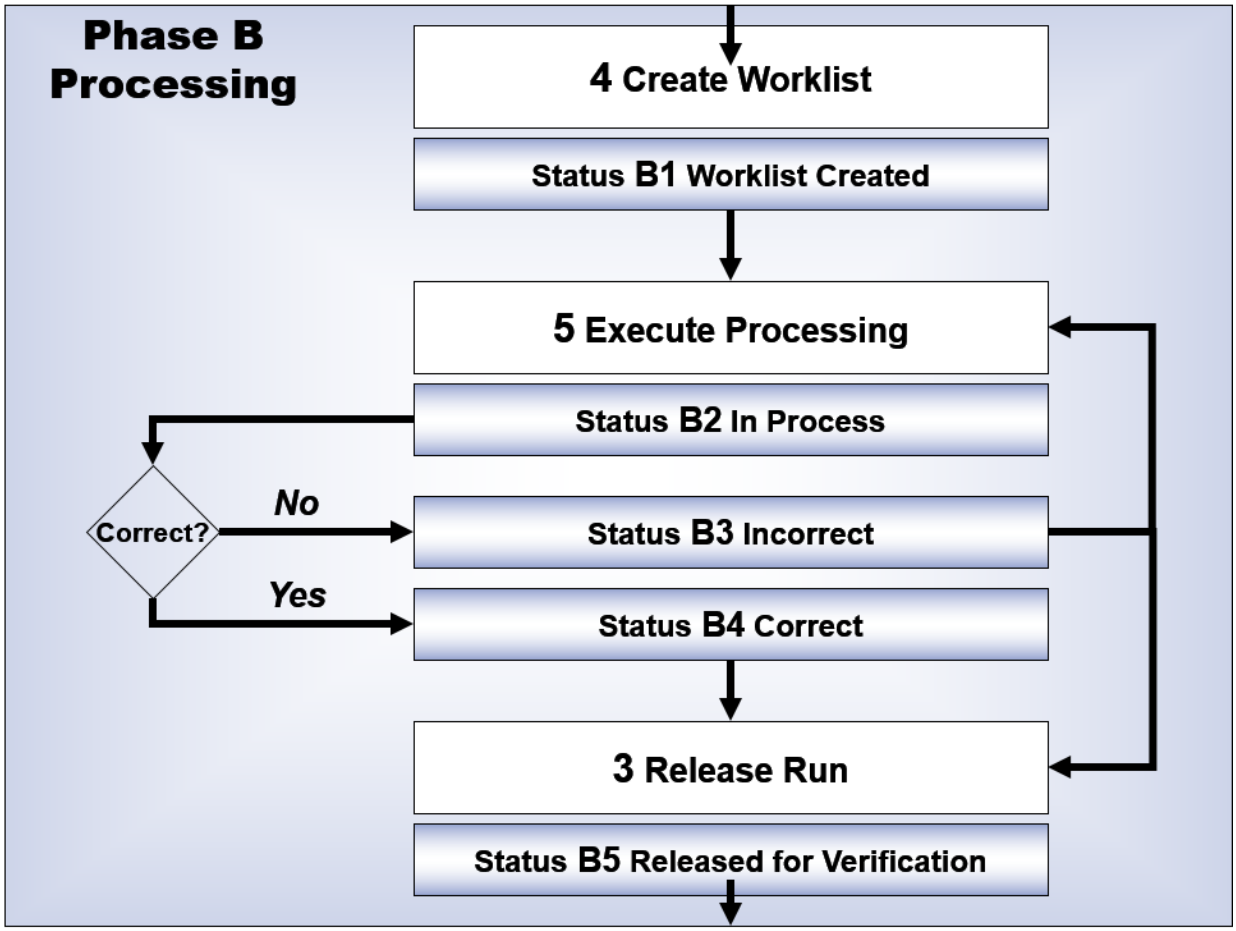
The following figures outline the processing phases and statuses:

Phase A: Initialization

Processing is initialized by calling up the *Copy Start ID* or *Create New Run* program followed by *Release Run*.

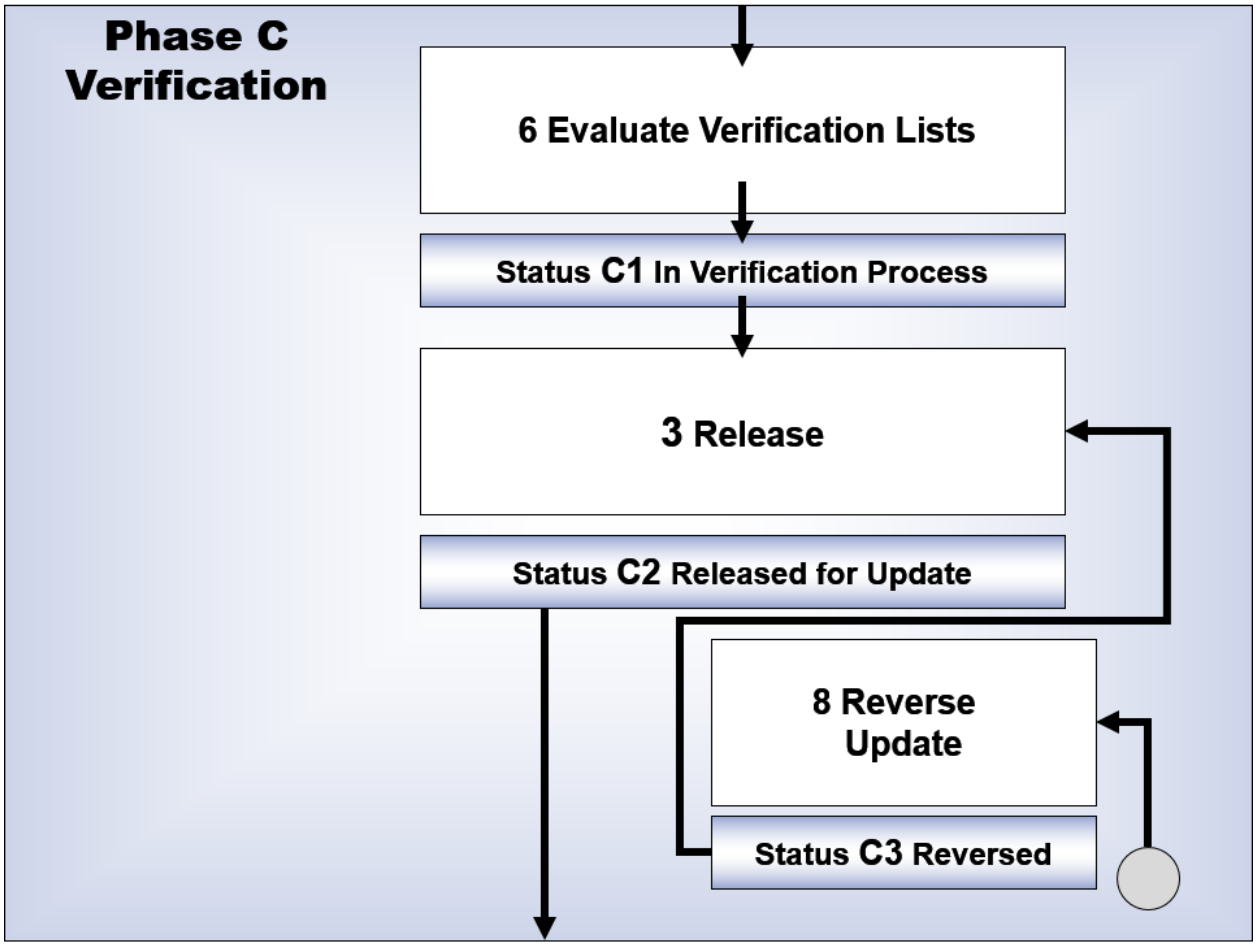
- 

Phase B: Processing Once the run has been released in phase A, the system creates the worklist and starts processing. Incorrect as well as correct runs can be released for verification. The processing phase can be triggered again for incorrect runs (after correction, for example in the processing rules).

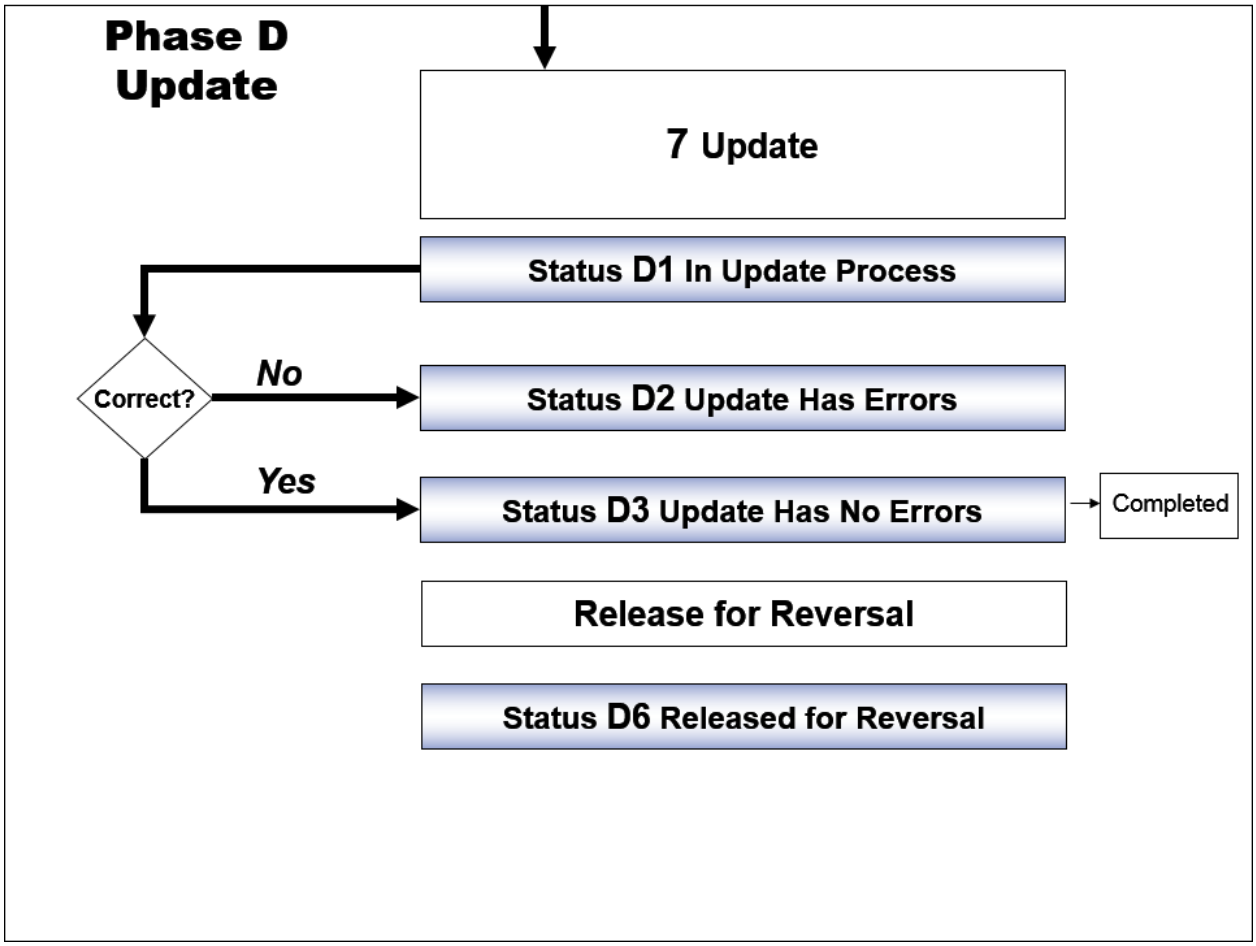


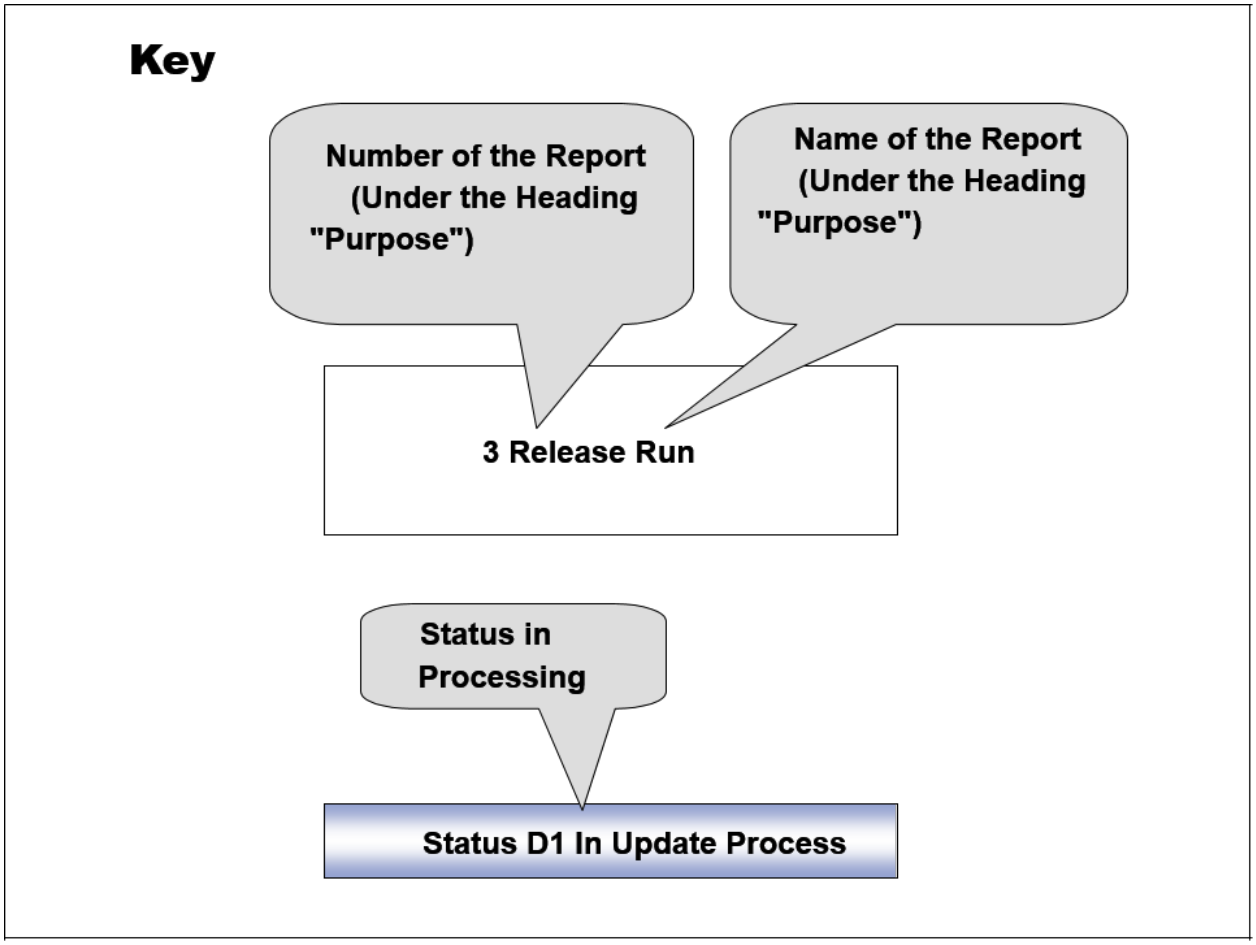
Phase C: Verification The results are checked using verification lists. After the verification phase, results can be released for updating. Updating can take place once only for each start parameter. This ensures that the results (for example, the profitability analysis) are not written several times to a data drain, which would lead to incorrect evaluations. Processing can end with the status C1 (in verification process). This status is achieved when the result data records are not to be updated because the check determined that the results are not plausible from a business perspective. You can correct the results in the Source Data Layer, for instance, or you can correct the processing rules, and then create a new run. The data from the incorrect run remains in the temporary buffer. If the results records have already been updated in the data drain, you can reverse the update.

> **Note**
>
> The *Reverse Update* program actually reverses the data and does not delete it. Creating reversal records has an impact on the volume of data in the data drain.



Update (Phase D): The system checks whether the update phase was successful and sets an appropriate status.


Phase D Update
7 Update
Status D1 In Update Process
Correct?
No
Status D2 Update Has Errors
Yes
Status D3 Update Has No Errors
Completed
Release for Reversal
Status D6 Released for Reversal



Note that the statuses that are not generated by calling a program (B3 and B4) are set by the system. While the statuses in phases A and B can be scheduled as jobs in the Schedule Manager and thereby automated, verifying the results in phase C must take place manually using verification lists. Phase C can be automated using a workflow.

Periodic Processing

The procedure described above applies to one processing package. Multiple processing areas, levels, and packages are normally used. This means that the programs listed above have to be executed for each package and start ID.

Processing for a processing package can take place automatically by bundling the individual steps to form processes in the SAP Schedule Manager. The individual processes can, for example, be included in general processing activities at the end of the month. In this case, variants have to be created for the programs for each processing package and start parameter.

You can use the *Copy Start ID* program to copy existing start parameters. Variants for the existing start parameters can be adjusted automatically to suit the new start parameters. This enables you to use the process that has already been defined for processing a processing package for dates of your choice. The *Copy Start ID* program can also be scheduled at the beginning of a new processing run for each processing package, so that a recurring, periodic processing run has to be set up only once and can be used in other periods.

7.4.3.5.1.1 Copying the Start ID

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *Export of Cash Flows* > *Define Processing Framework for Cash Flow Export*.
- You have already created a start ID for the processing package.

Context

This transaction creates a new start ID by copying one that already exists for a processing package. It reads all the start parameters [page 1660] of the start ID that you want to copy. You can give the start ID and start parameters names of your choice. This program is executed as the first step (if you do not create a processing run directly) in the initialization phase of the processing process for cash flow export.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *Export of Cash Flows* > *Processing* > *Copy Start ID*.
2. Select a processing area, processing level, and processing package.
3. Select the start ID you want to copy from the input help.
4. Enter a name for the new start ID you are creating.
5. Select the new start parameters.
6. Choose *Execute*. The new start ID is generated.

7.4.3.5.1.2 Create New Run

Prerequisites

You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *Export of Cash Flows* > *Define Processing Framework for Cash Flow Export*.

Context

This transaction generates a new run for a processing package [page 1658] and a start ID with associated start parameters [page 1660]. This transaction is the first step (if you have not already called this step automatically by copying a start ID) in the initialization phase of the cash flow processing procedure.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *Export of Cash Flows* > *Processing* > *Create New Run*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Choose *Execute*.

   A new processing run is created for the start ID and associated processing package that you selected.

7.4.3.5.1.3 Release Run

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *Export of Cash Flows* > *Define Processing Framework for Cash Flow Export*.

- In the *Create New Run* transaction, you have created a processing run for the processing package defined in Customizing.

Context

The following release status changes are executed within this transaction:

- Release for editing: from status A1 to status A3 (released for editing)
- Release for verification: from status B4 to status B5 (released for verification)
- Release for update: from status B5 (released for verification) or C1 (in verification process) or C3 (reversed) to status C2 (released for update)
- Reverse release for update: from status C2 to status C1 (in verification process)
- Release for reversal: from status D3 (update has no errors) to status D6 (released for reversal)

You use this transaction to:

- Release a processing run defined in the *Create New Run* transaction
- Release for verification the data determined in the *Execute Processing* transaction

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *Export of Cash Flows* > *Processing* > *Release Run*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Select the sequential number of the run you want to release.
5. Choose:
   - Initialization if you want to release a newly created run
   - Processing if you want to release the results data for verification
6. Select the appropriate status for the phase.
7. Choose *Execute*.

7.4.3.5.1.4 Create Worklist

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *Export of Cash Flows* > *Define Processing Framework for Cash Flow Export*.
- In the *Create New Run* transaction, you have defined a processing run for the processing package defined in Customizing, and released it in the *Release Run* transaction.

Context

The program for creating a worklist reads the settings for the processing package [page 1658] and uses them to determine how the initial data for the processing rule [page 1661] is to be selected. The results of this selection are used to generate the worklist. This program is the first step in the processing phase of the cash flow export process [page 1683].

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *Export of Cash Flows* > *Processing* > *Create Worklist*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Select the sequential number of the run for which you want to create a worklist.
5. Choose *Execute*.The worklist is created.

7.4.3.5.1.5 Execute Processing

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional*

and Alternative Settings > *Data Export* > *Export of Cash Flows* > *Define Processing Framework for Cash Flow Export* .

- In the *Create New Run* transaction, you have defined a processing run for the processing package defined in Customizing, and released it in the *Release Run* transaction.
- In the *Create Worklist* transaction, you have generated a worklist for the run you have released.

Context

This program reads data from the worklist, which was compiled from primary data sources [page 1593]. It then calls the processing rule [page 1661]. The cash flows extracted by the processing rule are stored in the buffer. The additional information is read and the initial data processed in parallel. The block size defined previously is used here. This program is the second step in the processing phase of the cash flow export process [page 1683].

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *Export of Cash Flows* > *Processing* > *Execute Processing* .
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Select the sequential number of the run you want to process.
5. Choose *Execute*.

   The run is processed and the data that is determined is written to the buffer.

7.4.3.5.1.6 Analyze Verification Lists

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *Export of Cash Flows* > *Define Processing Framework for Cash Flow Export* .
- In the *Create New Run* transaction, you have defined a processing run for the processing package defined in Customizing, and released it in the *Release Run* transaction.
- In the *Create Worklist* transaction, you have generated a worklist for the run you have released.

- You have carried out the *Execute Processing* transaction.

Context

You use verification lists [page 1658] to verify data that has been written to the buffer.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *Export of Cash Flows* > *Processing* > *Evaluate Verification Lists*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Choose *Execute*.
5. On the *Select List* screen, expand the *Verification Lists* folder and select a verification list that you defined in Customizing.
6. Choose *Execute*.
7. The verification list is set up using the characteristics and key figures in the buffer.
8. On the application toolbar, choose *Layout Settings* and *Change Layout*.
9. Select the additional columns you need to adjust your layout.
10. Save your entries.

7.4.3.5.1.7 Update Data

Prerequisites

- You have defined a processing package, processing level, and processing area in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *Export of Cash Flows* > *Define Processing Framework for Cash Flow Export*.
- You have defined a data drain in Customizing for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Settings* > *Data Export* > *Export of Cash Flows* > *Data Drains* > *Define Data Drains - Category 2 (File)*.
- In the *Create New Run* transaction, you have defined a processing run for the processing package defined in Customizing, and released it in the *Release Run* transaction.

- In the *Create Worklist* transaction, you have generated a worklist for the run you have released.
- You have carried out the *Execute Processing* transaction.

Context

You use this transaction to update data that you have stored in a buffer during a processing run in a data drain of category 02 (file).

> **Note**
>
> You can update exactly one run per start ID. If you have created several runs for one start ID and have already updated one of the runs, you have to reverse the updated run before you can update a different run using the same start ID.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *Export of Cash Flows* > *Processing* > *Update*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Select the sequential number of the run you want to update.
5. Choose *Execute*.

7.4.3.5.1.8 Reverse Update

Prerequisites

You have set the status to *Released for Reversal* under *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Export of Cash Flows* > *Processing* > *Release Run*.

You have updated the data records in a buffer to a data drain under *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *Data Export* > *Processing* > *Update*.

Context

You use this transaction to reverse buffered data records that you updated to a data drain of category 02 (file) using the *Update* transaction.

> **Note**
>
> You can reverse the update run, but the updated data records in the data drains are not reversed at the same time. This means that the data records still exist in the data drain and are not deleted.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *Export of Cash Flows* > *Processing* > *Reverse Update*.
2. Select a processing area, processing level, and processing package.
3. Select a start parameter ID using input help.
4. Select the sequential number of the run whose update you want to reverse.
5. Choose *Execute*.

7.4.3.5.2 Status Management in Cash Flow Export

This function enables you to trigger the processing procedure for cash flow exports in the processing framework [page 1651] and control all the required processing steps

at processing package [page 1658] level using the relevant status.

This means you can use the following transactions for the cash flow export processing procedure in status management:

- Copy start ID
- Create new run
- Release run
- Create worklist
- Execute processing
- Analyze verification lists
- Update data
- Reverse update

7.4.3.5.2.1 Calling Status Management

Prerequisites

You have set up and activated a processing area, a processing level, and a processing package. The data sources are filled with initial data that can be transferred to a processing rule.

Context

You can use the status management facility to trigger the processing of a processing run and control the necessary processing phases and status.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Source Data and Cash Flows: Additional and Alternative Functions* > *Data Export* > *Export of Cash Flows* > *Status Management* > *Call Status Management*.
2. Select a processing area, processing level, and processing package.
3. Specify suitable start parameters (*To-Date*,*Changes As of a Specific Date*, *Cash Flow Generation Type*, *Market Data Area*) for selecting processing runs.

   > **Note**
   >
   > If you do not specify any start parameters, the system will display all processing runs.

4. Choose *Execute*.
5. Expand the file structure. The folders represent the selected processing area, processing level, processing package, and start parameters. Runs that have already been created are displayed below the start parameters.
6. To create a new run, start the appropriate program in the context menu (right mouse click) for the processing package. The new run is displayed below the *Start Parameters* folder.

   You can call the context menu for a processing package in any run, and start the transactions from the cash flow export process for status management. In the context menu, you can also call the log for the run.

7.4.4 Extract Finance and Risk Data Model

Use

You can use the transaction **/BA1/FR_DM_EXTRACT** to extract finance and risk data model information from the Bank Analyzer system in the form of XML files. You can then import these XML files to the , for example, where you can display the data model graphically, enhance it, or document it.

You can find the transaction on the SAP Easy Access screen for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Data Model Extraction* > *Extract Finance and Risk Data Model*.

You can select the following areas for extraction:

- Reference data
  - Analytical business partner data
  - Market data

    > **Caution**
    >
    > Market data can be extracted. However the system ignores it when it is imported to SAP PowerDesigner.
- Source data
  - Master data
  - Generic market data
  - Flow data
- Results data

The system creates a separate XML file for each area selected. Under *Output Options*, specify whether you want to save or display the XML files. If you want to save the data, you need to specify the file directory where you want the system to save the XML files. If you want to import the file to the *FRDP PowerDesigner Extension* afterwards, we recommend that you select the `XML` directory in your PowerDesigner Workspace directory.

Features

The following Customizing activities are extracted as XML files:

- *Create Market Data Area*: The XML file tag `OBJ_CONFIGN_AREA` is assigned to the market data area.

  > **Note**
  >
  > You make Customizing settings for market data under *Financial Services* > *Foundation* > *Market Data*.
- *Interest* node
  - *Edit Reference Interest Rates and Yield Curves*: The reference interest rates and the interest rate spread types are extracted here, but the yield curve types are **not**.
- *Exchange Rates* node
  - *Define Exchange Rate Types and Translation Ratios*: The exchange rate factors are modeled in the XML file as application data so that the Customizing data is **not** included. However, the database table `/BA1/`

`TF4_FXCVFCT` is included as an application table. The yield curve type for forward exchange rates is not taken into account.

- *Prices for Financial Instruments* node
  - *Edit Customizing for Security Prices*
  - *Assign Price Units to Security*
- *Volatilities* node
  - *Edit Volatility Types*
  - *Create Cluster from Volatility Types*
  - *Create Non-Underlying-Based Volatility Structures*

The following Customizing activities are not extracted:

- *Interest* node
  - *Edit Spread Curve Types*
  - *Edit Maturity Structures*
  - *BAdI: Model for External Yield Curve*
- *Exchange Rates* node
  - *Check Currency Codes*
  - *Define Standard Quotation for Exchange Rates*
  - *Set Decimal Places for Currencies*
  - *Map Exchange Rate Category to Exchange Rate Type*
  - *Configure Currency Translation for Accounting*
- *Prices for Financial Instruments* node
  - *Corporate Action for Security Prices and Cash Flows*
- *General Scenario Administration* node
- *Market Data Scenarios* node
- *Create Market Data Set*
- *Create Application*

7.5 Outbound Connections Using Third-Party Web Services

Use

You use the *Outbound Connections Using Third-Party Web Services* (FS-BA-IF-OUT) component to deliver data asynchrously **from** Bank Analyzer to third-party systems using customer-specific Web services.

You can do this for mass data and for data at single record level. In addition, method calls from third-party systems can be called up synchronously from Bank Analyzer.

In both cases, the component calls the Web service from Bank Analyzer.

Integration

Exits for converting and validating data are available for sending data and for synchronously calling methods that are integrated into third-party systems. These exits are based on the module editor and on Business Add-Ins (BAdIs), and have to be implemented according to customer requirements.

Ideally, you use the *Calculation and Valuation Process Manager* (CVPM) to create the relevant processes. The component is integrated into CVPM as a dedicated step category.

Features

- Data delivery from Bank Analyzer to third-party systems:
  - Asynchronoulsy or synchronously
  - Mass or single record data
  - With or without aggregation
  - With or without writing reconciliation data
- Synchronous call of methods that are provided by third-party systems

Constraints

You cannot deliver data from third-party systems **to** Bank Analyzer using this component. To deliver data from third-party systems, use the provided enterprise services or BAPIs, for example, or the *Data Load Layer* (DLL).

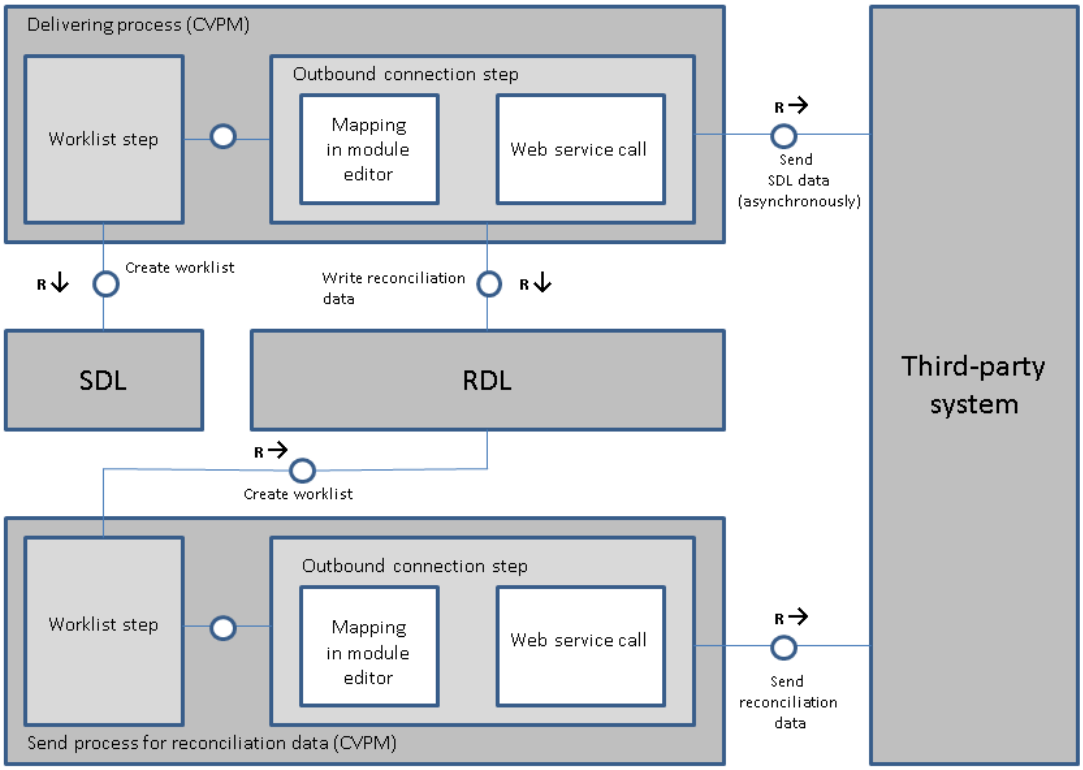
The integration with third-party systems is based on web services only.

More Information

- Example: Data Provision and Data Reconciliation [page 1716]
- Process Configuration with Outbound Connections Using Third-Party Web Services [page 1718]
- SAP Note 1623707

7.5.1 Example: Data Provision and Data Reconciliation

The following example illustrates a possible configuration for delivering data to a third-party system from Bank Analyzer, and the subsequent data reconciliation.



The *Calculation and Valuation Process Manager* (CVPM) process first delivers a delivery step and a worklist step to the third-party system. In the delivery step, the system calls the outbound connections framework. In this example, the worklist is created using the *Source Data Layer* (SDL). In a module of the module editor application for *Outbound Connections Using Third-Party Web Services*, the system converts the worklist data to the data structure of the outbound Web service operation.

In this module, the system also enriches the fields that are saved in the *Results Data Layer* (RDL) for data reconciliation at a later date. The system writes this reconciliation data to the RDL after data delivery (in this case asynchronous data delivery).

Once the delivering CVPM process is completed, the system starts the CVPM process to send the reconciliation data. You use this process to synchronize data. The system creates a worklist for this based on the previously written reconciliation data. The system converts this worklist to the data structure of the outbound Web service operation intended for data reconciliation, and then sends it.

This enables you to reconcile data in the third-party system with the previously sent data. Examples of reconciliation data are check sums (for example, the total of the nominal values of all sent loans) or the number of sent packages or financial transactions.

7.5.2 Process Configuration with Outbound Connections Using Third-Party Web Services

Use

You can use the *Outbound Connections Using Third-Party Web Services* framework with the Calculation and Valuation Process Manager (CVPM) t connect third-party systems to Bank Analyzer using Web services. The framework is already stored as a separate step category in CVPM so that you can integrate it within the step sequence of a process without any additional implementation effort.

Process

1. Generate proxy class
   You need to generate a proxy class in Bank Analyzer to enable communication using Web services. You can use a WSDL file to do this, for example. In the *Object Navigator*, select a package and in the context menu, choose *Create* > *Enterprise Service*. A wizard helps you to enter the information relevant for the proxy class.
2. Configure RDL for reconciliation data
   You configure the RDL in Customizing for *Bank Analyzer* by making settings in all the Customizing activities under *Results Data Layer* > *Basic Settings*.
3. Define outbound connections
   The outbound connection ID connects the CVPM and the *Outbound Connections Using Third-Party Web Services* framework. The Customizing activity *Create Outbound Connection ID* is particularly important for the outbound connection ID. You can find this activity in Customizing for Bank Analyzer under *Infrastructure* > *Outbound Connections Using Third-Party Web Services*. You also need to make settings in the other Customizing activities in this node so that you can create the outbound connection ID. We recommend that you process the Customizing activities in the sequence in which they appear in Customizing.
4. Create CVPM process and step sequences
   You create one or more analytical processes with the relevant step sequences in the Calculation and Valuation Process Manager (CVPM). Proceed as described in Process Configuration [page 1622]. When making these settings, you define the *Outbound Connections Using Third-Party Web Services* step by entering the outbound connection ID that you created beforehand. You do this in the Customizing activity *Edit Step Sequences for Analytical Processes* in Customizing for Bank Analyzer under *Infrastructure* > *Calculation and Valuation Process Manager (CVPM)*.
5. Connect systems
   To connect third-party systems to Bank Analyzer using Web services, you define the system connections under *SOA Management* (Transaction `SOAMANAGER`).
6. Monitor and log CVPM processes
   You can analyze all of the runs for your analytical processes in the CVPM Process Monitor. For more information, see CVPM Process Monitor [page 1635].

7.6 Correction Server

Definition

A central service function in Bank Analyzer. The correction server records corrections and can find and display any entities belonging to these corrections, provided the relevant system settings are made.

> ⚠ Caution
>
> The functions of the correction server are not available for master data framework objects that are persisted in a flat table.

> ℹ Note
>
> Corrections are made in the relevant application components only.

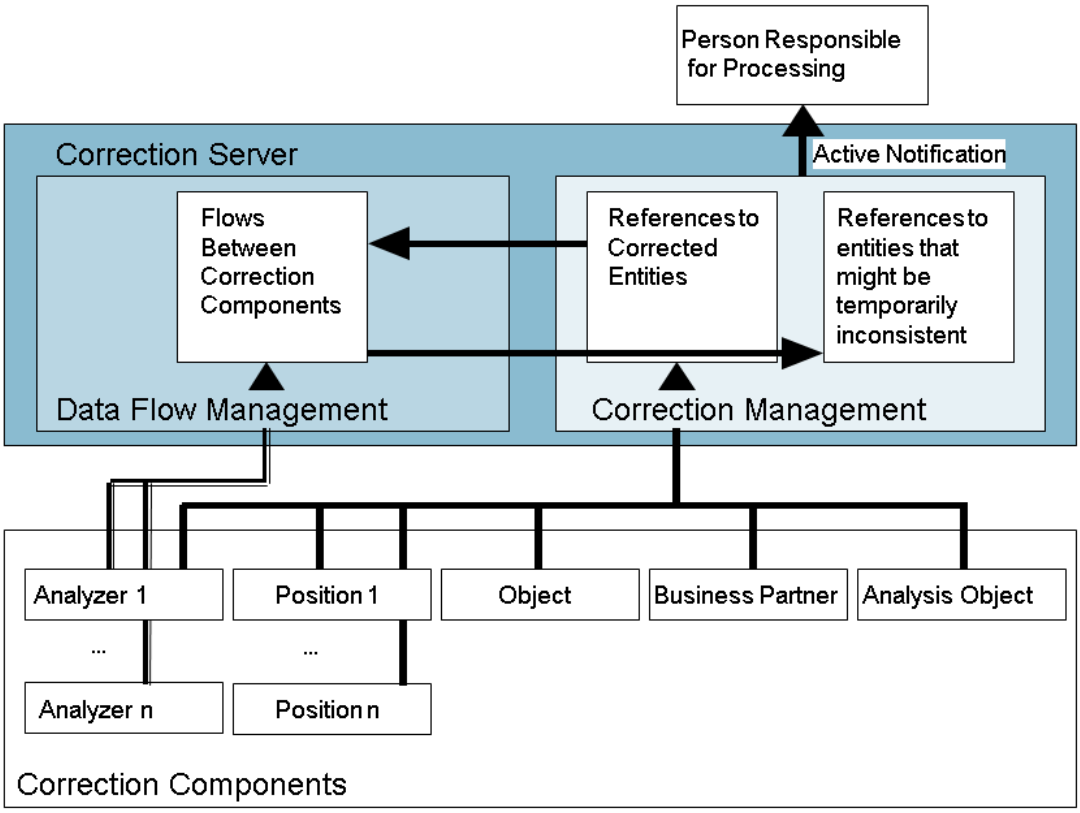
Structure

The correction server consists of data flow management and correction management.

The main functions of the correction server are:

- Logging the data flow in Bank Analyzer using data flow management
- Managing corrected entities and their dependent entities that are affected by corrections
- Determining the entities that are dependent on the corrections by using data flow management and correction management
- Actively notifying the persons responsible of the correction component dependent on a correction

The following graphic illustrates the functions of the correction server:



> **Note**
>
> You can make the appropriate settings for controlling the correction server in Customizing for *Bank Analyzer* under *Infrastructure* > *Correction Server*.

7.6.1 Data Flow Management

Use

The correction server consists of data flow management and correction management. Data is written to data flow management from the components by:

- Reading the Source Data Layer (business partner data, contract data, or actual data)
- Communicating with other applications of the Bank Analyzer system

Integration

When data flow management has been activated, data flow management and correction management can be used to refer to possibly temporarily inconsistent entities.

Prerequisites

You can only activate data flow management in Customizing for *Bank Analyzer* under *Infrastructure* > *Correction Services* > *Edit Dependencies in Correction Components*. In the Customizing activity, set the *DFM Active* checkbox.

- If the checkbox is set, you can control whether communication is logged in data flow management for each of the dependencies defined between the correction components.
- If the checkbox is not set, the dependent entities are not displayed.

Features

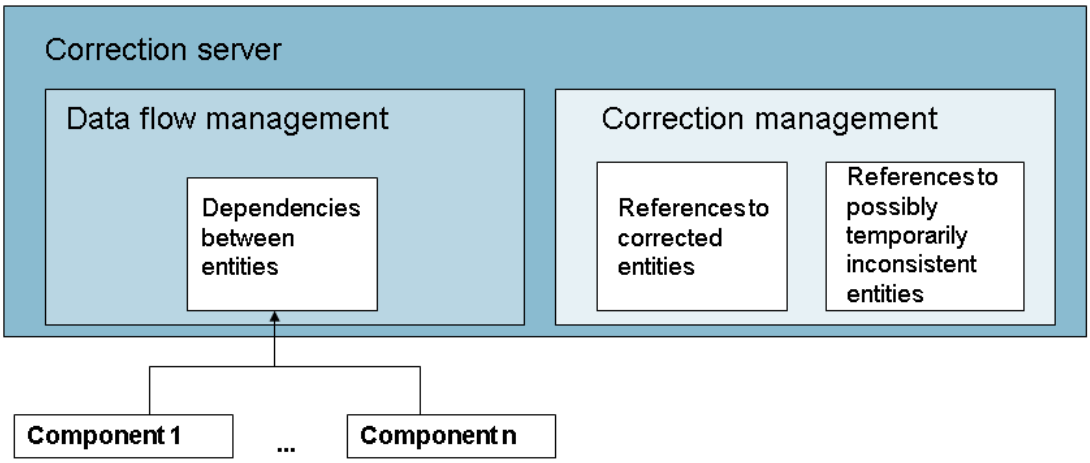
In data flow management, the system depicts the dependencies between exactly two entities. Each read process, therefore, creates a dependency between entities of components. The reading entity/component is identified as dependent and the read entity/component as the main entity/component. Each read process, that is, each resulting dependency, represents a data flow in Bank Analyzer.

The data flow can take place:

- Between SDL objects (for example, financial transaction) and an application of the Bank Analyzer system
- Between two applications of the Bank Analyzer system

These data flows are recorded in data flow management. This enables you to carry out evaluations, reports, and logging at a later date.

The following graphic illustrates data flow management:



Note: If a delta read is run on an InfoSet, then several versions of a primary object can be selected for a *"From" System Date* and a *"To" System Date*. The application of the Bank Analyzer system that is supplied with these versions might not, however, need all of these versions to calculate a result. For technical reasons, data flow management contains information stating that all of the versions supplied were required for calculating the results from the application of the Bank Analyzer system. If a correction is made to one of the versions that was not used to calculate the result, the result will still appear as a possibly temporarily inconsistent entity.

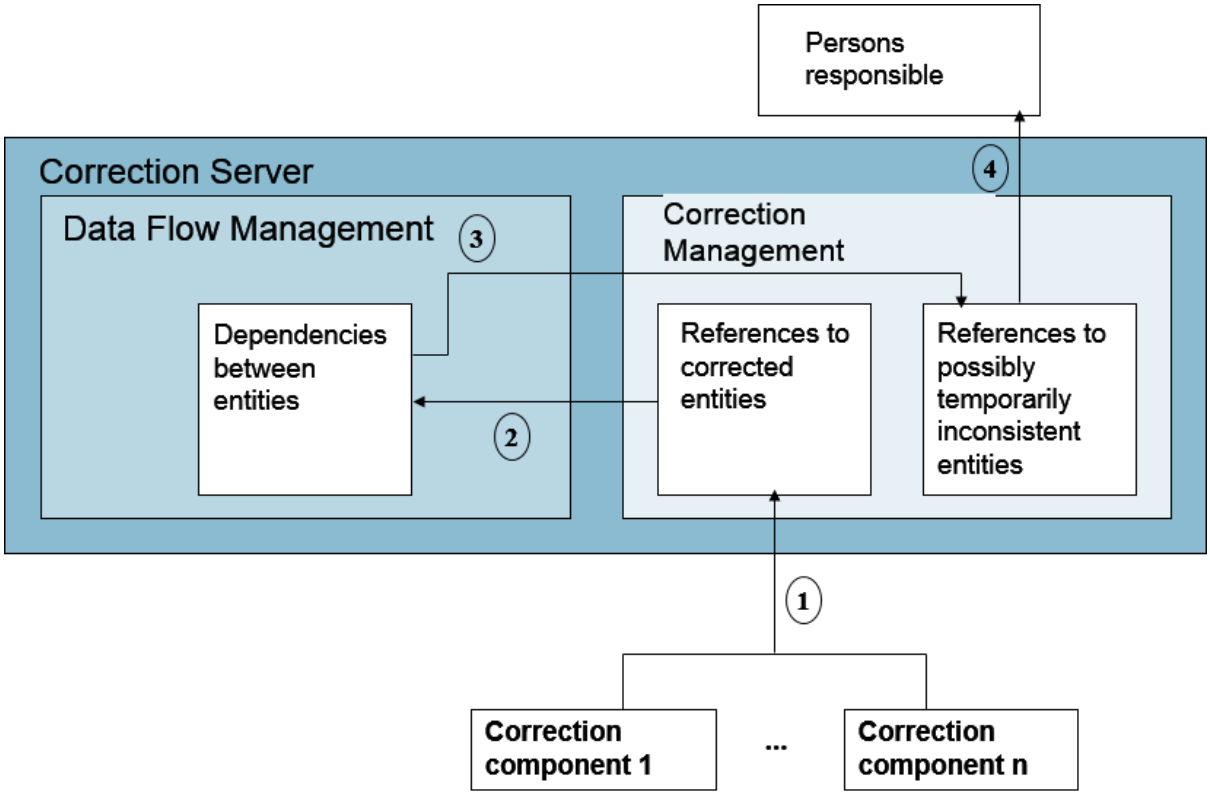
7.6.2 Correction Management

Use

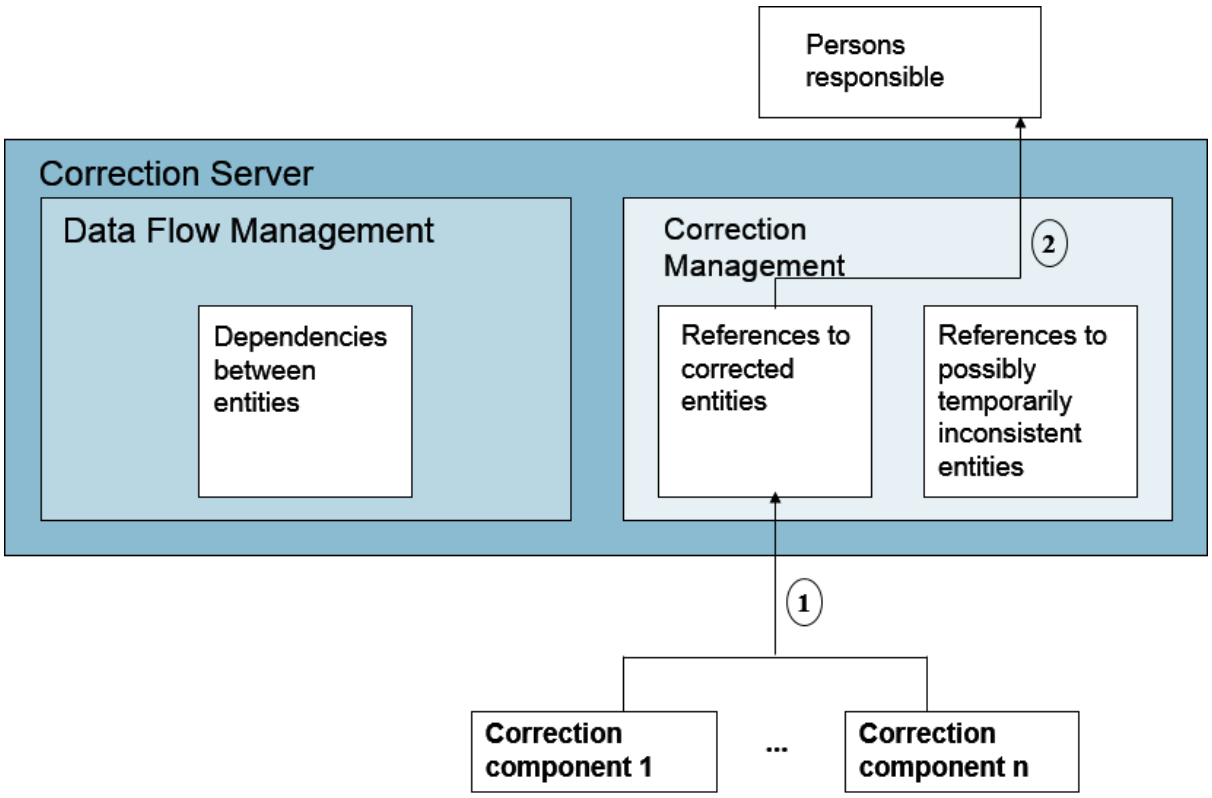
The correction server consists of data flow management and correction management. The correction management function records any changes made to objects (for example, primary objects or results of Bank Analyzer applications) and keeps a log of references to corrected entities and dependent entities of correction components. The dependent entities are determined using data flow management.

The correction management tasks differ depending on whether data flow management is activated (active) or deactivated (inactive):

- **Active Data Flow Management**
  When a correction is made, the references are written to the data pool of the corrected entities.
  The entities dependent on the correction are determined.
  The references to the dependent entities of the correction are saved, which means that they are identified as *possibly temporarily inconsistent*.
  The persons responsible for the relevant correction component receive notification in their office inbox, with information about the dependent entities.



- **Inactive Data Flow Management**
  The corrections made to leading correction components are saved, that is, when they are saved, a reference is written to the data pool of the corrected entities.
  The persons responsible for the relevant correction component receive notification in their office inbox, without information about the dependent entities.



7.6.3 Correction Components

Use

This function is technically a combination of a main component and a subcomponent. It is a business entity that manages its own data, provides functions for correcting it, and reads and uses data from other entities. A correction component can be an object in the Source Data Layer or an application of the Bank Analyzer system, or parts of these. An SDL object or application of the Bank Analyzer system can have several correction components, as is the case with positions of the Source Data Layer, for example. In this case, each position class is a correction component.

You can use subcomponents to divide and subdivide the main components. By using a subcomponent, for example, you can focus on just the relevant parts of the leading and dependent entity in the dependency relationships between the correction components.

The following table is a list of main components and subcomponents:

SDL Object/Application of the Bank Analyzer System	Main Component	Subcomponent
SDL objects	Business partner	- subcomponent only -

SDL Object/Application of the Bank Analyzer System	Main Component	Subcomponent
	Analysis object	Business partner Position Financial transaction Financial instrument Settlement account Securities position account G/L account Customer position account
	Position	Position class 1 ... Position class n
	Financial transaction	- subcomponent only -
	Financial instrument	- subcomponent only -
	Settlement account	- subcomponent only -
	Securities position account	- subcomponent only -
	Customer position account	- subcomponent only -
	G/L account	- subcomponent only -
Profit Analyzer	- main component only -	Rating level 1 ... Rating level n

Actual characteristics of a correction component are called entities of this component. When you correct the entity of a leading correction component, the values of the entities of the dependent correction components may become inconsistent with the values of the leading entity.

Note

If an inconsistency occurs, you also have to change the values of the dependent entities. The inconsistency exists temporarily until you correct the dependent entity.

Not every change to a leading entity has to result in an inconsistency. For this reason, the dependent entities that are affected when you correct a leading entity are only referred to as “possibly temporarily inconsistent entities” in the correction server.

Prerequisites

You define correction components in Customizing for *Bank Analyzer* under *Infrastructure* > *Correction Services* > *Edit Correction Components*.

7.6.4 Mass Corrections

Use

Mass corrections are made in Bank Analyzer for two reasons:

- Results are recalculated in applications of the Bank Analyzer system
- SDL objects are transferred using external interfaces

Features

When external interfaces are used to transfer SDL objects, the following Customizing settings are taken into account:

- Updating the possibly inconsistent entities and correction management
- Permission for correction
- Notification of the persons responsible
- Writing the log

> **Note**
>
> These functions are extremely performance-intensive, so you can expect long runtimes. We recommend that you deactivate these functions if you are working with large quantities of data.

During mass processing, import parameter I_CORR_SERV_ACT of the import BAPIs can be used to determine whether or not the correction server should be used and if so, how. You have the following options:

- The correction server is deactivated
- The correction server is activated with the available user settings
- The correction server is set individually for each of the objects to be transferred

> **Note**
>
> If the correction server is set individually for each of the objects to be transferred, you can use import structure I_CUST_SETTING of the import BAPIs to override the available user settings. This enables you to avoid the user settings at short notice. You might want to do this if, for example, you want to transfer a very large number of SDL objects shortly before the end of a period.
>
> Note that when overriding user settings, you can only deactivate available functions in order to accelerate mass processing at short notice. When you set the parameters to explicitly activate a function, the correction server uses the user settings available in Customizing instead of the settings you required.

7.6.5 Corrections

You can make corrections only in the corresponding application of the Bank Analyzer system or to the primary objects.

The SAP Easy Access menu contains the following transactions for the correction server:

- Resolve Inconsistent Entities [page 1727]
- Display Reports [page 1728]

7.6.5.1 Resolving Inconsistent Entities

Context

You can use this transaction to display the possible temporarily inconsistent entities in a list, or to set the selected entities to *Resolved*.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Infrastructure* > *Correction Services* > *Resolve Inconsistent Entities*.
2. Choose the inconsistent entities that you want to process. The following selection criteria are available:
   - Main component
   - Subcomponent
   - Time stamp
3. Set the checkbox
   - If you want to set the inconsistent entities to *Resolved*. In this case, no overview is displayed. A message appears that informs you that the status has changed.
   - If you want to *display* inconsistent entities. Also set the indicator if you want to *also display resolved inconsistent entities*.Next, choose *Execute* to display a list of the inconsistent entities. From this list, you can display the transaction for the corrected entity for an inconsistent entity, for example.

7.6.5.2 Displaying Reports

Context

The system uses this report to generate a list that displays all of the corrections stored in Correction Management [page 1722] as references.

Procedure

1. On the SAP Easy Access screen, choose *Bank Analyzer* > *Infrastructure* > *Correction Services* > *Display Reports*.
2. Enter the following selection criteria:
   - Main component and subcomponent
   - User name
   - Time stamp
3. In addition to entering the selection criteria, you can also set the *More Selection Criteria* checkbox. When you set the checkbox, the following options are available:
   - If you want to use the *component-specific selection*, enter an InfoSet and an InfoSet profile in the relevant *InfoSet Selection* dialog box.
   - If you want to use a selection for specific field changes, choose *Selection Criteria for Field Changes/Fields to Be Displayed* and enter the required selection criteria in the *More Selection Criteria* dialog box.
4. Choose *Execute*.

Results

The system displays a list containing all the corrected entities that match the selection criteria.

> **Note**
>
> Fields that undergo two field changes (for example, a change of the value and the currency) are flagged as cannot be determined clearly by the system.

The following functions are now available:

- Choose *Version Comparison* to display any changes made to the entity of a correction component that refers to one of the following SDL objects: financial transaction, financial instrument, position, business partner, or analysis object. The fields affected by the correction are compared in two columns with the values “before” and “after” the correction.
- Choose *Display Object* to display these correction components in display mode.

> **Note**
>
> When you choose to display a business partner from the list of corrections, the system always displays the latest version of the business partner. In the case of all other SDL objects, the system displays the version valid at that time.

- Choose *Inconsistent Entities* to go from a corrected entity to a list of dependent entities that are stored as possibly temporarily inconsistent entities.
- You can create notes for any corrections you make in a Bank Analyzer component. Choose *Display Note* to display these notes.

> **Note**
>
> Any objects currently in the release process will be linked up to the correction server, but you will not be able to call up the notes function for them.

- If a log was written for a corrected entity, you can choose *Display Log* to go from the list to the log. The log contains the following information:
  - Log ID
  - Date and time
  - User (who makes the change)
  - Correction component with the ID of the relevant entity
  - List of the IDs of the analysis objects
  - List of the entities for which permission to make corrections was requested
  - List of the entities for which permission to make corrections was denied
  - List of the entities that are dependent on the corrected entity (only entities that have been saved in data flow management can be logged here)
  - Note

7.6.6 Archiving Correction Server Data

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.

   The *Archive Administration: Initial Screen* appears.
2. Select the object name BA1_FK_COR *Bank Analyzer: Archiving of Correction Entries*.

> **Note**
>
> You use this archiving object to archive entries in correction management, and the relevant correction note. If you display an archived entry in correction management, you will not be able to display the relevant correction note. If the correction note has to be displayed, the relevant correction entry has to be reloaded beforehand.

3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a *Variant*.
5. You can specify settings for a variant by choosing *Maintain*. The *Maintain Variant* screen appears.
    1. Enter the following data:
        - *Main Component* and *Subcomponent*
        - Select whether you want to archive entries *By Time Interval* or *By Business Record Date*, or only those entries with a time stamp that is earlier than the time specified.
        - Specify whether you want to terminate the archiving run if there are still any possibly temporarily *inconsistent entities* (PTIE) for the corrections that you are trying to archive.
        - *Test Mode* or *Production Mode*: select the relevant checkbox (depending on whether or not you want to do a test archiving run).
    2. Save your entries and choose *Back*.
6. Specify the *Start Date* and the *Spool Parameters*.
7. Choose *Execute*. The system archives the data in the background.

> **Note**
>
> You archive PTIEs and corrections in one archiving object. Before you can do this, you must select the row containing a PTIE as soon as you have processed it. To do this, call the reporting functions for the PTIEs.
>
> When the corrections are archived, a preprocessing program checks whether the status of all the PTIEs has been set to *Completed*. If this condition is not fulfilled, the archiving procedure is canceled. You can deactivate this check if you wish. This will, however, mean that no conditions have to be met during the archiving process.
>
> You should, therefore, only archive corrections if you can assume that their PTIEs have the status *Completed*.
>
> For more information, see .

7.6.7 Archiving Data Flow Management Entries

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.

   The *Archive Administration: Initial Screen* appears.
2. Select the object name BA1_FK_DFW *Bank Analyzer: Archiving of Data Flow Entries*.
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a *Variant*.
5. You can specify settings for a variant by choosing *Maintain*. The *Maintain Variant* screen appears.
    1. Enter the following data:

- *Main Component* and *Subcomponent*
- Select whether you want to archive entries *By Time Interval* or *By Business Record Date*, or only those entries with a time stamp that is earlier than the time specified.
- *Test Mode* or *Production Mode*: select the relevant checkbox (depending on whether or not you want to do a test archiving run).

2. Save your entries and choose *Back*.

6. Specify the *Start Date* and the *Spool Parameters*.
7. Choose *Execute*. The system archives the data in the background.

   For more information, see .

7.7 Scenario

Definition

Scenario data is fictitious data that can be stored in the system in addition to the real data that exists for specific primary objects. The scenario data can then be used in analyses instead of, or alongside, real data and can therefore be used in simulations.

You can create scenarios for the following primary objects:

- Financial transaction
- Financial instrument
- Account
- Country
- Physical asset
- Position (including position master data)
- Business partner
- Market data

Scenario data is stored within the Source Data Layer in the respective primary object. The system additionally saves a scenario version of the existing real version of the primary object [page 103] (exception: scenario data is not time-dependent and not versioned for classical market data).

Use

A combined data version of real data and scenario data (**stress data**) is generated for the objects of the Source Data Layer (SDL) to support the stress tests [page 426] in the Basel II solution. You can access this stress data in the following evaluations:

- Support for stress tests in general methods [page 429]
- Support for stress tests in Credit Exposure [page 1261]
- Cash flow generation

> ⚠ Caution
>
> Scenario versions and stress data are not available for master data framework objects that are persisted in a flat table.

Structure

To **store** scenario data, you have to differentiate between single scenarios and composite scenarios when you create scenarios (in Customizing for Bank Analyzer under *Infrastructure* > *General Scenario Administration* > *Create Scenarios*). Single scenarios receive information about where the system stores the scenario data and about the fields for which you can store scenario data by means of their assignment to a data category (respective primary object of the SDL).

To **evaluate** the resulting stress data [page 1734], you require composite scenarios (groupings of several single scenarios in Customizing for Bank Analyzer under *Infrastructure* > *General Scenario Administration* > *Edit Composite Scenarios*).

Integration

Processing

You define scenario data manually for the primary objects listed in the following table on the SAP Easy Access screen under *Bank Analyzer* > *Infrastructure* > *General Scenario Administration*.

Object/Data Category	Considerations for the Storage of Stress Data
Financial transaction, financial instrument, country, physical asset, account, position master data	You cannot create a scenario unless at least one real data version exists for the primary object (even if it is not valid until a later date). For these object categories, you can store scenario data for all data fields (except for those that contain header data that is not based on the version, such as the external number and the authorization group). You control the fields that belong to a specific scenario using the template version (in Customizing for Bank Analyzer under *Source Data Layer* > *Primary Objects* > *Master Data* > *Templates* > *Edit Template Version*). You can use different template versions for scenario data than for real data.
Business partner	See Scenario Data for Business Partners [page 222].
Classical market data	For classical market data and, for example, interest and exchange rates, see .

Generic market data	For generic market data, you can store scenario data for all key figures. There are two categories here: • Key figure values • Absolute and relative shifts You can also combine both categories. In this case, the system uses the shifts for the key figure values in the scenario.
Positions	The scenario data includes key figures and descriptive characteristics. You can define the key figures and characteristics that are to be changed by the scenario for each position class and scenario (in Customizing for Bank Analyzer under *Source Data Layer* > *Primary Objects* > *Transaction Data* > *Positions* > *Edit Position Class*).

In addition to the manual processing of scenarios, you can also automatically provide the system with the scenario data (exception: there is no interface for classical market data). You can do this using the existing methods of the respective BAPI for the primary objects that contain the additional optional fields *Scenario* and *Scenario Enabled* or *VersionCategory* (for generic market data). For the business partner, the `Scenario` and `Read Scenario Data Only` fields are added to method `Complex Interface`.

> **Note**
>
> When you create scenario versions, the system does **not** update the index tables or the relevancy indicators.
>
> Authorization object F_BAF4_SCN is required to process scenario data (classical market data is an exception here).

Archiving and Deleting

You can archive scenario data in the same way as real data. However, you can also delete the scenario data at any time without archiving it. You find this function on the SAP Easy Access screen under *Bank Analyzer* > *Primary Objects*.

Example

You store scenario data that contains different classes of creditworthiness for a business partner. When you evaluate, this allows you to simulate not only the default risk, but also the credit rating risk in other words, the movement of the business partner from one class of creditworthiness to another.

7.7.1 Retrieving Stress Data

Use

When the system retrieves data for a stress test, it reads the real data and the scenario data and uses it to generate a combined data version (**stress data**). The system does not save this stress data permanently.

> **Note**
>
> Additionally, you can generate stress data in the Historical Database (HDB) and save this data in the Source Data Layer. For more information, see Generating Scenario Data in the Source Data Layer [page 1506].

Prerequisites

You have defined composite scenarios in Customizing for Bank Analyzer under *Infrastructure* > *General Scenario Administration* > *Edit Composite Scenarios*.

On the *SAP Easy Access* screen under *Financial Services* > *Foundation* > *Market Data* > *Edit Market Data Scenarios*, you have created market data scenarios, or on the *SAP Easy Access* screen under *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *<relevant primary object>* > *Edit Scenario Version*, you have stored scenario data for the primary objects.

Features

For market data and for the Source Data Layer, versions are created for scenario data and for real data. This is done using the time dimensions *business validity* and *time of entry in the system*. As a result, the system can read the related versions of the real data and the scenario data respectively and generate the stress data from them for a request with a key date and a time stamp. If the system does not find a suitable scenario version, it transfers the real data (unchanged) to the evaluation. If the system does not find a suitable real data version, it does not read any data.

Object/Data Category	Considerations for the Generation of Stress Data
Financial transaction, financial instrument, country, physical asset, account, position master data	Only the fields that are also activated in the scenario version are changed by the scenario. For service modules with multiple-value data (for example, valuations) the system always changes all of the data of the service module provided that this module is selected in the scenario template version.

Business partner	See Generating Stress Data for the Business Partner [page 226].
Classical market data	For classical market data and, for example, interest and exchange rates, see .
Generic market data	When the stress data is retrieved, all of the key figures of the relevant market data class are always changed by means of the scenario data (replaced and/or shifted). If both key figure values and shifts are specified in the scenario, the system applies the shifts to the key figure values in the scenario. Shifts for individual key figures can also have a value of zero. If this is the case, these key figures remain unchanged.

You can find an overview of the generated stress data on the SAP Easy Access screen for Bank Analyzer under *Source Data Layer* > *Primary Objects* > *<relevant primary object>* > *<relevant primary object>* > *Display Stress Data*.

7.8 XI Structure Mapping

Definition

A tool for using XI services in Bank Analyzer.

Use

In **inbound mapping**, XI structure mapping transfers the content of XML documents (XI messages) to Bank Analyzer structures. This XML document was first sent from an external system to an inbound XI service in Bank Analyzer.

In **outbound mapping**, XI structure mapping transfers the content of a Bank Analyzer structure to an XML document. Then, this document is sent to an external system using an outbound XI service in Bank Analyzer.

> **Note**
>
> Note that not all Bank Analyzer Web services use XI structure mapping.
>
> It is mainly the Web services that read data from the Results Data Layer (RDL) or write data to the RDL that use XI structure mapping.

For more information, see

- XML Document Structure, Representation in BA, BA Structure [page 1736]
- Mapping Rules [page 1737]

- Generic XI Source Fields [page 1739]

For more general information, see SAP Exchange Infrastructure.

7.8.1 XML Document Structure, Representation in BA, BA Structure

Use

In inbound mapping, XI structure mapping transfers an **XML document** - to be more precise, the **representation** of that XML document in Bank Analyzer - to a **Bank Analyzer structure**. In outbound mapping the reverse is the case.

Structure

You can display the structure of an XI message for a specific XI service in Enterprise Services Repository (ESR). The arrangement and the cardinality of the XML nodes in the XML document is determined in ESR.

For more information, see the Design section in the SAP NetWeaver Exchange Infrastructure.

The following is an excerpt from an XML document for the XI service for importing subledger documents:

```
<?xml version="1.0" encoding="UTF-8"?>
```

<FinancialInstrumentLedgerDocumentNotification_V1>

```
<MessageHeader><ID>MSG_ID_001</ID><CreationDateTime>2006-01-01</CreationDateTime></
MessageHeader><FinancialInstrumentLedgerDocument><ID
schemeAgencyID="SUBLEDGER_001">DOC_001</ID><SetOfBooksCode>IAS</
SetOfBooksCode><GroupID>GROUP_001</GroupID><CompanyID>LE_001</
CompanyID><PostingDate>2005-12-24</PostingDate>
```

[...]

Representation of an XML Document in Bank Analyzer

An XML document is first represented in the integrated systems as a nested ABAP structure. The field names of the structure are not the same as the names of the XML names due to the length restriction of 30 characters and the fact that the ABAP structure is not case sensitive. For example, the field name of the ABAP structure for XML node `<FinancialInstrumentLedgerDocument>` is `DOCUMENT`.

XI structure mapping identifies the XML nodes of an XI message by means of these field names and the context of the field. The context is made up of the superordinate field names of the respective node. The field names are run together and separated by a period. Therefore, the context of XI field `POSTING_DATE` for XML node `<PostingDate>` is `FIN_INSTRUMENT_LEDGER_DOC_MSG.DOCUMENT`.

> ⚠ Caution
>
> In this document, the term XML node refers both to an XML node in the usual sense (in other words, to an XML tag with opening and closing parentheses <>), and to the attributes that are listed within the tag (for example, schemeAgencyID in the example document above).

You can use transaction `SPROXY` to display the technical name of the ABAP structures as well as the generated ABAP classes for XI. The Bank Analyzer objects can be found under component `FSAPPL`.

Bank Analyzer Structure

Compared to the XML document, the Bank Analyzer structure is much less complicated. It often consists of flat DDIC structures, whose fields usually have the same technical names as characteristics and key figures. The structure is determined by Customizing for Bank Analyzer. Customizing for XI structure mapping requires the information from the fields of the Bank Analyzer entity.

Example

In the case of the inbound XI service for importing subledger documents, the Bank Analyzer structure is defined by Customizing for characteristics and key figures of an RDL result type under *Bank Analyzer* > *Results Data Layer* > *Basic Settings* > *Edit Data Structures in the Results Data Area*.

7.8.2 Mapping Rules

Mapping rules define the transfer of XML documents to Bank Analyzer structures and vice versa.

In both inbound and outbound mapping, XI structure mapping identifies fields from the XML document by means of field names and field contexts, and fields from the Bank Analyzer structure by means of a simple name. The latter is usually a characteristic or a key figure name.

For more information, see XML Document Structure, Representation in BA, BA Structure [page 1736].

> ℹ Note
>
> You can use a mapping rule set to group several mapping rules. By means of this group you can use different mapping rules depending on the Bank Analyzer entity. An example of this would be the result type used in Customizing for Bank Analyzer under *Results Data Layer* > *Settings for PI Interfaces* > *Define Results Data Area for PI Inbound Delivery*.

A mapping rule is used to populate exactly one target field. A mapping rule consists of a strategy and names one or more source fields. The strategy defines how XI structure mapping determines the value of the target field from the values of the source fields.

For example, you can choose the following strategies.

In each case only one source field is permitted:

- Copy
  This strategy copies the value of the XML node one-to-one to the field of a Bank Analyzer structure.
- Currency
  This strategy copies a currency in the XML document to a currency field of Bank Analyzer and transforms ISO currencies in the XML document to the (largely identical) SAP currency codes.
- Unit
  This strategy copies a unit in the XML document to a currency field of Bank Analyzer and transforms ISO units in the XML document to the (largely identical) SAP unit codes.
- Debit/Credit Indicator
  This strategy assigns the values 1 or 2 of the XI data type *DebitCreditCode* to the values D or C.

Up to five source fields are permitted for the following strategy:

- Derivation

As many source fields as required are permitted for the following strategy:

- BAdI

You define the strategies and the source fields in Customizing for Bank Analyzer under *Infrastructure* > *Settings for Enterprise Services* > *PI Structure Mapping*.

Use of the Derivation Strategy

When using the derivation strategy, note that the XI source fields or the XI target field (depending on the inbound or outbound mapping process used) are represented by placeholders. In other words, in the derivation tool, you do not use field **POSTING_DATE** and context **FIN_INSTRUMENT_LEDGER_DOC_MSG.DOCUMENT**, but a placeholder.

In inbound mapping, the name of the placeholder is the source field number and the field length that has to be considered in Customizing.

Additionally, the system calls the derivation tool for each target field that uses the derivation strategy separately.

Use of the BAdI Strategy

You can use the BAdI strategy to call a BAdI that transfers the content of the source fields to the respective target field. The number of source fields that can be used is not restricted in this case. You can use the BAdI to implement a mapping logic that cannot be implemented using the other strategies.

The prerequisite for using the BAdI Strategy is the implementation of BAdI definition **/BA1/AL_FX_STRATEGY** (enhancement spot **/BA1/AL_FX_STRATEGY_BADI** , ABAP interface **/BA1/IF_AL_FX_STRATEGY**).

Interface **/BA1/IF_AL_FX_STRATEGY** contains both method **EXECUTE_INBOUND** (inbound mapping) and **EXECUTE_OUTBOUND** (outbound mapping).

You have to register the BAdI implementation with a filter value and refer it to the ABAP class that implements the ABAP interface mentioned above.

In Customizing for XI Structure Mapping, you have to specify this filter value when you are defining the strategy.

7.8.3 Generic XI Source Fields

Use

A generic XI source field is an XML node that does not have fixed semantics. In other words, the content of generic XML nodes is not already known and can take on different values when the message is sent.

In contrast, there are XML nodes with fixed semantics that determine the meaning of the content. For example, the XML node **<PostingDate>** contains the posting date of a subledger document. Unlike generic XML nodes, this XML node cannot be used to transfer other information.

Example

The XML document for subledger documents has the generic XML node **<Property>**. In this case the semantics is determined by means of the content of the lower-level XML node **<ID>**. The content is stored in the XML node **<Value>**.

In the following example, holding category **HC_001** is transferred by means of the generic XML node **Property**. This is necessary as there is no dedicated XML node in the XML document for the holding category.

[...]

```
<Property><ID>HoldingCategory</ID><Value><NameSpecification><Name
languageCode="DE">HC_001</Name></NameSpecification></Value></Property>
```

[...]

Caution

You can use a generic XML node as a generic XI source field in inbound mapping only. You cannot populate generic XML nodes in outbound mapping.

Activities

If you want to use a generic XML node as a source field in inbound mapping, you have to make the following entries in Customizing for Bank Analyzer under *Infrastructure* > *Settings for Enterprise Services* > *PI Structure Mapping* > *Edit PI Structure Mapping* >:

1. Specify the XML node that contains the ID for the content.
   In the example above, the ID of the content is in XML node <ID>.
2. Specify the ID you require.
   In the example above, the required ID is *HoldingCategory*.
3. Specify where the actual content can be found.
   In the example, the content is in the XML node <Name> and the value is HC_001.

7.9 Portfolios

Definition

In Bank Analyzer, a portfolio generally represents a collection of transactions arranged in a hierarchy. These transactions can be processed, analyzed, and evaluated within the portfolio. You have the option of defining partial portfolios within a portfolio.

A portfolio in Bank Analyzer consists of the following entities:

- Portfolio set
- Portfolio hierarchy
- Composite portfolio
- Leaf portfolio
- Portfolio element

Structure

Bank Analyzer uses portfolios for the following applications:

- Portfolio for Credit Portfolio Data Processing [page 1740] (CPDP)
  This application defines portfolios using selection criteria.

7.9.1 Portfolios for Credit Portfolio Data Processing

Use

You can send portfolios to a connected credit risk model (for example, RiskFrontier from Moody's KMV). This allows you to calculate credit risk key figures for a particular portfolio. In addition to receiving regular evaluations with constant portfolios, you can use what-if scenarios to change the definition of portfolios and find out how these changes affect the credit risk key figures. Since this process may be iterative and involve frequent adjustments to portfolio definitions, portfolios are defined in the operational system via the application menu directly. To define portfolios, go to the SAP Easy Access screen and choose *Bank Analyzer* > *Infrastructure* > *Portfolios* > *Current Settings* > *Define Portfolios by means of Selection Criteria*.

Portfolios can therefore be considered as extremely flexible collections of financial instruments, transactions, and positions that can change very frequently. Since the analysis only ever covers one portfolio, hierarchies are not necessary here.

The portfolio set defined in Customizing acts as a technical bracket for the different portfolio defintions in this context.

Within a portfolio set, the following provisions apply in relation to a credit risk model:

- Only leaf portfolios exist; there are no composite portfolios
- Each leaf portfolio contains a set of selection criteria. All selection options together define a single entity of portfolio elements (financial transactions, instruments, and positions)
- You can define leaf portfolios independently of each other. The leaf portfolios can overlap. There are no restrictions on the definition of selection criteria

Prerequisites

You have made the required settings in Customizing for Bank Analyzer under *Infrastructure* > *Portfolios*.

Process

The selection criteria are linked to the leaf portfolios; there is no persistent connection between portfolio elements and leaf portfolios. This means the system uses the selection criteria for the respective process to find the portfolio elements to be analyzed and extract them to the external credit risk model.

7.9.1.1 Definition of the Portfolio Using Selection Criteria

Use

You can use this function to define a portfolio for Credit Portfolio Data Processing [page 1292] (CPDP) by specifying suitable selection criteria. The portfolio you create here forms part of a greater portfolio that you have already sent to the connected credit portfolio model. This allows you to integrate parts of a greater portfolio separately.

In CPDP, you can use the ad hoc extraction transaction on the *SAP Easy Access* screen to send the portfolio you create here to the credit portfolio model. To do so, choose *Processes and Methods* > *Credit Risk* > *Credit Portfolio Data Processing* > *Ad-Hoc Calculation*. This does not cause the system to transfer the data of each individual transaction to the connected credit portfolio model again. Instead, the connected credit portfolio model uses the transaction data that the system has already sent together with the definition of the greater portfolio.

Prerequisites

You have defined a portfolio set in Customizing for *Bank Analyzer* under *Infrastructure* > *Portfolios* > *Define Portfolio Set*.

You have successfully executed a credit exposure run and have already sent a portfolio of which the portfolio created here is a part to the connected credit portfolio model.

7.10 Data Aging in Bank Analyzer

You can use data aging in Bank Analyzer to move large volumes of data in the Results Data Layer (RDL) and the Source Data Layer (SDL) within a database in order to gain more working memory. You control the movement of data by specifying a data temperature. Data that is only used rarely or that is no longer accessed can be moved from a "hot area" (partition in the working memory) to a "cold area" (partitions on the hard drive). Data that is frequently used remains in the hot area.

Prerequisites

You have familiarized yourself with the general data aging process and with the general prerequisites for data aging. For more information, see http://help.sap.com/nw *SAP NetWeaver Platform* *Function-Oriented View* *Solution Life Cycle Management*.

Data Aging Process in Bank Analyzer

Planning

Database tables involved

For the SDL, the data aging objects and partitioning objects are provided **with** assigned database tables. If you use SGS characteristic structures in business transaction classes and item classes, and you have set the *Data Aging Field in SGS* indicator in the Customizing activity *Edit Characteristic Structure* and have **not** set the *Unique* indicator, the system generates additional enhancements for the data aging objects and generated SGS database tables are assigned to them. Partitioning objects are also generated for each of the characteristic structures to which generated database tables are also assigned. The generated partitioning objects are assigned below the delivered partitioning object `/BA1/F2_035`.

> **Note**
>
> The report *Generation of Data Aging Objects for BT Classes and Item Classes* (`/BA1/F2_BT_DAAG_GENERATE`) is provided to generate the enhancements for data aging objects and partitioning objects in the SDL across the system and to find and correct any inconsistencies.

For the RDL, the data aging objects and partitioning objects are provided **without** assigned database tables. If you make settings in Customizing for the Results Data Layer, the system generates enhancements for the data aging objects to which generated database tables are assigned. Partitioning objects are also generated for the individual Customizing entries to which generated database tables are also assigned. The generated partitioning objects are assigned below delivered partitioning objects.

Configuring rules for residence times

Define the condition fields as well as the derivation rules and the residence times for the individual data aging objects in Customizing for Bank Analyzer under *Infrastructure* *Data Aging* *Edit Rules for Residence Times for Data Aging Objects*. When you make changes in SDL Customizing, for every supported business transaction

class, the system creates a derivation module each for production control and for the instrument type. When you change RDL Customizing the system generates a derivation module for each result type that is supported and for each aggregation level.

You have to define the residence times for each data aging object and derivation module ID according to fixed condition fields. If necessary, you can also use optional condition fields to define different residence times due to varying increases in the volume of data or because the data needs to be accessed for different reasons.

Example

You need to access the data for RDL business transactions differently depending on the financial position type. Retroactive business transactions or reversals with a posting date older than one month are rare for a specific financial position type. However, retroactive events of up to three months are common for all the other financial position types. For this reason, you define the *Financial Position Type* field as an optional condition field and you define residence times of `30 days` and `90 days` for the different financial position types.

In the case of RDL business transactions, note the dependency between preaggregation and data aging. For more information, see the documentation for the data aging object RDL: Business Transaction [page 1748].

Migration of Run Administration for RDL Preaggregation

When you implement data aging, under certain conditions you can continue to use existing aggregation levels. To do so, you may need to migrate the run administration for RDL preaggregation for the aggregation levels using the report `/BA1/R_AL_HW_DAAG_PREAGG_MIGR`. For more information, see the documentation for the reports in the system.

Execution

During data aging runs, the system selects the data records whose residence time has expired and that can therefore be moved. In Bank Analyzer, this selection is made using fixed and, if necessary, optional condition fields. Bank Analyzer-specific statistics are updated to ensure optimum access to data in the hot and cold area for each data aging object. For more information, see Bank Analyzer-Specific Statistics for Data Aging [page 1744].

For runs to undo data aging, you can use the data aging selection ID in Bank Analyzer as a selection reference. The data aging selection IDs that have been defined for a data aging object are provided in the input help for the *Selection Reference* field.

Available Data Aging Objects

You can use the following data aging objects:

- SDL: Business Transaction [page 1746]
- RDL: Business Transaction [page 1748]
- RDL: Aggregation Level Category "Balance Level" [page 1751]
- RDL: Aggregation Level Category “Balances with Two Dates” [page 1752]
- RDL: Generic Aggregation Level [page 1753]

Related Information

Aggregation Levels [page 1751]

7.10.1 Bank Analyzer-Specific Statistics for Data Aging

Definition

To ensure optimum access to data, it is vital that the system can recognize the date from which it has to access not only the hot area but also the cold area. When the system accesses the cold area, it also has to be able to recognize the lower date limit up to which data has to be accessed. To ensure this optimum access, statistics are updated during data aging runs. In addition to the standard statistics, two Bank Analyzer-specific statistics are required since data aging takes place in Bank Analyzer with the use of fixed and, if necessary, optional condition fields.

Boundary Date Statistics

The business reference date is saved from these statistics for each data aging object and data aging run. This is the date as of which the system has to select data from the hot area and also from the cold area. The boundary date can be determined in different ways depending on the data aging object.

Example

In the case of the *RDL: Business Transaction* data aging object, the boundary date is whichever posting date or value date of all data records moved in a data aging run is the latest (most recent). Business transactions whose posting date and value date are earlier or the same as the boundary date most likely lie in the cold area.

In addition to the **run-dependent** boundary date, the latest **run-independent** boundary date is also determined when the boundary date statistics are accessed.

Example

You want to display all the RDL business transactions whose posting date lies between January 1, 2015 and March 31, 2015 and that meet other selection criteria. The run-independent boundary date on which business transactions that meet your selection criteria were moved to the cold area is January 31, 2015. Therefore, the data temperature is set to January 1, 2015. The system then accesses the hot area for the business transactions that were posted from February 1, 2015 to March 31, 2015 and also accesses the cold area for the business transactions that were posted from January 1, 2015 to January 31, 2015.

Note

From one data aging run to the next data aging run, the boundary date (for the same data aging object) can shift to the future if the residence time for data records with a later reference date has since elapsed. The boundary date can also shift into the past if data records have since been moved back to the hot area by a run to

undo data aging. For this reason, the latest, run-independent boundary date is always used as the basis for selecting data.

Creation Date Statistics

These statistics combine technical information about the creation date and the business-related reference date. The system saves the minimum reference date in these statistics as a technical creation date for each data aging object and data aging run. This is the date up to which the system has to select data from the cold area. This minimum reference date is always relevant when the system uses the technical creation date to select data. It can be determined in different ways depending on the data aging object. To ensure that the minimum reference date lies in the cold area, the system checks whether the date is earlier than the run-independent boundary date (from the boundary date statistics).

In addition to the **run-dependent** minimum reference date, the latest **run-independent** minimum reference date is also determined when the creation date statistics are accessed.

> **Note**
>
> From one data aging run to the next data aging run, the minimum reference date (for the same data aging object) can shift to the past if there is a return posting or a reversal, for example. For this reason, the earliest run-independent minimum reference date is always used as the basis for selecting data.

Display Statistics

You can display the statistics as follows:

1. Call transaction `/BA1/DA_STATS`.
2. Enter the required data on the initial screen.

   > **Note**
   >
   > If you also want to display statistics records that have lost their validity due to an undo run in the boundary date statistics, select the *Include Records from Undo Run* checkbox in the *Options* group box. Through the boundary date, the checkbox also indirectly has an effect on the creation date statistics.

3. Choose *Execute*.
4. On the results screen, select the *Boundary Date Statistics* tab page or the *Creation Date Statistics* tab page. The system provides an overview of all data aging runs, of the related optional fields, and of the boundary or creation dates.
   The *Validity* column contains the following information:
   - Green traffic light: The data aging run is current.
   - Yellow traffic light:
     - Boundary date statistics: The boundary date displayed in the *Latest Date* field is no longer current because there is at least one data aging run with a more recent boundary date. The current boundary date is displayed in the *Run-Independent Latest Date* field.

- Creation date statistics: The minimum reference date displayed in the *Oldest Reference Date* field is no longer current because there is at least one data aging run with an earlier minimum reference date. The current minimum reference date is displayed in the *Run-Ind. Minimum Ref. Date* field.
- Red traffic light:
  - Boundary Date Statistics: The boundary date is no longer valid because (at least) one run to undo data aging exists that has moved the valid boundary date into the past.
  - Creation Date Statistics: The minimum reference date does not lead to accesses in the cold area because there is no run-independent boundary date that is more recent than the minimum reference date.

Example

You want to transfer all RDL business transaction that were created up to the technical creation time 2015-01-02, 23:59:59 and that meet further selection criteria to the general ledger. The last transfer to the general ledger included business transactions up to technical creation date 2014-12-31, 23:59:59. For the dates January 1, 2015 and January 2, 2015 the lowest run-independent minimum reference date October 15, 2014 is determined from the creation date statistics. The data temperature is therefore set to October 15, 2014. The run-independent boundary date determined from the boundary date statistics is January 2, 2015. Because the system finds data records with a reference date earlier or the same as January 2, 2014, it accesses the cold area as well as the hot area. When it selects data from the cold area, the system considers only business transactions that have a reference date that is after or the same as October 15, 2014.

7.10.2 SDL: Business Transaction

Technical Data

Primary Object	Business transaction
Name of data aging object	SDL: Business Transaction
Technical name of data aging object	`/BA1/F2_035`
Category of data aging object	Application
Reference date	Business record date
Fixed condition fields	Business transaction class
Optional condition fields	• Source system • Legal entity • For business transactions for financial transactions: Production control • For business transactions for financial instruments: Instrument type
Versioning schema	Single versioning with reversal

Participating tables	Static tables: • `/BA1/F2_BT_H` (header table for business transaction) • `/BA1/F2_BT_P` (item table for business transaction) • `/BA1/F2_BT_KF` (key figures for business transactions) • `/BA1/FG_ABT_ASG` (assignment table for aggregation business transactions) Generated tables for the segmentation service (SGS): • Header table • Item table • Table for consistent access

Use

You can use this data aging object to specify that the system moves business transactions in the SDL to the cold area during a data aging run.

Prerequisites

- You have made sure that processing of the business transactions has been completed **before** the corresponding data aging runs. This ensures that regular jobs, such as source data aggregation, do not have to access the cold area unnecessarily.

> **Note**
>
> Business transactions that are persisted in a flat table cannot be moved to the cold area.

Dependencies

Segmentation service (SGS)

The data temperature for entries in the SGS table for consistent access is set to the business record date during data aging. The data temperature for entries in the SGS table for header and item data is set to the system date of the data aging run during data aging.

Residence times

Reversals are subject to the same residence times as the underlying business transactions. A business transaction, the corresponding reversal business transaction and any reversal of this reversal business transaction are not included in data aging until the residence time has passed for the newest reversal.

External business transactions are included in data aging only if the posting has been completed successfully.

Result

If you want to access business transactions in dialog mode and do not enter any dates or times, a dialog box appears in which you are prompted to enter the lower date limit for the selection of data in the cold area. When you post external business transactions, you can set the new *Read Access to Hot Only* checkbox on the initial screen if required.

If you want to access business transactions using Business Application Programming Interfaces (BAPIs) or Enterprise Services and do not transfer any dates or times, the system reads the data from the hot **and** cold area.

Related Information

Data Aging in Bank Analyzer [page 1742]

7.10.3 RDL: Business Transaction

Technical Data

Results data object	Flow result (HF)
Name of data aging object	RDL: Business Transaction
Technical name of data aging object	`/BA1/HF`
Category of data aging object	Application
Reference date	Posting date or value date (depending on which is the latest date)
Fixed condition fields	Results data area, result type
Optional condition fields	Legal entity, financial position type
Versioning schema	Single versioning with reversal
Participating tables	Generated

Use

You can use this data aging object to specify that the system moves business transactions in the RDL to the cold area during a data aging run.

Prerequisites

- You have made sure that processing of the business transactions has been completed **before** the corresponding data aging runs. This ensures that regular jobs, such as general ledger transfer, do not have to access the cold area unnecessarily.
- In the job administration, you have scheduled the preaggregation runs for business transactions **before** the corresponding data aging runs. This is necessary because when it selects business transactions during the data aging run the system takes into account the residence time **and** the corresponding preaggregation runs. We recommend that you schedule preaggregation runs and data aging runs at the end of posting periods.
- If you have defined different residence times for a data aging object according to the combination of values in the condition fields mentioned above, you have to define data aging selection IDs in Customizing. You can select these data aging selection IDs as a selection criterion on the initial screen for the preaggregation run.

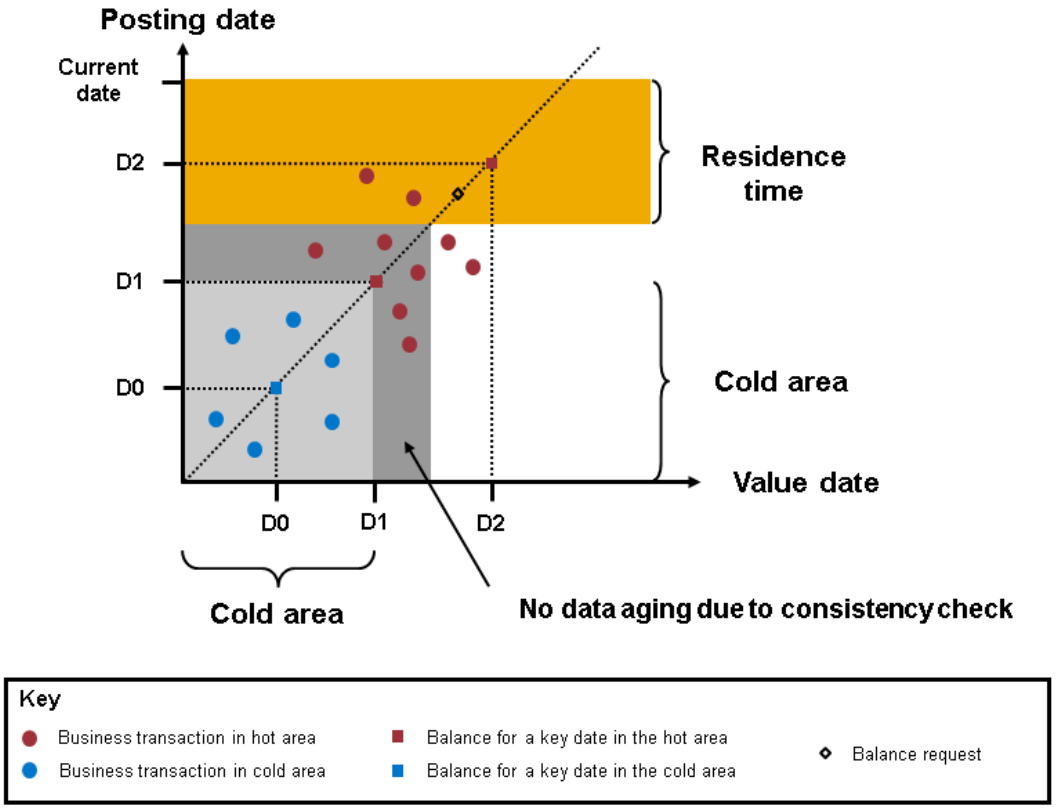
> **Note**
>
> Business transactions that are persisted in a flat table cannot be moved to the cold area.

Dependency Between Preaggregation and Data Aging

The applications do not access individual RDL business transactions in the cold area but only preaggregates in the cold area. The preaggregates are used as starting points to keep the number of accesses to the cold area to a minimum. Therefore, in the data aging process, only business transactions are included for which a preaggregation run has been executed.

To also avoid errors when balances are calculated, the system ensures that RDL business transactions that belong to the same aggregation level either remain in the hot area or are moved completely to the cold area.

The figure below provides an example of the system's handling of RDL business transactions and the aggregation level category *Balance Level*.



The figure shows the results of preaggregation runs for the three key dates D0, D1, and D2. The RDL business transactions whose residence time has expired and whose latest posting and value date is earlier than the current system date are in the light gray **and** dark gray area. For the business transactions in the light gray area, preaggregation runs have already taken place on the key dates D0 and D1 after the residence time has expired. For the business transactions in the dark gray area, a preaggregation run has taken place during the residence time on key date D2. If the business transactions in the dark gray area were also moved into the cold area, it would lead to incorrect results in the calculation of balances for key dates between D1 and D2. During calculation, the system would access only the preaggregates in the cold area (light gray area) but not individual business transactions in the cold area (dark gray area). For this reason, the system runs a consistency check during data aging to make sure that only the RDL business transactions that lie below balance level value D1 are moved to the cold area. When it calculates balances, the system then accesses the preaggragates in the cold area and the individual business transactions in the hot area.

Related Information

Data Aging in Bank Analyzer [page 1742]
Preaggregation [page 1408]

7.10.4 Aggregation Levels

During data aging the system supports aggregation levels for which the following conditions apply:

- The underlying result type is based on the result category *Business Transaction* (`HF`).
  The aggregation level is assigned to the result category *BALLV*, to the level category *QUBAL*, or to no level category (generic level category).
- Neither the *Direct Aggregation Without Preaggregation* or the *Obsolete Aggregation Level* checkbox has been selected for the aggregation levels in the Customizing activity *Edit Data Structures in Results Data Area*.

Prerequisites

- You have moved the corresponding business transactions to the cold area.
- If you have defined optional condition fields for the data aging object *RDL: Business Transaction* in Customizing for Bank Analyzer under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*, note the following:
  - You have to define the optional condition fields also as optional condition fields for the data aging objects for the aggregation levels.
  - You have to define the optional condition fields also as granularity characteristics in the corresponding aggregation levels.

More Information

For more information, see Data Aging in Bank Analyzer [page 1742] and the documentation for the following data aging objects:

- RDL: Aggregation Level Category "Balance Level" [page 1751]
  RDL: Aggregation Level Category “Balances with Two Dates” [page 1752]
- RDL: Generic Aggregation Level [page 1753]

7.10.4.1 RDL: Aggregation Level Category “Balance Level”

Technical Data

Results data object	Aggregation level (BALLV - balance level)
Name of data aging object	RDL: Aggregation Level Category: Balance Level
Technical name of data aging object	/BA1/HL_BALLV

Category of data aging object	Application
Reference date	Key date (from RDL versioning schema)
Fixed condition fields	Results data area, aggregation level
Optional condition fields	Fields as for underlying result type
Versioning schema	Two-dimensional versioning
Participating tables	Generated

Use

You can use this data aging object to move to the cold area aggregation levels that belong to the aggregation level category *Balance Level*.

Related Information

Aggregation Levels [page 1751]

7.10.4.2 RDL: Aggregation Level Category “Balances with Two Dates”

Technical Data

Results data object	Aggregation level (QUBAL - balance with two dates)
Name of data aging object	RDL: Aggregation Level Category: Balances with Two Dates
Technical name of data aging object	/BA1/HL_QUBAL
Category of data aging object	Application
Reference date	Key date (from RDL versioning schema)
Fixed condition fields	Results data area, aggregation level
Optional condition fields	Fields as for underlying result type
Versioning schema	Two-dimensional versioning

Participating tables	Generated

Use

You can use this data aging object to move to the cold area aggregation levels that belong to the aggregation level category *Balances with Two Dates*.

Related Information

Aggregation Levels [page 1751]

7.10.4.3 RDL: Generic Aggregation Level

Technical Data

Results data object	Aggregation level (initial)
Name of data aging object	RDL: Generic Aggregation Level
Technical name of data aging object	/BA1/HL_TOTAL1D
Category of data aging object	Application
Reference date	Is determined based on granularity characteristics (such as the last date of the period or the financial year)
Fixed condition fields	Results data area, aggregation level
Optional condition fields	Fields as for underlying result type
Versioning schema	One-dimensional versioning
Participating tables	Generated

Use

You can use this data aging object to move the total flow results to the cold area.

Prerequisites

If required, in Customizing for Bank Analyzer under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*, you have selected the *Data Aging Reference Date* checkbox for the dates that you want to use to determine the reference date.

Related Information

Aggregation Levels [page 1751]

7.11 Optimizing Main Memory for Smart Accounting for Financial Instruments

If you want to optimize the main memory usage in SAP HANA for Smart AFI, contact your SAP contact person or create a customer incident under the component `FS-BA-PM-SFA`.

7.12 Data Archiving and Data Deletion in Bank Analyzer

7.12.1 Archiving with ILM-Enabled Archiving Objects

In Bank Analyzer, you can use the following archiving objects with *SAP Information Lifecycle Management (SAP ILM)*. You can use SAP ILM to control the blocking and deletion of business partner data.

The ILM objects in Bank Analyzer support retention rules (policy category `RTP`).

Moreover, all ILM objects in Bank Analyzer support the following ILM actions:

- Archiving
- Snapshot
- Data destruction

ILM-Enabled Archiving Objects in Bank Analyzer

Archiving Object/ILM Object	Available Time References	Available Condition Fields	Business Object
BA1_F1_040 Bank Analyzer: Archiving of Financial Transactions	CREATION_DATE Creation Date	ACC_LEGENT Legal Entity	/BA1/B100 Financial Transaction
BA1_F1_041 Bank Analyzer: Archiving of Financial Instruments	CREATION_DATE Creation Date	ACC_LEGENT Legal Entity	/BA1/B101 Financial Instrument
BA1_F1_042 Bank Analyzer: Archiving of Settlement Accounts	CREATION_DATE Creation Date	None	/BA1/B104 Account
BA1_F1_043 Bank Analyzer: Archiving of Securities Pos. Acct	CREATION_DATE Creation Date	ACC_LEGENT Legal Entity	/BA1/B104 Account
BA1_F1_044 Bank Analyzer: Archiving of G/L Accounts	CREATION_DATE Creation Date	ACC_LEGENT Legal Entity	/BA1/B104 Account
BA1_F1_045 Bank Analyzer: Archiving Customer Position Acct.	CREATION_DATE Creation Date	ACC_LEGENT Legal Entity	/BA1/B104 Account
BA1_F1_046 Bank Analyzer: Archiving of Country	CREATION_DATE Creation Date	ACC_LEGENT Legal Entity	/BA1/B113
BA1_F1_048 Bank Analyzer: Archiving for Physical Assets	CREATION_DATE Creation Date	ACC_LEGENT Legal Entity	/BA1/B118 Physical Asset
BA1_F1_049 Bank Analyzer: Archiving of Position Master Data	CREATION_DATE Creation Date	ACC_LEGENT Legal Entity	None
BA1_F2_030 Bank Analyzer: Position	CREATION_DATE Creation Date	POSITION_CLASS Position Class	/BA1/B102 Position Data

Archiving Object/ILM Object	Available Time References	Available Condition Fields	Business Object
BA1_F2_035 Bank Analyzer: Business Transaction	CREATION_DATE Creation Date	ACC_LEGENT Legal Entity	/BA1/B111 Business Transaction
BA1_F5_016 Bank Analyzer: Archiving of Generated Market Data	CREATION_DATE Creation Date	GMD_AREA Generic Market Data Area	/BA1/B128 Generic Market Data
BA1_R4_054 RDB: Archiving for Balance Object Manager	CREATION_DATE Creation Date	None	None
BA1_R4_055 RDB: Archiving for Balance Analyzer	CREATION_DATE Creation Date	None	None
BA1_R4_063* RDB: Archiving for Credit Exposure 2	CREATION_DATE Creation Date	None	None
BA1_R4_064* RDB: Archiving of HDB - Scenario Data Storage	CREATION_DATE Creation Date	None	None
BA1_R4_065* RDB: Archiving HDB - Review Data Storage	CREATION_DATE Creation Date	None	None
BA1_R4_066* RDB: Archiving HDB - Historical Data Storage	CREATION_DATE Creation Date	None	None
BA1_R4_067* RDB: Archiving HDB - Data Processing Storage	CREATION_DATE Creation Date	None	None
BA1_R4_068* RDB: Archiving of GM Data - Default Data	CREATION_DATE Creation Date	None	None
BA1_R4_069* RDB: Archiving of GM Data - Default Stress Data	CREATION_DATE Creation Date	None	None

Archiving Object/ILM Object	Available Time References	Available Condition Fields	Business Object
BA1_R4_071* RDB: Archiving for Pooling	CREATION_DATE Creation Date	None	None
BA1_R4_072* RDB: Archiving for Facility Distribution	CREATION_DATE Creation Date	None	None
BA1_R4_073* RDB: Archiving for Collateral Distribution	CREATION_DATE Creation Date	None	None
BA1_R4_074* RDB: Archiving for Distribution of Free Line	CREATION_DATE Creation Date	None	None
BA1_PFW_F1* Bank Analyzer: PFW Runs (FDB1)	CREATION_DATE Creation Date	None	None
BA1_PFW_F2* Bank Analyzer: PFW Runs (FDB2)	CREATION_DATE Creation Date	None	None
BA1_PFW_F3* Bank Analyzer: PFW Runs (FDB3)	CREATION_DATE Creation Date	None	None
BA1_PFW_S1* Bank Analyzer: Processing Framework Runs (S1RR)	CREATION_DATE Creation Date	None	None
BA1_R3_LM* Archive Limits	CREATION_DATE Creation Date	None	None
BA1_R3_UT* Archive Limit Utilizations	CREATION_DATE Creation Date	None	None
BA1_B0_FPO Bank Analyzer: Financial Position Object	LAST_CHANGE_DATE Last Change Date	• ACC_LEGENT Legal Entity • ACC_SYSTEM Accounting System	None

Archiving Object/ILM Object	Available Time References	Available Condition Fields	Business Object
BA1_B7_AT Bank Analyzer: Aggregation Transaction	CREATION_DATE Creation Date	None	None
BA1_B8_GL Bank Analyzer: GL Connector	CREATION_DATE Creation Date	ACC_LEGENT Legal Entity	None
BA1_P4_DT Bank Analyzer: Rule Entries of Derivation Tool	CREATION_DATE Creation Date	None	None
BA1_FK_COR Bank Analyzer: Archiving of Correction Entries	CREATION_DATE Creation Date	CORR_CMP Correction Component	None
BA1_FK_DFW Bank Analyzer: Archiving of Data Flow Entries	CREATION_DATE Creation Date	LEAD_CORR_CMP Leading Correction Component	None
BA1_R0_FVS Bank Analyzer: Detail Log for Fair Value Server	CREATION_DATE Creation Date	ACC_LEGENT Legal Entity	None
BA1_R4_056 RDB: Archiving for Effectiveness Test	CREATION_DATE Creation Date	None	None
BA1_R4_057 RDB: Archiving of Portfolio Items	CREATION_DATE Creation Date	None	None
BA1HM_HR Hedging Relationships in Hedge Management	LAST_CHANGE_DATE Last Change Date	• ACC_LEGENT Legal Entity • ACC_SYSTEM Accounting System • BUSINESS_STATE Business Status	None

Archiving Object/ILM Object	Available Time References	Available Condition Fields	Business Object
BABP_FULL Analytical Business Partner (Complete)	START_RET_DATE Start of Retention Period	• APPL_NAME Application Name • APPL_RULE_VARIANT Application Rule Variant • BP_TYPE Partner Category • LEGAL_ENTITY Legal Entity	BUS1006 Business Partner
BABP_SCENA Analytical Business Partner (Scenario)	CREATION_DATE Creation Date	• LEGAL_ENTITY Legal Entity • SCENARIO Scenario	BUS1006 Business Partner
FSBP_VERSN Archive BP Versions Based on Key Date	CREATION_DATE Creation Date	LEGAL_ENTITY Legal Entity	BUS1006 Business Partner
Generated archiving objects or ILM objects RDL results data	See Archiving and Destroying Results Data [page 1395]	LEGAL_ENTITY Legal Entity	None

> **Note**
>
> * SAP does not provide defined condition fields for these archiving objects. You can define individual condition fields using an implementation of the Business Add-In BADI_IRM_OT_FLD.

Prerequisites

You must do the following before you can use the block and delete function:

- Configure *SAP Information Lifecycle Management*.
- Activate the business function *Information Lifecycle Management (ILM)*.

More Information

-
-
-
-

7.12.1.1 Archiving of Rule Entries of the Derivation Tool

Definition

Archiving object `/BA1/P4_DT` (*Bank Analyzer: Rule Entries of Derivation Tool*) is used to archive and delete rule entries of the derivation tool.

Use

The data for rule entries for the derivation tool that can be archived are is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Table

The system uses archiving object `/BA1/P4_DT` to archive data from various tables. For information about how to display table names, see .

Programs

The following programs are available for /BA1/P4_DT:

Program	Function
`/BA1/P4_R_ARCHIVE_WRITE`	Write
`/BA1/P4_R_ARCHIVE_DELETE`	Delete

The standard variants SAP PROD (productive mode) and SAP TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.12.1.2 Archiving and Destroying Results Data

Use

You can archive results data using the Archiving Engine and destroy it as part of Information Lifecycle Management (ILM).

The number of generated archiving objects and ILM objects for the results data depends on the versioning [page 1378] used and the Persistence [page 1393]:

- For results data that uses single versioning with reversal, there is only one archiving object or one ILM object for each result type because the system always archives or destroys one result completely with all of its versions.
- For results data that uses versioning (one-dimensional versioning, two-dimensional versioning or result group versioning) with persistence schema *00*, there are two archiving objects and two ILM objects.
  One of the archiving objects or ILM objects is used for archiving or destroying result versions, while the other is used for archiving or destroying the header data that groups all versions. The header data can only be archived or destroyed if the result is to be archived completely and all versions have previously been archived or destroyed.
- For results data that uses one-dimensional versioning with persistence schema *03*, two-dimensional versioning with persistence schema *02*, *03* and *04*, and result group versioning with persistence schema *03* there is one archiving object and one ILM object. These objects are used for the archiving and destruction of data records from the RDL. The number of generated archiving objects and ILM objects for the results data depends on the versioning used.
- For results data that uses result type HFSPD, there is only one archiving object or one ILM object for three database tables.
  For more information about this result type, see Result Types and Result Categories [page 1367].

Prerequisites

- To execute archiving or data destruction, there must first be archiving objects and ILM objects in the system for each result type. You enter the data for aggregation in Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Technical Settings in Results Data Area*. In the *Archiving for Result Type* dialog, you can set the indicator for archiving. When you save, the system automatically generates archiving objects and ILM objects. You can display these objects for each result type in Customizing for *Bank Analyzer* under *Results Data Layer* > *Basic Settings* > *Edit Data Structures in Results Data Area*.
  When you save, the system also generates the required Customizing settings for the ILM objects. The transaction `IRM_CUST` contains the Legal Entity condition field (`ACC_LEGENT`) for ILM objects of versions. Condition fields are not required for the generated ILM objects of header tables.
- In transaction `ILMARA` (Audit Area Processing), you need to assign an audit area to each ILM object.
- In transaction `IRMPOL` (ILM Policies), you define the retention rules for your audit area for each ILM object.

Procedure

1. Archiving analysis phase:
   The system determines the archiving status of an object to be archived. It does this by running the following checks:
   - Residence time check: An object can be archived only if the actual date is later than the posting date, or of the key date of the object plus the residence time in the system. The former is the case for flow results and imported subledger documents and the latter is the case for the other result categories. When it calculates the residence time, the system uses the system date as the start date for header data and the business key date as the start date for versions.

- Business check: The system can archive an object only if it is no longer used. The versions of an object are always checked. The system uses BAdIs to check a complete result. The following BAdIs are available for each archiving object in Customizing for *Bank Analyzer* under *Results Data Layer* > *Archiving*:
  `/BA1/AL_HW_ARCH_ODV` for one-dimensional versioning
  `/BA1/AL_HW_ARCH_ODV_PS3` for one-dimensional versioning with persistence schema *03*
  `/BA1/AL_HW_ARCH_MV` for two-dimensional versioning
  `/BA1/AL_HW_ARCH_MV_PS02` for two-dimensional versioning with persistence schema *02*
  `/BA1/AL_HW_ARCH_MV_PS34` for two-dimensional versioning with persistence schema *03* and *04*
  `/BA1/AL_HW_ARCH_SV` for single versioning with reversal
  `/BA1/AL_HW_ARCH_RGV` for result group versioning
  `/BA1/AL_HW_ARCH_RGV_PS3` for result group versioning with persistence schema *03*

  The system sets only objects that are run successfully through both checks to *can be archived*. The other objects are set to *cannot be archived* and a resubmission date is set. The resubmission date determines when a new archiving check for an object is run. In this way the amount of data to be archived can be reduced to specific time periods.
  The resubmission date is found by comparing the following two date combinations and taking the later date: Reference date + residence time + 1 and current date + resubmission period.
  You can define the resubmission period in the archiving settings in Customizing.
2. Write: The system writes the objects that can be archived from the database into the archive.
3. Delete: The system deletes the objects that were written to the archive from the database.
4. Destroy: The objects are marked as destructible and can then be destroyed with the deletion run. The objects are also deleted from the archive; they cannot be reloaded.

7.12.1.3 Archiving of Limit Data

A large amount of data is saved in the system when Limit Manager is in productive use. To prevent evaluations being slowed down by too much data, you can use the to delete the following data from the system:

- `/BA1/R3_LM` (*Archive Limits*)
- `/BA1/R3_UT` (*Archive Limit Utilizations*)

When data is archived, it is removed from the system. You can display your archived data by calling the Archive Information System. Displaying this archived data can seriously impair system performance, however.

> **Note**
>
> In addition to archiving data, you can delete data permanently using one of the following reorganization functions in Customizing under *Bank Analyzer* > *Analytics* > *Limit Manager* > *Reorganization in Limit Manager*:
>
> - Reorganize Limit Types
> - Reorganize Utilizations

7.12.1.4 Archiving of Financial Position Objects

Definition

Archiving object `BA1_B0_FPO` (*Bank Analyzer: Financial Position Object*) is used to archive, delete, and reload financial position objects.

Use

There are archive writing and deletion programs for this object. Archived data can also be read. The data that can be archived for financial position objects is written to one or multiple archive files. So that the system archives only the data that is no longer required, the following criterion has to be met:

- The corresponding financial position balances (object `BA1_B1_FPT`) have been archived.

Structure

Tables

The archiving object archives data from the following financial position object tables:

- `/BA1/B1_FP_MD`
- `/BA1/B1_FP_MDH`
- `/BA1/B1_FP_STAT`

For information about how to display table names, see .

Programs

The following programs are available for BA1_B0_FP0.

Program	Function
`/BA1/RB1_FP_AR_WRITE`	Write
`/BA1/RB1_FP_AR_DELETE`	Delete
`/BA1/RB1_FP_AR_RELOAD`	Reload

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

Procedure for writing data to the archive

Make sure that the options such as the size of the archive file, variants for the test run and productive run for the deletion program are correct (see Environment -> Customizing).

Recommendation

We recommend the following settings:

- Maximum MB: 50-100
- Maximum number of data objects: 0

Hence only the file size is important.

Make sure that you select the *Test Mode* option in the test run variant, and the *Production Mode* option in the variant for the production run. You use the *Detail Log* option in your variants for the deletion run to define whether the system is to display the deleted financial position objects in a list.

You start archiving runs for one legal entity and one accounting system at a time. The system archives all the financial position objects for the accounting system and legal entity whose accounting objects have become inactive because the residence time specified in Customizing was exceeded.

You write data to the archive as follows:

1. Create a variant and choose *Maintain*. The system displays the *Variant Maintenance* menu.
2. Enter data for the following fields:
   Required fields:
   - *Legal Entity*
   - *Accounting System*

   Optional fields:
   - Detail Log
   - *Processing Options* (test or production mode)
   - Archiving session note so you can identify the run at a later point in time.
3. Save your variant, and schedule the job.

Procedure for reloading data

Recommendation

We recommend that you reload archived data only in exceptional circumstances.

1. Create a variant for the test or productive run, and save the variant.
2. Choose *Archive Selection* to select an archiving run.
3. Enter the reload variant and choose *Execute*.
   The system reloads the financial positions objects from the archiving run you selected.

Note

If you carried out a test run, the system does not reload the data to the database. If the system cannot reload a financial position object back to the database, the system opens a new archive file. For this reason make sure you do not set the *Prohibit New Session During Reload* indicator.

Note

The system can reload archived runs only after you have carried out the write and deletion processes in production mode.

Procedure for deleting data

If you want to archive or delete data records separately, you can start the deletion program separately for each archive file.

Make sure that you select the *Test Mode* option in the test run variant, and the *Production Mode* option in the variant for the production run. You use the *Detail Log* option in your variants for the deletion run to define whether the system is to display the deleted financial position objects in a list.

1. Choose *Archive Selection* to select an archiving run.
2. Enter the deletion variant and choose *Execute*.

> **Note**
>
> If you select the test mode, the system does not remove the data from the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

7.12.1.5 Archiving of Aggregation Transaction

Definition

Archiving object `BA1_B7_AT` (*Bank Analyzer: Aggregation Transaction*) is used to archive, delete, and reload aggregation transaction data.

Use

The data that can be archived for aggregation transactions is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Tables

The system uses archiving object `BA1_B7_AT` to archive data from the following tables:

- `/BA1/B7_ATB_FIXF`
- `/BA1/B7_ATB_VARF`
- `/BA1/B7_ATTOTALS`

For information about how to display table names, see .

Programs

The following programs are available for BA1_B7_AT:

Program	Function
/BA1/RB7_AT_AR_WRITE	Write
/BA1/RB7_AT_AR_DELETE	Reload
/BA1/RB7_AT_AR_RELOAD	Delete

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.12.1.6 Archiving of General Ledger Connector

Definition

Archiving object BA1_B8_GL (*Bank Analyzer: GL Connector*) is used to archive, delete, and reload GL connector data.

Use

The data that can be archived for the General Ledger Connector is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Tables

The system uses archiving object BA1_B8_GL to archive data from the following tables:

- /BA1/B8GLCDOCH
- /BA1/B8GLCITEM

For information about how to display table names, see .

Programs

The following programs are available for BA1_B8_GL:

Program	Function

/BA1/RB8_GL_AR_WRITE	Write
/BA1/RB8_GL_AR_DELETE	Reload
/BA1/RB8_GL_AR_RELOAD	Delete

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.12.1.7 Archiving and Destroying Financial Transaction Data

You can archive financial transactions using archiving object *BA1_F1_040* (**Bank Analyzer: Archiving of Financial Transactions**), and destroy them using SAP Information Lifecycle Management (ILM).

For this ILM-enabled archiving object, you can execute the following programs using Archive Administration (transaction SARA):

- /BA1/F1_ARCHIVE_040_WRITE to write the archive files
- /BA1/F1_ARCHIVE_040_READ to read the archive files
- /BA1/F1_ARCHIVE_040_DELETE to delete data from the operational system
- /BA1/F1_ARCHIVE_040_RELOAD to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction SARA), enter the archiving object, and choose Database Tables.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.
   The *Archive Administration: Initial Screen* appears.
2. Select the object name *BA1_F1_040* (**Bank Analyzer: Archiving of Financial Transactions**)
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* field to select the financial transactions to be archived.
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.
    The system displays a list of financial transactions. From this list, you can select the financial transactions that you want to archive.
    For more information, see Creating Archive Files.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

7.12.1.8 Archiving and Destroying Financial Instrument Data

You can archive financial instruments using archiving object *BA1_F1_041* (**Bank Analyzer: Archiving of Financial Instruments**), and destroy them using SAP Information Lifecycle Management (ILM).

For this ILM-enabled archiving object, you can execute the following programs using Archive Administration (transaction `SARA`):

- `/BA1/F1_ARCHIVE_041_WRITE` to write the archive files
- `/BA1/F1_ARCHIVE_041_READ` to read the archive files
- `/BA1/F1_ARCHIVE_041_DELETE` to delete data from the operational system
- `/BA1/F1_ARCHIVE_041_RELOAD` to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction `SARA`), enter the archiving object, and choose Database Tables.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.
   The *Archive Administration: Initial Screen* appears.
2. Select the object name *BA1_F1_041* (**Bank Analyzer: Archiving of Financial Instruments**)
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* field to select the financial instruments to be archived.
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.
    The system displays a list of financial instruments. From this list, you can select the financial instruments that you want to archive.
    For more information, see Creating Archive Files.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

7.12.1.9 Archiving and Destroying Settlement Account Data

You can archive settlement accounts using archiving object *BA1_F1_042* (**Bank Analyzer: Archiving of Settlement Accounts**), and destroy them using SAP Information Lifecycle Management (ILM).

For this ILM-enabled archiving object, you can execute the following programs using Archive Administration (transaction `SARA`):

- `/BA1/F1_ARCHIVE_042_WRITE` to write the archive files
- `/BA1/F1_ARCHIVE_042_READ` to read the archive files
- `/BA1/F1_ARCHIVE_042_DELETE` to delete data from the operational system
- `/BA1/F1_ARCHIVE_042_RELOAD` to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction `SARA`), enter the archiving object, and choose Database Tables.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.
   The *Archive Administration: Initial Screen* appears.
2. Select the object name *BA1_F1_042* (Bank Analyzer: Archiving of Settlement Accounts)
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* field to select the settlement accounts to be archived.
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.
    The system displays a list of settlement accounts. From this list, you can select the settlement accounts that you want to archive.
    For more information, see Creating Archive Files.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

7.12.1.10 Archiving and Destroying Securities Position Account Data

You can archive securities position accounts using archiving object *BA1_F1_043* (**Bank Analyzer: Archiving of Securities Position Account**) and destroy them using SAP Information Lifecycle Management (ILM).

For this ILM-enabled archiving object, you can execute the following programs using Archive Administration (transaction `SARA`):

- `/BA1/F1_ARCHIVE_043_WRITE` to write the archive files
- `/BA1/F1_ARCHIVE_043_READ` to read the archive files
- `/BA1/F1_ARCHIVE_043_DELETE` to delete data from the operational system
- `/BA1/F1_ARCHIVE_043_RELOAD` to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction `SARA`), enter the archiving object, and choose Database Tables.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.
   The *Archive Administration: Initial Screen* appears.
2. Select the object name *BA1_F1_043* (Bank Analyzer: Archiving of Securities Position Account)
3. To generate an archive file, choose *Write*. The *Achive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* field to select the securities position account to be archived.
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.
    The system displays a list of securities position accounts. From this list, you can select the securities position accounts that you want to archive.
    For more information, see Creating Archive Files.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

7.12.1.11 Archiving and Destroying G/L Account Data

You can archive G/L accounts using archiving object *BA1_F1_044* (**Bank Analyzer: Archiving of G/L Accounts**), and destroy them using SAP Information Lifecycle Management (ILM).

For this ILM-enabled archiving object, you can execute the following programs using Archive Administration (transaction `SARA`):

- `/BA1/F1_ARCHIVE_044_WRITE` to write the archive files
- `/BA1/F1_ARCHIVE_044_READ` to read the archive files
- `/BA1/F1_ARCHIVE_044_DELETE` to delete data from the operational system
- `/BA1/F1_ARCHIVE_044_RELOAD` to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction `SARA`), enter the archiving object, and choose Database Tables.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.
   The *Archive Administration: Initial Screen* appears.
2. Select the object name *BA1_F1_044* (Bank Analyzer: Archiving of G/L Accounts)
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* field to select the G/L accounts to be archived.
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.
    The system displays a list of G/L accounts. From this list, you can select the G/L accounts that you want to archive.
    For more information, see Creating Archive Files.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

7.12.1.12 Archiving and Destroying Customer Position Account Data

You can archive customer position account data using archiving object *BA1_F1_045* (**Bank Analyzer: Archiving Customer Position Acct.**) and destroy it using SAP Information Lifecycle Management (ILM).

For this ILM-enabled archiving object, you can execute the following programs using Archive Administration (transaction `SARA`):

- `/BA1/F1_ARCHIVE_045_WRITE` to write the archive files
- `/BA1/F1_ARCHIVE_045_READ` to read the archive files
- `/BA1/F1_ARCHIVE_045_DELETE` to delete data from the operational system
- `/BA1/F1_ARCHIVE_045_RELOAD` to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction `SARA`), enter the archiving object, and choose *Database Tables*.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*. The *Archive Administration: Initial Screen* appears.
2. Select the object name *BA1_F1_045* (**Bank Analyzer: Archiving Customer Position Acct.**).
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* field to select the position master data to be archived.
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

7.12.1.13 Archiving and Destroying Country Data

You can archive country data using archiving object *BA1_F1_046* (**Bank Analyzer: Archiving of Country**) and destroy it using SAP Information Lifecycle Management (ILM).

For this ILM-enabled archiving object, you can execute the following programs using Archive Administration (transaction SARA):

- /BA1/F1_ARCHIVE_046_WRITE to write the archive files
- /BA1/F1_ARCHIVE_046_READ to read the archive files
- /BA1/F1_ARCHIVE_046_DELETE to delete data from the operational system
- /BA1/F1_ARCHIVE_046_RELOAD to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction SARA), enter the archiving object, and choose Database Tables.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.
   The *Archive Administration: Initial Screen* appears.
2. Select the object name *BA1_F1_046* (Bank Analyzer: Archiving of Country)
3. To generate an archive file, choose *Write*. The *Achive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* field to select the country data to be archived.
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.
    The system displays a list of countries. From this list, you can select the countries that you want to archive. For more information, see Creating Archive Files.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

7.12.1.14 Archiving and Destroying Physical Asset Data

You can archive physical assets with the archiving object *BA1_F1_048* (**FDB PO: Physical Asset**) and destroy them in SAP Information Lifecycle Management (ILM).

For this ILM-enabled archiving object, you can execute the following programs using Archive Administration (transaction SARA):

- /BA1/F1_ARCHIVE_048_WRITE to write the archive files
- /BA1/F1_ARCHIVE_048_READ to read the archive files

- `/BA1/F1_ARCHIVE_048_DELETE` to delete data from the operational system
- `/BA1/F1_ARCHIVE_048_RELOAD` to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction `SARA`), enter the archiving object, and choose Database Tables.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.
   The *Archive Administration: Initial Screen* appears.
2. Choose the object name *BA1_F1_048* (**FDB PO: Physical Asset**)
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* field to select the physical assets to be archived.
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.
    The system displays a list with physical assets. In this list, you can select individual physical assets that you want to archive.
    For more information, see Creating Archive Files.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

7.12.1.15 Archiving and Destroying Position Master Data

You can archive position master data using archiving object *BA1_F1_049* (**Bank Analyzer: Archiving of Position Master Data**) destroy it using SAP Information Lifecycle Management (ILM).

For this ILM-enabled archiving object, you can execute the following programs using Archive Administration (transaction `SARA`):

- `/BA1/F1_ARCHIVE_049_WRITE` to write the archive files
- `/BA1/F1_ARCHIVE_049_READ` to read the archive files
- `/BA1/F1_ARCHIVE_049_DELETE` to delete data from the operational system
- `/BA1/F1_ARCHIVE_049_RELOAD` to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction `SARA`), enter the archiving object, and choose Database Tables.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*. The *Archive Administration: Initial Screen* appears.
2. Select the object name *BA1_F1_049* (**Bank Analyzer: Archiving of Position Master Data**).
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* field to select the position master data to be archived.
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

7.12.1.16 Archiving and Destroying Position Data

Use

You can archive position data using archiving object `BA1_F2_030` *Bank Analyzer: Position* and destroy it as part of SAP Information Lifecycle Management (ILM).

You can use archiving object `BA1_F2_030` to archive classically stored positions as well as positions that are persisted in flat tables. In Customizing for Bank Analyzer under *Financial Services* > *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Flow Data* > *Positions* > *Edit Positions*, you use the *Positions in Flat Table* checkbox to indicate whether these are classically stored positions or positions in flat tables.

For this ILM-enabled archiving object, you can execute the following programs using Archive Administration (transaction SARA):

- `/BA1/RF2_ARCHIVE_030_WRITE` to write the archive files
- `/BA1/RF2_ARCHIVE_030_READ` to read the archive files
- `/BA1/RF2_ARCHIVE_030_DELETE` to delete data from the operational system
- `/BA1/RF2_ARCHIVE_030_RELOAD` to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction `SARA`), enter the archiving object, and choose *Database Tables*.

The following tables are used for classically stored positions:

- `/BA1/F2_TAB_POSH`
- `/BA1/F2_TAB_POSI`
- `/BA1/F2_TAB_POSV`
- `/BA1/F2_POS_SKEY`
- `/BA1/F2_POSI_S`

Table `/BA1/F2_POS_FLAT` is used for business transactions that are stored in flat tables.

The following archive information structures and corresponding field catalogs are available in the standard system:

- Archive information structure <sap-technical-name>SAP_BA1_F2_035</sap-technical-name> and field catalog `SAP_BA1_F2_030A` are used for classically stored business transactions
- Archive information structure <sap-technical-name>SAP_BA1_F2_035F</sap-technical-name> and field catalog `SAP_BA1_F2_030F` are used for business transactions stored in flat tables

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.
   The *Archive Administration: Initial Screen* appears.
2. Select the object name `BA1_F2_030` *Bank Analyzer: Position*.
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a *Variant*.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use InfoSets and the *Archiving Key Date* field to select the position data to be archived.
7. Whichever tab page you use, activate the appropriate dependency check radio button, and test/production mode radio button.
8. Save your entries and choose *Back*.
9. Specify the *Start Date* and the *Spool Parameters*.
10. Choose *Execute*.
    The system displays a list of positions from which you can select the positions you wish to archive.
    For more information, see .

More Information

For more information, see ILM Objects for Bank Analyzer [page 1754].

7.12.1.17 Archiving and Destroying Business Transaction Data

Context

You can archive data for business transactions using archiving object `BA1_F2_035` *Bank Analyzer: Business Transaction* and destroy it as part of SAP Information Lifecycle Management (ILM).

You can use archiving object `BA1_F2_035` to archive classically stored business transactions as well as business transactions that are persisted in flat tables. In Customizing for Bank Analyzer under *Financial Services* > *Bank Analyzer* > *Source Data Layer* > *Primary Objects* > *Flow Data* > *Business Transactions* > *Edit Business Transaction Classes*, you use the *Business Transactions in Flat Table* checkbox to indicate whether these are classically stored business transactions or business transactions in flat tables.

For this ILM-enabled archiving object, you can execute the following programs using Archive Administration (transaction SARA):

- `/BA1/RF2_ARCHIVE_035_WRITE` to write the archive files
- `/BA1/RF2_ARCHIVE_035_READ` to read the archive files
- `/BA1/RF2_ARCHIVE_035_DELETE` to delete data from the operational system
- `/BA1/RF2_ARCHIVE_035_RELOAD` to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction `SARA`), enter the archiving object, and choose *Database Tables*.

The following tables are used for classically stored business transactions:

- `/BA1/F2_BT_H`
- `/BA1/F2_BT_P`
- `/BA1/F2_BT_KF`
- `/BA1/FG_ABT_ASG`

Table `/BA1/F2_BT_FLAT` is used for business transactions that are stored in flat tables.

The following archive information structures and corresponding field catalogs are available in the standard system:

- Archive information structure `SAP_BA1_F2_035` and field catalog `SAP_BA1_F2_035` are used for classically stored business transactions
- Archive information structure `SAP_BA1_F2_035F` and field catalog `SAP_BA1_F2_035F` are used for business transactions stored in flat tables

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.

   The *Archive Administration: Initial Screen* appears.
2. Select the object name `BA1_F2_035` *Bank Analyzer: Business Transaction*.

3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a *Variant*.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can select the business transactions to be archived on the basis of a *key*, *header characteristics*, or *item characteristics*.
    - If you use the *Selection Using Key* tab page, specify the *business transaction ID* and the *source system* as selection criteria.
    - If you use the *Selection Using Header Characteristics* tab page, specify at least the *business transaction class* as a selection criterion. You can enter more selection criteria if you wish, and in this way limit even further the amount of business transaction data selected.
    - If you use the *Selection Using Item Characteristics* tab page, specify at least the *business transaction class* and the *item class* as selection criteria. You can enter more selection criteria if you wish, and in this way limit even further the amount of business transaction data selected.
7. Whichever tab page you use, activate the appropriate dependency check radio button, and test/production mode radio button.
8. Save your entries and choose *Back*.
9. Specify the *Start Date* and the *Spool Parameters*.
10. Choose *Execute*.

    The system displays a list of business transactions. from which you can select the business transactions you wish to archive.

    For more information, see .

Next Steps

For more information, see ILM Objects for Bank Analyzer [page 1754].

7.12.1.18 Archiving and Destroying Generic Market Data

You can archive generic market data using archiving object *BA1_F5_016* (**Bank Analyzer: Archiving of Generated Market Data**) and destroy it using SAP Information Lifecycle Management (ILM).

For this ILM-enabled archiving object, you can execute the following programs using Archive Administration (transaction `SARA`):

- `/BA1/F5_GMD_ARCH_WRITE` to write the archive files
- `/BA1/F5_GMD_ARCH_DELETE` to delete data from the operational system
- `/BA1/F5_GMD_ARCH_RELOAD` to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction `SARA`), enter the archiving object, and choose Database Tables.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.
   The *Archive Administration: Initial Screen* appears.
2. Select the object name *BA1_F5_016* (**Bank Analyzer: Archiving of Generated Market Data**).
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* field to select the generic market data to be archived.
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.
    The system displays a list of generic market data. From this list, you can select the generic market data that you want to archive.
    For more information, see Creating Archive Files.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

7.12.1.19 Archiving of Correction Entries

Definition

Archiving object `BA1_FK_COR` (*Bank Analyzer: Archiving of Correction Entries*) is used to archive, delete, and reload correction entries.

Use

The data that can be archived for correction entries is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Tables

The system uses archiving object BA1_FK_COR to archive data from the following tables:

- /BA1/FK_COR_ENT
- /BA1/FK_FLD_CHG
- /BA1/FK_INC_ENT
- STXB
- STXH
- STXL

For information about how to display table names, see .

Programs

The following programs are available for BA1_FK_COR:

Program	Function
/BA1/FK_ARCHIVE_COR_WRITE	Write
/BA1/FK_ARCHIVE_COR_DELETE	Reload
/BA1/FK_ARCHIVE_COR_RELOAD	Delete

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.12.1.20 Archiving of Data Flow Entries

Definition

Archiving object BA1_FK_DFW (*Bank Analyzer: Archiving of Data Flow Entries*) is used to archive, delete, and reload data flow entries.

Use

The data that can be archived for data flow entries is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Tables

The system uses archiving object BA1_FK_DFW to archive data from the following table:

- /BA1/FK_DATAFLOW

For information about how to display table names, see .

Programs

The following programs are available for BA1_FK_DFW:

Program	Function
/BA1/FK_ARCHIVE_DFW_WRITE	Write
/BA1/FK_ARCHIVE_DFW_DELETE	Reload
/BA1/FK_ARCHIVE_DFW_RELOAD	Delete

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.12.1.21 Archiving of Processing Framework Runs (FDB1)

Definition

Archiving object BA1_PFW_F1 is used to archive, delete, and reload processing framework runs.

Use

The data that can be archived for processing framework runs are written to one or multiple archive files. Before you can archive runs in Profit Engine, the runs must exist in the status management for the processing framework.

Structure

Tables

The system uses the archiving object to archive data from the following table:

- /BA1/P1_TA_BLADM
- /BA1/P1_TA_CBLAD
- /BA1/P1_TA_CPNRE
- /BA1/P1_TA_CTRAN
- /BA1/P1_TA_CWLAD
- /BA1/P1_TA_CWLCP

- /BA1/P1_TA_LOGS
- /BA1/P1_TA_PNREF
- /BA1/P1_TA_RPADM
- /BA1/P1_TA_RPAPP
- /BA1/P1_TA_RPTAC
- /BA1/P1_TA_TRANC
- /BA1/P1_TA_WLADM
- /BA1/P1_TA_WLCPL

For more information about how to display table names, see .

Programs

The following programs are available for BA1_PFW_F1:

Program	Function
/BA1/FP_R_PFW_FDB1_ARC_RUN_WRI	Write
/BA1/FP_R_PFW_FDB1_ARC_RUN_DEL	Delete
/BA1/FP_R_PFW_FDB1_ARC_RUN_REL	Reload

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

Procedure for writing data to the archive

1. Create a variant and choose *Maintain*. The *Variant Maintenance* menu appears.
2. Enter data for the following fields:
   Required fields:
   - *Application*
   - *Area*
   - *Date* For application (PA), enter the date in the format YYYY0MM in the *Date for Profit Analyzer* field. Or *Date for FDB*; use the input help.

   Optional fields:
   - Level(s)
   - Package(s)
   - Status
   - *Processing Options* (test or production mode)
   - Archiving session note so you can identify the run at a later point in time.
3. Save your variant, and schedule the job.

> **Note**
>
> If you did not enter any data in the fields for the levels, the system uses all levels and packages that belong to the area in the archiving process.
>
> If you do not enter data in the fields for the level, you cannot enter data for the fields for packages.
>
> Only choose multiple packages if you have selected at least one level.

Procedure for reloading data

You can reload complete archiving runs only. If an error occurs when you are reloading archived data, the system rejects the entire run.

You have to have finished reloading data before you can read archived runs.

> **Recommendation**
>
> We recommend that you reload archived data only in exceptional circumstances.

1. Create a variant for the test or productive run, and save the variant.
2. Choose *Archive Selection* to select an archiving run.
3. Enter the reload variant and choose *Execute*.

> **Note**
>
> If you carried out a test run, the system does not reload the data to the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

Procedure for deleting data

1. Choose *Archive Selection* to select an archiving run.
2. Enter the deletion variant and choose *Execute*.

> **Note**
>
> If you select the test mode, the system does not remove the data from the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

7.12.1.22 Archiving of Processing Framework Runs (FDB2)

Definition

Archiving object `BA1_PFW_F2` is used to archive, delete, and reload processing framework runs.

Use

The data that can be archived for processing framework runs are written to one or multiple archive files. Before you can archive runs in Profit Engine, the runs must exist in the status management for the processing framework.

Structure

Tables

The system uses the archiving object to archive data from the following table:

- /BA1/P1_TA_BLADM
- /BA1/P1_TA_CBLAD
- /BA1/P1_TA_CPNRE
- /BA1/P1_TA_CTRAN
- /BA1/P1_TA_CWLAD
- /BA1/P1_TA_CWLCP
- /BA1/P1_TA_LOGS
- /BA1/P1_TA_PNREF
- /BA1/P1_TA_RPADM
- /BA1/P1_TA_RPAPP
- /BA1/P1_TA_RPTAC
- /BA1/P1_TA_TRANC
- /BA1/P1_TA_WLADM
- /BA1/P1_TA_WLCPL

For more information about how to display table names, see .

Programs

The following programs are available for BA1_PFW_F2:

Program	Function
/BA1/FP_R_PFW_FDB2_ARC_RUN_WRI	Write
/BA1/FP_R_PFW_FDB2_ARC_RUN_DEL	Delete
/BA1/FP_R_PFW_FDB2_ARC_RUN_REL	Reload

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

Procedure for writing data to the archive

1. Create a variant and choose *Maintain*. The *Variant Maintenance* menu appears.

2. Enter data for the following fields:
   Required fields:
   - *Application*
   - *Area*
   - *Date* For application (PA), enter the date in the format YYYY0MM in the *Date for Profit Analyzer* field. Or *Date for FDB*; use the input help.

   Optional fields:
   - Level(s)
   - Package(s)
   - Status
   - *Processing Options* (test or production mode)
   - Archiving session note so you can identify the run at a later point in time.
3. Save your variant, and schedule the job.

> **Note**
>
> If you did not enter any data in the fields for the levels, the system uses all levels and packages that belong to the area in the archiving process.
>
> If you do not enter data in the fields for the level, you cannot enter data for the fields for packages.
>
> Only choose multiple packages if you have selected at least one level.

Procedure for reloading data

You can reload complete archiving runs only. If an error occurs when you are reloading archived data, the system rejects the entire run.

You have to have finished reloading data before you can read archived runs.

> **Recommendation**
>
> We recommend that you reload archived data only in exceptional circumstances.

1. Create a variant for the test or productive run, and save the variant.
2. Choose *Archive Selection* to select an archiving run.
3. Enter the reload variant and choose *Execute*.

> **Note**
>
> If you carried out a test run, the system does not reload the data to the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

Procedure for deleting data

1. Choose *Archive Selection* to select an archiving run.
2. Enter the deletion variant and choose *Execute*.

> **Note**
>
> If you select the test mode, the system does not remove the data from the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

7.12.1.23 Archiving of Processing Framework Runs (FDB3)

Definition

Archiving object `BA1_PFW_F3` is used to archive, delete, and reload processing framework runs.

Use

The data that can be archived for processing framework runs are written to one or multiple archive files. Before you can archive runs in Profit Engine, the runs must exist in the status management for the processing framework.

Structure

Tables

The system uses the archiving object to archive data from the following table:

- `/BA1/P1_TA_BLADM`
- `/BA1/P1_TA_CBLAD`
- `/BA1/P1_TA_CPNRE`
- `/BA1/P1_TA_CTRAN`
- `/BA1/P1_TA_CWLAD`
- `/BA1/P1_TA_CWLCP`
- `/BA1/P1_TA_LOGS`
- `/BA1/P1_TA_PNREF`
- `/BA1/P1_TA_RPADM`
- `/BA1/P1_TA_RPAPP`
- `/BA1/P1_TA_RPTAC`
- `/BA1/P1_TA_TRANC`
- `/BA1/P1_TA_WLADM`

- `/BA1/P1_TA_WLCPL`

For more information about how to display table names, see .

Programs

The following programs are available for BA1_PFW_F3:

Program	Function
`/BA1/FP_R_PFW_FDB3_ARC_RUN_WRI`	Write
`/BA1/FP_R_PFW_FDB3_ARC_RUN_DEL`	Delete
`/BA1/FP_R_PFW_FDB3_ARC_RUN_REL`	Reload

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

Procedure for writing data to the archive

1. Create a variant and choose *Maintain*. The *Variant Maintenance* menu appears.
2. Enter data for the following fields:
   Required fields:
   - *Application*
   - *Area*
   - *Date* For application (PA), enter the date in the format YYYY0MM in the *Date for Profit Analyzer* field. Or *Date for FDB*; use the input help.

   Optional fields:
   - Level(s)
   - Package(s)
   - Status
   - *Processing Options* (test or production mode)
   - Archiving session note so you can identify the run at a later point in time.
3. Save your variant, and schedule the job.

> **Note**
>
> If you did not enter any data in the fields for the levels, the system uses all levels and packages that belong to the area in the archiving process.
>
> If you do not enter data in the fields for the level, you cannot enter data for the fields for packages.
>
> Only choose multiple packages if you have selected at least one level.

Procedure for reloading data

You can reload complete archiving runs only. If an error occurs when you are reloading archived data, the system rejects the entire run.

You have to have finished reloading data before you can read archived runs.

> **Recommendation**
>
> We recommend that you reload archived data only in exceptional circumstances.

1. Create a variant for the test or productive run, and save the variant.
2. Choose *Archive Selection* to select an archiving run.
3. Enter the reload variant and choose *Execute*.

> **Note**
>
> If you carried out a test run, the system does not reload the data to the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

Procedure for deleting data

1. Choose *Archive Selection* to select an archiving run.
2. Enter the deletion variant and choose *Execute*.

> **Note**
>
> If you select the test mode, the system does not remove the data from the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

7.12.1.24 Archiving of Processing Framework Runs (S1RR)

Definition

Archiving object `BA1_PFW_S1` is used to archive, delete, and reload processing framework runs.

Use

The data that can be archived for processing framework runs are written to one or multiple archive files. Before you can archive runs in Profit Engine, the runs must exist in the status management for the processing framework.

Structure

Tables

The system uses the archiving object to archive data from the following tables:

- /BA1/P1_TA_BLADM
- /BA1/P1_TA_CBLAD
- /BA1/P1_TA_CPNRE
- /BA1/P1_TA_CTRAN
- /BA1/P1_TA_CWLAD
- /BA1/P1_TA_CWLCP
- /BA1/P1_TA_LOGS
- /BA1/P1_TA_PNREF
- /BA1/P1_TA_RPADM
- /BA1/P1_TA_RPAPP
- /BA1/P1_TA_RPTAC
- /BA1/P1_TA_TRANC
- /BA1/P1_TA_WLADM
- /BA1/P1_TA_WLCPL

For more information about how to display table names, see .

Programs

The following programs are available for BA1_PFW_S1:

Program	Function
/BA1/RS1_PFW_ARC_RUN_WRI	Write
/BA1/RS1_PFW_ARC_RUN_DEL	Delete
/BA1/RS1_PFW_ARC_RUN_REL	Reload

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

Procedure for writing data to the archive

1. Create a variant and choose *Maintain*. The *Variant Maintenance* menu appears.
2. Enter data for the following fields:
   Required fields:
   - *Application*
   - *Area*
   - *Date* For application (PA), enter the date in the format YYYY0MM in the *Date for Profit Analyzer* field. Or *Date for FDB*; use the input help.

   Optional fields:
   - Level(s)
   - Package(s)

- Status
- *Processing Options* (test or production mode)
- Archiving session note so you can identify the run at a later point in time.

3. Save your variant, and schedule the job.

> **Note**
>
> If you did not enter any data in the fields for the levels, the system uses all levels and packages that belong to the area in the archiving process.
>
> If you do not enter data in the fields for the level, you cannot enter data for the fields for packages.
>
> Only choose multiple packages if you have selected at least one level.

Procedure for reloading data

You can reload complete archiving runs only. If an error occurs when you are reloading archived data, the system rejects the entire run.

You have to have finished reloading data before you can read archived runs.

> **Recommendation**
>
> We recommend that you reload archived data only in exceptional circumstances.

1. Create a variant for the test or productive run, and save the variant.
2. Choose *Archive Selection* to select an archiving run.
3. Enter the reload variant and choose *Execute*.

> **Note**
>
> If you carried out a test run, the system does not reload the data to the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

Procedure for deleting data

1. Choose *Archive Selection* to select an archiving run.
2. Enter the deletion variant and choose *Execute*.

> **Note**
>
> If you select the test mode, the system does not remove the data from the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

7.12.1.25 Archiving of Detail Logs for Fair Value Server

Definition

Archiving object `BA1_R0_FVS` (*Bank Analyzer: Detail Log for Fair Value Server*) is used to archive, delete, and reload data for the detail logs for the fair value server.

Use

The detail log data for the fair value server that can be archived is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Tables

The system uses archiving object `BA1_R0_FVS` to archive data from various tables.

For information about how to display table names, see .

Programs

The following programs are available for BA1_R0_FVS:

Program	Function
`/BA1/R0_LOG_ARCHIVE_WRITE`	Write
`/BA1/R0_LOG_ARCHIVE_RELOAD`	Reload
`/BA1/R0_LOG_ARCHIVE_DELETE`	Delete

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.12.1.26 Archiving of Accounting Balances, Balance Objects, and Financial Statement Items

Definition

Archiving object `BA1_R4_055` (*RDB: Archiving for Balance Analyzer*) is used to archive, delete, and reload accounting balances, balance objects, and financial statement items (runs and packages in the Result Database).

Use

There are archive writing and deletion programs for this object. Archived data can also be read. The data that can be archived for financial position objects is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Criteria for archiving accounting balances

- The accounting documents on which the balances are based have already been archived.
- All the periods in question are closed.

Criteria for archiving balance objects and financial statement items

You can archive balance objects and financial statement items only if the relevant periods have been write-protected.

Structure

Tables

The system uses archiving object `BA1_R4_055` to archive data from the following tables:

- `/BA1/R4_KFSTRC`
- `/BA1/R4_PACK`
- `/BA1/R4_REC_BP`
- `/BA1/R4_RUN`

For information about how to display table names, see .

Programs

The following programs are available for BA1_R4_055:

Program	Function
`/BA1/RB2_RESDAT_AR_WRITE`	Write
`/BA1/R4_ARCHIVE_DELETE`	Delete
`/BA1/R4_ARCHIVE_RELOAD`	Reload

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

Procedure for writing data to the archive

Make sure that the options such as the size of the archive file, variants for the test run and productive run for the deletion program are correct (see Environment -> Customizing).

➡ Recommendation

We recommend the following settings:

- Maximum MB: 50-100
- Maximum number of data objects: 0

Hence only the file size is important.

Make sure that you select the *Test Mode* option in the test run variant, and the *Production Mode* option in the variant for the production run. You use the *Detail Log* option in your variants for the deletion run to define whether the system is to display the deleted accounting objects in a list.

You start archiving runs for one legal entity and one accounting system at a time. The system archives all the data that relates to periods that became inactive when the residence time specified in Customizing was exceeded.

You write data to the archive as follows:

1. Create a variant and choose *Maintain*. The system displays the *Variant Maintenance* menu.
2. Enter data for the following fields:
   Required fields:
   - *Legal Entity*
   - *Accounting System*

   Optional fields:
   - Detail Log
   - *Processing Options* (test or production mode)
   - Archiving session note to enable you to identify the archiving run at a later point in time. The system adds the following information:
     - `BD` (balance objects)
     - `FSI` (financial statement items)
     - `ACCT` (accounting balances)
3. Save your variant, and schedule the job.

Note

For technical reasons, an archiving run is created for each RDB run and RDB packages. This means that there may be multiple archiving runs each time you start the job. This is why the system displays the same archiving session note for multiple archiving runs after you have written data to the archive.

Procedure for reloading data

Recommendation

We recommend that you reload archived data only in exceptional circumstances.

1. Create a variant for the test or productive run, and save the variant.
2. Choose *Archive Selection* to select an archiving run.
3. Enter the reload variant and choose *Execute*.

Note

The system can reload archived runs only after you have carried out the write and deletion processes in production mode.

Procedure for deleting data

If you want to archive or delete data records separately, you can start the deletion program separately for each archive file.

Make sure that you select the *Test Mode* option in the test run variant, and the *Production Mode* option in the variant for the production run.

1. Choose *Archive Selection* to select an archiving run.
2. Enter the deletion variant and choose *Execute*.

> **Note**
>
> If you select the test mode, the system does not remove the data from the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

7.12.1.27 Archiving of Effectiveness Tests

Definition

Archiving object `BA1_R4_056` (*RDB: Archiving of Effectiveness Test*) is used to archive, delete, and reload data relating to the effectiveness test that is stored in the Result Database (RDB).

Use

The data that can be archived for the effectiveness test is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Tables

The system uses archiving object `BA1_R4_056` to archive data from various tables.

- `/BA1/R4_FVE_AT`
- `/BA1/R4_FVE_EFFE`
- `/BA1/R4_FVE_EXTR`
- `/BA1/R4_FVE_FAT1`
- `/BA1/R4_FVE_MARK`
- `/BA1/R4_FVE_OFFS`

- /BA1/R4_FVE_REGR
- /BA1/R4_FVE_TVAL
- /BA1/R4_FVE_XYAT
- /BA1/R4_FVE_XYPR
- /BA1/R4_FVE_XYRR
- /BA1/R4_KFSTRC
- /BA1/R4_PACK
- /BA1/R4_REC_FVE
- /BA1/R4_RUN

For information about how to display table names, see .

Programs

The following programs are available for BA1_R4_056:

Program	Function
/BA1/R4_ARCHIVE_WRITE	Write
/BA1/R4_ARCHIVE_DELETE	Reload
/BA1/R4_ARCHIVE_RELOAD	Delete

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.12.1.28 Archiving of Portfolio Items

Definition

Archiving object BA1_R4_057 (*RDB: Archiving of Portfolio Items*) is used to archive, delete, and reload portfolio items that are stored in the Result Database (RDB).

Use

The data that can be archived for portfolio items is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Tables

The system uses archiving object BA1_R4_057 to archive data from the following tables:

- /BA1/R4_KFSTRC
- /BA1/R4_PACK
- /BA1/R4_PFVH_VP
- /BA1/R4_REC_PFVH
- /BA1/R4_RUN

For information about how to display table names, see .

Programs

The following programs are available for BA1_R4_057:

Program	Function
/BA1/R4_ARCHIVE_WRITE	Write
/BA1/R4_ARCHIVE_DELETE	Reload
/BA1/R4_ARCHIVE_RELOAD	Delete

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.12.1.29 Archiving and Destroying Credit Exposure Data

You can archive credit exposure data (results from the calculations for Basel II) using the archiving object BA1_R4_063 *(RDB: Archiving for Credit Exposure 2)* and destroy it as part of SAP Information Lifecycle Management (ILM).

For this ILM-enabled archiving object, you can execute the following programs using Archive Administration (transaction SARA):

- /BA1/R2_ARCHIVE_WRITE to write the archive files
- /BA1/R4_ARCHIVE_DELETE to delete data from the operational system
- /BA1/R4_ARCHIVE_RELOAD to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction SARA), enter the archiving object, and choose Database Tables.

Procedure

On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.

1. The *Archive Administration: Initial Screen* appears.
2. Choose the object BA1_R4_063 *(RDB: Archiving for Credit Exposure 2)*.

3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* field to select the credit exposure results to be archived.
   .
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.
    The system displays a list of credit exposure results. From this list, you can select the credit exposure results that you want to archive.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

7.12.1.30 Archiving and Destroying Historical Database Data

You can archive Historical Database (HDB) data using the following archiving objects, and destroy it as part of SAP Information Lifecycle Management (ILM):

- `BA1_R4_064` *(RDB: Archiving of HDB - Scenario Data Storage)*
- `BA1_R4_065` *(RDB: Archiving HDB - Review Data Storage)*
- `BA1_R4_066` *(RDB: Archiving HDB - Historical Data Storage)*
- `BA1_R4_067` *(RDB: Archiving HDB - Data Processing Storage)*

> **Note**
>
> Note that you cannot archive individual HDB runs. The archiving function always archives complete RDB runs, each of which usually consists of multiple HDB runs.

You can execute the following write, delete, and reload programs for these ILM-enabled archiving objects in archiving administration (transaction SARA):

Archiving Object	Programs

`BA1_R4_064`	`/BA1/R6_ARCHIVING_WRITE_SCEN`
	`/BA1/R6_ARCHIVING_DELETE_SCEN`
	`/BA1/R6_ARCHIVING_RELOAD_SCEN`

Archiving Object	Programs
BA1_R4_065	/BA1/R6_ARCHIVING_WRITE_REV /BA1/R6_ARCHIVING_RELOAD_REV /BA1/R6_ARCHIVING_DELETE_REV
BA1_R4_066	/BA1/R6_ARCHIVING_WRITE_HIST /BA1/R6_ARCHIVING_DELETE_HIST /BA1/R6_ARCHIVING_RELOAD_HIST
BA1_R4_067	/BA1/R6_ARCHIVING_WRITE_COL /BA1/R6_ARCHIVING_DELETE_COL /BA1/R6_ARCHIVING_RELOAD_COL

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction SARA), enter the name of the archiving object, and choose *Database Tables*.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.
   The *Archive Administration: Initial Screen* appears.
2. Choose the required object name. For example, BA1_R4_064 *(RDB: Archiving of HDB - Scenario Data Storage)*.
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* fields to select the data to be archived (for example, HDB scenario data storage).
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.
    The system displays a list of data (for example, HDB scenario data storage). In this list, you can select individual items (for example, HDB scenario data storage items) that you want to archive.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

7.12.1.31 Archiving and Destroying Default Data

You can archive default data using the following archiving objects, and destroy it as part of SAP Information Lifecycle Management (ILM):

- `BA1_R4_068` *(RDB: Archiving of GM Data - Default Data)*
- `BA1_R4_069` *(RDB: Archiving of GM Data - Default Stress Data)*

> **Note**
>
> Note that you cannot archive individual default determination runs. The archiving function always archives complete RDB runs, each of which usually consists of multiple default determination runs or stress runs in Determination of Default.

You can execute the following write, delete, and reload programs for these ILM-enabled archiving objects in archiving administration (transaction `SARA`):

Archiving Object	Programs

`BA1_R4_068`	`/BA1/RA_ARCHIVING_WRITE_DEF`
	`/BA1/RA_ARCHIVING_DELETE_DEF`
	`/BA1/RA_ARCHIVING_RELOAD_DEF`

`BA1_R4_069`	`/BA1/RA_ARCHIVING_WRITE_SCEN`
	`/BA1/RA_ARCHIVING_DELETE_SCEN`
	`/BA1/RA_ARCHIVING_RELOAD_SCEN`

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction `SARA`), enter the name of the archiving object, and choose *Database Tables*.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*. The *Archive Administration: Initial Screen* appears.
2. Choose the required object name. For example, `BA1_R4_068` *(RDB: Archiving of GM Data - Default Data)*.
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* fields to select the data to be archived (for example, GM default data).
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.

11. Choose *Execute*.
    The system displays a list of data (for example, GM default data). In this list, you can select individual items (for example, GM default data) that you want to archive.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]

7.12.1.32 Archiving of Data From General Calculation and Valuation Methods

You can use the following archiving objects to archive data for general calculation and valuation methods and destroy it as part of SAP Information Lifecycle Management (ILM):

- `BA1_R4_071` *(RDB: Archiving for Pooling)*
- `BA1_R4_072` *(RDB: Archiving for Facility Distribution)*
- `BA1_R4_073` *(RDB: Archiving for Collateral Distribution)*
- `BA1_R4_074` *(RDB: Archiving for Distribution of Free Line)*

> **Note**
>
> Use archiving objects `BA1_R4_068`*(RDB: Archiving of GM Data - Default Data)* and`BA1_R4_069`*(RDB: Archiving of GM Data - Default Stress Data)* for data relating to the determination of default. For more information, see *Related Links*.

For these ILM-enabled archiving objects, you can execute the following programs using Archive Administration (transaction `SARA`):

- `/BA1/DE_ARCHIVE_WRITE` to write the archive files
- `/BA1/DE_ARCHIVE_DELETE` to delete data from the operational system
- `/BA1/R4_ARCHIVE_RELOAD` to reload data from the archive files into the operational system

To view the tables from which data for this archiving object is deleted, go to Archive Administration (transaction `SARA`), enter the archiving object, and choose Database Tables.

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.
   The *Archive Administration: Initial Screen* appears.
2. Choose the required object name. For example, `BA1_R4_071` *(RDB: Archiving for Pooling)*.
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a variant.

5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
6. You can use *InfoSets* and the *Archiving Key Date* fields to select the data to be archived (for example, pooling data).
7. In the *Selection Options* group box, choose *With Dependency Check*.
8. In the *Processing Options* group box, choose *Production Mode*.
9. Save your entries and choose *Back*.
10. Specify the start date and the spool parameters.
11. Choose *Execute*.
    The system displays a list of data (for example, pooling data). In this list, you can select individual items (for example, pooling data) that you want to archive.

Related Information

Archiving with ILM-Enabled Archiving Objects [page 1754]
Archiving and Destroying Default Data [page 422]

7.12.1.33 Archiving of Hedging Relationships in Hedge Management

Definition

You use archiving object `BA1HM_HR` (*Hedging Relationship in Hedge Management*) to archive and preprocess hedging relationships in Hedge Management.

Use

The hedge data in Hedge Management that can be archived is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Table

The system uses archiving object `BA1HM_HR` to archive data from various tables. For information about how to display table names, see .

Programs

The following programs are available for BA1HM_HR:

Program	Function
/BA1HM/RAH_M_HR_ARCH_WRITE	Write
/BA1HM/RAH_M_HR_ARCH_PREPROC	Preprocess

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.12.1.34 Archiving of Balance Object Manager Data

Definition

Archiving object BA1_R4_054 (*RDB: Archiving for Balance Object Manager*) is used to archive, delete, and reload Balance Object Manager data that is stored in the Result Database (RDB).

Use

The data that can be archived for the balance object manager is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Table

The system uses archiving object BA1_R4_054 to archive data from the following tables:

- /BA1/R4_KFSTRC
- /BA1/R4_PACK
- /BA1/R4_REC_BOM
- /BA1/R4_RUN

For information about how to display table names, see .

Programs

The following programs are available for BA1_R4_054:

Program	Function

/BA1/RB6_RESDAT_AR_WRITE	Write
/BA1/R4_ARCHIVE_DELETE	Delete
/BA1/R4_ARCHIVE_RELOAD	Reload

The standard variants SAP PROD (productive mode) and SAP TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.12.2 Archiving with Archiving Objects

7.12.2.1 Archiving of CVPM Process Monitor

To control the size of the dataset, you can archive data at run level using Archiving Engine.

If you archive data by mistake, you can use Archiving Engine to reload it.

7.12.2.2 Archiving of Basic Data for Customer Migration Analysis

Definition

Archiving object /BA1/P6_BD (*BA: Basic Data for Customer Migration Analysis*) is used to archive, delete, and reload basic data for customer migration analysis.

Use

The basic data that can be archived for customer migration analysis is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Tables

The system uses archiving object /BA1/P6_BD to archive data from various tables. For information about how to display table names, see .

Programs

The following programs are available for /BA1/P6_BD:

Program	Function
/BA1/P6_ARCHIVE_BD_WRITE	Write
/BA1/P6_ARCHIVE_BD_RELOAD	Reload
/BA1/P6_ARCHIVE_BD_DELETE	Delete

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

Procedure for writing data to the archive

1. Create a variant and choose *Maintain*. The system displays the *Variant Maintenance* menu.
2. Enter data for the following fields:
   Optional fields:*Processing Options* (test or productive mode) and a comment so you can identify the archiving run at a later point in time.
3. Save your variant, and schedule the job.
4. You can save parameters for printing. Even if you do not enter any spool parameters, you have to call the menu so that the system sets the traffic light to green.

Recommendation

We recommend that you do not select a fiscal year period for archiving that is equal to or older than the fiscal year period you entered in your variant. The system still needs this data to calculate results data.

Procedure for reloading data

You can reload complete archiving runs only. You cannot reload the data for individual fiscal year periods. If an error occurs when you are reloading archived data, the system rejects the entire run. The system can reject a run, if, for example, a dataset in the past was calculated more than once, and the value of a key has been duplicated.

To read archived runs, you must not have already reloaded the data.

Recommendation

We recommend that you reload archived data only in exceptional circumstances.

1. Create a variant for the test or productive run, and save the variant.
2. Choose *Archive Selection* to select an archiving run.
3. Enter the reload variant and choose *Execute*.

Note

If you carried out a test run, the system does not reload the data to the database.

Note

The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

Procedure for deleting data

1. Choose *Archive Selection* to select an archiving run.
2. Enter the deletion variant and choose *Execute*.

> **Note**
>
> If you select the test mode, the system does not remove the data from the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

7.12.2.3 Archiving of Result Data from Customer Migration Analysis

Definition

Archiving object `/BA1/P6_RD` (*Bank Analyzer: Result Data From Customer Migration Analysis*) is used to archive and delete result data from customer migration analysis.

Use

The result data that can be archived for customer migration analysis is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Tables

The system uses archiving object `/BA1/P6_RD` to archive data from various tables. For information about how to display table names, see .

Programs

The following programs are available for /BA1/P6_RD:

Program	Function
`/BA1/P6_ARCHIVE_RD_WRITE`	Write

/BA1/P6_ARCHIVE_RD_RELOAD	Reload
/BA1/P6_ARCHIVE_RD_DELETE	Delete

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

Procedure for writing data to the archive

1. Create a variant and choose *Maintain*. The system displays the *Variant Maintenance* menu.
2. Enter data for the following fields:
   Required fields:*Fiscal Year Period* The system archives all the fiscal year periods that are equal to or older than the fiscal year period you select.
   Optional fields:*Processing Options* (test or productive mode) and a comment so you can identify the archiving run at a later point in time.
3. Save your variant, and schedule the job.
4. You can save parameters for printing. Even if you do not enter any spool parameters, you have to call the menu so that the system sets the traffic light to green.

Procedure for reloading data

You can reload complete archiving runs only. You cannot reload the data for individual fiscal year periods. If an error occurs when you are reloading archived data, the system rejects the entire run. The system can reject a run, if, for example, a dataset in the past was calculated more than once, and the value of a key has been duplicated.

To read archived runs, you must not have already reloaded the data.

> **Recommendation**
>
> We recommend that you reload archived data only in exceptional circumstances.

1. Create a variant for the test or productive run, and save the variant.
2. Choose *Archive Selection* to select an archiving run.
3. Enter the reload variant and choose *Execute*.

> **Note**
>
> If you carried out a test run, the system does not reload the data to the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

Procedure for deleting data

1. Choose *Archive Selection* to select an archiving run.
2. Enter the deletion variant and choose *Execute*.

> **Note**
>
> If you select the test mode, the system does not remove the data from the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

7.12.2.4 Archiving of Accounting Objects

Definition

Archiving object `BA1_B0_AO3` (*Bank Analyzer: Accounting Object*) is used to archive, delete, and reload accounting objects.

Use

There are archive writing and deletion programs for this object. Archived data can also be read. The data that can be archived for accounting objects is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

- The corresponding financial position objects (object `BA1_B0_FPO`) have been archived.
- The corresponding balance objects and financial statement items (object `BA1_R4_055`) have been archived.

Structure

Tables

The archiving object archives data from the following accounting object tables:

- `/BA1/B0_AOH3`
- `/BA1/B0_AOT3`
- `/BA1/B0_AOS3`

For information about how to display table names, see .

Programs

The following programs are available for BA1_B0_AO3:

Program	Function

/BA1/RB0_AO_AR_WRITE3	Write
/BA1/RB0_AO_AR_DELETE3	Delete
/BA1/RB0_AO_AR_RELOAD3	Reload

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

Procedure for writing data to the archive

Make sure that the options such as the size of the archive file, variants for the test run and productive run for the deletion program are correct (see Environment -> Customizing).

Recommendation

We recommend the following settings:

- Maximum MB: 50-100
- Maximum number of data objects: 0

Hence only the file size is important.

Make sure that you select the *Test Mode* option in the test run variant, and the *Production Mode* option in the variant for the production run. You use the *Detail Log* option in your variants for the deletion run to define whether the system is to display the deleted accounting objects in a list.

You start archiving runs for one legal entity at a time. The system archives all the accounting objects that became inactive when the residence time specified in Customizing was exceeded.

You write data to the archive as follows:

1. Create a variant and choose *Maintain*. The system displays the *Variant Maintenance* menu.
2. Enter data for the following fields:
   Required fields:
   - *Legal entity*

   Optional fields:
   - Detail Log
   - *Processing Options* (test or production mode)
   - Archiving session note so you can identify the run at a later point in time.
3. Save your variant, and schedule the job.

Procedure for reloading data

Recommendation

We recommend that you reload archived data only in exceptional circumstances.

1. Create a variant for the test or productive run, and save the variant.
2. Choose *Archive Selection* to select an archiving run.
3. Enter the reload variant and choose *Execute*.

The system reloads the accounting object from the archiving run you selected. If the system cannot reload an accounting object back to the database, the system opens a new archive file. For this reason make sure you do not set the *Prohibit New Session During Reload* indicator.

> **Note**
>
> If you carried out a test run, the system does not reload the data to the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode. You can reload archived runs only after you have carried out the write and deletion process in production mode.

Procedure for deleting data

If you want to archive or delete data records separately, you can start the deletion program separately for each archive file.

Make sure that you select the *Test Mode* option in the test run variant, and the *Production Mode* option in the variant for the production run.

1. Choose *Archive Selection* to select an archiving run.
2. Enter the deletion variant and choose *Execute*.

> **Note**
>
> If you select the test mode, the system does not remove the data from the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

Procedure for displaying archived data

You can display archived accounting objects. You do this by selecting the relevant archive runs.

> **Note**
>
> The system searches for the archived accounting objects in sequence. However, this can impair system performance.

- Enter data for the following fields:
  Required fields:
  - *Legal Entity*
  - *Key Date* (set by default to today's date)
  - *Timepoint* (contains the current time)

  The system displays all the accounting objects from the archive files selected that meet your selection criteria.

7.12.2.5 Archiving of Financial Position Balances

Definition

Archiving object `BA1_B1_FPT` (*Bank Analyzer: Financial Position Balances*) is used to archive, delete, and reload financial position balances.

Use

There are archive writing and deletion programs for this object. Archived data can also be read. The data that can be archived for financial position balances is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

- The corresponding accounting documents have already been archived.
- The periods in question have been closed.

Structure

Tables

The archiving object archives data from the following financial position balance tables:

- `/BA1/B1_FPTOTALS`

For information about how to display table names, see .

Programs

The following programs are available for BA1_B1_FPT:

Program	Function
`/BA1/RB1_FPT_AR_WRITE`	Write
`/BA1/RB1_FPT_AR_DELETE`	Delete
`/BA1/RB1_FPT_AR_RELOAD`	Reload

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

Procedure for writing data to the archive

Make sure that the options such as the size of the archive file, variants for the test run and productive run for the deletion program are correct (see Environment -> Customizing).

Recommendation

We recommend the following settings:

- Maximum MB: 50-100
- Maximum number of data objects: 0

Hence only the file size is important.

Make sure that you select the *Test Mode* option in the test run variant, and the *Production Mode* option in the variant for the production run. You use the *Detail Log* option in your variants for the deletion run to define whether the system is to display the deleted financial position balances in a list.

You start archiving runs for one legal entity and one accounting system at a time. The system archives all the financial position balances that relate to periods that became inactive when the residence time specified in Customizing was exceeded.

You write data to the archive as follows:

1. Create a variant and choose *Maintain*. The system displays the *Variant Maintenance* menu.
2. Enter data for the following fields:
   Required fields:
   - *Legal Entity*
   - *Accounting System*

   Optional fields:
   - Detail Log
   - *Processing Options* (test or production mode)
   - Archiving session note so you can identify the run at a later point in time.
3. Save your variant, and schedule the job.

Procedure for reloading data

Recommendation

We recommend that you reload archived data only in exceptional circumstances.

1. Create a variant for the test or productive run, and save the variant.
2. Choose *Archive Selection* to select an archiving run.
3. Enter the reload variant and choose *Execute*.
   The system reloads the financial positions balances from the archiving run you selected.

Note

If you carried out a test run, the system does not reload the data to the database.

Note

The system can reload archived runs only after you have carried out the write and deletion processes in production mode.

Procedure for deleting data

If you want to archive or delete data records separately, you can start the deletion program separately for each archive file.

Make sure that you select the *Test Mode* option in the test run variant, and the *Production Mode* option in the variant for the production run. You use the *Detail Log* option in your variants for the deletion run to define whether the system is to display the deleted financial position balances in a list.

1. Choose *Archive Selection* to select an archiving run.
2. Enter the deletion variant and choose *Execute*.

> **Note**
>
> If you select the test mode, the system does not remove the data from the database.

> **Note**
>
> The system displays the archived runs only after you have carried out the write and deletion processes in production mode.

7.12.2.6 Archiving of Postings from Balance Processing

Definition

Archiving object `BA1_B2_BPA` (*Balance Processing Postings*) is used to archive, delete, and reload data for postings from Balance Processing.

Use

The data that can be archived for postings from Balance Processing is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Tables

The system uses archiving object `BA1_B2_BPA` to archive data from the following tables:

- `/BA1/B2_T_BPPHDA`
- `/BA1/B2_T_BPPKFA`

For information about how to display table names, see .

Programs

The following programs are available for BA1_B2_BPA:

Program	Function
/BA1/RB2_BPP_AR_A_WRITE	Write
/BA1/RB2_BPP_AR_A_DELETE	Reload
/BA1/RB2_BPP_AR_A_RELOAD	Delete

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.12.2.7 Archiving of Balance Object Manager

Definition

Archiving object BA1_B6_BO (*Bank Analyzer: Balance Objects Manager*) is used to archive, delete, and reload Balance Object Manager data.

Use

The data that can be archived for the balance object manager is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Tables

The system uses archiving object BA1/B6_BO to archive data from the following tables:

- /BA1/B2_BSOFSI_A
- /BA1/B6_SLS_BO
- /BA1/B6_SLS_BSO

For information about how to display table names, see .

Programs

The following programs are available for BA1_B6_BO:

Program	Function

/BA1/RB6_BOM_AR_WRITE	Write
/BA1/RB6_BOM_AR_RELOAD	Reload
/BA1/RB6_BOM_AR_DELETE	Delete

The standard variants SAP&PROD (productive mode) and SAP&TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.12.2.8 Archiving Market Data

Procedure

1. On the *SAP Easy Access* screen, choose *Tools* > *Administration* > *Administration* > *Data Archiving*.

   The *Archive Administration: Initial Screen* appears.
2. Specify the name of the archiving object. You can choose from the following object names for archiving market data:
   - BA1_F4_FX Bank Analyzer: Archiving of Exchange Rates
   - BA1_F4_FXV Bank Analyzer: Archiving of Exchange Rate Volatilities
   - BA1_F4_IR Bank Analyzer: Archiving of Interest Rates
   - BA1_F4_IRS Bank Analyzer: Archiving of Interest Rate Spreads
   - BA1_F4_IRV Bank Analyzer: Archiving of Interest Volatilities
   - BA1_F4_SEC Bank Analyzer: Archiving of Security Prices
   - BA1_F4_SEV Bank Analyzer: Archiving of Security Price Volatilities
3. To generate an archive file, choose *Write*. The *Archive Administration: Create Archive Files* screen appears.
4. Select a *Variant*.
5. You can enter settings for a variant by choosing *Edit*. The *Maintain Variant* screen appears.
   1. Specify a *Market Data Area*, and other relevant selection criteria for the archiving object in question.
   2. *Test Mode* or *Production Mode*: select the relevant checkbox (depending on whether or not you want to do a test archiving run).
   3. Save your entries and choose *Back*.
6. Specify the *Start Date* and the *Spool Parameters*.
7. Choose *Execute*. The system archives the data in the background.

   For more information, see .

7.12.2.9 Archiving of Counterparty/Issuer Risk Data

Definition

Archiving object `BA1_R4_060` (*RDB: Archiving for Counterparty/Issuer Risk*) is used to archive, delete, and reload counterparty/issuer risk data that is stored in the Result Database (RDB).

Use

The data that can be archived for counterparty/issuer risk is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Table

The system uses archiving object `BA1_R4_060` to archive data from the following tables:

- `/BA1/R4_KFSTRC`
- `/BA1/R4_PACK`
- `/BA1/R4_REC_CR`
- `/BA1/R4_RUN`

For information about how to display table names, see .

Programs

The following programs are available for BA1_R4_060:

Program	Function
`/BA1/RB6_RESDAT_AR_WRITE`	Write
`/BA1/R4_ARCHIVE_DELETE`	Delete
`/BA1/R4_ARCHIVE_RELOAD`	Reload

The standard variants SAP PROD (productive mode) and SAP TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.12.2.10 Archiving of Credit Exposure Data

Definition

Archiving object `BA1_R4_062` (*RDB: Archiving for Credit Exposure*) is used to archive, delete, and reload Credit Exposure data for country risk that is stored in the Result Database (RDB).

Use

The data that can be archived for Credit Exposure is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Table

The system uses archiving object `BA1_R4_062` to archive data from the following tables:

- `/BA1/R2_LAYOUT01`
- `/BA1/R4_KFSTRC`
- `/BA1/R4_PACK`
- `/BA/REC_EXP`
- `/BA1/R4_RUN`

For information about how to display table names, see .

Programs

The following programs are available for BA1_R4_062:

Program	Function
`/BA1/R4_ARCHIVE_WRITE`	Write
`/BA1/R4_ARCHIVE_DELETE`	Delete
`/BA1/R4_ARCHIVE_RELOAD`	Reload

The standard variants SAP PROD (productive mode) and SAP TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.12.2.11 Archiving of Test Analyzer Data for Cash Flow Generation

Definition

Archiving object `BA1_R4_099` (*RDB: Archiving for Test Analyzer*) is used to archive, delete, and reload Test Analyzer data for cash flow generation that is stored in the Result Database (RDB).

Use

The data that can be archived for the Test Analyzer is written to one or multiple archive files. So that the system archives only the data that is no longer required, various criteria have to be met.

Structure

Table

The system uses archiving object `BA1_R4_099` to archive data from the following tables:

- `/BA1/R4_KFSTRC`
- `/BA1/R4_PACK`
- `/BA1/R4_RUN`

For information about how to display table names, see .

Programs

The following programs are available for BA1_R4_099.

Program	Function
`/BA1/R4_ARCHIVE_WRITE`	Write
`/BA1/R4_ARCHIVE_DELETE`	Delete
`/BA1/R4_ARCHIVE_RELOAD`	Reload

The standard variants SAP PROD (productive mode) and SAP TEST (test mode) are available for the deletion program. The system displays a status message in the job log and on the screen (status line) regularly during the write and deletion runs.

7.13 Data Destruction in Bank Analyzer

7.13.1 Destroy Process Controller Data

The *Destroy Process Controller Data* report allows you to delete data records from the Process Controller's status management (for example, for data protection reasons or to reduce the data volume).

Prerequisites

You can delete data records under the following conditions:

- You can only delete entries from closed periods if there are no earlier open periods.
- The contract has already expired. You can determine when a contract is considered expired from the end of the contract term in the Source Data layer (SDL).
- The contract is active but there is a another entry in the Process Controller on a later date than the posting date in the destruction report.

Context

To delete, you use the data destruction object `/BA1/RBR_PROC_CTRL` and proceed as follows:

Procedure

1. On the SAP Easy Access screen, choose *Bank Analyzer* > *Processes and Methods* > *Smart Accounting for Financial Instruments* > *Subledger Accounting* > *Tools*. Call the *Destroy Process Controller Data* transaction.
2. Choose the *Destroy* pushbutton.
3. On the selection screen, choose an existing variant or create a new variant. The system processes variants based on the report `/BA1/RBR_PROC_CTRL_DES`. In a variant, you can limit deletion to various Process Controller contract attributes:
   - Accounting system, legal entity and posting date are mandatory.
   - You can also define whether only expired contracts or all active contracts are deleted.
4. Enter a start date and spool parameters (background printing parameters).
5. Finally, choose F8 or the *Schedule Job* pushbutton to schedule the destruction run.

Results

When the run is completed, you can view the logs either in the data destruction object or in the CVPM process monitor for the process `/BA1/RBR_PROC_CTRL_DESTRUCT`.

> **Note**
>
> Do not execute the process `/BA1/RBR_PROC_CTRL_DESTRUCT` directly because the connection to the data destruction run can be lost.

7.13.2 Destroying Application Event Data of Application Event Manager with BA1_R3_EVENTS_DES

You use the destruction object `BA1_R3_EVENTS_DES` to destroy content from the table of *application events of the application event manager* (`/BA1/IA_AE`). It destroys the legacy data from this table that is no longer relevant. It removes data from this table once the retention period defined in the *Information Retention Management (IRM)* policy has expired and the process is completed.

There is no relationship to other destruction or archiving objects.

The destruction object contains the data destruction program `/BA1/R3_AEP_DES`.

ILM-Related Information for BA1_R3_EVENTS_DES

This object has an ILM object called `BA1_R3_EVENTS`.

Allowed Start:

The start time is the creation date. The retention period of the data starts on the creation date (in ILM, this is `CREATION_DATE`). Optionally, the following time offsets are available:

- `END_OF_MONTH`
- `END_OF_QUARTER`
- `END_OF_YEAR`

Allowed Fields:

- `ACC_SYST` (data element `ACC_SYSTEM`)
- `LEGAL_ENTITY` (data element `ACC_LEGENT`)

Allowed Policy Categories:

`RTP` (retention rules)

Prerequisites

- The data destruction object is assigned to an ILM object in transaction IRM_CUST.
- You can put a legal hold on the business object related to the ILM-object. For more information about legal holds, see Putting Holds on Data for Legal Cases.
- You have activated the Information Lifecycle Management (ILM) business function in the Switch Framework (transaction SFW5).
- You are familiar with the functions for ILM policies and rules and have defined the retention rules and the residence rules for the destruction object in transaction IRMPOL.

Procedure

Finding the Destruction Object for Cash Flow Data

1. Enter the transaction DOBJ.
2. Choose the Position button to search for the destruction object.
3. Double-click the destruction object BA1_R3_EVENTS_DES to view information such as the data destruction program and component.
4. Double-click the Structure Definition in the Dialog Structure to view tables from which data is destroyed.

Customizing the Destruction Object

Proceed as follows to customize the data destruction object and define retention rules:

1. In transaction ILMARA, define an audit area and assign the ILM object.
2. In transaction IRMPOL, define ILM rules for the data destruction object.
3. In transaction ILMSIM, check the retention period rules for the data destruction object by running a simulation (evaluation of the Retention Management definitions).
4. If required, implement the Business Add-In *Reference Date: Determination from Time Reference* (BADI_IRM_OT_STT) to determine a specific reference date from the time reference.

For more information about the Customizing settings for destruction objects, see:

Executing the Destruction Object

You can destroy data in the database by executing a destruction run in transaction ILM_DESTRUCTION for an appropriate data destruction object. For more information about how to execute the destruction object, see Data Destruction with a Data Destruction Object. In step 2 of the procedure, you need to enter the Information Lifecycle Management (ILM) object BA1_R3_EVENTS for the data destruction object BA1_R3_EVENTS_DES.

8 Tools (FS-BA-TO)

Use

You can use this component to call functions that are used in various places in Customizing for Bank Analyzer.

In addition, the following tools are available:

-
- Schedule Manager
- Segmentation Service [page 1843]

Features

Derivation Tool (FS-BA-TO-DE)

The derivation tool enables you to control how the system derives characteristics and key figures from other characteristics and key figures, and how it derives the fixed fields of a field catalog. In Bank Analyzer the system calls derivations from the coding or by using a secondary data source. You can create this secondary data source with the module editor in Customizing for Bank Analyzer.

You can state the derivation environment for deriving the validity of a hedging relationship, for example, in Customizing for *Bank Analyzer* under *Process and Methods* > *Hedge Processes* > *Portfolio Fair Value Hedge* > *Configuration* > *Derivation of Validity*. You use this derivation process in the secondary data source in order to use the characteristics of a transaction to derive whether the transaction is one of the qualified positions or unqualified positions in Hedge Accounting.

Module Editor (FS-BA-TO-ME)

The module editor generates modules that contain a sequence of processing steps. The modules are used to enrich user-defined information and provide the system with secondary data sources.

An application makes entries into the fields of an input structure and calls the module. The system applies each processing step of the module in the sequence defined in Customizing. The system can call function modules, derivations, or primary data sources within the module. The system then makes entries into fields of the output structure.

Modules can have various functions. The selection module of Strategy Analyzer, for example, selects data using the *Primary Data Source* processing step. The calculation module of Profit Analyzer carries out complex calculations for the processing steps *Formula, Derivation, and Function Module*.

You can find the settings for the module editor, for example, in Customizing for *Bank Analyzer* under *Analytics* > *Profit Analyzer* > *Profit Engine* > *Calculation* > *Edit Modular Costing*.

Result Database (FS-BA-TO-RDB)

The Result Database [page 1823] (RDB) is a database in which the system saves results data permanently. These results are then available for further processing in reporting, for example, or for additional calculation runs.

> **Note**
>
> The RDB and the Results Data Layer [page 1364] (RDL) are two different results databases in which the system can store results data. Each database is based on different principles. The RDB is found in a variety of forms in Bank Analyzer. These forms depend on the various areas (Financial Accounting, Basel II). The RDL is a standardized results data store for accounting and risk-based analyses of financial transactions or financial instruments.
>
> For the long-term we recommend that you use the central RDL to store results data in a standardized way. In Customizing for Bank Analyzer you can choose whether the system is to store Basel II-specific results data in the RDB or the RDL.

Processing Framework (FS-BA-TO-PFW)

The processing framework supplies the processing rules with data from various data source categories. The calculation and allocation processing rules are available in Profit Analyzer, for example. The system uses suitable selection conditions to create a worklist. The system can also add further information using a secondary data source. The results records generated by the processing rules are forwarded to temporary storage. The system provides verification lists which you can use to check whether the result records are plausible from a business perspective. The result records are then updated in data drains.

You can also start the processing steps manually. In a typical scenario, you include the processing steps in the Schedule Manager which then carries out an automatic month-end processing on the basis of this.

Run Administration (FS-BA-TO-RUN)

Run Administration [page 430] provides you with various processing functions for the runs in the individual Bank Analyzer applications. Run administration therefore enables standard, general run administration.

Aggregation Tool (FS-BA-TO-AGT)

The aggregation tool is used to aggregate data from primary and secondary data sources, BAPIs, and the Data Processing Framework. The aggregation type is determined using granularities such as the industry or the business partner. Possible aggregation functions are determining minima, maxima, totals, or the number of occurrences of a certain value.

You can find the settings for aggregation, for example, in Customizing for *Bank Analyzer* under *Analytics* > *Historical Database* > *General Settings for the Historical Database* > *General Settings for Data Selection* > *Settings for Aggregation Processes*. You can use the *Aggregation* Business Add-In (BAdI) to override the results from the aggregation process you defined in the Customizing activity *Edit Aggregation*. This enables you to change individual results.

Data Processing Framework (FS-BA-TO-DPF)

The Data Processing Framework provides selection processes for processing data to the Historical Database [page 1478], the Limit Manager [page 1529], and Bank Analyzer-wide to Generic BI Data Extraction [page 1638] and Generic Ad Hoc Reporting [page 1649]. For example, you determine the selection settings in Customizing for the Historical Database in the *Edit Basic Settings for Data Sources* section. Every selection is assigned to a fixed context (application of the Data Processing Framework) which is, in turn, assigned to a certain application of the module editor. Data processing that is either triggered by a report or by generic data extraction, for example, can contain both selection BAdIs as well as aggregations and general selection criteria.

Configurator (FS-BA-TO-CON)

The configurator reads characteristics and key figures and generates customer-specific database tables and field structures for further processing. The system calls these processes "generation". The system currently uses only both *Bank Analyzer* accounting scenarios for generation. For more information, see Generation [page 713].

8.1 Result Database (RDB)

Use

The Analyzers store their results data (key figure values) persistently in the Bank Analyzer Result Database (RDB).

Unlike the Source Data Layer (SDL), the Result Database (RDB) is not used to select transaction data for processing. Instead, it provides result data for reporting, and for extraction to SAP NetWeaver Business Intelligence (BI).

Features

- Saving Key Figures and Characteristics in the RDB
  The RDB has one write interface, which can be used by the RDB clients only. In order to save data in the RDB, the RDB client has to supply this write interface with data during a run.
- Reading Key Figures and Characteristics from the RDB
- Extracting Data
  You can extract the following data to SAP NetWeaver Business Intelligence (BI):
  - RDB administrative information, such as the business status, RDB client, ID of runs, reference object category, version of the RDB run, and process steps.
  - Transaction data
  - Master data that contains the text tables for RDB characteristics
- Archiving
  The Result Database (RDB) contains an archiving function that you can use to save, read, and reload result data. You find the archiving function in each of the applications. In the archiving function of Strategy Analyzer, you can, for example, write RDB runs, archive RDB runs, or manage your RDB runs.
- Displaying Generated Objects [page 1823]
  The RDB generates views and tables for storing data. In the RDB, objects are generated in background processing.

8.1.1 Displaying Generated Objects in the RDB

Use

When Bank Analyzer is used as your productive system, you may need to make client copies. This takes place without checking the consistency of the data. To check whether the client was copied successfully, the data in the

source client and target client has to be compared. To prevent errors occurring in the Result Database (RDB) [page 1823] , you need to ensure that the generated objects have the same status in both clients. If they do not, this could be because particular characteristics no longer exist in the target client, for example.

> **Note**
>
> This function can be used only to display generated objects. It cannot be used to generate objects.

Procedure

1. You call this function from the *SAP Easy Access* screen by choosing *Bank Analyzer* > *Tools* > *Display RDB Generation* .
   In the dialog structure, the system displays all the RDB clients that are available.
2. To display the status of the generated objects, choose the relevant RDB client (double click).
3. The system then displays an overview of the statuses of the generated objects, which are sorted by package type.
4. To display more information about a package type, select the appropriate row.

You have the following options:

Action	Description
Key Figure Structure	The system displays the key figure structure of the object, the Customizing, and the type of generation that was used.
Layout Categories	In addition to key figures and characteristics, the system uses layout categories. These tables are static, and are predefined in the system.
Display Characteristic Structure and DDIC View	You can display the structure and object view for the key figures and characteristics of a package type. You do this by choosing the required object in the relevant column in the table. The system displays a detail screen.

You need the appropriate authorization in order to display generated objects. The system checks authorization object /BA1/R4_CL for the activity *Display* .

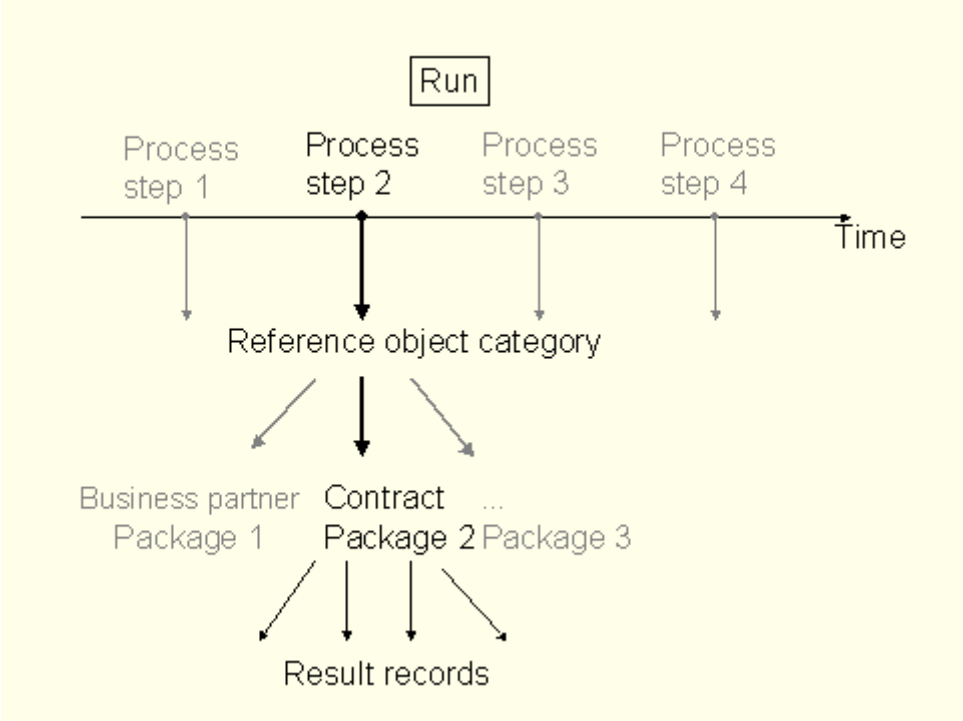
8.1.2 Extraction of Data to BI (Prototype Function, Counterparty Risk)

Purpose

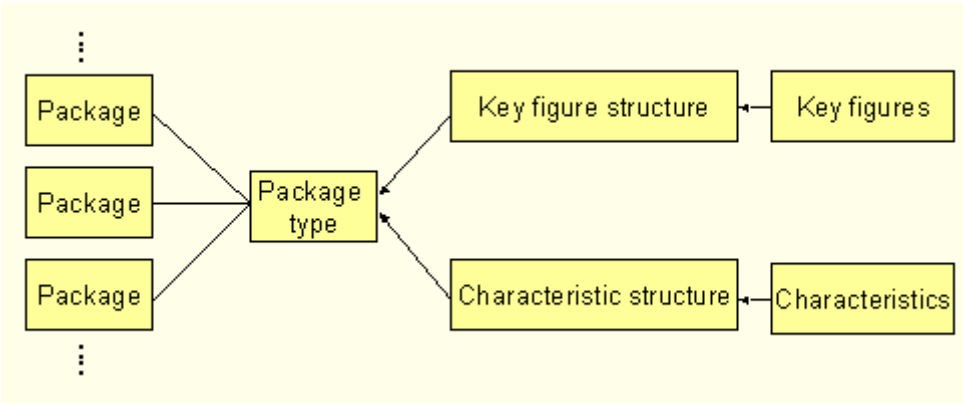
Results data from the risk applications is extracted to SAP NetWeaver Business Intelligence (BI) for reporting purposes.

Architecture

The data from an RDB run is sorted into result packages, which contain result records.



Each package that is written from a client of the RDB to the Result Database itself, is assigned a result package type. You assign the package type to a package in Customizing for the relevant applications. You use the package type to assign packages the characteristics and key figures that the system is to write to the packages of that package type.



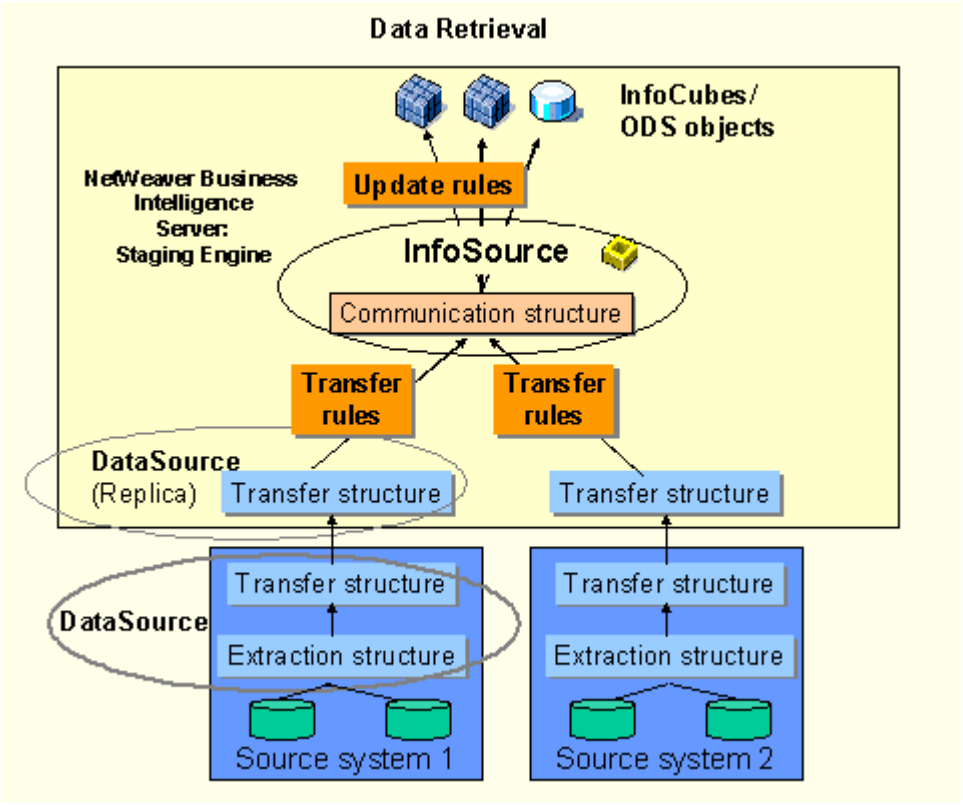
Since all the packages with the same package type always contain the same characteristics and key figures, data is extracted to SAP NetWeaver Business Intelligence (BI) at package type level. In other words, the system generates a data source for the extraction of data to BI for each package type and for each database layout type. For more information about the generation process see the IMG documentation under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Results Management* > *Generate and Edit DDIC Objects.*

Prerequisites for Data Extraction

The DataSource of the package type must have already been generated successfully in the generation transaction for the Result Database. You can find the generation transaction in Customizing under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Generate and Edit DDIC Objects*.

Process Flow

The following sections describe the individual steps involved in extracting a package type from the RDB to BI. The following graphic provides an overview of the BI entities:



Extracting RDB Administrative Information [page 1827]

Extracting Flow Data [page 1828]

Extracting Texts for RDB Fields (Master Data) [page 1829]

8.1.2.1 Extracting RDB Administrative Information

Use

In order to interpret results data from the result database (RDB) in SAP NetWeaver Business Intelligence (BI), you also need to extract RDB administrative information from Bank Analyzer. RDB administrative information includes the business status, the RDB client, the run ID, the reference object type, the version of the RDB run, and the process step.

Procedure

Carry out the following activity once for each BI system:

Create InfoObjects (more specifically, characteristics) in BI for the RDB administrative information. The following table contains a proposed name as well as the data type and length of the characteristics to be created:

Proposed Name for InfoObject	Data Type	Length	Proposed Description	Conversion Routine	Text table
RDB_BSTXX	CHAR	3	RDB: Business Status	- without -	Required
RDB_CLNT	CHAR	2	RDB: Result Database Client	- without -	Required
RDB_NAME	CHAR	45	RDB: Run ID	- without -	- none -
RDB_ROTXX	CHAR	4	RDB: Reference Object Type	- without -	Required
RDB_RUNVERS	NUMC	5	RDB: Version of the RDB Run	- without -	- none -
RDB_PRCST	CHAR	5	RDB: Process Step	- without -	- none -

Caution

Conversion Routine: When you create the above InfoObjects, make sure that you create them without a conversion routine. ALPHA is the default conversion routine. Delete this conversion routine.

Note

Proposed Name for InfoObject: The characteristic values of the InfoObjects RDB_BSTXX and RDB_ROTXX depend on the RDB client. For this reason, the last two characters of the name are reserved for an abbreviation for the RDB client. Proposed name for counterparty/issuer risk: RDB_BSTCR and RDB_ROTCR. We also recommend using the name (abbreviation) of the RDB client for the description of the InfoObject.

Text Table: You can export texts from Bank Analyzer to SAP NetWeaver Business Intelligence (BI) for the InfoObjects RDB_BSTXX, RDB_CLNT, and RDB_ROTXX. For more information, see Extracting Texts for RDB Fields (Master Data) [page 1829].

8.1.2.2 Extracting Flow Data

Once the system has generated the DataSources, you have to carry out the following activities for each DataSource. You can find the generation transaction in Customizing under *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Prototype Functions* > *Generate and Edit DDIC Objects* .

1. In order to extract the flow data you need to know the name of the relevant DataSource for your package type. The name of the DataSource is contained in the results database (RDB) generation transaction (*Generate and Edit DDIC Objects*) in the Bank Analyzer system. Select the package type and choose *Details* . The system displays a list of DataSources for all five database layout categories. Since the counterparty/issuer risk uses only the database layout category 5, this is currently the only relevant DataSource.
   Alternatively, you can access the name of the DataSource for counterparty/issuer risk data in the administrative data for counterparty/issuer risk runs on the *SAP Easy Access* screen by choosing *Bank Analyzer* > *Processes and Methods* > *Credit Risk* > *Credit Risk Portfolio* > *Tools* > *Administration of Counterparty Risk Runs* > *Edit Counterparty Risk Run* . Choose *Goto* > *Results Packages* .
   Make the following settings in your BI system:
2. In the area menu of the BI system choose *SAP Menu* > *Administration* > *Data Warehousing Workbench: Administration* . If multiple RFC destinations exist, the *RFC Destination* dialog box appears. Choose the RFC destination you require. The system then creates the Administrator Workbench.
3. In the *Modeling* screen area, choose *Source Systems* . Select the source system you require and choose *Replicate DataSources* in the context menu (right-click).
4. Create an InfoSource in the Data Warehousing Workbench by choosing *InfoSources* in the *Modeling* screen area. Select the Bank Analyzer application component and choose *Create InfoSource* in the context menu. Select the radio button *Flexible Update In Any Data Target (Apart From Hierarchies)* . Enter an appropriate name for the InfoSource.
5. Double-click the newly created InfoSource to select it. To expand the data area *Transfer Structure/Transfer Rules* , choose *Expand* . Enter the source system and the DataSource (choose *Assign DS*). Save your entries.

   On the *Transfer Rules* tab page, assign an InfoObject to each field. Choose *Propose Transfer Rules* . The communication structure at the top of the screen is now filled. On the *Transfer Rules* tab page in the data area *Transfer Structure/Transfer Rules* , a traffic light appears (green).
   Activate the InfoSource.
6. In the Data Warehousing Workbench, create an InfoCube or a Data Store object. To do so, choose *InfoProvider* in the *Modeling* screen area. Select the InfoArea you require and choose *Create InfoCube* or *Create DataStore Object* in the context menu. If the system does not propose a suitable InfoArea, create a new InfoArea.

   InfoCube: On the *Characteristics* tab page, choose *InfoSources* . To transfer the InfoSource InfoObjects to the InfoCube structure, select the InfoSource you require and choose *Continue* . You can then remove InfoObjects from the structure or add any missing InfoObjects.
   Assign dimensions to the characteristics. Choose *Dimensions* . On the *Define* tab page, you can create dimensions. On the *Assign* tab page, you can assign characteristics to the dimensions. When you have finished assigning characteristics to the dimensions, choose *Continue* .

Activate the InfoCube.

> **Note**
>
> The performance of the reports depends on how you assign the characteristics in the BI InfoCube to the dimensions, since this defines the structure of the database tables for the InfoCube (BI star schema).

DataStore object: Choose *InfoSources* . Select the InfoSource you require and choose *Continue* . Assign the characteristics and key figures to the relevant ODS objects. Activate the DataStore object.

7. In the Data Warehousing Workbench, create an update rule for the InfoCube or DataStore object. To do so, select the InfoCube or DataStore object and choose *Create Update Rules* in the context menu. In the *DataSource* screen area, enter the InfoSource and activate the update rule.

Carry out the following activities each time you want to extract data to BI once you have carried out multiple runs successfully:

1. In the *Modeling* screen area, choose *InfoSources* . Create an InfoPackage for the source system of the InfoSource. To do so, select the source system assigned to the InfoSource and choose *Create InfoPackage* in the context menu.
   Choose the *Data Selection* tab page and define the selection conditions. Check that the correct InfoCube has been entered as the data target. Choose the *Schedule* tab page . Start the job.
2. Choose *Extraction Monitor* . Check whether the job has been completed successfully.

> **Note**
>
> For more information about the settings in BI, see the documentation for SAP NetWeaver Business Intelligence (BI).

You can now use BEX Analyzer or Web Reporting to create reports as required using the data from the InfoCube/ DataStore object.

If the RDB client writes data to a package of the same package type during a different run, you do not have to repeat all the steps and need only retrigger the extraction process (steps 8 and 9).

8.1.2.3 Extracting Texts for RDB Fields (Master Data)

If text tables exist in Bank Analyzer for result database (RDB) characteristics, you can extract these text tables to SAP NetWeaver Business Intelligence (BI).

These RDB characteristics are listed in the table below:

Proposed Name for InfoObject	Proposed Name for DataSource	Text Table or View in Bank Analyzer	Data Element in Bank Analyzer
RDB_BSTXX	BA1_R4_DS_BUS_STAT_XX	/BA1/VR4_BSTATXX	/BA1/R4_DTE_BUSSTAT
RDB_CLNT	BA1_R4_DS_RDB_CLNT	/BA1/R4_RDBCLNTT	/BA1/R4_DTE_RDBCLNT

RDB_ROTXX	BA1_R4_DS_ROBJ_TYPE_XX	/BA1/VR4_ROCATXX	/BA1/R4_DTE_REFOBJ_CATE

> **Note**
>
> The DataSource in the table above is not the DataSource generated in Bank Analyzer in the generation transaction.
>
> Proposed Name for the InfoObject: The characteristic values of the InfoObjects RDB_BSTXX and RDB_ROTXX depend on the RDB client. For this reason, the last two characters of the name are reserved for an abbreviation for the RDB client. Proposed name for counterparty/issuer risk: RDB_BSTCR and RDB_ROTCR.

When the system has generated the DataSources, you have to carry out the following activities once for each DataSource.

1. In Bank Analyzer, create the DataSources for the texts using transaction RS02. Select the *Texts* radio button.Enter the name of the data source. (note the proposed names in the *DataSource Proposed Name* column in the table above). Then choose *Create*.
   Choose the Bank Analyzer application component as the application component.
   In the *View/Table* field in the screen area *Extraction from DB View*, enter the table name or view name (note the proposed names in the *Text Table or View in Bank Analyzer* column in the table above).
   Make the following settings in your BI system:
2. In the area menu for the BI system, choose *SAP Menu* > *Information Systems* > *Business Information Warehouse* > *Administration* > *Data Warehousing Workbench: Administration* .If multiple RFC destinations exist, the *RFC Destination* dialog box appears. Choose the RFC destination you require. The system then sets up the Data Warehousing Workbench.
3. In the *Modeling* screen area, choose *Source Systems*. Select the source system you require and choose *Replicate DataSources* in the context menu (right-click).
4. Create an InfoSource in the Administrator Workbench by choosing *InfoSources* in the *Modeling* screen area. Select the Bank Analyzer application component and choose *Create InfoSource* in the context menu. Select the radio button *Direct Update of Master Data*. The InfoSources have to be defined as *direct update of master data* for the InfoObject for which you want to extract the texts (note the proposed names in the *InfoObject Proposed Name* column in the table above).
5. Double-click the newly created InfoSource to select it. To expand the data area *Transfer Structure/Transfer Rules*, choose *Expand*. Enter the source system and the DataSource (choose *Assign DS*). Save your entries. On the *Transfer Rules* tab page, assign an InfoObject to each field. Choose *Propose Transfer Rules*. The communication structure at the top of the screen is now filled. On the *Transfer Rules* tab page in the data area *Transfer Structure/Transfer Rules*, a traffic light appears (green).
   Activate the InfoSource.

Carry out the following activities each time you want to extract data to BI once you have executed several runs successfully.

1. In the *Modeling* screen area, choose *InfoSources*. Create an InfoPackage for the source system of the InfoSource. To do so, select the source system assigned to the InfoSource and choose *Create InfoPackage* in the context menu.
   Check that the correct InfoCube has been entered as the data target. Choose the *Schedule* tab page. Start the job.
2. Choose *Extraction Monitor*. Check whether the job has been completed successfully.

> **Note**
>
> For more information about the settings in BI, see the SAP NetWeaver Business Intelligence documentation.

The entries selected from Bank Analyzer should now be available in the InfoObject text table.

8.2 Generation Report for Calculation

You can use this report to generate runtime programs manually for the calculation function. These programs are usually generated when a module version is saved in Customizing.

However, the automatic generation may fail after a transport, or in a client copy. In such cases, you can use the report to generate the runtime programs for the calculation function manually.

To access the report, on the *SAP Easy Access* screen choose *Bank Analyzer* > *Tools* > *Generation Report for Calculation*

Generation status:

Depending on the mode (test/update run), the generation report can display one of the following statuses:

- Generated (update mode): The runtime program for the calculation function is up-to-date.
- Generation required (test mode): To update the runtime program for the calculation function, you must start the generation report in update mode, and run the report for the module versions listed.
- Generation errors (update mode): The generation report was required, and running the report has resulted in errors. In this case, you have to check the relevant Customizing settings.

Activity

1. Use the input help to choose the application.
2. Use the input help to choose an environment.
3. Use the input help to choose a module.
4. Use the input help to choose a module version.
5. To start the report in test mode, make sure that the *No Generation* indicator is set. Then choose *Execute* to start the report in test mode.
6. Depending on the generation status, you can start the report in update mode. To do so, make sure the *No Generation* indicator is **not** set. Then choose Execute.
   If you want to run the report for all module versions in the application, specify only the *Application*, and leave all other fields blank.

8.3 Transport and Client Copy

Use

You use the client copy tool to set up a new client with the same business processes as an existing client.

You use the transport tool to transport the Customizing settings for a client to another client. You have the additional option of transporting global Customizing between systems.

Integration

Bank Analyzer uses the standard functions for client copy and for transport. However, there are points that you need to bear in mind for the system landscape for Bank Analyzer [page 13], and the postprocessing of the client copy or of the transport. See also:

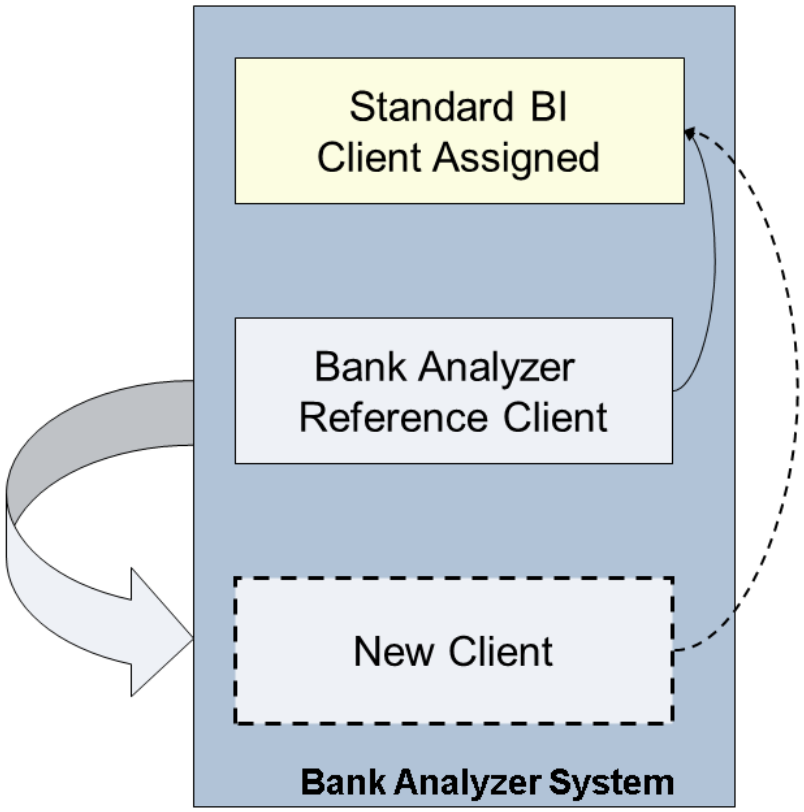
- Client Copy in Bank Analyzer [page 1833]
- Transport in Bank Analyzer [page 1838]
- Risk of Data Loss by Changes to Generated Tables [page 1839]

8.3.1 Client Copy in Bank Analyzer

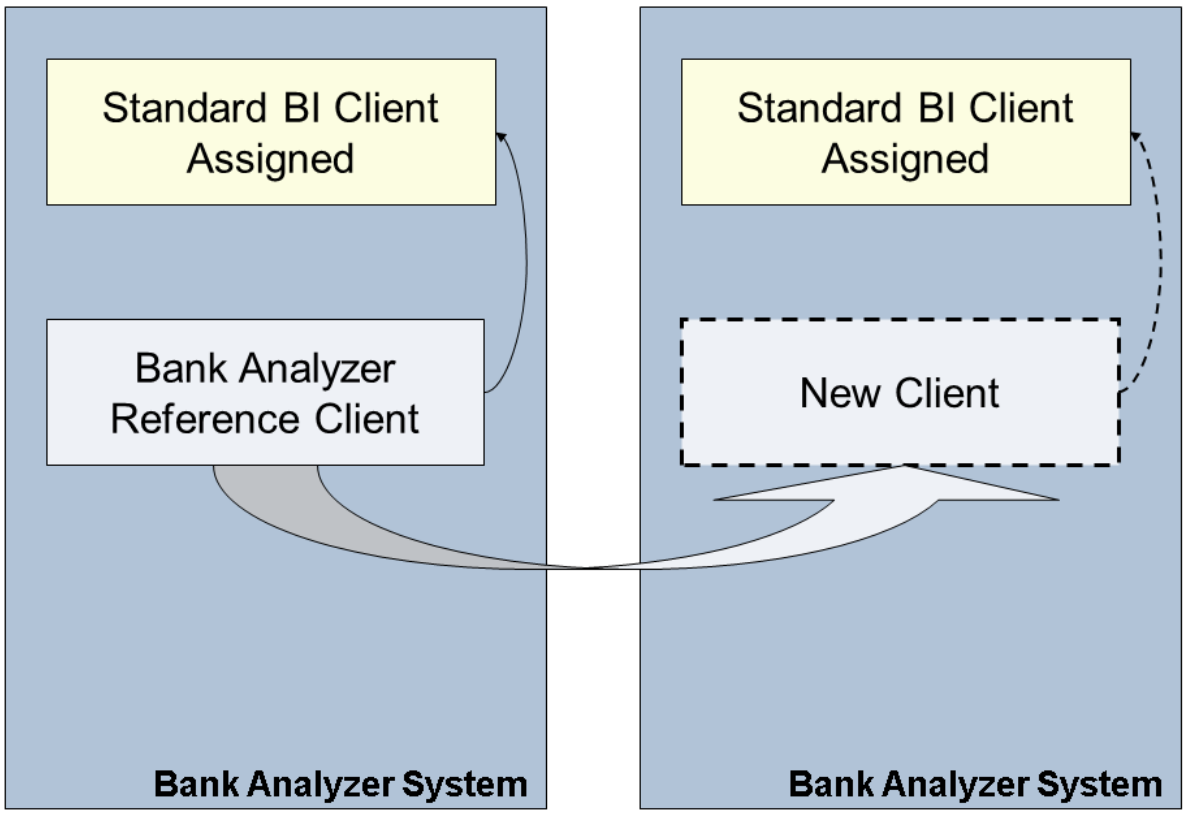
Use

The client copy function in Bank Analyzer (BA) is based on the standard client copy function (see). In Bank Analyzer, the client copy needs additional preprocessing and postprocessing. Clients can be copied in the following ways in Bank Analyzer:

All Bank Analyzer clients refer to the standard BI client for the Bank Analyzer system. A local client copy can be illustrated as follows:



For a remote client copy, you have to have first created or set up a standard BI client. Bear in mind that only one standard BI client from the source system is copied to the standard BI client of the target system. A remote client copy can be illustrated as follows:



Activities

There are different procedures to be followed, depending on which type of client copy is used. The process of copying a client is divided into the following steps:

1. Preprocess for client copy
2. Make the standard client copy
3. Postprocess the standard client copy
4. Import the InfoSets
5. Copy or import master data
6. Copy the application data to generated tables
7. Copy index data

> **Note**
>
> Note that each step is a prerequisite for the following step. If errors occur within one step, you might not be able to continue without encountering further errors.

Explanation of the Individual Steps

1. **Preprocess for client copy**
   First define a new client for a local or a remote client copy on the *SAP Easy Access* screen under *Tools* → *Administration* → *Administration* → *Client Administration* → *Client Maintenance*. Assign the new client to a logical system.

   > **Note**
   >
   > Since the same standard BI client is referenced in the local client copy, you do not have to make any further settings. You can now copy the client.

   However, for remote client copy, you have to bear the following points in mind:
   - Both the source system and the target system must have the same support packages at all relevant levels (for example, *FIN-B*, *Bank Analyzer*).
   - A standard BI client must be set up and available in the target system.
   - The source and the target systems must have the same DDIC status. Therefore, you have to transport all non SAP DDIC and all non ABAP objects from the source system to the target system.

   When you have met these prerequisites, proceed as follows:
   Configure the tool BI system as explained in SAP Note 631416. You should note the following here:
   - The standard BI client must already have been defined.
   - The logical name of the client must already be defined.
   - The BI batch user must be set up.

   If you have the relevant administrator authorization, go to Customizing for *Bank Analyzer* and choose *Basic Settings* → *Settings for Meta Data* → *Call Data Warehousing Workbench in BI*. The system automatically creates the Myself connection. The Myself connection is the connection from the BI to the standard BI client - in other words, to itself. To test the Myself connection, choose *Source Systems* in the Data Warehousing Workbench of the tool BI system and then *Check*.
2. **Make the standard client copy**
   This is the procedure for both the local and the remote client copy:
   First check whether the RFC destinations are correctly set in the source system. On the *SAP Easy Access* screen, choose *Bank Analyzer* → *Tools* → *Transport and Client Copy*, and then in the Customizing screen area choose *Check RFC Destinations*.
   Then run the standard client copy in the target system in the same way as described in the standard function .
3. **Postprocess the standard client copy**
   As a prerequisite for postprocessing, you have to work through the following steps for the local and for the remote client copy.
   Log on in the new client. You need sufficient authorization to do this (for example, `SAP_ALL`).
   Define or check the system landscape. You do this in Customizing for *Bank Analyzer* under *Basic Settings* → *Global Settings* → *Define System Landscape for Bank Analyzer*.
   Define and check the RFC destination in the source client.
   Define and check the RFC destination in the target client.
   Define and check the logical system of the new client on the *SAP Easy Access* screen by choosing *Tools* → *Administration* → *Administration* → *Client Administration* → *Client Maintenance*.
   Based on the system settings, assign the RFC destinations in Customizing for *Bank Analyzer* by choosing *Analytics* → *Accounting: Merge Scenario* → *Basic Settings* → *Technical Settings* → *Assign RFC Destinations*.
   For remote client copy, these additional steps must be followed in the target client:
   In the tool BI system, generate the InfoObjects for the fixed fields. On the *SAP Easy Access* menu, choose *Bank Analyzer* → *Tools* → *Transport and Client Copy*. Call the report under *Postprocessing* and choose the *Objects for BI Initial Generation* function. Then start the report.

Using the transport tools, import the InfoObjects from the source tool BI system.
You start postprocessing from the *SAP Easy Access* screen by choosing *Bank Analyzer* → *Tools* → *Transport and Client Copy* . There you choose *Postprocess*. To call processing logs, choose the *Logs* icon. The most up-to-date logs are in the top node of the table. If you want more detailed information about the postprocessing, double-click on the log. You can choose from the following types of logs:

- Success messages: These contain additional information about the data integrated in the process.
- Warnings: These check the places where warnings occurred.
- Error messages: These indicate inconsistent Customizing in the new client. Check whether these error messages were already available in the source client. If necessary, check the source client.

4. **Import the InfoSets**
   For the remote client copy, import the relevant InfoSets from the source system. In Customizing for *Bank Analyzer*, choose *Basic Settings* → *Settings for Meta Data* → *Call Data Warehousing Workbench in BI*, and then choose the *Transport Connection* function. From there, you can select and transport the relevant InfoSets.
5. **Copy or import master data**
   Copy or import the master data of the InfoObjects. For remote client copy, you have to import the master data by means of the BI extraction function. You have the following options:
   Choose the relevant method from BI.
   From the *SAP Easy Access* screen, choose *Bank Analyzer* → *Tools* → *Transport and Client Copy*, and then choose *Copy BI Master Data*.
   For the local client copy, you also choose the *Copy BI Master Data* function. Using the report that is executed, the data is replicated and, as a result, the value of the `0SOURSYSTEM` characteristic is changed according to the settings in the System Landscape for Bank Analyzer [page 13].
6. **Copy the application data to generated tables**
   This step is necessary if the following copy profiles are being used for the technical client copy:
   - `SAP_ALL` All client-dependent data without change documents
   - `SAP_APPL` Customizing and application data without change documents
   - `SAP_APPX SAP_APPL` without authorization profiles and roles

   You have to copy the application data that is saved in the generated tables for the following reason: The technical copy cannot copy application data that is saved in the generated tables. The reason for this is that, at the time of the technical copy, the system has not yet run the Bank Analyzer postprocessing in which the tables are generated.
   To copy the application data that is saved in the generated tables, on the *SAP Easy Access* screen choose *Bank Analyzer* → *Tools* → *Transport and Client Copy* , and then *Copy Application Data* (Report `FINB_TR_CC_COPY_DATA`). Do not start the report until the Bank Analyzer postprocessing is completed (including any restarts that might be necessary). Otherwise the generated tables will not yet exist in the target client.
   The technical background of the report is as follows: Since the generated tables are named arbitrarily, the system first has to identify the technical names of the table pairs that are logically connected. To carry out this step, select the *Collect Trans. Data* option. When the system has identified the table pairs, start the copying process by selecting *Copy Transaction Data*. During the copying process, the system copies the table entries, pair by pair. You use the package size to determine the maximum number of table entries to be processed simultaneously in one block. Schedule this copying process as a background process. The system terminates this process if some tables have already been completed or partly copied. The system writes this procedure to the log to enable new starts and to ensure the consistency of the data. If you want to start the report a number of times, set the *Restart* indicator.

> **Recommendation**
>
> We recommend that you carry out a simulation run first. To do this, set the *Test Run* indicator.

If you require a list, set the *List Output* indicator. To display the result of the processing steps in the application log, from the *SAP Easy Access* screen, choose *Bank Analyzer* → *Tools* → *Transport and Client Copy*. Alternatively, call transaction *SLG1 (Analyze Application Log)* and enter object `FINB_TR` and subobject `FINB_TR`.

> **Note**
>
> If you encounter serious problems, you might have to completely reset all of the settings using *Reset Administration*. This action deletes the information about the table pair relationships and about which tables the system has already copied. However, this action does not delete the generated tables and any existing table entries in the generated table. Use the *Reset Administration* option with caution as, in exceptional cases, it can result in data inconsistencies. Therefore, we strongly recommend that you use this option only after consulting your SAP support team.

7. **Copy index data**

   This step is necessary if the following copy profiles are being used for the technical client copy:

   - `SAP_ALL` All client-dependent data without change documents
   - `SAP_APPL` Customizing and application data without change documents
   - `SAP_APPX SAP_APPL` without authorization profiles and roles

   You have to copy the index data that was saved in the DataStore objects of the tool BI system in a separate step for the following reasons: The technical copy cannot copy any index data. The reason for this is that the system has not yet run the Bank Analyzer postprocessing in which the DataStore objects are generated at the time of the technical copy. The technical copy cannot employ the Bank Analyzer-specific modeling that uses the technical field `OSOURSYSTEM` for client-specific data.

   To complete the index data, go to the *SAP Easy Access* menu and choose *Bank Analyzer* → *Tools* → *Transport and Client Copy*. There you choose *Copy Index Data* (Report `/BA1/RF3_ODS_CLIENT_COPY`).

   > **Recommendation**
   >
   > We recommend that you carry out a simulation run first. To do this, set the *Test Run* indicator. Do not set the indicators *Synchronous Call* and *Execute Online* as these are for the use of the SAP support team.

   Note that you have to call the report in the target client. The report first checks which DataStore objects exist in the source client, and then checks whether these objects are available and activated in the target client. The report additionally checks whether the DataStore objects already contain data for the source system ID of the target client. You can start the report online only, but as it has a short runtime, this poses no problems. When the report has completed all of these checks, background processes are automatically generated. These processes copy the index data to packages of predefined sizes. You can configure the background processes in Customizing for *Bank Analyzer* under: *Basic Settings* → *Parallel Processing* → *Maintain Job Distribution*. You do this for Application Type `F3_ODS_CC`. For a remote client copy, the report uses the RFC connection to the tool BI system of the source client. You must have previously created this connection.

8.3.2 Transport in Bank Analyzer

Use

Transport in Bank Analyzer is based on the SAP correction and transport system.

> **Note**
>
> In Bank Analyzer, there are some points in transport postprocessing that you should bear in mind.

Activities

The transport process in Bank Analyzer is divided into the following steps:

1. Determine the RFC destinations
2. Release the transport requests
3. Postprocess transport requests

1. Determine the RFC destinations

Each client that is the target of an import has to have an RFC destination assigned to it. To check the assignment of the RFC destinations, proceed as follows:

1. From the *SAP Easy Access* menu, choose *Bank Analyzer* > *Tools* > *Transport and Client Copy*.
2. To start the report to display the results, choose *Check RFC Destinations*.

> **Note**
>
> You cannot transport the following settings. They have to be processed locally in each client:

   - RFC destinations of the transport tools
   - RFC destinations of the accounting component

2. Release the transport requests

The system tests the transport request before it releases it. If error messages are issued during the test, the system can stop the release of transports. To identify the errors, analyze the logs. Correct the errors or remove the faulty transport objects from the transport request.

3. Postprocess transport requests

You often have to activate or check objects and data from Bank Analyzer after the import process. For every after-import process, the system writes transport logs that contain important information about whether all of the activation processes were correctly completed, or about whether problems have occurred. If the process needs to be restarted, proceed as follows:

1. From the *SAP Easy Access* menu, choose *Bank Analyzer* > *Tools* > *Transport and Client Copy*.
2. Choose *Postprocess Trans. Req.* to start the report.
3. Check the logs on the report.

8.3.3 Risk of Data Loss by Changes to Generated Tables

Bank Analyzer uses generation in many places to attain the maximum flexibility of the data model without having a negative effect on the performance of the runtime. There are some places where the system **generates database tables** that are later used to store live application data.

Note for the transport of changes to Customizing that when you import this Customizing to the target system, the program might make automatic changes to tables that have already been generated. The result might be irreversible database conversion as a result of the deletion of fields or changes to field categories in the configuration. Existing application data might then be lost. When you transport deletions of configuration, the system might completely delete associated generated tables. In addition, references to application data might be lost which, in turn, means that you can no longer read this data.

The system does not provide support here because it checks whether the application data (dependent data) is available in the generated tables for the source system only. Incompatible changes are therefore **not permitted**.

If there is no application data in the source system, the change or deletion is permitted. The system cannot check the target system because the target system is unknown during the creation of the transport request. The transport request can be transported to any system.

> ⚠ Caution
>
> To minimize the risk of data loss, use a fast backup mechanism for live Bank Analyzer systems.

8.3.4 Language Activation

Use

Language activation is used for multilingual systems. Translated texts can thus be transferred to generated objects.

> Example
>
> Texts from technical fields and hierarchy types from segment hierarchies can be written to the related InfoObjects of the Tool BI system [page 16] in several languages.

Prerequisites

To be able to activate a language, the following prerequisites must be met:

- The settings must have been configured in a source system and, by means of an appropriate translation procedure, have been translated into a second language. This procedure must be complete before a language can be activated.
- All of the target languages for translation must be installed in the system.

- The relevant Customizing settings have been activated.

Activities

To activate texts in translated languages, proceed as follows:

1. From the *SAP Easy Access* menu, choose *Bank Analyzer* > *Tools* > *Transport and Client Copy*.
2. In the *Activate Languages* section, choose *Activate*.

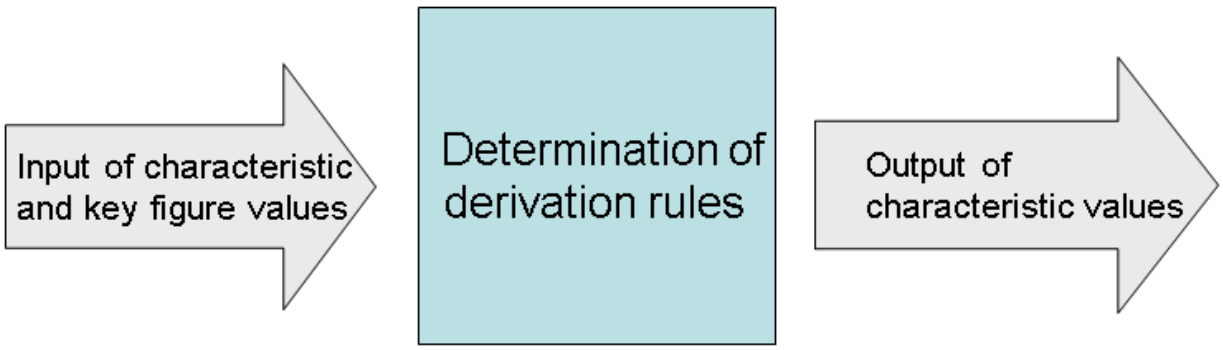
   > **Note**
   >
   > For application *BATO* (Bank Analyzer Tools & Services), you can select the objects */BA1/V_TF3TFLD* (technical fields) and */BA1/F0_H_IOBCREATE* (hierarchy types of segment hierarchies) to activate the languages.

3. Define the language for the text copying process.
4. Start the report. The system displays the log that you can edit.

8.4 Derivation

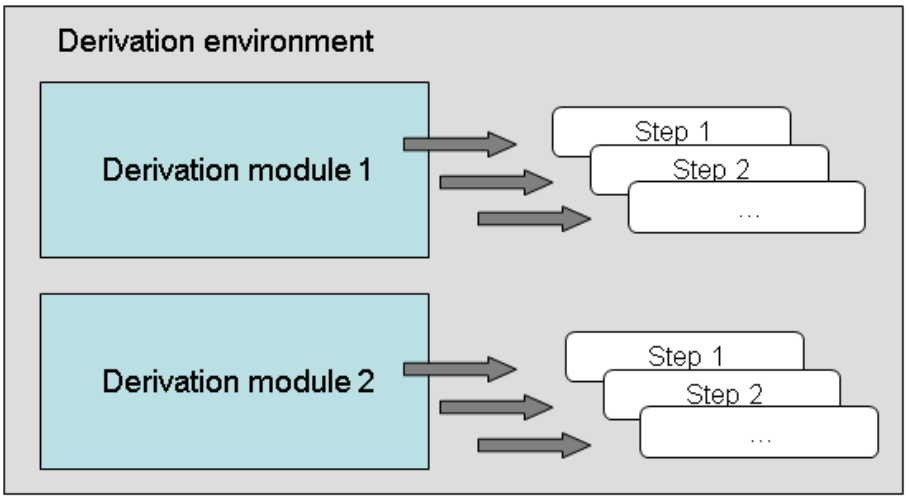
Use

You use derivation to calculate the content of fields (characteristic values) from the values of other characteristics and key figures. You can define your own derivation rules.



For more information, see Generation Report for Derivation [page 1842] and Copying Rule Entries [page 1842].

General Derivation Process



The general derivation process is hierarchically structured:

The derivation environment is a container for the settings of the derivation run. You create derivation modules in the derivation environment. These modules contain derivation steps that determine the strategy of the derivation process.

You can use the following derivation steps:

- Derivation Rule
- Value assignment
- Initialization
- Function modules
- Business Add-In (BAdI)

You can create additional function modules and BAdIs and use them in the derivation process.

> **Note**
>
> You can use key figures for the *Function Module* step only. They are not supported by the other steps.

Integration

Derivation can be used as a module across Bank Analyzer (for the Source Data Layer and for other components of Bank Analyzer).

- Risk Exposure
- Regulatory Reporting
- Balance Analyzer

8.4.1 Generation Report for Derivation

Use

This report generates structures for the derivation environment. You use this report when automatic generation or postprocessing using client copy fails, for example.

Prerequisites

You have defined a derivation environment in Customizing of Bank Analyzer.

You do not define the derivation steps until after the automatic generation.

Activities

1. From the *SAP Easy Access* screen, choose *Bank Analyzer* > *General Functions* > *Tools* > *Derivation* > *Derivation Report for Derivation Tool*.

2. Select an *Application* and (optionally) an *Environment* using the input help (F4).

If you do not select an environment, the system generates the report for the complete application.

3. To carry out the generation run in test mode, set *Test Run* and choose *Execute*. The system displays the status messages *Generated* (generation is not necessary) or *Generation Required* for the respective applications and derivation environments. If you do not carry out a test run, the system displays the status messages *Generated* or *Generation Error* (generation proposed) for the respective applications and derivation environments.

We strongly recommend that you carry out the test run so that the system recognizes the structures that are required for the generation.

If an error occurs during generation, check or edit the Customizing settings for the derivation environment or for the characteristics and key figures.

8.4.2 Copying Rule Entries

Context

You use this report to call up rule entries, and to attach them to the report.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Tools* > *Derivation* > *Copy Rule Entries*.
2. In the *RFC Data* box, use the input help to select a logical source system.
3. In the *Data Selection* box, select the *application* and the *rule ID*. If you choose the *Selection Criteria* pushbutton, the system lists all of the characteristics that are used as source fields in the derivation rule.
4. Specify a business record date.

   > **Note**
   >
   > You can include or ignore versioning during the derivation.

   - Without versioning: Specify the business record date only. The system then includes all rule entries that were produced as of this date.
   - With versioning: Specify the business record date and the time stamp (range). If several versions meet the selection criteria, the system proposes the most recent version.
5. In the *Correction Server* box, enter your data as required.
6. Choose *Execute*.
7. Check to see if the rule entries have been copied to the current system.

8.5 Segmentation Service

Use

This function is a technical service that applications (such as primary objects, Analyzers) can use to store characteristics and their values in a flexible way. The applications can get the segmentation service to store and manage a selection of characteristics for them. The application itself only needs to store a reference (in the form of a unique key) in its own objects. The combinations of characteristic values are stored in the segmentation service in the background.

Prerequisites

You have configured the use of the segmentation service in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Segmentation Service*.

Features

In addition to the generic storage of characteristics, the segmentation service provides other service functions, which enable applications to make flexible selections according to characteristic combinations, for example.

Furthermore, the segmentation service provides applications with the option of defining field control (optional entry fields, required entry fields, display fields, invisible, checks for value tables, entering default values) for characteristic structures.

Characteristic structures, defined as part of the segmentation service, are used in several different places in Bank Analyzer, for example:

- In the attribution service for primary objects
  This gives users the option of enhancing the data model for primary objects to include customer-specific characteristics.
- To define position classes in the Source Data Layer (SDL)
- In the Analyzers

8.5.1 Reorganizing Data in a Structure

Prerequisites

You have defined characteristic structures in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Segmentation Service*.

Context

You use this transaction to delete the data in a characteristic structure from the segmentation service. This data is redundant because it is no longer used by the relevant primary object, for example, when the primary object data has been archived.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Tools* > *Segmentation Service* > *Reorganize Data of a Structure*.
2. Select a *structure*.
3. If you wish to choose additional selection criteria, choose *Field Selection*.
4. Choose *Execute*. The system reorganizes the data in the background. The results of the reorganization are displayed in a log.

8.5.2 Reorganizing Data of Multiple Structures

Prerequisites

You have defined characteristic structures in Customizing for *Bank Analyzer* under *Basic Settings* > *Settings for Meta Data* > *Characteristics* > *Segmentation Service*.

Context

You use this transaction to delete the data in several characteristic structures from the segmentation service. This data is redundant because it is no longer used by the relevant primary object, for example, when the primary object data has been archived.

Procedure

1. On the *SAP Easy Access* screen, choose *Bank Analyzer* > *Tools* > *Segmentation Service* > *Reorganize Data of Multiple Structures*.
2. Select several structures by making entries in the *Structure ... to* fields.
3. Choose *Execute*. The system reorganizes the data in the background. The results of the reorganization are displayed in a log.

8.6 Reorganization

The following reorganization tools are available for the Data Processing Framework and the primary data sources:

- Delete temporary data (see pushbutton with quick info text *Program Documentation*)
- Display and delete logs (see pushbutton with quick info text *Program Documentation*)
- Deleting Data Packages for Primary Data Sources [page 1621]
- Display logs for deleted data packages (see Delete Data Packages for Primary Data Sources)

8.7 Schedule Manager

Definition

A tool in Bank Analyzer that you can use to execute and monitor complex business processes, such as period-end closing. When you schedule a run, you always have to specify the program with which the system calls the run. There is a separate program for each run category in an application.

The table below shows for which runs and run categories the Schedule Manager can be used, and their corresponding programs.

Application	Run Category	Program
Account Pooling	Account Pooling [page 383]	/BA1/DE_GM_MAIN_COMP
Account Pooling	Stress Run for Account Pooling [page 429]	/BA1/DE_GM_MAIN_COMP_SCEN
Determination of the Free Line for Facilities	Determination of the Free Line for Facilities [page 395]	/BA1/DE_GM_MAIN_FL
Determination of the Free Line for Facilities	Stress Run for Determination of the Free Line for Facilities [page 429]	/BA1/DE_GM_MAIN_FL_SCEN
Distribution of Facilities	Distribution of Facilities [page 399]	/BA1/DE_GM_MAIN_RLV
Distribution of Facilities	Stress Run for Facility Distribution [page 429]	/BA1/DE_GM_MAIN_RLV_SCEN
Collateral Distribution	Collateral Distribution [page 404]	/BA1/DE_GM_MAIN_SV
Collateral Distribution	Stress Run for Collateral Distribution [page 429]	/BA1/DE_GM_MAIN_SV_SCEN
Determination of Default	Determination of Default [page 411]	/BA1/RA_DFLT4280_RUN
Determination of Default	Stress Run for Determination of Default [page 429]	/BA1/RA_DFLT4280_RUN_SCENARIO
Fair Value Effectiveness Test [page 1036]	Effectiveness Test Run [page 1039]	/BA1/R8_FVHE_MAIN_RUN
Fair Value Effectiveness Test	Generation of Time Series for Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MAIN_RUN
Fair Value Effectiveness Test	Modifying a Time Series for the Retrospective Regression Analysis [page 1084]	/BA1/R8_FVH_BTS_MODIFY
Cash Flow Hedge Analysis [page 1107]	Subvaluation Run [page 1128]	/BA1/R5_CFHA_BTC_RUN

Application	Run Category	Program
Cash Flow Hedge Analysis	Aggregation Run [page 1129]	/BA1/R5_START_BATCH_AGGR
Cash Flow Hedge Analysis	Joint call of Valuation Run [page 1111] and Aggregation Run [page 1123]	/BA1/R5_EXECUTE_CFHA
Portfolio Fair Value Hedge [page 1156]	Initial Generation Run [page 1169]	/BA1/R9_VPM_BTC_ID_RUN
Portfolio Fair Value Hedge	Portfolio Item Run (Merge Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_RUN
Portfolio Fair Value Hedge	Period-End Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PE_RUN
Portfolio Fair Value Hedge	Period-Start Run for Portfolio Items (Subledger Scenario) [page 1172]	/BA1/R9_VPM_BTC_MR_PS_RUN
Portfolio Fair Value Hedge	Define End of Valuation Period	/BA1HM/RAH_M_PFVH_SET_VP_END
Portfolio Fair Value Hedge	Adjust Initial Portfolio Fair Value Hedges for Next Valuation Period	/BA1HM/RAH_M_PFVH_ADJUST
Portfolio Fair Value Hedge	(Re-) Designation of Initial Portfolio Fair Value Hedges	/BA1HM/RAH_M_PFVH_DESIG
Credit Exposure [page 1192]	Credit Exposure Run [page 1238]	/BA1/R2_CRA_MAIN_CRE
Credit Exposure	Credit Exposure Stress Run [page 1261]	/BA1/R2_CRA_MAIN_CRE_SCEN
Credit Exposure	Credit Exposure Run for Revolving Retail	/BA1/RR2_CRE_REVRET
Credit Exposure	Approximate Stress Run [page 1274]	/BA1/RR2_CRE_SCENARIO
Credit Exposure	Aggregation Run for Ad Hoc Calculation [page 1276]	/BA1/RR2_SCEN_AGGR
Country Risk [page 1326]	Country Risk Run [page 1336]	/BA1/R2_CRE_MAIN
Historical Database [page 1478]	Historization Run for Data Layers [page 1481]	/BA1/R6_HIST_RUN
Historical Database	Historization Run for Bank's Models [page 1484]	/BA1/R6_HIST_RUN_MODEL
Historical Database	Determine Default Rates [page 1488]	/BA1/R6_CF_DEFAULT_4210
Historical Database	Determine Average Default Rates [page 1491]	/BA1/R6_CF_DEFAULT_4220
Historical Database	Determine Default Figures [page 1493]	/BA1/R6_CF_DEFAULT_KEYFIGURES

Application	Run Category	Program
Historical Database	Exporting Data to In-House Models [page 1498]	/BA1/R6_MOD_RUN
Historical Database	Calculation of Migration Matrices [page 1496]	/BA1/R6_CF_MIGRATIONS
Historical Database	Stress Run for Supplying Models with Data [page 1505]	/BA1/R6_MOD_RUN_SCENARIO
Historical Database	Stress Run for Supplying Data [page 1506]	/BA1/R6_FDB_RUN_SCENARIO
Regulatory Reporting Interface	Processing Framework Run Copying the Start ID	/BA1/S1_COPY_STARTID
Regulatory Reporting Interface	Processing Framework Run Creating Runs	/BA1/S1_CREATE_RUN
Regulatory Reporting Interface	Processing Framework Run Release Run	/BA1/S1_RELEASE_RUN
Regulatory Reporting Interface	Processing Framework Run Creating a Worklist	/BA1/S1_CREATE_WRKL
Regulatory Reporting Interface	Processing Framework Run Processing a Run	/BA1/S1_PROCESS
Regulatory Reporting Interface	Data Extraction Runs	/BA1/S3_EXTRACT_RUN
Limit Manager [page 1529]	Limit Utilization Run [page 1551]	/BA1/R3_UTILIZATION_RUN_DPF

More Information

Schedule Manager

8.8 Framework for the Principle of Dual Control (CA-GTF-TS-PDC)

Purpose

The *Framework for the Principle of Dual Control* provides an application the option of validating certain process flows with a principle of multiple control. The following options are then available for the application:

- A function for setting the number of users who have to give their approval ("release procedure"). One user enters the data, and up to three other users can then "release" or approve this data.
- Uniform design for display of all releases
- Framework for additional uses of the object

Implementation Considerations

Using the *Framework for the Principle of Dual Control*, you can define data entries that are subject to a special release rule.

According to the guidelines for banks, for example, one or more employees must be designated to check the entries made by a user creating data to ensure they are accurate. In addition, certain approval powers are defined for the employees in a bank (such as who is permitted to commit which limits to customers and enter them in the system). This release functionality for data entries can also support the correct handling of these approval power regulations. Moreover, the auditors and/or controllers can track which employees have released data entries, making them jointly responsible for the results.

Integration

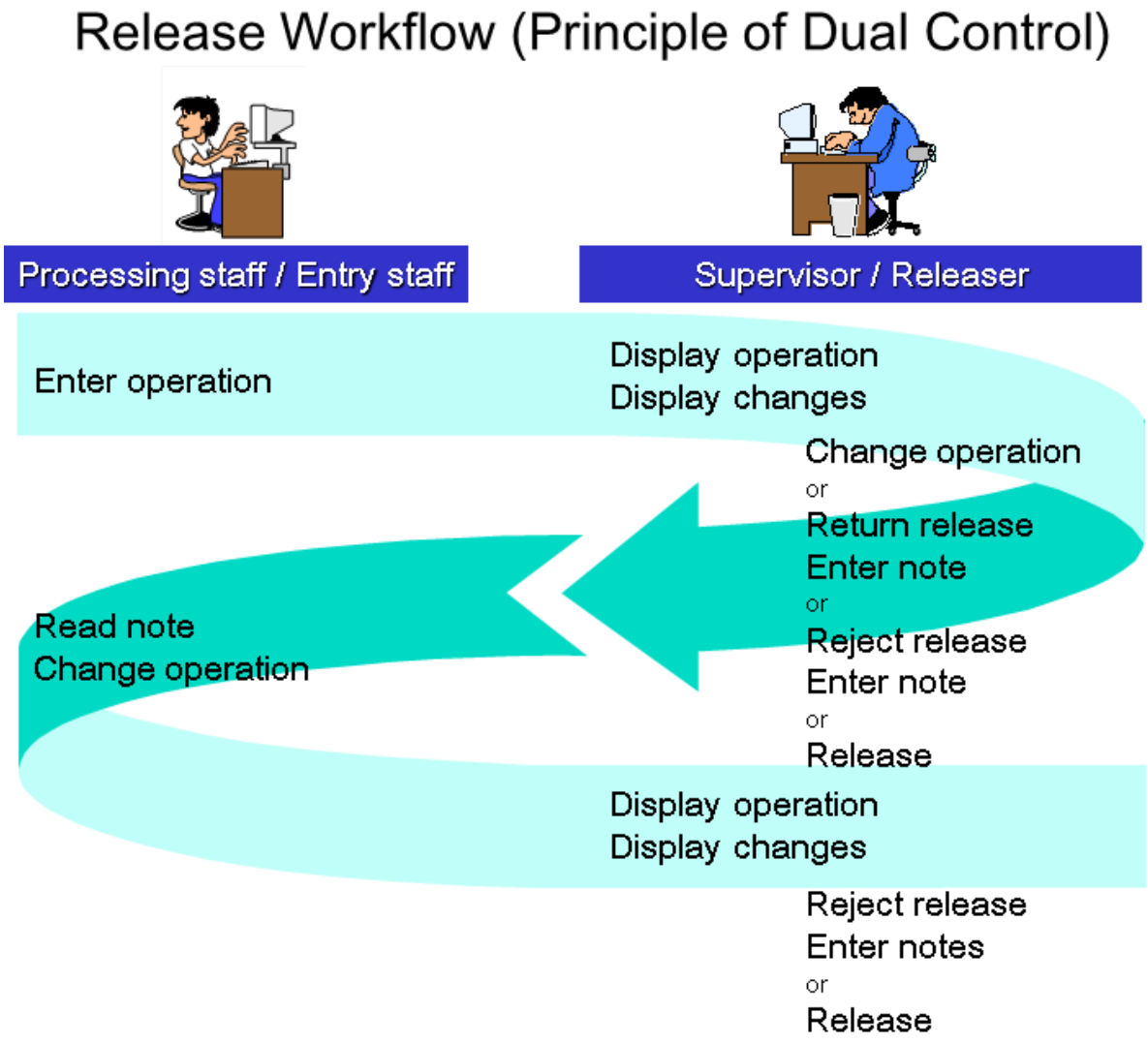
The *Framework for the Principle of Dual Control* uses the functionality provided by the SAP Business Workflow (BC-BMT-WFM). Consequently, we supply a workflow template for the release workflow that covers the functionality described. The functions described below require integration of application-specific function modules in the application. By modeling the workflow template, customers can then adapt it to suit their own requirements.

The application into which this *Framework for the Principle of Dual Control* is integrated uses the current status to decide how to handle the data that needs to be released.

You have the following three options when an ‘old’ data record in the system has been changed.

1. The new data record becomes active on entry. The system works with this data record as soon as it is entered. Example: When an address is changed you can immediately send correspondence to the new address, as the old one is definitely wrong.
2. The old data record remains active until the new data record has been released. The system works with the old status until the new data record has been released.
3. When the new data record is entered, no data record is active until the new one has been released. When a new data record is entered, the system cannot access it until it has been released.

Scope of Functions



The *Framework for the Principle of Dual Control* provides the following functions that can be implemented for a release object:

- Display data entered
  At this point, the application for displaying the data is called up.
- Change data entered
  At this point, the application for changing the data is called up. Changing a release-relevant object possibly triggers a new release process.
- Display of Change Documents
  At this point, the application for displaying the changes is called up.
- Return data entered
  If the data has not been entered correctly and the operation needs to be processed again, the releasing user can return the data to the user who entered it for it to be changed. In this case, the releaser must enter a note. The user who entered the data receives a message accordingly via the release workflow. The user who entered

the data can change the data of the release-relevant operation. This can mean that a new release workflow is then triggered.

- Reject the data entered
  If the data entered needs to be rejected, the releasing user can reject release. In this case, the releaser must enter a reason for doing so in a note.
  The application development team must check whether rejection, meaning deletion of the operation, or return to have the data changed is possible. In case of a total rejection, the settings in the application must be such that the old data status is restored.
- Release the data entered
  The system ensures that the user who entered the data is not also the one releasing it. When a user releases data, the release log is updated with his/her name, the time and the date.
  In the case of the principle of dual control, the workflow is now complete. In the case of the principle of treble and quadruple control, another releasing user must first process the operation before it can be finally released.
- Display the release history
  The *Framework for the Principle of Dual Control* records all the release steps. These are displayed in the *Business Workplace* for the workflow. The log can be displayed in the application.

The application development team decides which of these functions are meaningful from a business perspective and how they are realized.

Constraints

If the data for a release object is transferred to an application from an external system, it must include a release indicator. On the basis of this indicator, the application recognizes whether the data has already been released, or possibly still needs releasing. Once the data is transferred, release can only be made by the receiving application and the workflow, and not by an external system.

By means of user (agent) determination, the system distributes the release-relevant operations to the processing staff responsible. This means that users only receive an operation to release if they are stored as being responsible to do so in user (agent) determination. For this reason, the *Framework for the Principle of Dual Control* does not make any additional authorization checks when a release is carried out.

When the *Framework for the Principle of Dual Control* is connected, the respective development team decides how to restrict its functions provided by the *Framework for the Principle of Dual Control*.

Release by committees, such as the supervisory board, is not supported outside the system. You can use the note function to document the decision of persons with no user in the system.

8.8.1 Content of this Documentation

This documentation contains the generally applicable details on using a release object in the context of an application that uses the *Framework for the Principle of Dual Control* .

For more specific details on a release object, refer to the documentation of the corresponding application.

You use the sections under *Connecting an Application* if you wish to define another release object and connect another application to the *Framework for the Principle of Dual Control* .

8.8.2 Release Object “ReleaseProcessBCA” (General Release Object)

Definition

Business object of the *Framework for the Principle of Dual Control (CA-GTF-TS-PDC)* for defining the workflow template .

Use

You can use the general release object with its attributes, methods and events as basis for defining the workflow template used by the *Framework for the Principle of Dual Control* .

Structure

The structure of the general release object *ReleaseProcessBCA* is subdivided into key fields, attributes, methodsand events.

In addition, the workflow tasks and workflow template defined by the *Framework for the Principle of Dual Control* on this basis and the possible customer enhancements , the statuses and the release history are also displayed.

Key Field

The general release object has the following key field:

Key Field	Name	Short description
ReleaseProcessID	Release Process	Release process instance ID

Attributes

The attributes of the general release object are contained in structure BCA_STR_RTW_RELPROC:

Attribute*	Name	Short description
ObjectType	Object type	Object type
ObjectCategory	Object Category	Release object category
ObjectKey	Object Key	Release object key
CurrentStatus	Current Status	Current status of the release object
ReleaseProcedure	Release procedure	Release procedure

Attribute*	Name	Short description
Activity	Activity	Activity of the application on the release object
WorkItemText	Work Item Text	Work item textfor the inbox

Methods

The *Framework for the Principle of Dual Control* uses the following methods:

Method	Short description
ReleaseProcessBCA.Display	Display history of the release object
ReleaseProcessBCA.ExistenceCheck	Check for the existence of the release object
ReleaseProcessBCA.Release	Dialog for executing release process activities
ReleaseProcessBCA.GetReleaseSubWorkflow	Determine the release procedure workflow during the dynamic release procedure
ReleaseProcessBCA.EditApplicationObject	Edit returned release object
ReleaseProcessBCA.GetDetail	Display release object

Events

The *Framework for the Principle of Dual Control* uses the following events:

Event	Short description
ReleaseProcessBCA.Rejected	Application object returned during the release dialog
ReleaseProcessBCA.Returned	Application object returned during the release dialog
ReleaseProcessBCA.Initiated	Release process initiated for an application object
ReleaseProcessBCA.Terminated	Release process terminated
ReleaseProcessBCA.ChangedNoReleaseRequired	Application object changed but no additional release required
ReleaseProcessBCA.ChangedReleaseRequired	Application object changed and additional release required
ReleaseProcessBCA.CompletelyReleased	Application object completely released during the release process
ReleaseProcessBCA.PartiallyReleased	Application object released and additional release required

Workflow Tasks

Workflow Task	Short description
TS 50100025*	Release the work item
TS 50100026*	Edit returned release object
TS 50100027	Notification of final release
TS 50100028	Notification of rejection
TS 50100029	Determine sub-workflow ID
TS 50100066*	Add attachment after rejection
TS 50100075*	Add attachment after return

* Possible processing staff must be assigned to these tasks or the tasks must be identified as general tasks (refer to Preparation and Customizing [page 1856]) .

Workflow Template

A workflow template is available for designing the release process. Sub-workflowsare available for the special characteristic values of the various release procedures.

Workflow Template	Short description
WS 50100024	Release workflow
WS 50100021	Release procedure workflow (principle of dual control)
WS 50100022	Release procedure workflow (principle of treble control)
WS 50100023	Release procedure workflow (principle of quadruple control)

The applications adopt these workflow templates. Customers can use these workflow templates or copy them and adapt them to suit their requirements.

Customer Enhancements

The following Business Transaction Events(BTEs) are available to you for the general release object:

BTEProcess	Short description	Call-Up
0BANK002	Supply user selection and period for the release	You call this up before triggering the workflow. You can use it to define: • a processing period (only once for all release steps) • a preferred processor (per release step) You can supply preferred processors for all release steps at once.
0BANK004	User-specific release control	You call this up after evaluation of the release Customizing settings (*Assign Release Object to Release Procedures).* You can use this to control release, depending on who initiates an operation.

Status

Every release object has one of the following statuses:

Status	Name	Short description
00	Entered	In the release process
01	Release 1	Released, 1 st step
02	Release 2	Released, 2nd step
B	Return	Returned
C	Changed	Changed
F	Finally released	Finally released
R	Rejected	Rejected

You can freely choose the names per client (in the Customizing settings for the *Framework for the Principle of Dual Control*) by choosing *Cross-Application Components* > *General Application Functions* > *Edit Release Status Description.*

Release History

The *Framework for the Principle Dual Control* function records creation of and every change to the release object in a release history.

Contents	Explanation
Date, time	Date and time of the logged processing step
Release procedure	Release procedure
Initial status	Status of the release object before the logged processing step
Target status	Status of the release object after the logged processing step
First name, last name, department, telephone	Available data on the user who carried out the logged processing step
Activity	Activity of the application of the logged processing step

Integration

You make the settings for the release object in the application Customizing settings in the following IMG activities (refer also to Preparation and Customizing [page 1856]):

- *Assign Release Object to Release Procedure*
  Define whether and when the release object is to be forwarded to the release process and which release procedure is to be chosen on the grounds of which criteria.
- *Assign Standard Role to Release Steps*
  Assign a standard role for the release object to every release step of the release procedure used. Using the responsibilities involved in the role you define which processors are to receive a work item via the release workflow for which release step.
- *Assign Release WF and Release Procedure WF to Release Procedures.*
  Assign a release workflow and a release procedure workflow (sub-workflow) to every release procedure of a release object.

You may need to make other application-specific settings for a release object.

8.8.3 Preparation and Customizing

Purpose

Before you can use the functions described in an application that uses the Framework for the Principle of Dual Control, you must carry out these steps.

Prerequisites

Before you make the specific settings for an application for the *Framework for the Principle of Dual Control*, you must have made the standard Customizing settings for the workflow in the Customizing for the SAP Business Workflow (BC-BMT-WFM). This involves in particular the settings you make by choosing

Basic Settings (Organization Management) → *Maintain Prefix Numbers*.

Maintain Standard Settings for SAP Business Workflow

Basic Settings (System, SAP Business Workflow)

Process Flow

Maintain Event Type Linkage

From the *SAP Easy Access* screen, choose *Tools* → *Business Workflow* → *Development* → *Definition Tools* → *Events* → *Event Linkages* → *Type Linkages*.

This takes you to the *Change View "Event Type Linkages": Overview* screen.

Choose object type *BUSISB001* and then *Goto* → *Details*.

This takes you to the *Change View "Event Type Linkages": Details* screen.

All the information about event type linkage and the *Initiated* event for the general release object (BUSISB001) are displayed.

Check the accuracy of the settings displayed, particularly the assigned function modules (SWW_WI_CREATE_VIA_EVENT as receiver function module and BCA_OBJ_RTW_RECEIVER_TYPE as receiver type function module).

Set the *Linkage Activated* indicator and save the settings.

Go back.

You can also activate the event type linkage in the workflow template. In this case, however, no other settings are displayed.

Assign Possible Processors to the Standard Task

You must either assign possible processors to the standard tasks 50100025, 50100026, 50100066, 50100075 or you must classify these tasks as general tasks. Other standard tasks can also be subject to this condition for an application.

Proceed as follows:

From the *SAP Easy Access* screen, choose *Tools* → *Business Workflow* → *Development* → *Definition Tools* → *Taks/ Task Groups* → *Display*.

Select *Standard Task* as the task category.

Choose the standard task you wish to assign to processors.

Choose *Display*.

This takes you to the *Standard Task: Display* screen.

Choose *Additional DataAgent AssignmentMaintain*.

This takes you to the *Standard Task: Maintain Agent Assignment* screen.

You can classify this standard task as general task, making it accessible for all processors, or you can assign this task to special users.

If you wish to assign this task to special users, select the task and choose *Agent AssignmentCreate*.

Choose the agent type in the selection box. ...

Assign the users or organizational units you want to process this task to the standard task.

You can repeat this step several times.

If you wish to classify this task as general task, select the task and choose *EditProperties*.

In the selection box choose *General Task*, transfer these values and go back.

By making this setting, all users in the system are possible processors of this task. A task cannot be performed unless it has possible processors. These settings are also the basis for agent determination during execution of the workflow.

Customizing for the General Release Object

In the Customizing for the *Framework for the Principle of Dual Control* (*Cross-Application ComponentsGeneral Application Functions*) under *Release Status Description* you can adapt the descriptions of the release statuses to suit your requirements for the general release object.

Customizing for a Release Object

You make the settings for the release objects in the Customizing for the application in the following IMG activities:

Assign Release Procedure to Release Object

Define whether and when the release object is to be forwarded to the release process and which release procedure is to be chosen on the grounds of which criteria.

In the standard system delivered, release is not (never) set for the release object. You can choose the following variants:

Always

All operations relating to this release object are subjected to the release procedure set, regardless of other selection criteria and the possible release attributes.

Conditional

Only operations relating to this release object that fulfill a release reason are subjected to the release procedure set. You define release reasons by assigning conditions to the possible release attributes.

Statistical

Only the millionth part of the operations defined by you is subjected to the release procedure, regardless of other selection criteria and the possible release attributes.

Assign Standard Role to Release Steps

Assign a standard role for the release object to every release step of the release procedure used. By doing this you define which processors are to receive a work item via the release workflow for which release step.

Assign Release WF and Release Procedure WF to Release Procedures

Assign a release workflow and a release procedure workflow (sub-workflow) to every release procedure of a release object. You can use the workflow templates of the general release objects supplied for the application, or your own ones by copying those supplied and adapting them to suit your requirements.

You may need to make other application-specific settings for a release object. For more information, see the documentation for the application.

Result

You have now set up the *SAP Business Workflow*, and the *Framework for the Principle of Dual Control* is functional.

8.8.4 Processing Release Objects

Purpose

This process describes the general process flow for processing a release object of an application that uses the *Framework for the Principle of Dual Control (CA-GTF-TS-PDC)*.

Prerequisites

Before you can process release objects in an application using the *Framework for the Principle of Dual Control*, you must have made the settings for the *SAP Business Workflow*, the general release object of the *Framework for the Principle of Dual Control*, and the special release object of the application described in Preparation and Customizing [page 1856].

Process Flow

1. A member of the processing staff enters data.
2. The system (*Framework for the Principle of Dual Control*) checks the release-relevance according to the Customizing settings (Customizing *Assign Release Procedure to Release Object)*.
   If the operation is relevant for release,
   1. a. the *Framework for the Principle of Dual Control* writes the history. The release object has the status 00 (in release process).
   2. b. the *Framework for the Principle of Dual Control* triggers the event.

3. The system (SAP Business Workflow (BC_BMT_WFM)) determines the event receiver, meaning the release workflow set in the Customizing settings (Customizing *Assign Release WF and Release Procedure WF to Release Procedure*).
4. The system starts the workflow.
5. The system (release workflow) triggers the release procedure workflow determined (Customizing *Assign Release WF and Release Procedure WF to Release Procedure)*.
6. The system (release procedure workflow) generates a work item.
7. The system (release procedure workflow) searches for a releasing user (user determination according to the Customizing for *Assign Release Procedure to Release Object)*.
8. The system distributes the work items. Then the work item is available in the Business Workplace of the user determined, ready for editing.
9. The processor (user responsible/releaser) edits the work item received in the *Business Workplace*.
   Depending on the release object and status, multiple functions are available:
   - Display data entered
     The data to be released is displayed to the processor.
   - Change data entered
     The processor has the option of changing the data to be released. If the processor changes the data to be released, the work item is deleted and, if applicable, a new release workflow is initiated. The *Framework for the Principle of Dual Control* updates the history. The release object has the status C (changed).
   - Display of Change Documents
     At this point, the application for displaying the changes is called up.
   - Return data entered
     If the data has not been entered correctly and the operation needs to be processed again, the releasing user can return the data for it to be changed. In this case, the releaser must enter a note. The *Framework for the Principle of Dual Control* updates the history. The release object has the status B (returned). The user who entered the data receives a message accordingly via the release workflow. The user who entered the data can change the data of the release-relevant operation. This can mean that a new release workflow is then triggered.
   - Reject the data entered
     If the data entered needs to be rejected, the releasing user can reject release. In this case, the releaser must enter a reason for doing so in a note. The application determines how to proceed further with the data, such as delete, set delete indicator. The *Framework for the Principle of Dual Control* updates the history. The release object has the status R (rejected).
   - Release the data entered
     In the case of the principle of dual control, the workflow is now complete. In the case of the principle of treble and quadruple control, another releasing user must first process the operation before it can be finally released. Depending on the release procedure, the release object has the status 01 (release, level 1), 02 (release, level 2) or F (finally released).
     The *Framework for the Principle of Dual Control* updates the history.

8.8.5 Users Involved

From the SAP Business Workflow (BC-BMT-WFM) perspective, the processing staff/data entry staff and supervisors/ releasers are workflow agents, who function as end users in the context of the productive workflow. They start workflows and process work items. Their work area is the Business Workplace.

However, for support purposes they also require the other roles in the workflow.

SAP does not supply roles for the *Framework for the Principle of Dual Control*. You have to create the roles used in the documentation.

8.8.6 Connecting an Application

Purpose

With this process, the *Framework for the Principle of Dual Control* (CA-GTF-TS-PDC) offers you the option of releasing objects or operations of your application using a principle of dual. treble or quadruple control.

Use this process if you wish to define an additional release object or if you wish to connect a new application to the *Framework for the Principle of Dual Control*.

Process Flow

If you wish to connect an application to the *Framework for the Principle of Dual Control*, you need to carry out the following steps:

1. Define the name for the release object.
   The maximum length for the name is ten characters. This name is used as part of the following names:
   - Name of the transfer structure
     *Prefix*_STR_*release object*_RELEASE
   - Names of the function modules
     *Prefix*_API_*release object*_*method*_CB
     The maximum length for these names is thirty characters.
   - Names of the parameter transaction and view variants
     *Prefix*_*release object*_REL01, VV_TBCA_REL_RULE_*release object*, VV_TBCA_RTW_LINK_*release object*
     The prefix should not be more than four characters long. It identifies your application (BCA has already been used).
2. Create the transfer structure. Naming convention: *Prefix*_STR_*release object*_RELEASE
   This structure contains all the fields in the application that can be selected as release attributes in the Customizing settings. The *Framework for the Principle of Dual Control* uses the release attributes to decide whether an object or operation is relevant for release, and which release procedure is involved. The release attributes also serve as criteria for user determination.
3. Define the release object in the context of the *Framework for the Principle of Dual Control*:
   In view V_TBCA_OBJ_CAT, make an entry for the release object with the following values:

Field Name	Entry Required
Object Type	*Release object*
Long Text	Description for the release object (anything you want)

Field Name	Entry Required
Structure	Name of the previously defined transfer structure

4. Provide the function modules needed for the release object (see Functionality and Interfaces for the API Function Modules [page 1864]).
5. In view V_TBCA_REL_FM, make the entries for the API function modules for the release object for each activity with the following values:

Field Name	Entry/Action
Object Type	*Release object*
Activity	Activity
Function Module	Name of the function module that executes the activity
Function supported	Set the indicator if the activity is supported by the release object.

6. Create a parameter transaction for IMG activity *Assign Release Procedure to Release Object* with the following values:

Field Name	Entry/Action
Transaction Code	*Prefix_release object*_REL01
Short text	Assign Release Procedure to Release Object
Start object	Transaction with parameters (parameter transaction)
Transaction text	Keep proposed short text
Default values for transaction	BCA_CUS_REL_PROC
Skip initial screen	Select
Default values: Name of screen field	P_OBJCAT
Default values: Value	*Release object*

Using this parameter transaction, in the Customizing for the application you define if and when the release object is to be forwarded to the release process and which release procedure is to be chosen on the grounds of which criteria.

7. For view V_TBCA_REL_RULE, create a view variant for IMG activity *Assign Release Procedure to Release Object* with the following values:

Field Name	Entry/Action
Table/view	V_TBCA_REL_RULE
Maintenance view variant	VV_TBCA_REL_RULE_*release object* Due to the length of the name, you can only use part of the release object name
Short Description	Assign Rules to Release Steps
Tables/ join conditions: Tables	V_TBCA_REL_RULE
View fields: Table fields	Select all
Selection Conditions Table fields	OBJECT_CAT with selection condition EQ '*release object*'

See also Creating Maintenance View Variants.
Using this view variant, in the Customizing for the application you define which processors are to receive a work item via the release workflow for which release step.

8. For view V_TBCA_RTW_LINK create a view variant for IMG activity *Assign Release WF and Release Procedure WF to Release Procedures*.

Field Name	Entry/Action
Table/view	V_TBCA_REL_LINK
Maintenance view variant	VV_TBCA_RTW_ *release object* Due to the length of the name, you can only use part of the release object name
Short Description	**Assign workflow template to release procedure of assign release wf and release procedure wf to release procedures**
Tables/ join conditions: Tables	V_TBCA_REL_LINK
View fields: Table fields	Select all
Selection Conditions Table fields	OBJECT_CAT with selection condition EQ '*release object*'

See also Creating Maintenance View Variants.
Using this view variant, in the sample Customizing and/or in the Customizing for the application you assign a release workflow and a release procedure workflow (subworkflow) to every release procedure of a release object.

9. Insert the three IMG activities at an appropriate place in the Implementation Guide (IMG).

General texts PDC_PROCEDURE, PDC_RULE and PDC_WORKFLOW are available to you for documenting these IMG activities.

10. In the sample Customizing settings, for each release object assign a release workflow and a release procedure workflow (subworkflow) to every useable release procedure.
    To do this you can either use the workflow template provided with the *Framework for the Principle of Dual Control* (see Release Object ReleaseProcessBCA (General Release Object) [page 1852] or your own template, by copying the one provided and modifying it to suit your requirements.
11. In the application source text, insert the following calls of the function modules for the *Framework for the Principle of Dual Control* at the appropriate places:

Function Module	Insert
BCA_API_REL_CHECK_ATTRIB checks if the operation/object is relevant for release	Before the data for release is saved.
BCA_API_REL_CALL starts the release procedure if the operation/object is relevant for release	After the data for release is saved.

    This means that for operations involving release, before you save the data you must call the *Framework for the Principle of Dual Control* (BCA_API_REL_CHECK_ATTRIB) to check whether this operation or object is relevant for release. Then you save the data on the database and call the *Framework for the Principle of Dual Control* a second time (BCA_API_REL_CALL). With this call, the release procedure (workflow) is started, if applicable.
    You must always call the *Framework for the Principle of Dual Control* in this sequence and always using the two function modules. To generate a workflow event, after you call function module BCA_API_REL_CALL you must issue a COMMIT WORK for the application.
    You can find a description of the function modules and the corresponding parameters in the system.

12. In the application in the *Extras* menu point, insert the *Display Release History* function. Use function module BCA_API_REL_HISTORY_DISPLAY to do this. Use the ICON_HISTORY icon if you also want to offer this function via a pushbutton.
    You can find a description of the function module and the corresponding parameters in the system.

Result

You have created the option of releasing the processing of an object in the application using a principle of multiple control.

8.8.6.1 Functionality and Interfaces of the API Function Modules

Following is a description of the functionality and interfaces of the function modules that you must provide for connecting the application.

The following naming convention applies:

*Prefix*_API_*release object*_*method*_CB

The prefix should not be more than four characters long. It identifies your application (BCA has already been used).

You can provide function modules for the following methods:

DISPLAY Display

CHANGE Change

RELEASE Release

RET Return

RETURN Return

CHDOC Change Documents

LOCK Lock

REJECT Reject

GET_TXT Transfer work item text

CHECK_CUST Check Release reasons

MB Suppress Pushbuttons

FUNCTION<Präfix>_API_<Objekttyp>_DISPLAY_CB.

Display function for the object being released

PARAMETER:

IMPORT

I_REL_OBJECT_KEY TYPEBCA_DTE_OBJECT_KEY

EXCEPTIONS:

NO_SUCCESS

Note:

This module must be identified as remote-enabled, as it is called with the *Starting New Task* command.

FUNCTION <prefix>_API_<object type>_CHANGE_CB.

Change function for the object being released

PARAMETER:

IMPORT

I_REL_OBJECT_KEY TYPEBCA_DTE_OBJECT_KEY

EXPORT

E_FLAG_MODIFIED TYPEBCA_REL_DTE_BOOLEAN

EXCEPTIONS:

NO_SUCCESS

FUNCTION <prefix>_API_<object type>_RELEASE_CB.

Function module for calling after the final, meaning the last release.

PARAMETER:

IMPORT

I_REL_OBJECT_KEY TYPEBCA_DTE_OBJECT_KEY

I_FLG_INUPDATETASK TYPEBCA_REL_DTE_BOOLEAN DEFAULT CON_TRUE

EXCEPTIONS:

NO_SUCCESS

Note:

The callback module may not contain a COMMIT WORK or ROLLBACK WORK because the *Framework for the Principle Control* is the leading application at this point and issues a COMMIT WORK itself.

FUNCTION <prefix>_API_<object type>_REJECT_CB.

Function module for calling release function *Reject*

PARAMETER:

IMPORT

I_REL_OBJECT_KEY TYPEBCA_DTE_OBJECT_KEY

I_FLG_INUPDATETASK TYPEBCA_REL_DTE_BOOLEAN DEFAULT CON_TRUE

EXCEPTIONS:

NO_SUCCESS

Note:

The callback module may not contain a COMMIT WORK or ROLLBACK WORK because the *Framework for the Principle Control* is the leading application at this point and issues a COMMIT WORK itself.

FUNCTION <prefix>_API_REL_<object type>_CHDOC_CB.

Display function for changes to data/change documents

PARAMETER:

IMPORT

I_REL_OBJECT_KEY TYPEBCA_DTE_OBJECT_KEY

I_CHANGED_AT TYPEBCA_REL_DTE_CHNGAT

EXCEPTIONS:

NO_SUCCESS

Note:

In addition to the object key (I_REL_OBJECT_KEY), for this call a time stamp (I_CHANGED_AT) is displayed. This specifies the time at which the *Framework for the Principle of Dual Control* was called the first time, so that all changes made since can be displayed.

FUNCTION <prefix>_API_<object type>_LOCK_CB.

Function module for setting or deleting SAP locks for this object.

PARAMETER:

IMPORT

I_REL_OBJECT_KEY TYPEBCA_DTE_OBJECT_KEY

I_FLG_LOCKSET TYPEBCA_REL_DTE_BOOLEAN

EXCEPTIONS:

OBJECT_LOCKED

OBJECT_ERROR

Note:

Parameter I_FLG_LOCKSET specifies whether the lock is to be set ('X') or deleted (' ').

FUNCTION <prefix>_API_<object type>_GET_TXT_CB.

Function module for supplying the work item with a text from the application

This module is used for supplying the *SAP Business Workflow* with a text that is displayed as a work item text in the *Business Workplace*. The text is intended to provide exact details about which object and which application is involved. This can contain an external number, for example.

PARAMETER:

IMPORT

I_REL_OBJECT_KEY TYPEBCA_DTE_OBJECT_KEY

EXPORT

E_STR_ITEMTXT TYPEBCA_STR_REL_ITEMTXT_DATA

Note:

In this module it is not permitted to access data that is possibly written to the database only after the *Framework for the Principle of Dual Control* is called, as this would result in update terminations.

FUNCTION <prefix>_API_<object type>_RET_CB.

Function module for calling release function *Return*: Information that the work item was returned.

PARAMETER:

IMPORT

I_REL_OBJECT_KEY TYPEBCA_DTE_OBJECT_KEY

I_FLG_INUPDATETASK TYPEBCA_REL_DTE_BOOLEAN DEFAULT CON_TRUE

EXCEPTIONS:

NO_SUCCESS

Note:

The callback module may not contain a COMMIT WORK or ROLLBACK WORK because the *Framework for the Principle Control* is the leading application at this point and issues the COMMIT WORK command itself.

FUNCTION <prefix>_API_<object type>_RETURN_CB.

Function module for calling release function *Return*: Processing the returned work item

PARAMETER:

IMPORT

I_REL_OBJECT_KEY TYPE BCA_DTE_OBJECT_KEY

EXPORT

E_FLAG_MODIFIED TYPEBCA_REL_DTE_BOOLEAN

EXCEPTIONS:

NO_SUCCESS

Note:

For Release 4.6C and Release 6.20, this callback function module supports both parameter E_FLG_MODIFIED and parameter E_ORDERMODIFIED. From Release 6.40, only parameter E_FLG_MODIFIED is supported.

FUNCTION <prefix>_API_<object type>_BUTTON_CB.

Function module with which the application can suppress pushbuttons for individual functions of the *Framework for the Principle of Dual Control* in the *Business Workplace*.

Using this module you can modify the FUNCTIONFLAG indicator in the transferred table (C_TAH_RELFM) for individual callback function modules for the object just edited. If this indicator is set, the corresponding pushbutton appears on the interface, otherwise it is suppressed.

PARAMETER:

IMPORT

I_REL_OBJECT_KEY TYPEBCA_DTE_OBJECT_KEY

CHANGING

C_TAH_REL_FM TYPEBCA_TAH_REL_FM

FUNCTION <prefix>_API_<object type>_CHECK_CUST.

Function module for checking the settings in IMG activity *Assign Release Procedure to Release Object* after entry and before they are saved.

This function module is called when you enter a new release reason, when you change or delete an existing one, and also when you save. You have the option of making consistency checks and of integrating a dialog with any messages that occur, if applicable.

To do this, you transfer the following data on the interface of the callback function module:

- Before saving: A table with the current Customizing settings for this release object
- When inserting a new entry: A table with all the entries for this release object in the current version. This can contain changes made within this LUW that have been classified as consistent. A table with the new entries is also transferred.
- When changing an entry: A table with all the entries for this release object in the current version. This can contain changes made within this LUW that have been classified as consistent. A table with changed entries is also transferred.
- When you delete an entry, the table with the entries for changing is empty.

PARAMETER:

IMPORT

I_TAB_EXPR TYPEBCA_TAB_RELPROC_EXP

I_TAB_EXPR_NEWTYPEBCA_TAB_RELPROC_EXP

I_TAB_EXPR_CHGTYPEBCA_TAB_RELPROC_EXP

EXCEPTIONS:

INCONSISTENT

Note:

The callback module may not contain a COMMIT WORK or ROLLBACK WORK because the *Framework for the Principle Control* is the leading application at this point and issues the COMMIT WORK command itself.

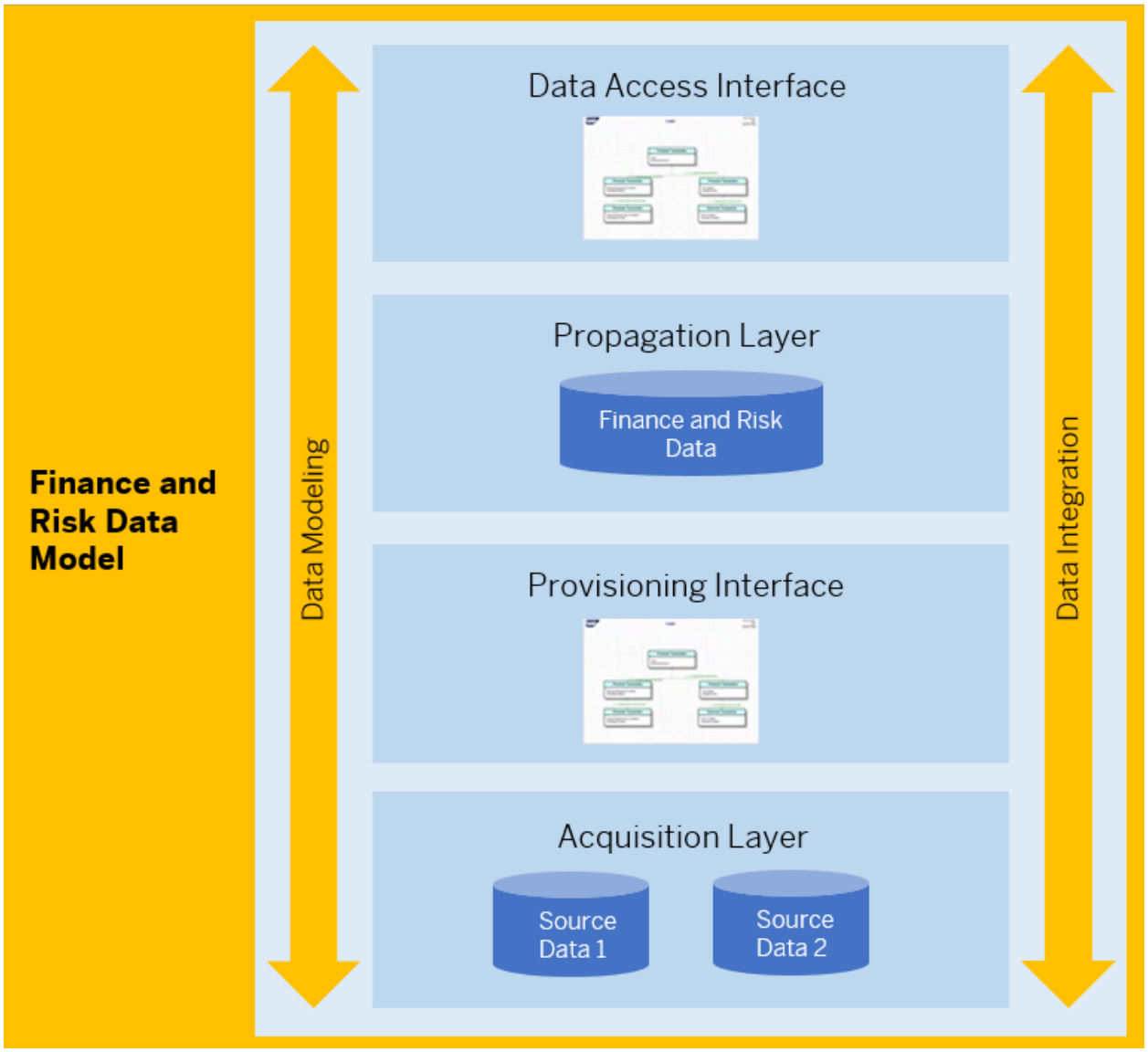
9 Finance and Risk Data Platform

The *Finance and Risk Data Platform (FRDP)* comprises several software components that together form a database that enables integrated reporting and supports the requirements of BCBS239 (Basel Committee on Banking Supervision: "Principles for effective risk data aggregation and risk reporting"). Moreover, the capital requirements for insurance in accordance with the Solvency II Directive (EU Directive 2009/138/EC) can be mapped in the insurance data model. Some existing basic entities that are required for IFRS 4 or IFRS 17 are defined for accounting for insurance contracts For more information, see .

Information from SAP Bank Analyzer can be accessed faster with the use of the Finance and Risk Data Platform (FRDP) and is therefore also available quickly for reporting purposes. Technically, this is supported through the use of an SAP HANA database, through the provision of a virtual data model based on the physical data model, and through the use of SQL as the query technique.

The Finance and Risk Data Platform is divided into the following software layers:

- **Acquisition Layer**: Receives the data extracted from a bank's source system and transforms it into the corresponding data format for the SAP system.
- **Provisioning Interface**: Provides consistent data for the Propagation Layer for further processing.
- **Propagation Layer**: Is the central data store ("single source of truth") for all FRDP data in the SAP system (the Source Data Layer (FS-BA-SD) [page 100] and Results Data Layer (FS-BA-RD) [page 1364] in SAP Bank Analyzer are currently used for this purpose).
- **Data Access Interface**: Allows you to access data in the propagation layer. You can also combine the data in different views.
- **Reporting Layer**: Allows you to view the data using BI tools, such as SAP Lumira.



Properties

A corresponding virtual data model is generated in the **Data Access Interface** based on the physical data model. Regardless of the database system being used, the technology of ABAP Core Data Services (CDS) is available here. If you are using an SAP HANA database, it is also possible to generate SAP HANA views to further optimize performance.

For more information, see Data Access Interface [page 1873].

The system generates views for the following frameworks and primary objects:

- Analytical business partner data

- Generic market data
- Master data
- Flow data
- Results data

For an overview of the different view types, see View Types in Data Access Interface [page 1876].

> ⚠ Caution
>
> Note that you must use only the basic views described in View Types in Data Access Interface [page 1876].
>
> It is the case for all view types that the system generates the views based on the Customizing settings in SAP Bank Analyzer. You have to regenerate the views as soon as these settings change in order to reflect the current system settings. A stable interface can be guaranteed only if you use the basic views. A stable interface means that the existing fields remain constant in future releases.

For more information about specific transport options for the views in the system landscape, see Transport of Views [page 1889].

Integration

SAP PowerDesigner has an *FRDP PowerDesigner Extension* that can be used to extract the data model and enables you to enhance the data model to meet your own needs.

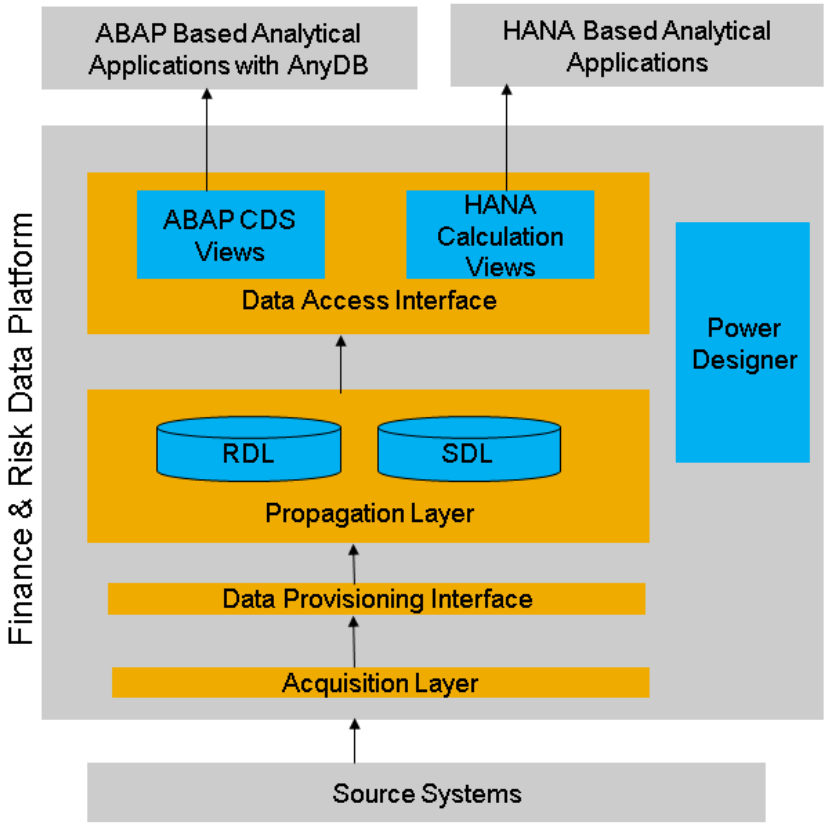
For more information, see .

You can generate your own views based on the data model in SAP PowerDesigner. You can also use SAP HANA Studio (to generate calculation views) or Eclipse (to generate ABAP CDS views). For more information about ABAP Core Data Services (CDS), see SAP Help Portal under *Technology Platform* > *SAP NetWeaver*`<Release >` > *CDS View Building*.

9.1 Data Access Interface

This layer in the Finance and Risk Data Platform architecture provides a virtual data model that allows to access data stored in Propagation Layer. This means you can build your reporting scenarios directly on SDL and RDL data.

The figure below gives an overview of the FRDP architecture and illustrates how analytical applications consume data from the Data Access Interface:



Features

The data in SDL and RDL is exposed for consumption to other analytical processes and applications using virtual data models. The data models are built using different views on the Data Access Interface.

You can generate views for the following primary objects:

- Reference Data
  - Analytical Business Partner Data
    - Analytical Business Partner
    - Analytical Business Partner Relationship
- Source Data
  - Master Data
    - Financial Transaction
    - Financial Instrument
    - Account
    - Country

    - Physical Asset
    - Positions Master Data
  - Flow Data
    - Business Transactions
    - Positions
  - Generic Market Data
- Result Data
  - <Result Data types>

For more information on views, see View Types in Data Access Interface [page 1876].

Activities

Depending on the database of the application, the data models can be generated in two ways:

- **SAP HANA calculation views**: These views support SAP HANA based analytical applications. To generate these views, go to Customizing under *Financial Services* > *Bank Analyzer* > *Data Access Interface* > *Generate SAP HANA Views for Data Model*.
- **ABAP Core Data Services (CDS) views**: These views support ABAP based analytical applications with anyDB. To generate ABAP CDS views, go to Customizing under *Financial Services* > *Bank Analyzer* > *Data Access Interface* > *Generate ABAP CDS Views for Data Model*.

The system writes to a log after generation. You can display the logs from *SAP Easy Access* under *Financial Services* > *Bank Analyzer* > *Infrastructure* > *Extraction and Reporting Services* > *Data Access Interface* > *Display Generation Logs*.

You can also schedule background jobs to generate SAP HANA views and ABAP CDS views. To schedule jobs to generate views, go to Customizing under *Financial Services* > *Bank Analyzer* > *Data Access Interface* > *Generate Views as Background Job*.

Result

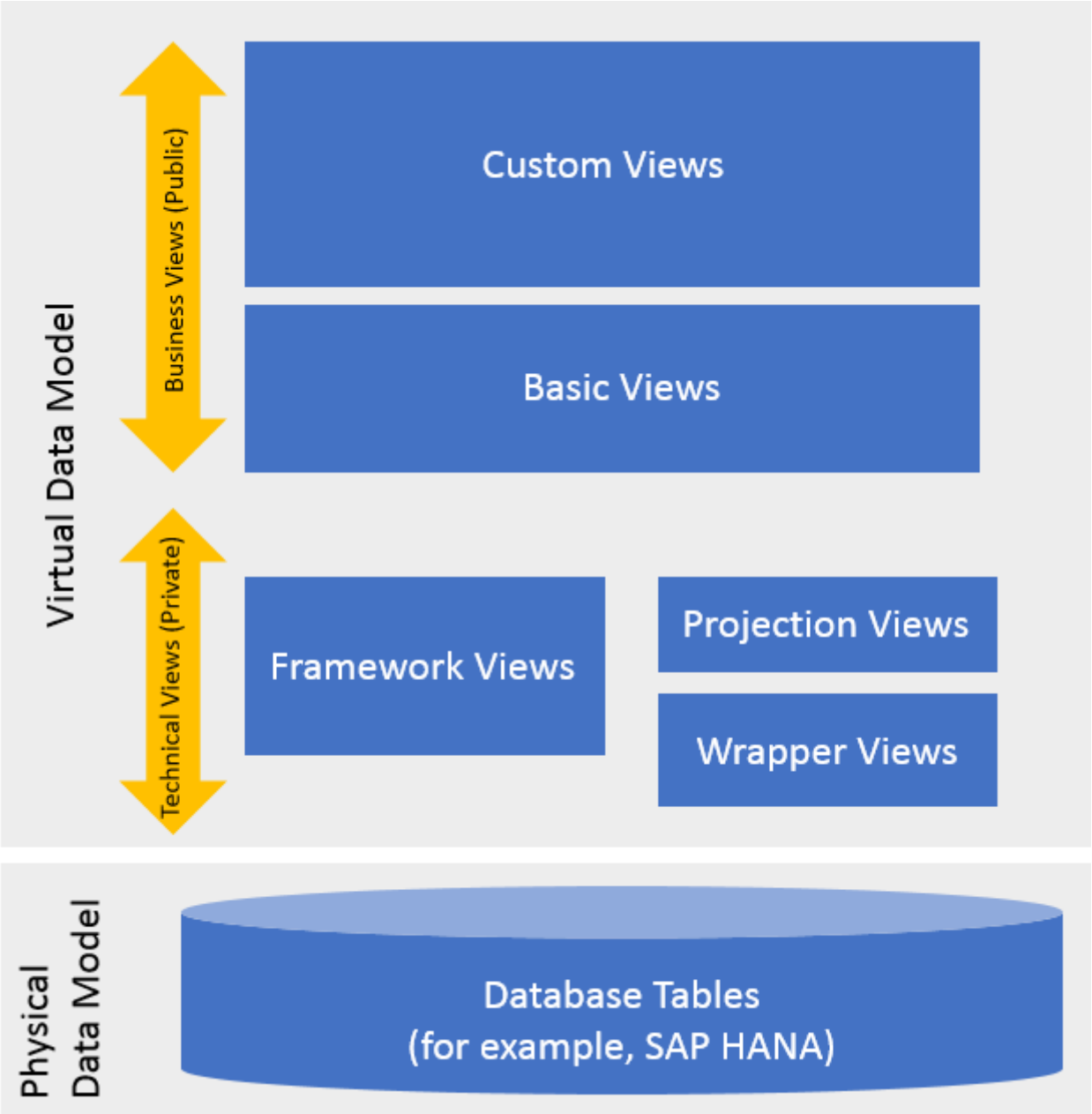
SAP HANA calculation views are generated in SAP HANA studio in the package `sap.fs.frdp.<SubPackageName>`.

> **Note**
>
> You can get the package details in SAP HANA studio from the view descriptions.

ABAP CDS views are generated in the local package `$TMP`.

9.1.1 View Types in Data Access Interface

Views with the following view types are generated in the Data Access Interface:



View Category	Description	Name of Generated Views	Packages for Generation of Calculation Views in SAP HANA
Table wrapper views (WV)	The view selects directly from a table; it is not possible to add additional fields.	Name 1: AB_X Name 2: Z Resulting Name: AB_X__Z Name 1: AB Name 2: X_Z Resulting Name: AB___X_Z The name of the first view is filled to its maximum length with underscore characters, then the name of the second view is added. WV_<framework>_<table name>	The table wrapper views are generated in the "Wrapper" folder, which is divided into subfolders according to the framework: "Business Partner", "Financial Transaction", "Generic Market Data", "Master Data Framework", "Results Data Layer", "Market Data", "Position Classes".
Projected views	The view selects the information for a specific Bank Analyzer template from the table wrapper views.	DV_Framework_TemplateName_TemplateVersion_TableName	The projected views are generated in their own folder, in the same way as the framework.
Framework views (FV)	The view implements implicit logic from the SAP Bank Analyzer framework (such as the versioning in the master data framework) and is created based on table wrapper views.		The views are generated in the same folder as the projected views.
Basic views [page 1878]	These views represent detailed entities (such as financial transaction, results data), are created based on a framework view and projected views, and do not contain any client information.		The basic views are stored in a `pub` folder as a subfolder of the projected views.
Custom views	These views are created by customers based on the SAP data model.		

⚠ Caution

Note that the system generates the views based on the Customizing settings in SAP Bank Analyzer. You have to regenerate the views as soon as these settings change in order to reflect the current system settings.

More Information

Basic Views [page 1878]

Package Hierarchy in SAP HANA Studio [page 1885]

9.1.2 Basic Views

Basic Views

In Customizing you can generate two types of basic views:

- SAP HANA views: *Financial Services* > *Bank Analyzer* > *Data Access Interface* > *Generate SAP HANA CDS Views for Data Model*
- ABAP CDS views: *Financial Services* > *Bank Analyzer* > *Data Access Interface* > *Generate ABAP CDS Views for Data Model*

The generated views are stored in a local package belonging to the user.

Basic Views for Business Partners

Under *Reference Data* > *Analytical BP Data* you can generate a total of 26 basic views with the entry of a business date. In doing so, the system includes only business partner data that had the status “Active” on the specified date.

The generated basic views reside in the package `sap.fs.frdp.<Mandant>.<User>.BP`.

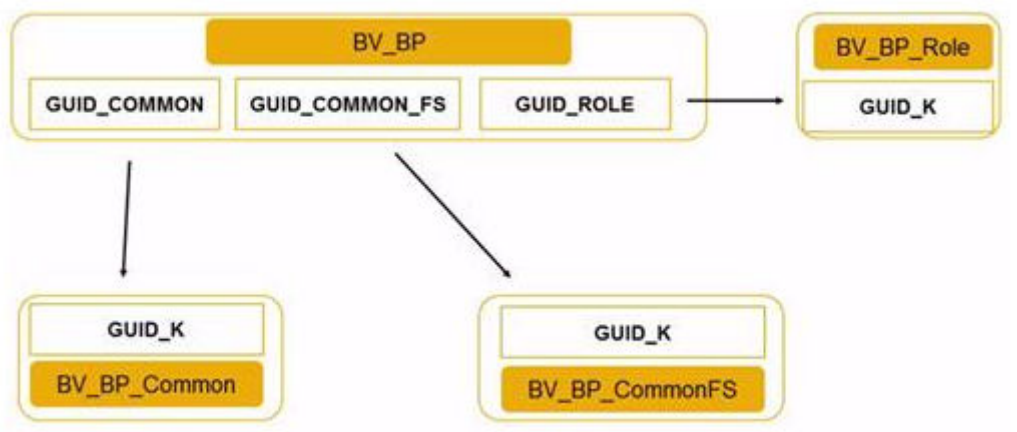
Number	Name of SAP HANA View	Name of ABAP CDS View	Description
1	BV_BP	/1BC/DA_<Client>_ABP	Basic view for business partner
2	BV_BP_AnalyticalBusinessPartner	/1BC/DA_ABP_<Client>_ABUSINESSPRTNR	Basic view for analytical business partner
3	BV_BP_Common	/1BC/DA_ABP_<Client>_COMMON	Basic view for general data
4	BV_BP_CommonFS	/1BC/DA_ABP_<Client>_COMMONFS	Basic view for general FS data
5	BV_BP_AddressInformation	/1BC/DA_ABP_<Client>_ADDRINFORM	Basic view for address information

Number	Name of SAP HANA View	Name of ABAP CDS View	Description
6	BV_BP_AddressUsage	/1BC/DA_ABP_<Client>_ADDRESSUSAGE	Basic view for address usage
7	BV_BP_PostalAddress	/1BC/DA_ABP_<Client>_POSTALADDRESS	Basic view for postal address
8	BV_BP_Telephone	/1BC/DA_ABP_<Client>_TELEPHONE	Basic view for telephone number
9	BV_BP_Facsimile	/1BC/DA_ABP_<Client>_FACSIMILE	Basic view for fax
10	BV_BP_Email	/1BC/DA_ABP_<Client>_EMAIL	Basic view for e-mail
11	BV_BP_Web	/1BC/DA_ABP_<Client>_WEB	Basic view for Web address
12	BV_BP_Role	/1BC/DA_ABP_<Client>_ROLE	Basic view for role
13	BV_BP_BankDetails	/1BC/DA_ABP_<Client>_BANKDETAILS	Basic view for bank data
14	BV_BP_IndustrySector	/1BC/DA_ABP_<Client>_INDUSTRYSECTOR	Basic view for industry sector
15	BV_BP_Identification	/1BC/DA_ABP_<Client>_IDENTIFICATION	Basic view for identification
16	BV_BP_FinancialReporting	/1BC/DA_ABP_<Client>_FINREPO	Basic view for financial reporting
17	BV_BP_CreditRating	/1BC/DA_ABP_<Client>_CREDITRATING	Basic view for credit standing check
18	BV_BP_CreditStanding	/1BC/DA_ABP_<Client>_CREDITSTANDING	Basic view for credit standing

Number	Name of SAP HANA View	Name of ABAP CDS View	Description
19	BV_BP_Occupation	/1BC/DA_ABP_<Client>_OCCUPATION	Basic view for occupation
20	BV_BP_FinancialStatement	/1BC/DA_ABP_<Client>_FINANCSTMNT	Basic view for balance sheet and income statement
21	BV_BP_BankRegulations	/1BC/DA_ABP_<Client>_BANKREGULATION	Basic view for bank regulations
22	BV_BP_CorrelationModel	/1BC/DA_ABP_<Client>_CORRELMODEL	Basic view for correlation model
23	BV_BP_Weight	/1BC/DA_ABP_<Client>_WEIGHT	Basic view for weighting
24	BV_BP_ProbabilityOfDefault	/1BC/DA_ABP_<Client>_PROBABLTYOFDEF	Basic view for probability of default
25	BV_BP_BaselAttributes	/1BC/DA_ABP_<Client>_BASELATTRIBUTE	Basic view for Basel attributes
26	BV_BPR	/1BC/DA_<Client>_ABPR	Basic view for business partner relationship

Note

The general basic view `BV_BP` can group several service module views (for example, *BV_BP_Common*, *BV_BP_CommonFS*, and so on) in one view using `GUID_K`.



Basic Views for Yield Curves

You can generate the following basic views under *Reference Data* > *Market Data* > *Yield Curve*:

Name of SAP HANA View	Name of ABAP CDS View	Package	Comment
BV_MD__BA1_F4_DF_PERS	/1BC/DAMD<Mandant>_BA1_F4_DF_PERS	sap.fs.frdp.<Mandant>.<User>.MD	

The generated basic views reside in the package `sap.fs.frdp.<Mandant>.<User>.MD`.

Basic Views for Master Data Framework Objects

The basic views listed apply only to master data framework objects that are **persisted in flat tables**.

You can generate the following basic views under *Source Data* > *Flow Data* > *Master Data* > *Financial Transaction* (for example):

Name of SAP HANA View	Name of ABAP CDS View	Package	Comment
BV_MDF_<Vorlage>_<Vorlageversion>	/1BC/BC<Mandant><Vorlage><Vorlageversion>	sap.fs.frdp.<Mandant>.<User>.MDF.<FT/FI/...>	This view shows the data for a specific template and template version based on the business date and system time that you entered.
BV_MDF_<Vorlageversion>_CURRENT	/1BC/BZ<Mandant><Vorlage><Vorlageversion>	sap.fs.frdp.<Mandant>.<User>.MDF	This view shows the data for a specific template and template version based on the business date that you entered. The system takes the system time into account during generation and selects the last table entry.
BV_MDF_BA1_F1_CON_FLAT	/1BC/BC<Mandant>_ALL	sap.fs.frdp.<Mandant>.<User>.MDF.misc	This view shows the data for all templates and template versions persisted in flat tables based on the business date and system time that you entered.
BV_MDF_BA1_F1_CON_FLAT_CURRENT	/1BC/BZ<Mandant>_ALL	sap.fs.frdp.<Mandant>.<User>.MDF.misc	This view shows the data for all templates and template versions persisted in flat tables based on the business date you entered. The system takes the system time into account during generation and selects the last table entry.

Basic Views for Business Transactions

The views listed apply only to business transactions that are **persisted in flat tables**.

You can generate the following basic views under *Source Data* > *Flow Data* > *Business Transactions*:

Name of SAP HANA View	Name of ABAP CDS View	Package	Comment
BV_BT_<BT_CLASS>		sap.fs.frdp.<Mandant>.<User>.BT	This view shows the data for all business transactions persisted in flat tables that belong to a specific business transaction class. This view is used both for reversal business transactions by reference and for reversal business transactions with values.
BV_BT_ORIGINAL_<BT_CLASS>		sap.fs.frdp.<Mandant>.<User>.BT	This view shows the data for a specific business transaction class. This view is used only for reversal business transactions by reference.
BV_BT_REVERSAL_<BT_CLASS>		sap.fs.frdp.<Mandant>.<User>.BT	This view is used to reverse business transactions belonging to a specific business transaction class and applies only to reversal business transactions by reference.
BV_BT_REV_OF_REV_<BT_CLASS>		sap.fs.frdp.<Mandant>.<User>.BT	This view is used to reverse the (abovementioned) reversal business transactions by reference that belong to a specific business transaction class.
BV_BT_FLAT	/1BC/DABT_821_FLAT	sap.fs.frdp.<Mandant>.<User>.BT	This view shows the data for all business transactions persisted in flat tables. This view is used both for reversal business transactions by reference and for reversal business transactions with values.
BV_BT_ORIGINAL_FLAT	/1BC/DABT_821_ORIGINAL_FLAT	sap.fs.frdp.<Mandant>.<User>.BT	This view shows the data for all business transactions that allow reversal business transactions by reference.

Name of SAP HANA View	Name of ABAP CDS View	Package	Comment
BV_BT_REVERSAL_FLAT	/1BC/DABT_821_REVERSAL_FLAT	sap.fs.frdp.<Mandant>.<User>.BT	This view shows the data for all reversal business transactions by reference.
BV_BT_REV_OF_REV_FLAT	/1BC/DABT_821_REV_OF_REV_FLAT	sap.fs.frdp.<Mandant>.<User>.BT	This view is used to reverse all reversal business transactions by reference.

Basic Views for Positions

The views listed apply only to positions that are **persisted in flat tables**.

You can generate the following basic views under *Source Data* > *Flow Data* > *Positions*:

Name of SAP HANA View	Name of ABAP CDS View	Package	Comment
BV_POS_<Bestandsklasse>	/1BC/DAPOS_<Mandant>_<Bestandsklasse>	sap.fs.frdp.<Mandant>.<User>.POS	This view shows the data for a specific position class based on the business date and system time that you entered.
BV_POS_<Bestandsklasse>_CURRENT	/1BC/DAPOS_<Mandant>_<Bestandsklasse>_CURRENT	sap.fs.frdp.<Mandant>.<User>.POS	This view shows the data for a specific position class based on the business date you entered. The system takes the system time into account during generation and selects the last table entry.
BV_POS_FLAT	/1BC/DAPOS_<Mandant>_FLAT	sap.fs.frdp.<Mandant>.<User>.POS	This view shows the data for all position classes persisted in flat tables based on the business date and system time that you entered.
BV_POS_CURRENT	/1BC/DAPOS_<Mandant>_FLAT_CURRENT	sap.fs.frdp.<Mandant>.<User>.POS	This view shows the data for all position classes persisted in flat tables based on the business date you entered. The system takes the system time into account during generation and selects the last table entry.

Basic Views for Results Data Area

The following basic views are generated under *Results Data Area*:

Name of SAP HANA View	Name of ABAP CDS View	Package	Comment
BV_RDL_<RDA>_<RT>_CURRENT	/1BC/BZ<Mandant><RDA>_<RT>	sap.fs.frdp.<Mandant>.<User>.RDL	These views show the data for a specific result category of a results data area based on the business date that you entered. The system takes the system time into account during generation and selects the last table entry.
BV_RDL_<RDA>_<RT>	/1BC/BR<Mandant><RDA>_<RT>	sap.fs.frdp.<Mandant>.<User>.RDL	The generated basic views reside in the package sap.fs.frdp.<Mandant>.<User>.RDL.

More Information

Package Hierarchy in SAP HANA Studio [page 1885]

9.1.3 Authorization Checks in Data Access Interface

You can perform authorization checks to restrict access to business partner data in data access interface views in Customizing for *Bank Analyzer* under *Data Access Interface* > *Enable Authorization Checks in DAI*. If you enable authorization checks, additional views in the SAP HANA views and ABAP CDS views will be generated. The data in these additional views will be protected based on the user's authorizations to view blocked data.

Prerequisite

Before you enable authorization checks, make sure that the analytical business partner is migrated from old business partner relation tables to the optimized data model for business partner versioning.

In Customizing for Bank Analyzer under *Basic Settings* > *Settings for Meta Data* > *Set Up Custom Characteristics for Blocks* you define custom characteristics that can contain personal data for personally identifiable information (PII category) and for which the system needs to run the checking and blocking of personal data.

Authorization Checks for SAP HANA Views

A graphical calculation view is generated that provides information about whether a business partner has reached its end of purpose. This view is common to all frameworks.

This graphical calculation view is joined with the existing basic views to generate a new set of basic views with the suffix `AUTH`. They are access-protected via analytic privileges. The following analytic privileges are generated to grant access to read data from these views depending on the roles assigned to users:

Analytic Privilege	Description
AU_GEN_USER	Analytic privilege for general users to restrict access to blocked data. They can view the data of only those business partners that are not blocked.
AU_AUDITOR	Analytic privilege for auditors to provide unrestricted access to data. They can also view the data of blocked business partners.

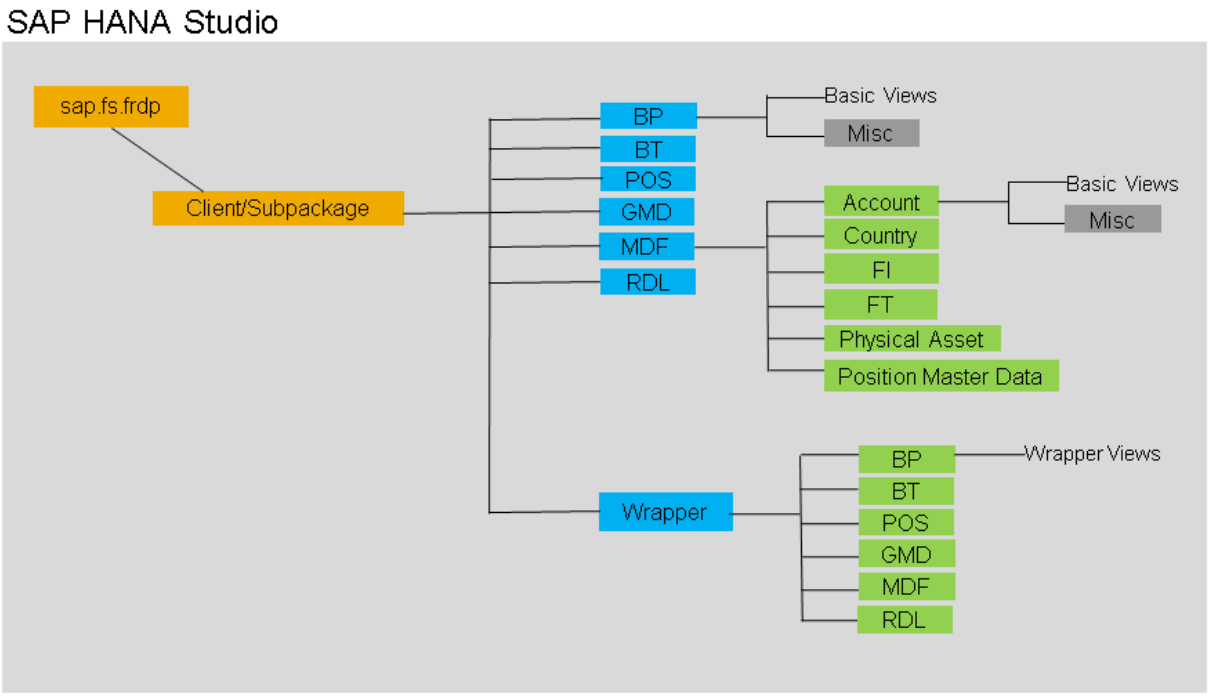
Authorization Checks for ABAP CDS Views

A new set of ABAP CDS basic views with names starting with `/1BC/B0<<>>` are generated with a DCL mapping role. These views and access controls are also generated in the $TMP folder of the user. The mapping role is associated with the authorization object `F_BAFR_AUT`. When a user requests access to data stored in an ABAP CDS view, an authorization check based on the authorization object will be performed. Users with this authorization can view both blocked and unblocked data, whereas users without this authorization can view only unblocked data.

For SAP HANA native consumption, a table function will be generated with the structured privilege check `SP_AUTHORIZATION`. Users can be granted the privilege to view blocked business partner data using the call GRANT_ACTIVATED_ANALYTICAL_PRIVILEGE (SP_AUTHORIZATION, '<<user_name>>).

9.2 Package Hierarchy in SAP HANA Studio

In SAP HANA Studio, you can find the generated views in a defined hierarchy as illustrated in the figure below.



All the generated views reside in the package `sap.fs.frdp`. In this package, you can find a subpackage, which is either the system client (for example, `821`) or the subpackage name maintained in the Customizing activity *Define Subpackages in a Transport Chain* (for example `PQR`). In this subpackage, you will find a folder for each framework and a folder for wrapper views.

Framework Folders

The following folders are framework folders:

- BP
- BT
- POS
- GMD
- MDF
- RDL

You can find the **basic views** directly in these folders, whereas all other views are kept in a separate subfolder called `Misc.`. Every framework folder has the same hierarchy except MDF framework. In the MDF framework folder, there is another level of folder structures for objects: *Account*, *Country*, *Financial Instruments*, *Financial Transaction*, *Physical Asset*, and *Position Master Data*. You can find the basic view directly in these folders. All other views are kept in a separate folder called `Misc.`.

Wrapper View Folder

The wrapper views of each framework are kept in a separate folder called *Wrapper*. This folder has subfolders based on the framework - *BP*, *BT*, *POS*, *GMD*, *MDF*, and *RDL*. All wrapper views are kept in these subfolders based on the framework.

Related Information

View Types in Data Access Interface [page 1876]
Basic Views [page 1878]

9.3 Extract Finance and Risk Data Model

Use

You can use the transaction **/BA1/FR_DM_EXTRACT** to extract finance and risk data model information from the Bank Analyzer system in the form of XML files. You can then import these XML files to the , for example, where you can display the data model graphically, enhance it, or document it.

You can find the transaction on the SAP Easy Access screen for *Bank Analyzer* under *Infrastructure* > *Extraction and Reporting Services* > *Data Model Extraction* > *Extract Finance and Risk Data Model*.

You can select the following areas for extraction:

- Reference data
  - Analytical business partner data
  - Market data

> **Caution**
>
> Market data can be extracted. However the system ignores it when it is imported to SAP PowerDesigner.

- Source data
  - Master data
  - Generic market data
  - Flow data
- Results data

The system creates a separate XML file for each area selected. Under *Output Options*, specify whether you want to save or display the XML files. If you want to save the data, you need to specify the file directory where you want the system to save the XML files. If you want to import the file to the *FRDP PowerDesigner Extension* afterwards, we recommend that you select the `XML` directory in your PowerDesigner Workspace directory.

Features

The following Customizing activities are extracted as XML files:

- *Create Market Data Area*: The XML file tag `OBJ_CONFIGN_AREA` is assigned to the market data area.

  > **Note**
  >
  > You make Customizing settings for market data under *Financial Services* > *Foundation* > *Market Data*.

- *Interest* node
  - *Edit Reference Interest Rates and Yield Curves*: The reference interest rates and the interest rate spread types are extracted here, but the yield curve types are **not**.
- *Exchange Rates* node
  - *Define Exchange Rate Types and Translation Ratios*: The exchange rate factors are modeled in the XML file as application data so that the Customizing data is **not** included. However, the database table `/BA1/TF4_FXCVFCT` is included as an application table. The yield curve type for forward exchange rates is not taken into account.
- *Prices for Financial Instruments* node
  - *Edit Customizing for Security Prices*
  - *Assign Price Units to Security*
- *Volatilities* node
  - *Edit Volatility Types*
  - *Create Cluster from Volatility Types*
  - *Create Non-Underlying-Based Volatility Structures*

The following Customizing activities are not extracted:

- *Interest* node
  - *Edit Spread Curve Types*
  - *Edit Maturity Structures*
  - *BAdI: Model for External Yield Curve*
- *Exchange Rates* node
  - *Check Currency Codes*
  - *Define Standard Quotation for Exchange Rates*
  - *Set Decimal Places for Currencies*
  - *Map Exchange Rate Category to Exchange Rate Type*
  - *Configure Currency Translation for Accounting*
- *Prices for Financial Instruments* node
  - *Corporate Action for Security Prices and Cash Flows*
- *General Scenario Administration* node
- *Market Data Scenarios* node
- *Create Market Data Set*
- *Create Application*

9.4 Transport of Views

Use

In FRDP system landscape, the generated views in the source system are transported to the target systems using transport tools. For applications based on SAP HANA database, you can visualize these views in SAP HANA Studio in a package hierarchy.

Concept

The ABAP Change and Transport System (CTS) enables you to save the primary objects selected for generation in a transport request. When you release the ABAP transport request, the after-import functionality of data access interface gets triggered in the target systems. The target system reads the primary objects locked in the transport request, extracts the metadata, and deploys the HANA calculation and ABAP CDS views. The system writes transport logs that contain important information about whether the process was completed correctly.

Prerequisites

You can create a subpackage for a combination of source and target systems in Customizing under *Financial Services* > *Bank Analyzer* > *Data Access Interface* > *Define Subpackages in a Transport Chain*.

This Customizing is important before you do an after-import and generation. If multiple systems/clients are connected to a source system, defining a subpackage ensures that the generation happens only in a particular target system. For applications based on SAP HANA database, this Customizing ensures that the HANA calculation views reside in the same package hierarchy in database of source and target systems, for example `sap.fs.frdp.<subpackage>.Wrapper.MDF`.

Defining a subpackage is also important when you build a typical sequence of systems in a productive landscape with development, consolidation and productive systems. It ensures that the view names remain stable even if you transport between different clients of a system. If you do not define a subpackage, the custom views you build in development system will not work in the productive system because the basic views will have other names in development and consolidation systems.

> **Note**
>
> In a test environment, it is not mandatory to define a subpackage. The client name will be used by default as the name of the subpackage.

More Information

Transport in Bank Analyzer [page 1838]

Important Disclaimers and Legal Information

Coding Samples

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended to better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, unless damages were caused by SAP intentionally or by SAP's gross negligence.

Gender-Neutral Language

As far as possible, SAP documentation is gender neutral. Depending on the context, the reader is addressed directly with "you", or a gender-neutral noun (such as "sales person" or "working days") is used. If when referring to members of both sexes, however, the third-person singular cannot be avoided or a gender-neutral noun does not exist, SAP reserves the right to use the masculine form of the noun and pronoun. This is to ensure that the documentation remains comprehensible.

Internet Hyperlinks

The SAP documentation may contain hyperlinks to the Internet. These hyperlinks are intended to serve as a hint about where to find related information. SAP does not warrant the availability and correctness of this related information or the ability of this information to serve a particular purpose. SAP shall not be liable for any damages caused by the use of related information unless damages have been caused by SAP's gross negligence or willful misconduct. All links are categorized for transparency (see: https://help.sap.com/viewer/disclaimer).

go.sap.com/registration/contact.html