TADM12_2

SAP NetWeaver AS Implementation & Operation II

SAP NetWeaver

Date ______________________

Training Center ______________________

Instructors ______________________

____

Education Website ______________________

Participant Handbook

Course Version: 62
Course Duration: 10 Day(s)
Material Number: 50089457

An SAP course - use it to learn, reference it for work

Copyright

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Trademarks

- Microsoft®, WINDOWS®, NT®, EXCEL®, Word®, PowerPoint® and SQL Server® are registered trademarks of Microsoft Corporation.
- IBM®, DB2®, OS/2®, DB2/6000®, Parallel Sysplex®, MVS/ESA®, RS/6000®, AIX®, S/390®, AS/400®, OS/390®, and OS/400® are registered trademarks of IBM Corporation.
- ORACLE® is a registered trademark of ORACLE Corporation.
- INFORMIX®-OnLine for SAP and INFORMIX® Dynamic ServerTM are registered trademarks of Informix Software Incorporated.
- UNIX®, X/Open®, OSF/1®, and Motif® are registered trademarks of the Open Group.
- Citrix®, the Citrix logo, ICA®, Program Neighborhood®, MetaFrame®, WinFrame®, VideoFrame®, MultiWin® and other Citrix product names referenced herein are trademarks of Citrix Systems, Inc.
- HTML, DHTML, XML, XHTML are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.
- JAVA® is a registered trademark of Sun Microsystems, Inc.
- JAVASCRIPT® is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.
- SAP, SAP Logo, R/2, RIVA, R/3, SAP ArchiveLink, SAP Business Workflow, WebFlow, SAP EarlyWatch, BAPI, SAPPHIRE, Management Cockpit, mySAP.com Logo and mySAP.com are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other products mentioned are trademarks or registered trademarks of their respective companies.

Disclaimer

THESE MATERIALS ARE PROVIDED BY SAP ON AN "AS IS" BASIS, AND SAP EXPRESSLY DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR APPLIED, INCLUDING WITHOUT LIMITATION WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WITH RESPECT TO THESE MATERIALS AND THE SERVICE, INFORMATION, TEXT, GRAPHICS, LINKS, OR ANY OTHER MATERIALS AND PRODUCTS CONTAINED HEREIN. IN NO EVENT SHALL SAP BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR PUNITIVE DAMAGES OF ANY KIND WHATSOEVER, INCLUDING WITHOUT LIMITATION LOST REVENUES OR LOST PROFITS, WHICH MAY RESULT FROM THE USE OF THESE MATERIALS OR INCLUDED SOFTWARE COMPONENTS.

About This Handbook

This handbook is intended to complement the instructor-led presentation of this course, and serve as a source of reference. It is not suitable for self-study.

Typographic Conventions

American English is the standard used in this handbook. The following typographic conventions are also used.

Type Style	Description
Example text	Words or characters that appear on the screen. These include field names, screen titles, pushbuttons as well as menu names, paths, and options. Also used for cross-references to other documentation both internal (in this documentation) and external (in other locations, such as SAPNet).
Example text	Emphasized words or phrases in body text, titles of graphics, and tables
EXAMPLE TEXT	Names of elements in the system. These include report names, program names, transaction codes, table names, and individual key words of a programming language, when surrounded by body text, for example SELECT and INCLUDE.
`Example text`	Screen output. This includes file and directory names and their paths, messages, names of variables and parameters, and passages of the source text of a program.
`Example text`	Exact user entry. These are words and characters that you enter in the system exactly as they appear in the documentation.
`<Example text>`	Variable user entry. Pointed brackets indicate that you replace these words and characters with appropriate entries.

Icons in Body Text

The following icons are used in this handbook.

Icon	Meaning
💡	For more information, tips, or background
➡	Note or further explanation of previous point
⚠	Exception or caution
1 2 3	Procedures
🖥	Indicates that the item is displayed in the instructor's presentation.

Contents

Course Overview

This course complements the course TADM10 for providing you with all the database- and operating system-independent knowledge that is needed for passing one part of the certification exam for **SAP Certified Technology Associate – System Administration – SAP NetWeaver 7.0 (<Database>)**.

To pass the exam to become a certified SAP Certified Technology Associate – System Administration – SAP NetWeaver 7.0 (<Database>), you must have good knowledge of the content of courses TADM10 and TADM12, as well as one of the following database-specific courses: TADM51 (Oracle), TADM53 (MS SQL Server), TADM56 (DB2 on Win/UX) or ADM515 (MaxDB).

TADM12 is **largely independent of the type of operating system and database technology.**.

Like the other TADM courses, TADM12 comprises several individual courses (or parts thereof), which are arranged here in a way that will enable you to gain the knowledge you require as an SAP Technology Consultant as efficiently as possible.

Week 1 of course TADM12 is based on content taken from the following courses:

1. ADM110 – Installation SAP ECC 6.0
2. ADM100 – Administration AS ABAP I
3. ADM102 – Administration AS ABAP II
4. ADM200 – Administration AS Java

Week 2 of course TADM12 is based on content taken from the following courses:

1. ADM325 - SAP Software Logistics for ABAP
2. ADM102 - Administration AS ABAP II

At the end of the database-specific part of this training (TADM5#, or after ADM515, to be booked separately), there is a three-hour certification exam that covers topics from courses TADM10 and TADM12 and TADM5#/ADM515.

Caution: Note that the certification exam has been designed in such a way that the answers to all of the exam questions are contained in the handbooks provided for courses TADM10, TADM12, TADM5# (or ADM515). Therefore, you do not require any additional course material even if the instructor hands out other books during the course or provides additional information not contained in the course handbooks.

Target Audience

This course is intended for the following audiences:

- SAP Technology Consultants (Associate Level)

Course Prerequisites

Required Knowledge

- Content of the class **TADM10 - SAP NetWeaver AS Implementation & Operation I**

Course Goals

This course will prepare you to:

- Work as an Associate Consultant (Junior Consultant) within the SAP technology environment
- Install, configure and maintain SAP systems based on SAP NetWeaver AS ABAP
- Install, configure and maintain SAP systems based on SAP NetWeaver AS Java

Course Objectives

After completing this course, you will be able to:

- perform basic tasks within the technology environment of SAP systems

Unit 1

Introduction to SAP Software Logistics

Unit Overview

In this unit, you will learn about SAP systems and instances, about the data types in the database and the client concept. You will also learn the purpose of different client roles and different system types in a system landscape recommended by SAP. Additionally, you will learn about the system and client change options. Finally, you will identify the components of the Change and Transport System (CTS).

Unit Objectives

After completing this unit, you will be able to:

- Describe SAP systems, instances and servers
- Identify the system landscape recommended by SAP
- Identify the types of data in an SAP system
- Explain the client concept in an SAP system
- Describe the system change options in an SAP system
- Describe the client change options in an SAP system

Unit Contents

Lesson: SAP System Landscape

Lesson Overview

In this lesson, you will learn about the SAP's product portfolio, SAP NetWeaver, SAP systems and instances. You will also identify the components of the system landscape that SAP recommends.

Lesson Objectives

After completing this lesson, you will be able to:

- Describe SAP systems, instances and servers
- Identify the system landscape recommended by SAP

Business Example

An SAP system administrator needs to understand the roles of different systems in the SAP landscape before deciding how to configure the Transport Management System.

SAP NetWeaver, SAP ERP

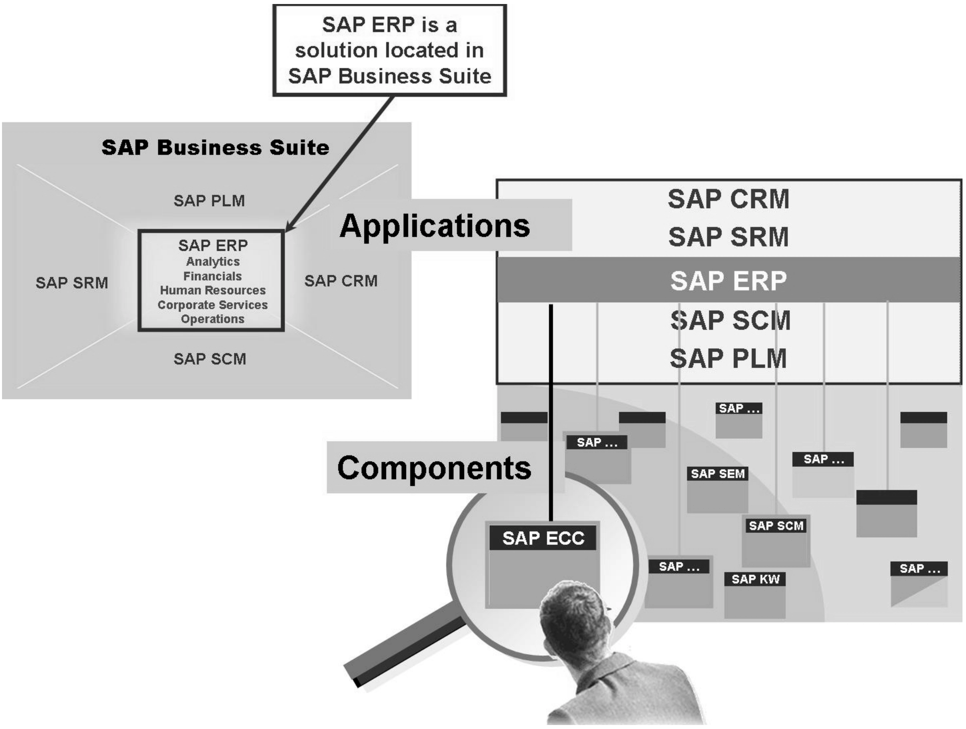


Figure 1: SAP Business Suite

As a first step, let's review the SAP product portfolio. SAP ERP is a SAP application. SAP offers other SAP applications for the areas of Customer Relationship Management (SAP CRM), Supplier Relationship Management (SAP SRM), Product Lifecycle Management (SAP PLM), and Supply Chain Management (SAP SCM). All these applications are part of the SAP Business Suite (formerly known as mySAP.com). As a result, a customer with a SAP Business Suite license has also the license for the applications that can be seen as subsets of the mSAP Business Suite.

Inside the SAP applications, SAP offers the SAP components, like SAP ECC, SAP SCM (formerly known as APO, Advanced Planner and Optimizer) or others. From a technology point of view, these are the components that a customer installs or upgrades.

Another important product to mention is SAP NetWeaver. It represents the technology stack provided by SAP to support the SAP Business Suite and its SAP applications, as well as other SAP products. Furthermore, SAP NetWeaver and its integration capabilities also allow the integration of non-SAP products. SAP NetWeaver is delivered with all of the SAP applications. As a result, customers can make use of the SAP NetWeaver functions for a specific application. SAP NetWeaver is a set of different components like SAP NetWeaver AS, SAP BI, SAP NetWeaver Portal, SAP PI, SAP MDM and others. These technical components of SAP NetWeaver can be installed as needed to serve different requirements.

The SAP for Industry applications, which are also part of the SAP product portfolio, can be seen as extensions to the SAP Business Suite and its SAP applications. SAP for Industry applications, which were historically add-ons to one single component called “SAP R/3”, now provide not just add-ons to SAP components but also preconfigurations and best practices.

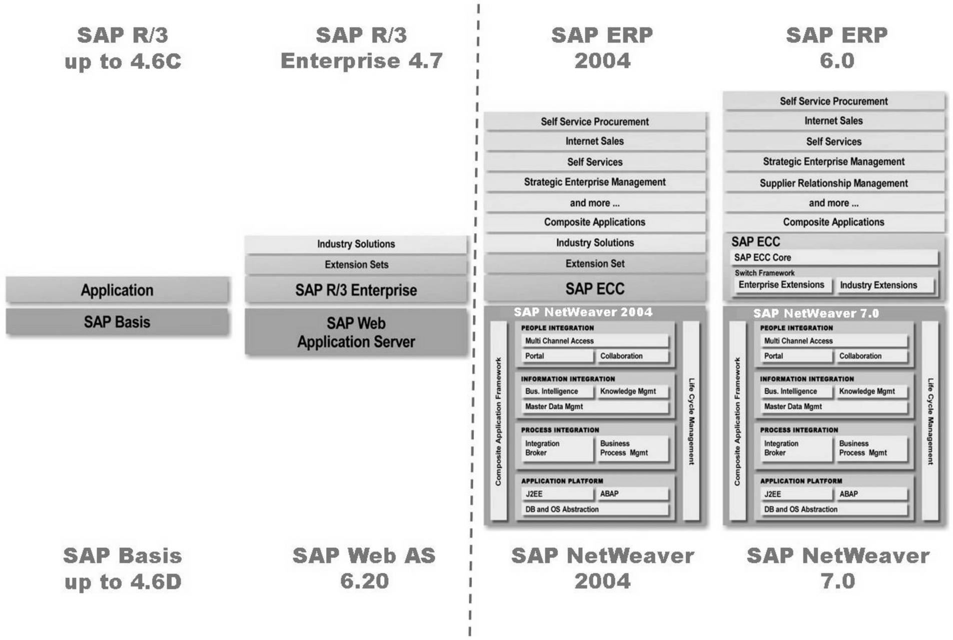
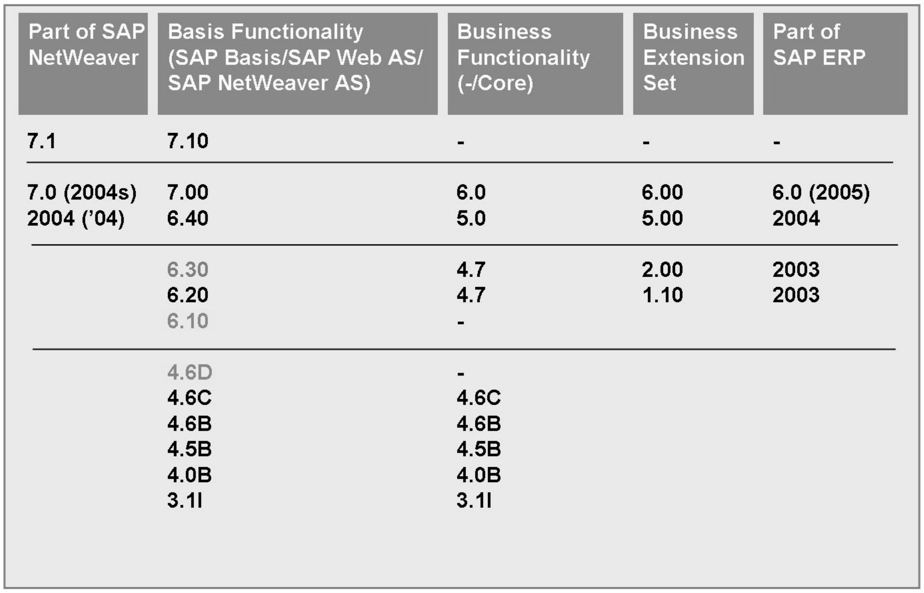


Figure 2: Example: Architecture of SAP ERP

As an example of an SAP application, this figure gives a good overview of the evolution of the ERP applications from SAP. After the SAP R/1 and SAP R/2 products, SAP introduced SAP R/3 in 1992. Now, there was a two-layer architecture (SAP Basis and SAP Application).

With SAP ERP, which has been available since March 2003, SAP offered a bundle of components supporting ERP business processes. At this point, SAP R/3 Enterprise 4.7 was one of those components.

The SAP ERP Central Component (SAP ECC)(ECC) represents the next step in this evolution. As you can see, different software components can be operated on an SAP ECC.



Part of SAP NetWeaver	Basis Functionality (SAP Basis/SAP Web AS/ SAP NetWeaver AS)	Business Functionality (-/Core)	Business Extension Set	Part of SAP ERP
7.1	7.10	-	-	-
7.0 (2004s)	7.00	6.0	6.00	6.0 (2005)
2004 ('04)	6.40	5.0	5.00	2004
	6.30	4.7	2.00	2003
	6.20	4.7	1.10	2003
	6.10	-		
	4.6D	-		
	4.6C	4.6C		
	4.6B	4.6B		
	4.5B	4.5B		
	4.0B	4.0B		
	3.1I	3.1I		

Figure 3: Get the 'Numbers' straight: SAP R/3 / ECC Releases

This slide gets rid of the confusion concerning the different R/3, R/3 Enterprise, ECC, Extension Set and ERP and Basis, SAP Web AS, SAP NetWeaver AS and SAP NetWeaver releases.

SAP Systems, Instances and Servers

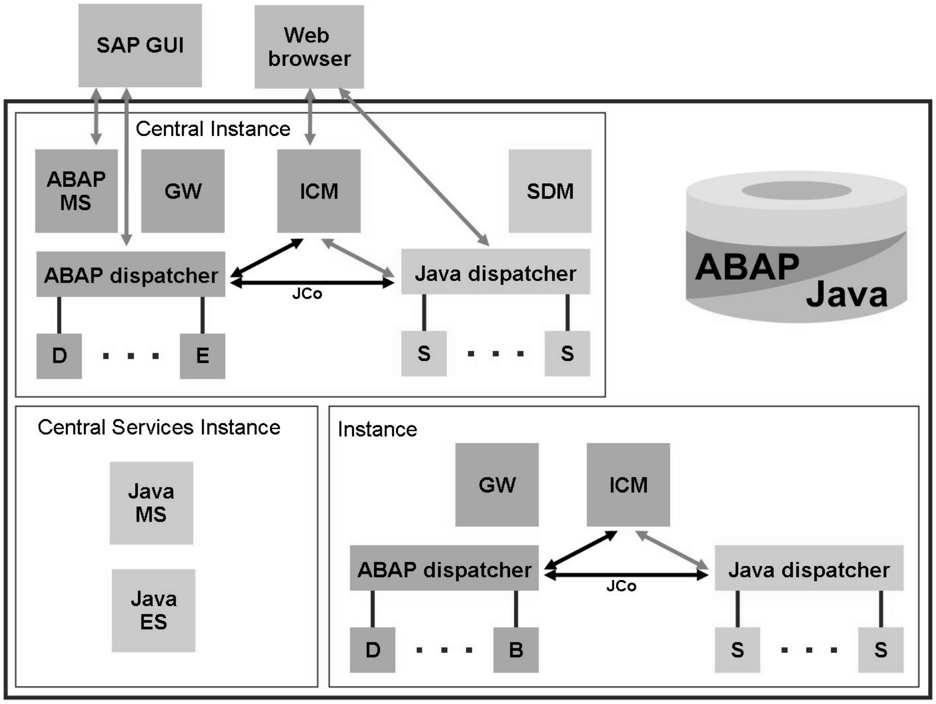


Figure 4: Architecture of the SAP NetWeaver AS ABAP+Java with two Instances

- The Internet Communication Manager (ICM) is the interface to the internet. It can deal with web requests as server and as client. It supports several protocols like HTTP, HTTPS and SMTP. The SAP NetWeaver AS can be used as a web server and as a web client.
- The ABAP-Dispatcher dispatches requests to the Work Processes. If all work processes are busy at the moment, the requests are queued in the Dispatcher-Queue.
- The ABAP Work Processes execute the ABAP coding.
- The Message Server exchanges messages and can perform a load balancing in the SAP system.
- On the Java-side of the SAP NetWeaver AS there are components like the Java-Dispatcher, Server Processes, the Software Deployment Manager (SDM) and the Central Services (SCS).
- The SAP Java Connector (JCo) provides communication between the ABAP-Stack and the Java-Stack of the SAP NetWeaver AS, in both directions.

Client requests to the SAP NetWeaver AS Java are handled by the Java-Dispatcher. It chooses a free Server Process for serving the request and builds up a connection between the client and the Server Process. The Java-Dispatcher works according to a 'Round Robin' algorithm.

The Software Deployment Manager (SDM) is the standard tool with which J2EE components are installed on the SAP NetWeaver AS Java.

The Central Services run on one server and forms an own Java instance. It consist of a Message Service and an Enqueue Service and are the basis for communication and synchronisation for the Java Cluster.

Each instance of a SAP NetWeaver AS ABAP+Java contains the ABAP Dispatcher with its Work Processes and the Java Dispatcher with its Server Processes. One of the instances is usually installed as an ABAP Central Instance, which means, that the Enqueue Work Process runs here. Alternatively the System can be installed without a Central Instance, but with a stand alone Enqueue Server. The applications and data for ABAP and Java both have their own database scheme in one common database.

Users can log on and communicate with the SAP system either using the SAP GUI or a web client (browser):

- Communication using the SAP GUI: users log on using the Message Server (for load balancing) or directly log on to a ABAP Dispatcher. The Work Processes execute the requests of the users.
- Execution of web requests: web requests are handled by the Internet Communication Manager (ICM). These HTTP(S) requests can be intended for the Internet Communication Framework (ICF), which means that they are executed by an ABAP Work Process (for example BSP applications). These requests can also be J2EE requests for the SAP NetWeaver AS Java. The ICM can direct the requests according to their URL.

System Landscape Recommended by SAP

From now on we focus on the SAP NetWeaver AS ABAP.

- **The system landscape should enable you to:**
  - **Protect application data using:**
    - **Client roles**
    - **Authorization concepts**
  - **Perform development, quality assurance, and production activities in separate environments to ensure:**
    - **Consistency of Repository objects**
    - **A secure and stable production environment**

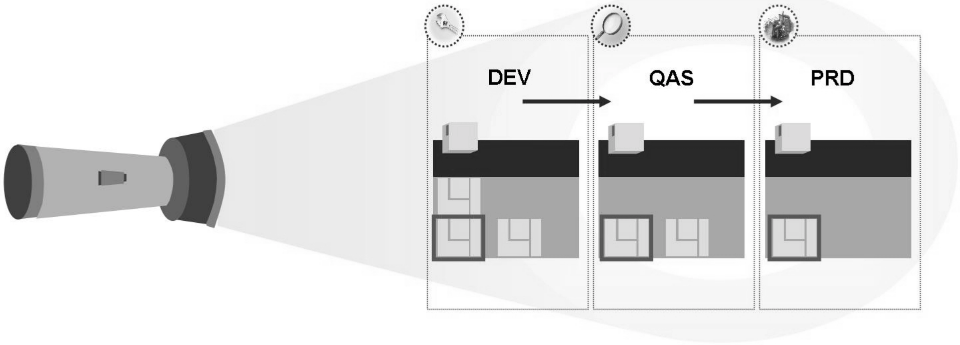


Figure 5: Aims of the System Landscape

To maintain data security, you should protect application data by using the client concept, which separates data according to clients, and the authorization concept, which separates data within one client according to the users.

In addition, you should strictly separate the environments for development, quality assurance, and production.

The performance of the production client should not be affected by development and test activities in the other clients.

Changes to Repository objects are client-independent, and immediately affect the runtime environment. Therefore, changes have to be tested before being transported to the production system.

If you had only one system, no further development would be possible after production had started. Changes to repository objects could be made only when production operations were stopped for development and testing.

A two-system landscape is not optimal because development and quality assurance testing both take place in the development system.

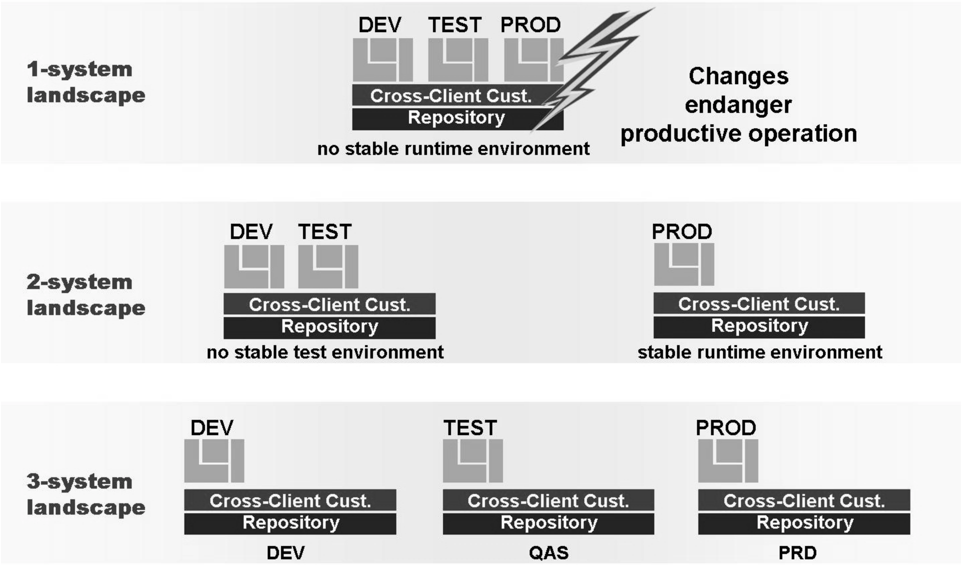


Figure 6: The SAP-Recommended System Landscape

To ensure smooth production operation, an SAP system landscape should contain the three SAP systems, the standard clients, and any required additional clients.

A three-system landscape is recommended and meets the needs of most customers.

Development takes place in the development system. Changed objects are then released to the quality assurance system.

In the quality assurance system, you can test new developments without affecting the production environment. A quality assurance system enables you to integrate development efforts and to check the validity and consistency of transported objects before moving the objects into production.

All objects imported into the quality assurance system can also be delivered to one or more production systems.

The three-system landscape is also a basis to apply Support Packages or to perform a release upgrade.

SAP systems in the same system landscape require unique system names.

SAP provides the tools required for the creation, documentation, and distribution of changes within a system landscape. To set up the system landscape so that it supports the management, verification, and testing of all changes:

- A single client is recommended for all customizing work. Set client change options appropriately before using your clients.
- A single SAP system (the development system) is recommended for all development work. Set the system change options appropriately.
- To create and assign appropriate user authorizations to developers and customizers, use the sample profiles included in the SAP system. These authorizations control whether a user can create, modify, or release change tasks and transport requests.
- Record and document all changes to a client or system.

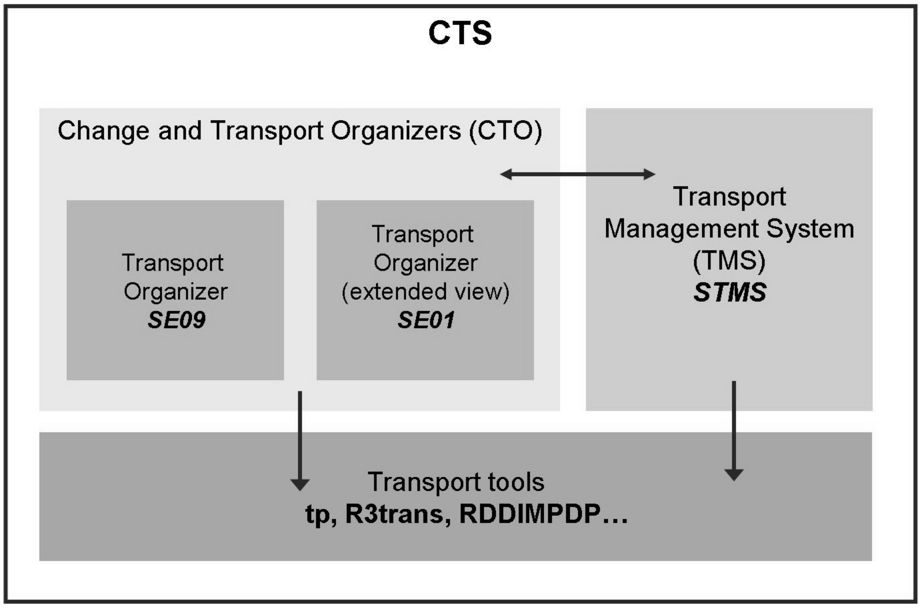

CTS
Change and Transport Organizers (CTO)
Transport Organizer SE09
Transport Organizer (extended view) SE01
Transport Management System (TMS) STMS
Transport tools
tp, R3trans, RDDIMPDP…

Figure 7: Change and Transport System (CTS)

In SAP Release 4.0, SAP has introduced a revised version of the Change and Transport System (CTS), comprising the following tools:

- The Change and Transport Organizer (CTO) provides functions for organizing software development projects. It is designed to support projects of all sizes, whether they are carried out centrally or in a distributed environment.
- The Transport Management System (TMS) organizes, monitors, and performs transports for all SAP systems within a system landscape. In addition, TMS is used to configure and manage the setting up of SAP systems and transport routes within a system landscape.
- Operating system transport tools are executables that communicate with the SAP system, the database, and files generated during the export process.

Lesson Summary

You should now be able to:

- Describe SAP systems, instances and servers
- Identify the system landscape recommended by SAP

Lesson: Client Concept

Lesson Overview

In this lesson, you will learn how to identify client roles in the system. In addition, you will learn about the concepts of change management within an SAP system environment.

Lesson Objectives

After completing this lesson, you will be able to:

- Identify the types of data in an SAP system
- Explain the client concept in an SAP system

Business Example

As an SAP system administrator or project team leader, you must monitor software configuration and program code changes to protect your production system from incorrect configuration settings or program bugs.

Data in an SAP System

Data in an SAP system can be divided into two categories: client-specific data and cross-client data.

1. Client-specific data, such as user master and application data, which affects only one client.
2. Cross-client data, such as cross-client customizing data and all Repository objects, which affects the whole system environment.

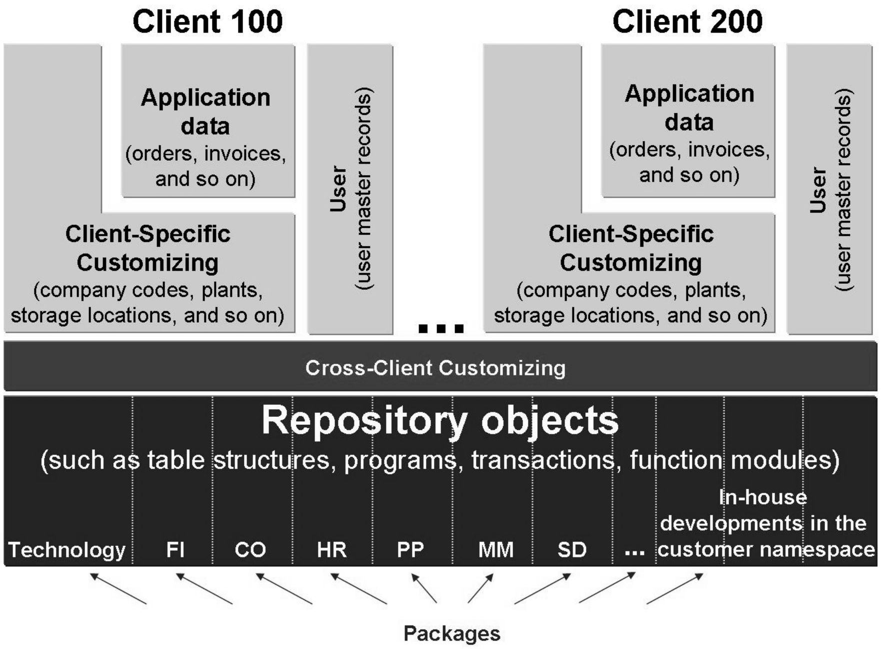

Client 100
Application data
(orders, invoices, and so on)
Client-Specific Customizing
(company codes, plants, storage locations, and so on)
User
(user master records)
Client 200
Application data
(orders, invoices, and so on)
Client-Specific Customizing
(company codes, plants, storage locations, and so on)
User
(user master records)
...
Cross-Client Customizing
Repository objects
(such as table structures, programs, transactions, function modules)
Technology
FI
CO
HR
PP
MM
SD
...
In-house developments in the customer namespace
Packages

Figure 8: Data Types in an SAP System

The ABAP Dictionary is a data dictionary that is part of the ABAP Repository. Each piece of the ABAP Dictionary information is entered only once and is then available anywhere in the system at any time. The ABAP Dictionary automatically supplies all new or changed information, thus providing current runtime objects and ensuring data consistency and security.

Clients in an SAP System

A **client** is a self-contained unit in commercial, organizational, and technical terms, with its own user master data and set of table key ranges.

- **A client:**
  - **Is a unit of the SAP system that is technically, organizationally, and commercially self-contained**
  - **Has its own set of user master data, application data, and Customizing based on table key ranges**

- **Technical implementation of clients:**
  - **All users must log on to a specific client within an SAP System**
  - **Application data (master and transaction data) belongs only to one specific client**

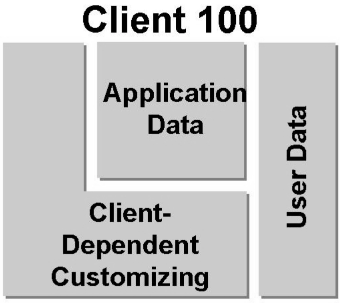


Figure 9: Clients in SAP Systems

Data from different clients is kept separate at the kernel level. SQL statements executed by an SAP application use the client number in the where-clause. Although a table may contain data from several different clients, the where-clause limits access to particular clients.

The following are examples of client-specific data:

- User master data, such as parameters, authorization, user groups
- Customizing data, such as organizational units, assignments, and document types
- Application data, such as business transaction data, and material master data

The SAP client concept can integrate several companies or subsidiaries in a single client by using company codes and the SAP authorization concept.

Company codes define the smallest corporate organizational units for which a complete self-contained set of accounts can be drawn up for external reporting.

The SAP authorization concept enables the parent company to access all subsidiaries for report purposes, while subsidiary-specific data is protected against access from other subsidiaries through company code definition.

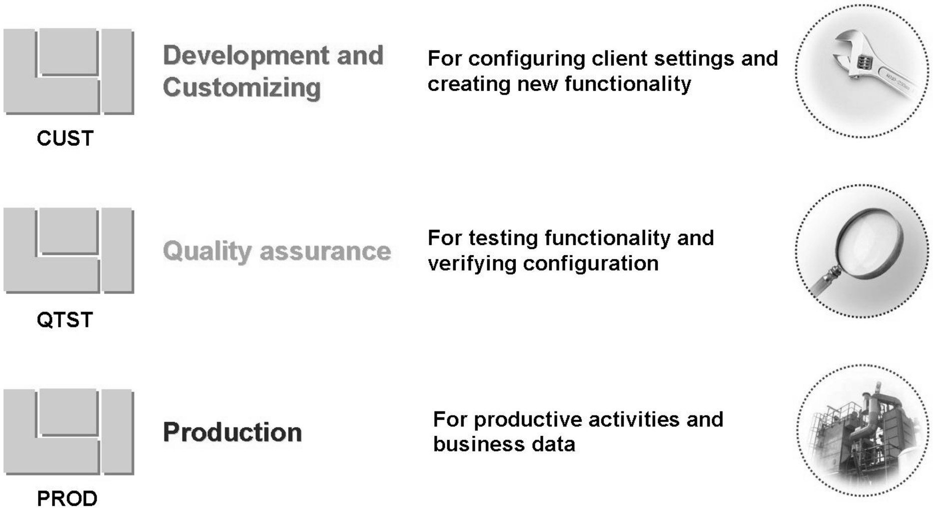

Development and Customizing
For configuring client settings and creating new functionality
CUST
Quality assurance
For testing functionality and verifying configuration
QTST
Production
For productive activities and business data
PROD

Figure 10: Standard Client Roles

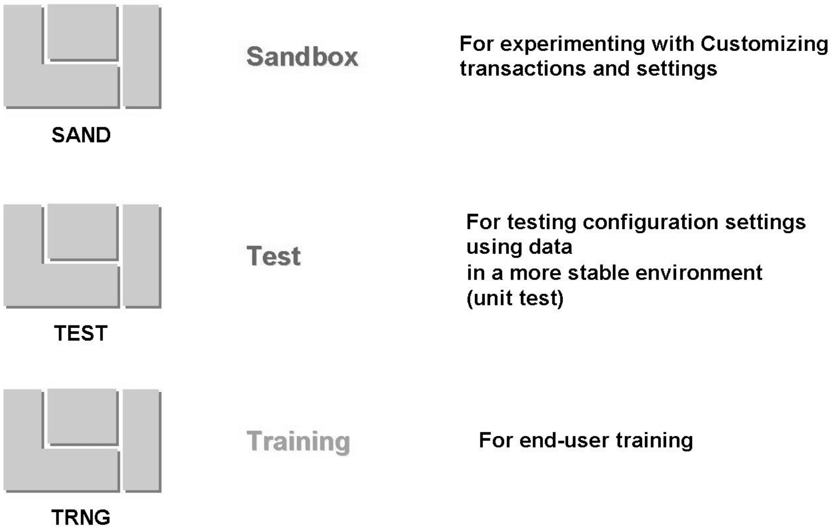
The standard client roles fulfill the optimal minimum requirements of your SAP system.

Client CUST, development and customizing, is the central customizing client where complete adaptation of the SAP system to customer-specific needs takes place. All changes performed in this client are recorded so they can be supplied to the other clients using the Transport Management System (TMS).

Client QTST, quality assurance, is used to test and verify the new customizing settings in the application.

Client PROD or production is the client for production activities, that is, where your company's business is carried out. Customizing changes imported into this client have to be first tested carefully in the QTST client in order to ensure that production operation is free of disruption.

To realize the full benefit of a three-system landscape, every critical client should have a separate SAP system.



SAND

Sandbox

For experimenting with Customizing transactions and settings

TEST

Test

For testing configuration settings using data in a more stable environment (unit test)

TRNG

Training

For end-user training

Figure 11: Additional Client Roles

Additional clients within a system landscape may include:

1. A sandbox client to experiment with transactions and settings.
2. A test client to verify customizing changes against sample data in a more stable environment. This can be done by 'copying' customizing settings from the customizing client to the sandbox client. This is called a unit test.
3. An end-user training environment.
4. A client that satisfies customer-specific requirements.

Lesson Summary

You should now be able to:

- Identify the types of data in an SAP system
- Explain the client concept in an SAP system

Lesson: System and Client Change Options

Lesson Overview

In this lesson, you will learn about the system and client change options of an SAP system.

Lesson Objectives

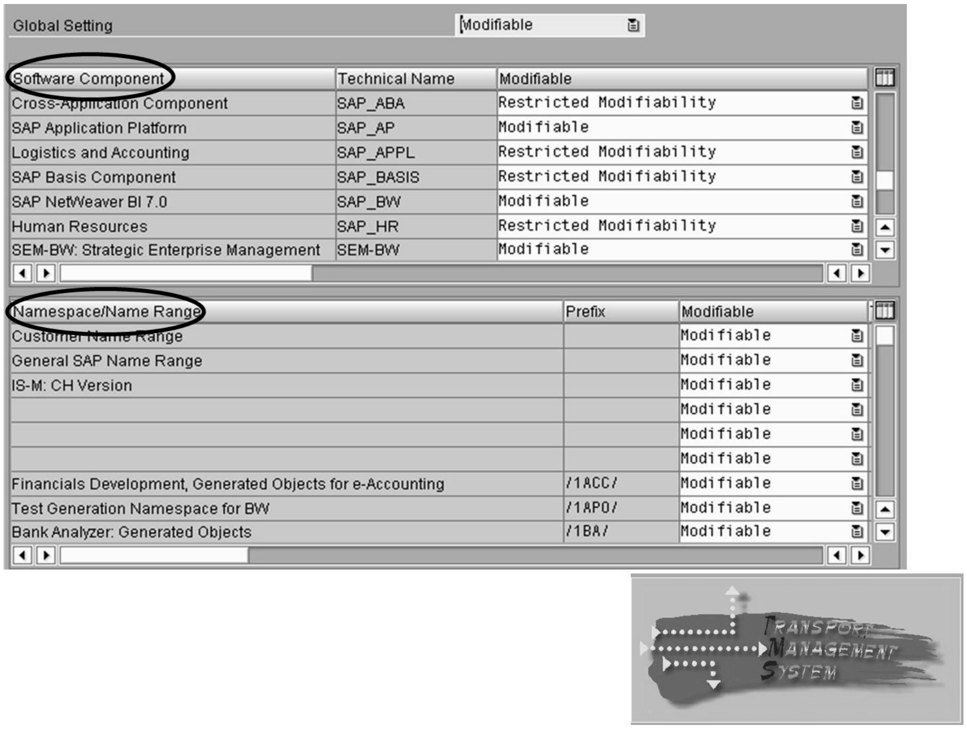
After completing this lesson, you will be able to:

- Describe the system change options in an SAP system
- Describe the client change options in an SAP system

Business Example

As a transport administrator, you need to set the correct system change options for each SAP system and the correct client change options for all clients.

System Change Options



Global Setting: Modifiable

Software Component	Technical Name	Modifiable
Cross-Application Component	SAP_ABA	Restricted Modifiability
SAP Application Platform	SAP_AP	Modifiable
Logistics and Accounting	SAP_APPL	Restricted Modifiability
SAP Basis Component	SAP_BASIS	Restricted Modifiability
SAP NetWeaver BI 7.0	SAP_BW	Modifiable
Human Resources	SAP_HR	Restricted Modifiability
SEM-BW: Strategic Enterprise Management	SEM-BW	Modifiable

Namespace/Name Range	Prefix	Modifiable
Customer Name Range		Modifiable
General SAP Name Range		Modifiable
IS-M: CH Version		Modifiable
		Modifiable
		Modifiable
		Modifiable
Financials Development, Generated Objects for e-Accounting	/1ACC/	Modifiable
Test Generation Namespace for BW	/1APO/	Modifiable
Bank Analyzer: Generated Objects	/1BA/	Modifiable

Figure 12: System Change Options

The system change option defines whether or not Repository objects and client-independent customizing objects are globally modifiable. If they are globally modifiable, you can specify whether or not each software component or namespace can be modified.

To reach the system change option use transaction SE06 and choose *System Change Option*. A software component is a set of dedicated development classes.

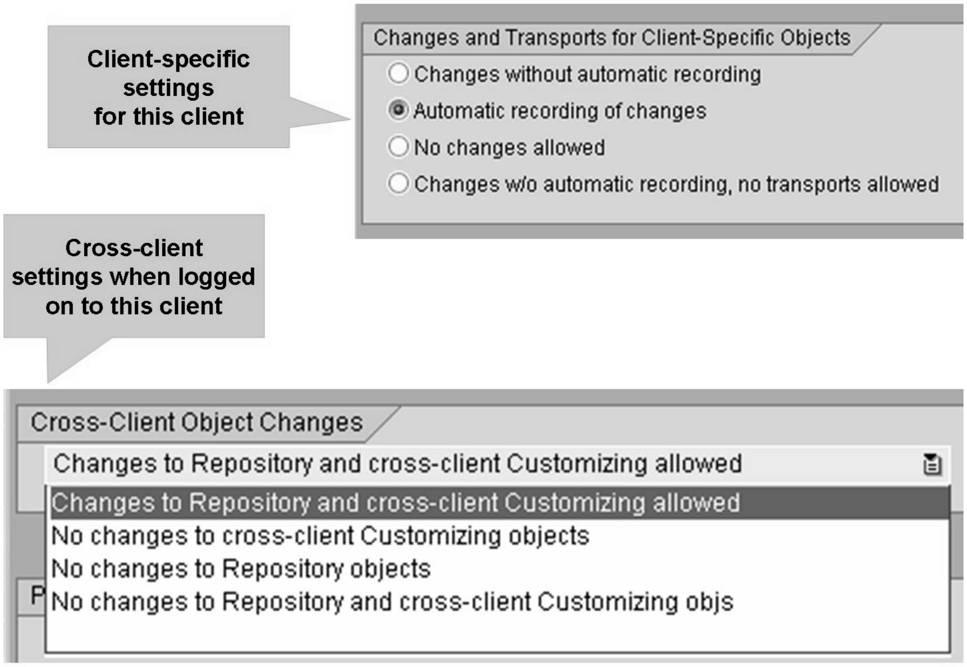
For the software components, there are four different possible settings:

- Modifiable
- Restricted modifiability (can only create objects as non-originals)
- Not modifiable; enhanceable only (changes not allowed; objects can be enhanced using the Enhancement Framework only)
- Not modifiable; not enhanceable (changes and enhancements not allowed)

You can reserve namespaces for software developed by SAP customers and partners. These namespaces are designed for large-scale customer enhancements as well as add-ons developed by partners.

The syntax for objects from reserved namespaces is: */<namespace>/<object name>*. For each SAP system, the customer can define whether or not the objects contained in the namespaces and name ranges can be changed. To enable you to change objects, the SAP system must not be globally locked to changes.

Client Change Options


Client-specific settings for this client
Cross-client settings when logged on to this client

Changes and Transports for Client-Specific Objects

○ Changes without automatic recording
● Automatic recording of changes
○ No changes allowed
○ Changes w/o automatic recording, no transports allowed

Cross-Client Object Changes

Changes to Repository and cross-client Customizing allowed

- Changes to Repository and cross-client Customizing allowed
- No changes to cross-client Customizing objects
- No changes to Repository objects
- No changes to Repository and cross-client Customizing objs

Figure 13: Client Change Options

Client change options that are found in the clients master table *T000*, can be maintained by using transaction SCC4.

The two settings that must be maintained to implement controls on where changes are made and enforce the changes being recorded to change requests are:

- Changes and transports for client-specific objects
- Cross-client object changes

The following list gives a brief description of what each option allows:

- Changes and transports for client-specific objects:
  - *Changes without automatic recording*

This flag means that changes in the Customizing settings of the client are permitted. The changes are not automatically recorded in a change request.

Customizing settings can be exported from the client by manual creation of a transport request. A manual transport can be carried out in the particular transaction for settings with the function *Transport*, which places the selected Customizing settings in a transport request.

This variant makes sense for pure test, training or demonstration clients.

– *Automatic recording of changes*

This flag means that changes are permitted in the customizing settings of the client and all the changes are automatically recorded in a change request.

It is also possible to export independently of the automatic recording from the client customizing settings by manual creation or recording of a transport request. A manual transport can be made in the particular transaction for settings using the function Transport, which places the selected customizing settings in a transport request.

In a productive client, settings that can be maintained as "current settings" are excluded from automatic recording; current settings (e.g. exchange rates, posting periods) are always modified without recording the changes.

This variant can be used for customizing clients and productive clients.

– *No changes allowed*

This flag means that changes in the customizing settings of the client are not permitted (client lock).

However, customizing settings can be exported from the client by manual creation of a transport request. A manual transport can be carried out in the particular transaction for settings with the function Transport, which places the selected customizing settings in a transport request.

In a productive client, the customizing settings that can be maintained as "current settings" are excluded from the client lock; this means that current settings (e.g. exchange rates, posting periods) can always be maintained in the productive client.

This variant can be used for productive clients, demonstration clients and training clients.

– *Changes w/o automatic recording, no transports allowed*

This flag means that changes are permitted in the customizing settings of the client. The changes are not automatically recorded in a change request, however.

No manual transport of Customizing settings from the client is possible.

This variant can be used for pure test clients.

- Cross-client object changes

  This feature controls in which clients you are allowed to maintain cross-client objects.

  It makes sense to disallow cross-client maintenance if the objects to be maintained in a client are exclusively application objects (all client-specific) or client-related Customizing objects. In this case (for example in a test client), you must be able to ensure that cross-client functionality cannot be changed.

  Cross-client functions include all cross-client Customizing objects (e.g. factory calendar, definition of price list conditions, printer controls). But in particular they also include all objects of the SAP Repository (Reports, module pools, screens, Dictionary etc.).

  – *Changes to repository and cross-client customizing allowed*

    There are no restrictions on the maintenance of cross-client objects for the client when this setting is used. Both cross-client Customizing objects and objects of the SAP Repository can be maintained.

    Use in customizing/development client only

  – *No changes to cross-client customizing objects*

    Cross-client Customizing objects cannot be maintained in the client with this setting.

    Only use this if development client is separate from customizing client

  – *No changes to repository objects*

    With this setting, objects of the SAP Repository cannot be maintained in the client.

    Only use this if customizing client is separate from development client.

  – *No changes to repository and cross-client customizing objects*

    Combination of both restrictions: Neither cross-client Customizing objects nor objects of the SAP Repository can be maintained in the client.

    Use this in every client in the landscape except customizing/development client(s)

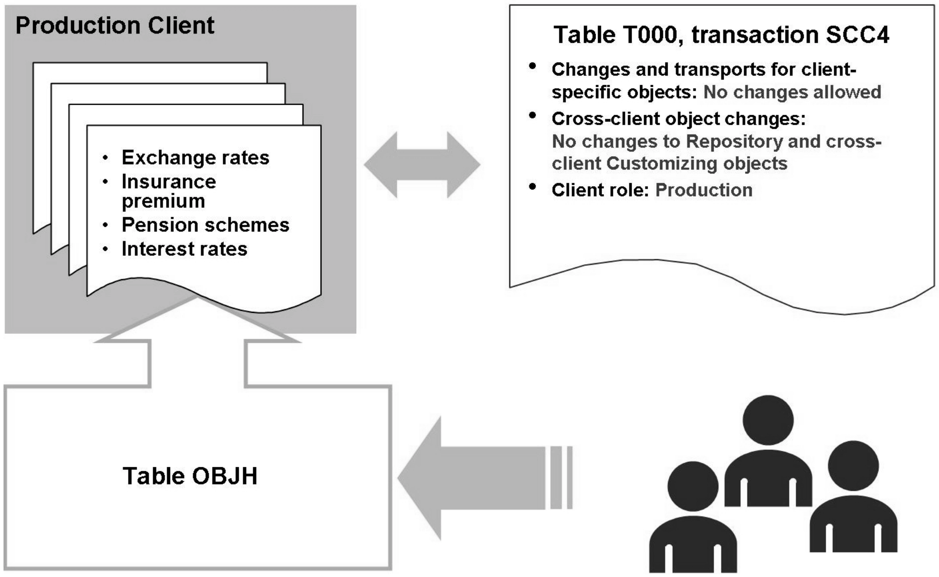

Production Client
Exchange rates
Insurance premium
Pension schemes
Interest rates
Table T000, transaction SCC4
Changes and transports for client-specific objects: No changes allowed
Cross-client object changes: No changes to Repository and cross-client Customizing objects
Client role: Production
Table OBJH

Figure 14: Customizing Using Current Settings

In a customizing project and also during the maintenance of a system all the customizing changes should be transported and tested in the quality assurance system. But certain kinds of customizing changes, known as data-only customizing changes, need to be carried out in a production client without being saved as transport requests. Examples of such data include interest rates, health insurance premiums, pension schemes, tax schemes, and currency exchange rates, which may require frequent adjustment in SAP systems.

Since these types of changes have a frequent change rate and they are not subject to extensive testing like other customizing changes, these changes are done directly in the productive system. To avoid having to use change requests for these changes, SAP has introduced the *Current Settings* function.

Current Settings may be used within a production client without impacting business flow-related Customizing objects.

SAP-approved *Current Settings* for customizing objects are kept in the field *CURSETTING* in the table *OBJH*. SAP recommends that no customer changes be made to the table.

When using *Current Settings* in a production environment:

- Client role is set to: *Production*
- Cross-client object changes are set to: *No changes to Repository and cross-client Customizing objects*
- Changes and transports for client-specific objects are set to: *No changes allowed*.

Lesson Summary

You should now be able to:

- Describe the system change options in an SAP system
- Describe the client change options in an SAP system

Unit Summary

You should now be able to:

- Describe SAP systems, instances and servers
- Identify the system landscape recommended by SAP
- Identify the types of data in an SAP system
- Explain the client concept in an SAP system
- Describe the system change options in an SAP system
- Describe the client change options in an SAP system

Test Your Knowledge

1. List the functions of Transport Management System (TMS).

   ____
   ____
   ____
   ____

2. List examples of client-specific data.

   ____
   ____
   ____
   ____

3. List categories in which data in an SAP system can be divided.

   ____
   ____
   ____
   ____

4. Define a client in an SAP system.

   ____
   ____
   ____
   ____

5. If the system change options are set to *not modifiable* and in the client change options allow the changing of repository objects, you can **not** change repository objects from within this client.

   Determine whether this statement is true or false.

   □ True

   □ False

Answers

1. List the functions of Transport Management System (TMS).

   Answer: The Transport Management System (TMS) organizes, monitors, and performs transports for all SAP systems within a system landscape.

2. List examples of client-specific data.

   Answer: Examples of client-specific data are:

   - User master data.
   - Client-specific Customizing data such as organizational units, assignments, and document types.
   - Application data such as business transaction data and material master data.

3. List categories in which data in an SAP system can be divided.

   Answer: Data in an SAP system can be divided into two categories: client-specific data and cross-client data. Client-specific data, such as user master and application data, affects only one client.

   Cross-client data, such as cross-client customizing data and all Repository objects, affects the whole system environment.

4. Define a client in an SAP system.

   Answer: A client in an SAP system is a self-contained unit in commercial, organizational, and technical terms. A client has its own user master data and set of table key ranges.

5. If the system change options are set to *not modifiable* and in the client change options allow the changing of repository objects, you can **not** change repository objects from within this client.

   Answer: True

   If the system change options are set to *not modifiable* and in the client change options allow the changing of repository objects, you can not change repository objects from within this client.

Unit 2

Setting Up an SAP System Landscape

Unit Overview

Before the development or Customizing can start, it is necessary to configure the Transport Management System (TMS) .

This unit focusses on how to configure the TMS for an SAP system landscape. In addition, you will learn about the implementation of Extended Transport Control and the Quality Assurance (QA) approval controls.

Unit Objectives

After completing this unit, you will be able to:

- explain the steps for setting up the transport management system (TMS)
- configure transport routes
- explain the use of transport target groups and client-dependent transport routes
- list the necessary steps to perform transports between different transport groups and transport domains
- list the prerequisites for CTS domain controller system, source system and target systems that are needed to perform transports of non-ABAP objects
- list the configuration steps to set up a transport landscape for transporting non-ABAP objects
- create non-ABAP systems in transaction STMS and define transport routes between them.

Unit Contents

Lesson: Setting Up the Transport Management System (TMS)

Lesson Overview

In this lesson, you will learn how to set up the transport management system (TMS). At first some basic terms such as system landscape, transport domain and transport group are introduced. Then you will learn how to set up a transport domain and how to configure a transport domain controller. After that, transport routes are defined. Finally the QA approval procedure will be introduced and the TMS setup will be checked.

Lesson Objectives

After completing this lesson, you will be able to:

- explain the steps for setting up the transport management system (TMS)
- configure transport routes

Business Example

An SAP system administrator needs to understand the roles of different systems and clients in the SAP system landscape before making decisions on how to configure the Transport Management System (TMS). After you install the first SAP system but before you make changes, you need to configure the TMS so that it will record, track and control Customizing and development changes for the subsequent transports to other systems in the SAP system landscape.

Overview and Prerequisites

The following list gives an overview of what has to be considered before, during and after installation of an SAP system.

- Before SAP system and database installation:
  - Define the network structure for the system landscape.
  - Install the hardware and the operating system.
- During SAP system and database installation:
  - Create one common transport directory (sometimes you want to have more then one).
- After SAP system and database installation:
  - If you have set up the system using database copy, initialize the Change and Transport Organizer (CTO) (transaction SE06).
  - In any case: set up the Transport Management System (TMS).

Hint: If you have performed a database copy, initialize the Change and Transport Organizer (CTO) using transaction SE06, choosing *Database Copy or Database Migration* and pressing the *Perform Post-Installation Actions* button.

In any case you have to configure the system landscape using the TMS using transaction STMS. The process of setting up the system landscape with transaction STMS will be described in detail in this lesson.

To set up your system landscape, it is sufficient to start with an existing SAP system as a development system; the quality assurance and production SAP systems are not required at this stage. All future systems can be represented at this stage as “virtual systems”. You will also have to create a transport directory at operating system level. This directory is required for the TMS.

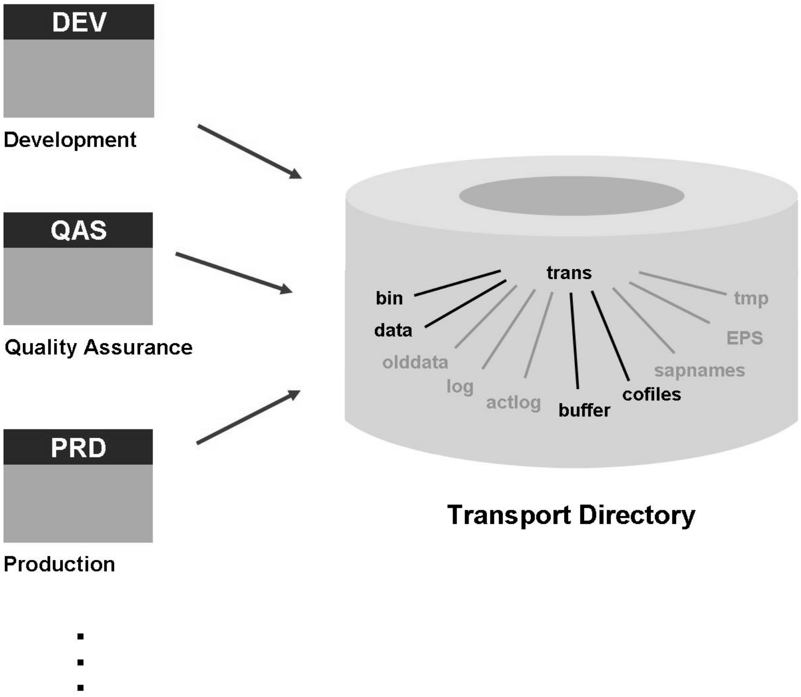


Figure 15: Creating a Transport Directory

Depending on your existing landscape, the global transport directory and all the necessary subdirectories are created automatically during the installation of the SAP system (see the installation guide for your SAP system for more details). The amount of disk space required depends on the amount of development work. Please refer to the installation guide of your SAP system.

The SAP system parameter *DIR_TRANS* has to point to the path of the transport directory. For Unix, the default path is */usr/sap/trans*. For Windows NT, the default path is *\\$(SAPTRANSHOST)\sapmnt\trans* and you must define the transport host with the alias SAPTRANSHOST on operating system level of the domain name server.

Hint: For example, in a Windows environment the corresponding file name is *%WINDIR%\system32\drivers\etc\hosts*.

The subdirectories required in the common transport directory include (among others):

- *bin*: configuration files for tp (TP_<domain name>.PFL) and TMS (DOMAIN.CFG).
- *buffer*: transport buffer for the each system, indicating which transport requests are to be imported.
- *data*: exported data.
- *cofiles*: command or transport request information files that include information on the transport type, object classes, required import steps, and post-processing exit codes.
- *log*: transport logs, trace files, and statistics.
- *tmp*: temporary data and log files.
- *actlog*: action logs for all tasks and requests.
- *sapnames*: information belonging to transport requests for each SAP user.
- *EPS*: download directory for SAP Support Packages.

> **Note:** To enable transports in a heterogeneous operating system environment, you must set up different transport directories. However, to enable transports in a heterogeneous operating system environment using a single transport directory, refer to SAP Note 83327 in the SAP Service Marketplace under the quick link */notes*.

Transporting enables you to synchronize customizing and development in multiple SAP systems through the transfer of changes from the development system to downstream systems. Transports along the transport routes must occur in only one direction.

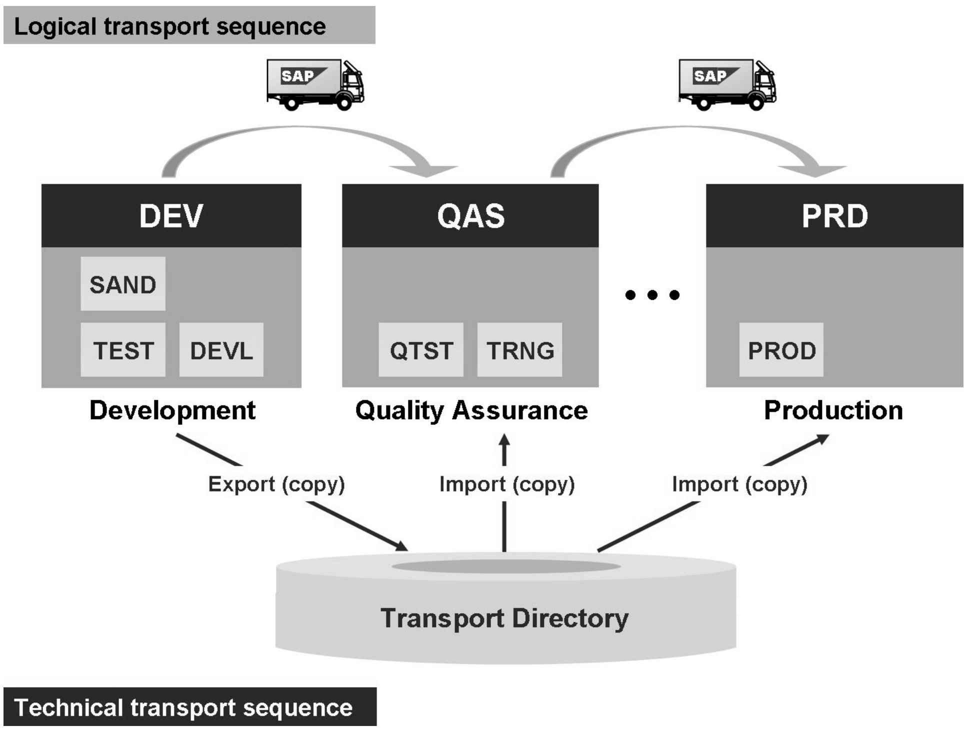


Figure 16: System Landscape Logical and Physical Transport Sequence

As mentioned above, transporting requires a transport directory to enable SAP systems in the system landscape to store and access various data files, command files, and log files.

Physically, changed objects in a three-system landscape are transported in three steps:

1. All objects in a (transportable) transport request that are to be released are exported – by being “copied” – from the database of the source system to the transport directory.
2. These objects are imported into the database of the quality assurance system. Here they are tested and validated.
3. After testing and verification, the objects can be imported into the database of the productive system.

Note: The terms “export” and “import” are meant as copy, not as move, of course. At the end, the objects exist four times: in the database of DEV, QAS and PRD and in the transport directory!

So actually the “truck” is a very bad visualization: because it does a move! A fax machine would be much better...

TMS: Concepts and Terminology

Depending on the size of an organization and the scope of its SAP implementation, there can be many different individuals responsible for carrying out customizing and development projects. Some will be involved in customizing specific application areas, others may be involved in development of new ABAP programs, others in the QA testing and acceptance of changes. There can be hundreds or even thousands of changes that have to be transported through the systems in the landscape. The transport of these changes will be done using the functions of the TMS (TMS).

The concepts behind TMS are:

- Centralized configuration of Change and Transport System (CTS) for all SAP systems
- Centralized management of transport requests and especially the import process
- Transport strategy based on predefined transport routes

The purpose of the TMS, accessed via transaction STMS, is to centrally control the propagation of changes through the system landscape based on predefined paths. This is designed to ensure the consistency of the SAP repository and the contents of the customizing tables in all systems in the landscape. All necessary activities can be done from within the SAP system (using the SAP authorization concept) and there is no need to execute scripts at operating system level by hand.

With TMS you are able to

- Define an SAP system's role within a system landscape or transport domain
- Configure the transport routes using either an editor or delivered standard configurations
- Configure the transport tool program's (*tp*) parameter profile
- Display the import queues of all systems in the transport domain
- Define quality assurance / acceptance procedures in the QA system
- Schedule the import of change requests in an import queue
- Perform transports between systems without a common transport directory
- Handle transport proposals of developers

TMS thus enables system administrators to centrally manage the transport configuration of multiple systems by defining transport domains, assigning domain controllers and defining transport routes.

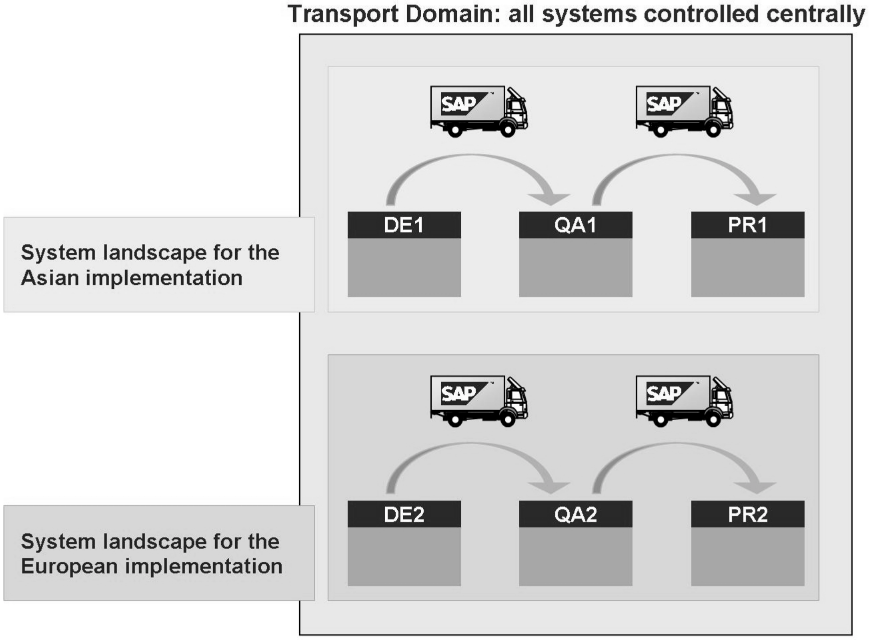

Transport Domain: all systems controlled centrally
System landscape for the Asian implementation
DE1
QA1
PR1
System landscape for the European implementation
DE2
QA2
PR2

Figure 17: What is a System Landscape?

A **transport domain** consists of all systems that you plan to manage using the same TMS. Within the transport domain, all systems must have unique system IDs (SID) and only one of these systems is identified as the (transport) domain controller.

The **transport domain controller** is the system where all TMS configuration settings are maintained. Any changes to the configuration settings are distributed to all systems in the landscape. This ensures that the TMS configuration settings are consistent throughout the domain. The domain controller stores the reference configuration and all other systems receive a copy of the reference configuration.

A **system landscape** is a set of systems that share customizing and repository objects via transported change requests. The typical landscape is made up of, but not limited to, a development, a quality assurance, and a production system. In most cases, the system landscape and the transport domain are made up of the same systems but it is not uncommon to have multiple landscapes within one transport domain.

Examples of a single transport domain with multiple landscapes are:

- A multi-national company may have separate landscapes for each subsidiary. DE1, QA1, and PR1 may be the landscape for the Asian implementation and DE2, QA2, and PR2 may be the landscape for the European implementation. Even though they are separate landscapes, both can still be controlled centrally in one domain.
- An SAP customer may have multiple landscapes for different SAP Business Suite products. DEE, QAE, and PRE may be the landscape for the ECC systems, while DEB, QAB, and PRB may be the landscape for the BI systems. Again, separate landscapes are controlled centrally using one transport domain.

A transport domain contains at least one **transport group**. Very simply, a transport group consists of one or more systems that share a common transport directory. The following figure shows the relation between a transport domain and a transport group.

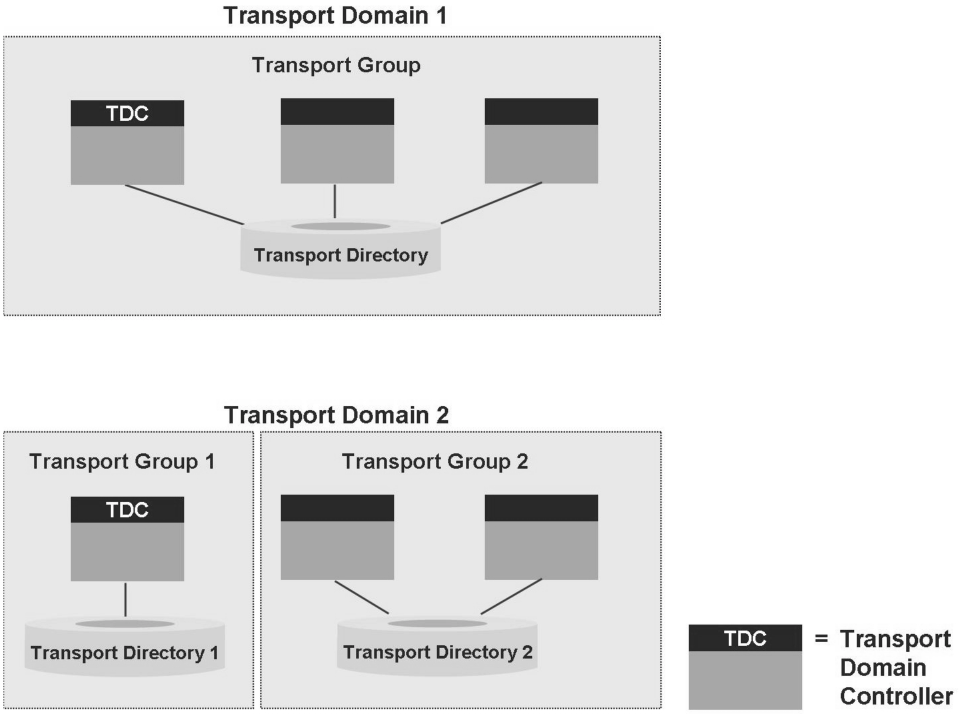


Figure 18: Transport Domain and Transport Group

TMS supports several transport directories within a transport domain. This might be the case e.g. for the multi-national company mentioned above where each subsidiary may have its own (local) transport directory. Or think about a transport domain containing a landscape that has a DEV system with its own transport directory and QAS and PRD system sharing a common transport directory, for security reasons. In this case, the domain would consist of two transport groups.

Note: The terms transport domain, domain controller, and transport group concern only the systems in the environment. They do not include the transport relationship between the systems, which is defined by the transport routes.

Establishing a Transport Domain

To set up the transport domain, first determine which systems should be included in the transport domain. The transport domain should contain all systems in all the system landscapes that will be centrally administered using TMS.

Note: Not all systems need to be installed and present when the domain is first configured but planning what the landscape will look like is necessary. These future systems can be represented by so called virtual systems.

Overview

The configuration of TMS can be broken down into three individual steps.

1. Configuring the transport domain defines which systems will be included in the domain.
2. Configuring the transport routes defines the system and client roles within the landscape(s).
3. Configuring the QA procedure defines who is responsible for approving the acceptance of changes and the promotion of those changes to the delivery system(s).

The following figure shows the three steps. Note that the first two steps are mandatory for setting up a system landscape whereas the last step (QA procedure configuration) is optional.

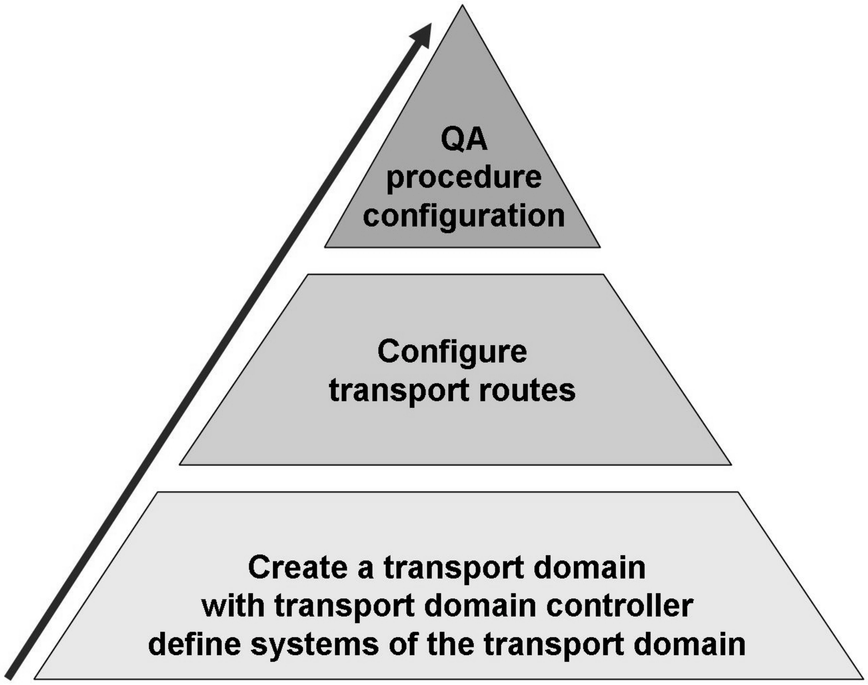

QA procedure configuration
Configure transport routes
Create a transport domain with transport domain controller define systems of the transport domain

Figure 19: Setting Up the Transport Management System

Initializing the transport domain controller

The first system you configure is automatically selected as the transport domain controller but you may later switch the role of the domain controller to a different system.

Transport Domain

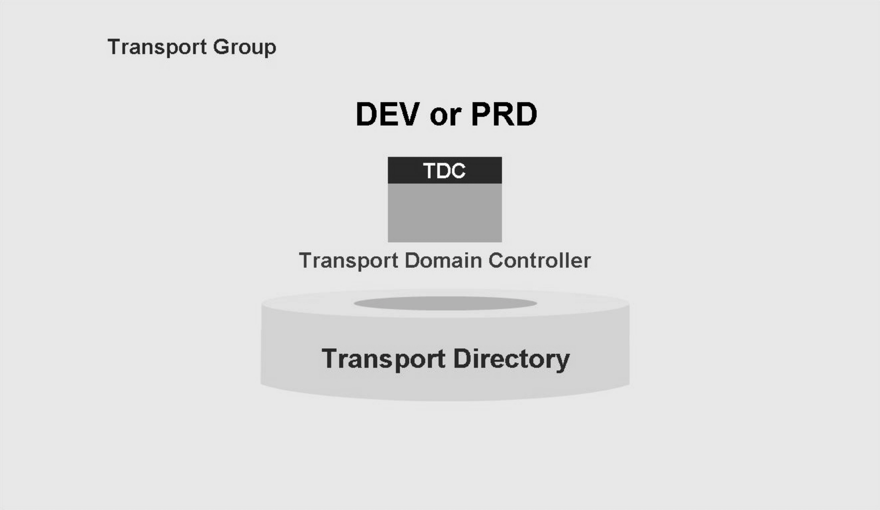


Figure 20: Establishing a Transport Domain

Since certain configuration tasks that are relevant to the entire transport domain (e.g. creating transport routes) can only be carried out on the transport domain controller, SAP recommends that the system chosen to be the domain controller have the following attributes:

- High availability
- High security precautions
- Highest level of maintenance

Therefore, a production system might be the ideal choice to be transport domain controller. Since the development system is usually installed before the quality assurance or production systems, the common practice is configuring the development system as the transport domain controller and then later moving the assignment of the transport domain controller to the production system. Another scenario might be to use a system that you use already for central monitoring and/or central user administration as transport domain controller.

Note: The system load on the SAP system caused by work on the transport domain controller is very low.

When using TMS for the first time after system installation, you are automatically prompted to initialize the TMS. You must initialize the TMS while logged on to client 000. For configuring TMS you need the authorization S_CTS_ADMIN.

- Transaction STMS in client 000 will:
  - Assign the SAP system as the transport domain controller
  - Create transport domain name DOMAIN_<SID>
  - Create transport group GROUP_<SID>
  - Create system user TMSADM in client 000
  - Create RFC destinations
  - Set up file DOMAIN.CFG in directory *bin* of the transport directory

When you initialize the TMS, the following actions will automatically be carried out by the SAP system:

- A transport group is created with the name GROUP_<SID>.
- In client 000, the system user TMSADM is created.
- The RFC destinations required for the TMS are generated.
- The TMS configuration file DOMAIN.CFG is stored in the transport subdirectory *bin*. This file contains the transport domain name and description as well as the transport domain controller's host name, instance number, SID, and transport group.
- The transport profile for the transport control program *tp* is generated and stored in the transport subdirectory *bin* under the name TP_<DOMAIN>.PFL. The parameters in this profile are maintained using the transaction STMS.

Note: The name of the transport domain may not contain blanks and may not be changed afterwards without reconfiguring the domain controller. By default, a transport domain will have the name DOMAIN_<SID>, where <SID> is the system ID of the domain controller. The name of the transport domain cannot be changed afterwards.

Adding additional systems to a domain

When other systems that will be part of an existing domain are ready for inclusion in the transport domain, you can extend the transport domain by adding these systems. To add new systems to a transport domain, you must perform configuration activities on both the new system and the transport domain controller.

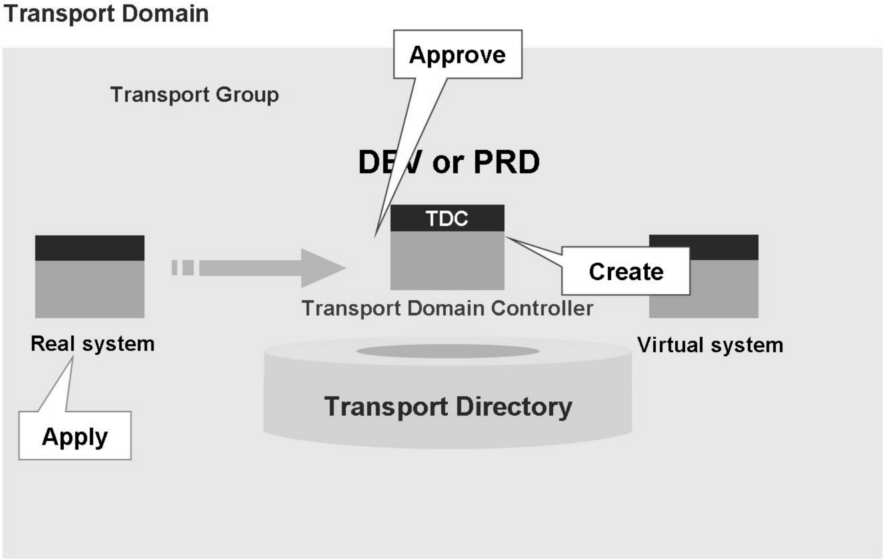


Figure 21: Adding Additional SAP Systems to a Domain

On the new system being added, the configuration process is known as requesting inclusion. Since configuration of the TMS only occurs on the domain controller, the domain controller would then have to approve the new system in the domain. Each additional SAP system must apply for inclusion to the transport domain controller. The transport domain controller then must confirm inclusion for each SAP system. The initial screen of transaction STMS indicates if an SAP system is waiting for inclusion in the transport domain.

After you have included an SAP system from one of the transport groups in the transport domain, the description of the transport domain is stored in the file DOMAIN.CFG in the *bin* subdirectory of the common transport directory. All other SAP systems connected to this transport directory read the file DOMAIN.CFG during STMS initialization to identify the domain controller and transport group address. If your SAP systems do not have a common transport directory, you can manually configure TMS with the address of the domain controller. In this case choose *Other configuration → Include system in domain* when entering transaction STMS for the first time. In the next popup, enter the target host and the system number of the transport domain controller. In any case, when an SAP system has been configured for the TMS and approved in the domain, you will find a newly created user TMSADM and generated RFC destinations required for the TMS.

Hint: When configuring the TMS on an SAP system, you can specify the application server (i.e. the instance) to be used for all TMS functions. Choose the instance with the highest availability. This is generally the instance that provides the enqueue service, the central instance.

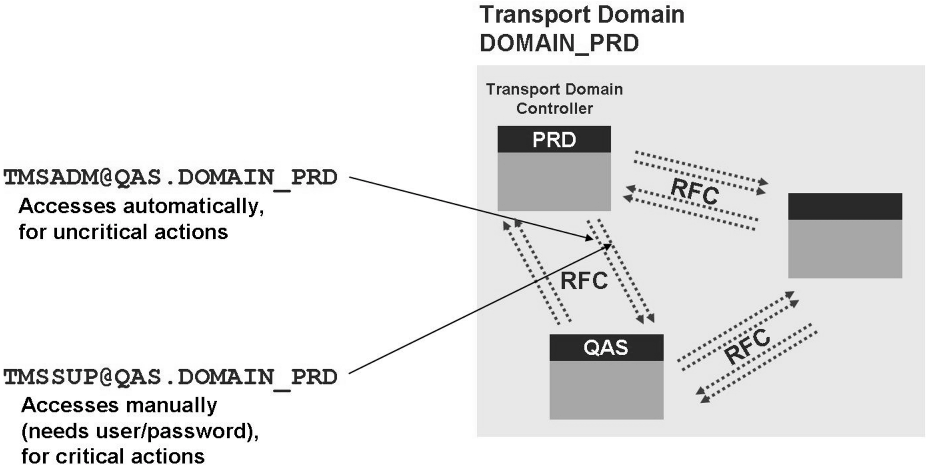

Transport Domain
DOMAIN_PRD
Transport Domain Controller
PRD
RFC
RFC
RFC
QAS
TMSADM@QAS.DOMAIN_PRD
Accesses automatically, for uncritical actions
TMSSUP@QAS.DOMAIN_PRD
Accesses manually (needs user/password), for critical actions

Figure 22: Centralized Configuration Data

The systems within a transport domain communicate with each other using Remote Function Calls (RFC). RFC communication requires user IDs to access target systems. When systems are added to a transport domain, the necessary RFC destinations and user IDs are automatically configured by the TMS tool. The domain configuration settings are distributed throughout the domain using RFC communication.

Changes to the transport domain configuration are made in the transport domain controller, and then distributed to all SAP systems in the transport domain. Each time you make a change to the domain controller, a dialog box is displayed asking whether to distribute the change. You can distribute several changes at once.

When systems are initially added to the TMS configuration, the communication user TMSADM is configured with limited authorizations. This user is utilized when distributing and activating the TMS configurations.

There are two types of RFC destinations created by the TMS:

TMSADM@<SID>.<domain name>

TMSSUP@<SID>.<domain name>.

The user TMSADM is inserted as system user into the TMSADM@<SID>.<domain name> destination but there is no RFC user inserted into the TMSSUP@<SID>.<domain name> destination. Therefore, if you try to execute a function that goes beyond the limited authorizations of the TMSADM user, for example, scheduling an import, the system uses the TMSSUP destination. As a result, the system prompts you for a user name and password on the target system that does have the necessary authorizations.

Virtual systems

Extending a transport domain is not restricted to physically installed systems. Virtual systems are often included as place holders for planned systems and are replaced by the planned system after it is physically installed. In addition, you can extend the transport domain to include external systems, for example, an SAP system from a different transport domain.

By creating virtual systems, you can model the transport routes of the planned system landscape and ensure that the import queues of subsequent systems will already be created and maintained for later use. In the beginning stages of a new SAP implementation, customers frequently have only the development system physically installed and store the development and customizing work in the import queues of the respective planned systems.

Backup Domain Controller

If the SAP System acting as the domain controller fails, no changes can be made to the TMS configuration. SAP therefore recommends that you configure a backup domain controller that can assume the function of the domain controller when required.

It may also be necessary to move the transport domain controller to another SAP system. In such a case, you must activate the backup domain controller. Then, the backup domain controller becomes the domain controller and vice versa.

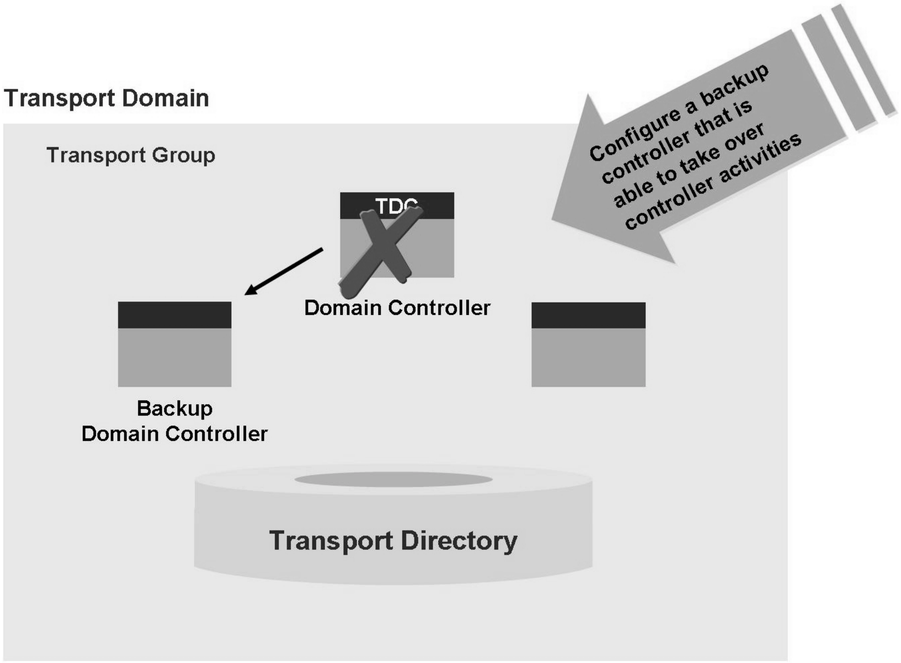


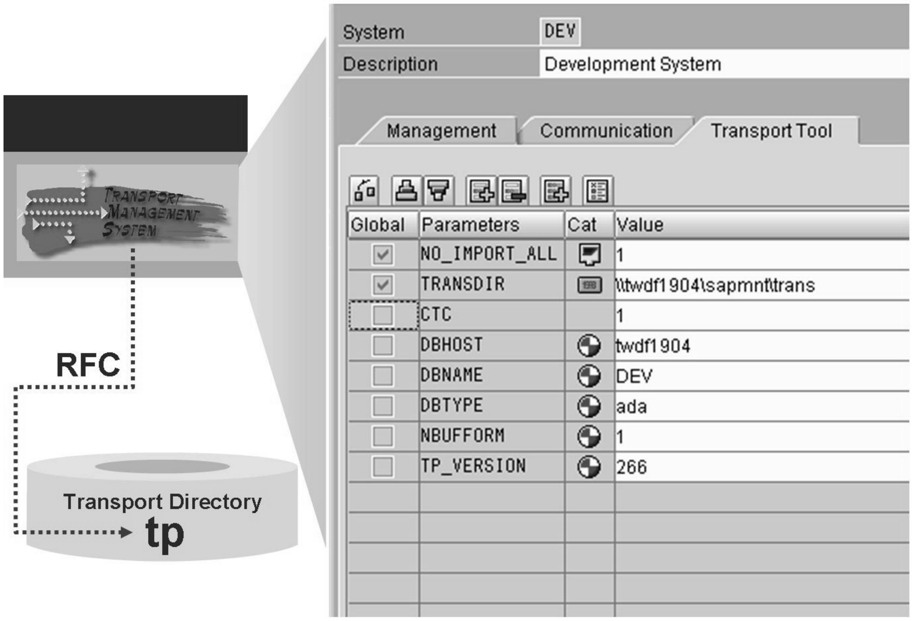
Figure 23: Transport Domain Management using a Backup Domain Controller

Hint: The system you select as a backup domain controller must be an existing installed system; it cannot be a virtual or external system.

To check the current status of the transport domain configuration for each SAP System in the transport domain, access the TMS overview. From the TMS initial screen, choose *Overview → Systems*. In this overview, you can also see whether the configuration is up to date, and whether any errors occurred when distributing the configuration.

Configuring tp

The transport control program tp requires a transport profile that contains information about establishing the database connection for all SAP Systems in the transport domain. TMS generates and manages this transport profile as a part of the transport domain configuration. Do not adjust the transport profile using a text editor at operating system level.



System DEV

Description Development System

Global	Parameters	Cat	Value
☑	NO_IMPORT_ALL	▭	1
☑	TRANSDIR	▭	\\twdf1904\sapmnt\trans
□	CTC		1
□	DBHOST	●	twdf1904
□	DBNAME	●	DEV
□	DBTYPE	●	ada
□	NBUFFORM	●	1
□	TP_VERSION	●	266

Figure 24: Configuring the Transport Control Program tp

To display the tp parameters of an SAP System, call transaction STMS. Choose *Overview* → *Systems*. Mark one system and choose *SAP System* → *Display*. Choose the tab *Transport Tool*. From the menu, choose *Goto* → *tp parameters*. This displays the parameters in TP_<domain name>.PFL and the default value of all parameters used by the programs *tp* and *R3trans*. If a parameter is not specified, a default value will be used. Global parameters override defaults. Local parameters override global parameters. For example, you can specify a global parameter value for all but one system if a local parameter applies to that system.

You can switch between the views *Display All Parameters* and *Display Substituted Parameters Only* (content of TP_<domain name>.PFL). You can switch by using the *Display Less* and *Display Less* button.

Note: In releases up to R/3 4.6C there was the configuration file *tpparam*. This is not used anymore. Although it was more or less ok, to maintain the *tpparam* with an text editor, this is not ok anymore for the *TP_<domain name>.PFL*

After performing an upgrade from R/3 4.6C to ECC 6.0, for example, you might want to transfer your settings from the old to the new configuration file. You can do this in the parameter overview by choosing *Extras -> Copy Configuration from TPPARAM*.

Configuring Transport Routes

Transport routes indicate the role of each system and the flow of change requests. The transport routes are what actually define your system landscape.

Overview

Initializing the TMS and setting up the transport domain define the physical systems in the environment only in terms of the transport domain, the domain controller, and the transport group. Next, you must define the transport relationship between each of these systems. Although the TMS has been initialized, you cannot perform transports until the transport routes have been configured and distributed.

After establishing a transport domain, you need to:

1. Model transport routes from the transport domain controller, using:
   - Default standard configurations (one-, two-, and three-system landscapes)
   - Graphical editor for non-standard configurations
2. Distribute and activate the new configuration data for all SAP systems within the transport domain

To reduce the effort of specifying individual transport routes, you can use standard configurations. Transport routes for the standard configurations are generated automatically.

You can choose from the following standard configurations:

Single system

Development and Production System

Three Systems in Group

If you use the standard configuration function, the current configurations for all SAP systems involved are replaced by the standard settings. Existing packages, transport layers, or objects created in the SAP system are not deleted. For complex system landscapes, define additional consolidation and delivery routes after setting up the transport route configuration on the basis of one of the standard configurations.

After you define a transport route, you must activate and distribute it to all systems in the transport domain.

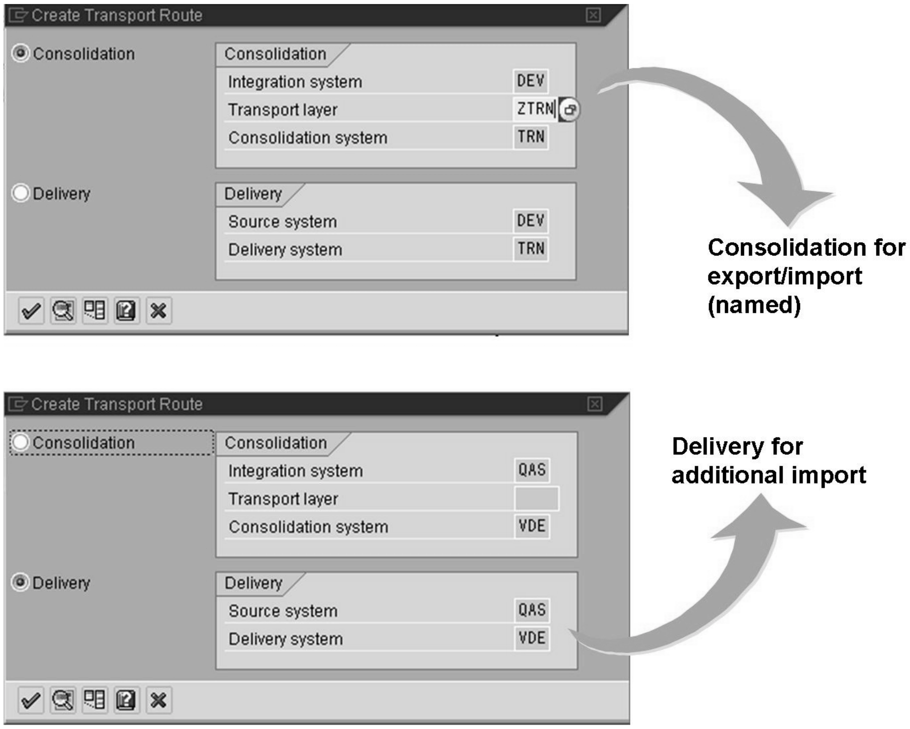
There is a version management of the stored configurations. If you modify and save an active configuration, the version counter increases. To activate a stored version, from the start screen of transaction STMS choose *Overview → Transport Routes →*

Configuration → Get Other version. This displays a list of all versions of the transport route configuration. Select the version you want from the list. To activate an older version, use the distribute and activate procedure.

To ensure consistency, transport routes can be configured only on the domain controller. To help define a transport route, the TMS provides a graphical editor and a hierarchical list editor, which can be used interchangeably, as well as standard configurations for a one-, two-, and three-system landscape.

Transport Layers and Transport Routes

As previously stated, transport routes define the flow of the change requests from one system to the next. These routes are called either consolidation or delivery routes.



Create Transport Route

● Consolidation

Consolidation	
Integration system	DEV
Transport layer	ZTRN
Consolidation system	TRN

○ Delivery

Delivery	
Source system	DEV
Delivery system	TRN

Consolidation for export/import (named)

Create Transport Route

○ Consolidation

Consolidation	
Integration system	QAS
Transport layer	
Consolidation system	VDE

● Delivery

Delivery	
Source system	QAS
Delivery system	VDE

Delivery for additional import

Figure 25: Transport Routes

A **consolidation route** is an "export / import" route. Typically, the consolidation route proceeds from the development system (where the transport request is exported from) to the quality assurance system (where the transport requested is imported into) in a standard three-system landscape. Consolidation routes have names in order to be addressed.

A **delivery route** is an “another import” route. In a standard three-system landscape, the delivery route is specified between the quality assurance system and the production system because there is no additional export from the quality assurance system but another import in the production system. Delivery routes have no names.

All repository objects are grouped into logical units called packages (formerly: development classes). The definition of each package contains an assignment to a transport layer. The transport layer is the name of a consolidation route. Objects, via the assignment to the package, inherit this transport layer assignment.

All SAP delivered objects are assigned according to their packages to the transport route *SAP*.

Customizing objects (and others) are not grouped into packages. They follow the standard transport route. This standard transport route is typically named *Z<SID of the development system>*.

In the context of transport routes, an SAP system may play the following roles:

Integration system

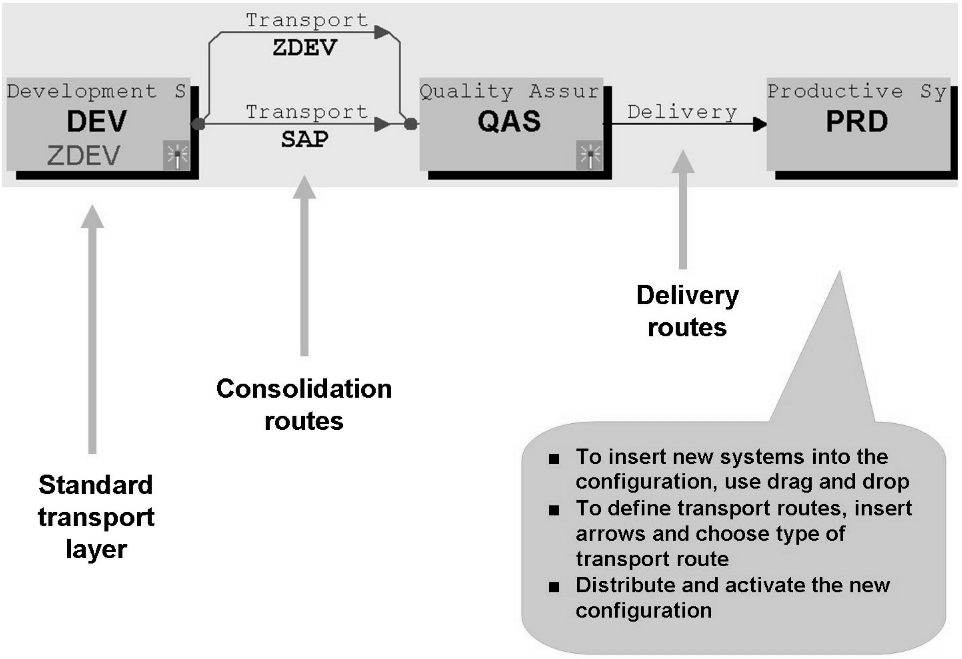
The system where changes originate and are assigned to transport requests. The system where customer changes are integrated with the SAP standard, customer developments and modifications take place.

Consolidation system

The target system of a consolidation route.

Delivery system

The target system of a delivery route.



- To insert new systems into the configuration, use drag and drop
- To define transport routes, insert arrows and choose type of transport route
- Distribute and activate the new configuration

Figure 26: Three System Landscape Using a Standard Configuration

You need to configure more transport layers and transport routes if you have a more complex landscape and you need to reroute certain objects away from the standard transport routes. For example, if a separate training system exists and there are certain programs that are to run there but you do not want those programs getting into the quality assurance or production systems.

If the standard configurations delivered by SAP do not meet your landscape needs, TMS has editors available to allow you to maintain the landscape.

The graphical editor delivers a visual drag and drop interface, where the systems in the domain can be positioned and linked using the mouse. This is the default editor in the transaction STMS. It is shown in the following figure.

To create transport routes, use the mouse to drag the appropriate SAP systems from the node area, and drop them in the display area. To create a transport route between these SAP Systems, from the screen *Display Transport Routes*, choose the *Add Transport Route* button. The mouse pointer becomes a stylus. Using the stylus, draw a line connecting the two SAP systems between which a transport route is desired. In the *Create Transport Route* popup, enter the required information for either the consolidation or delivery route. In case of a consolidation route you need to assign a transport layer. You can create the transport layer right here. Save your changes. The following figure shows a more complex landscape.

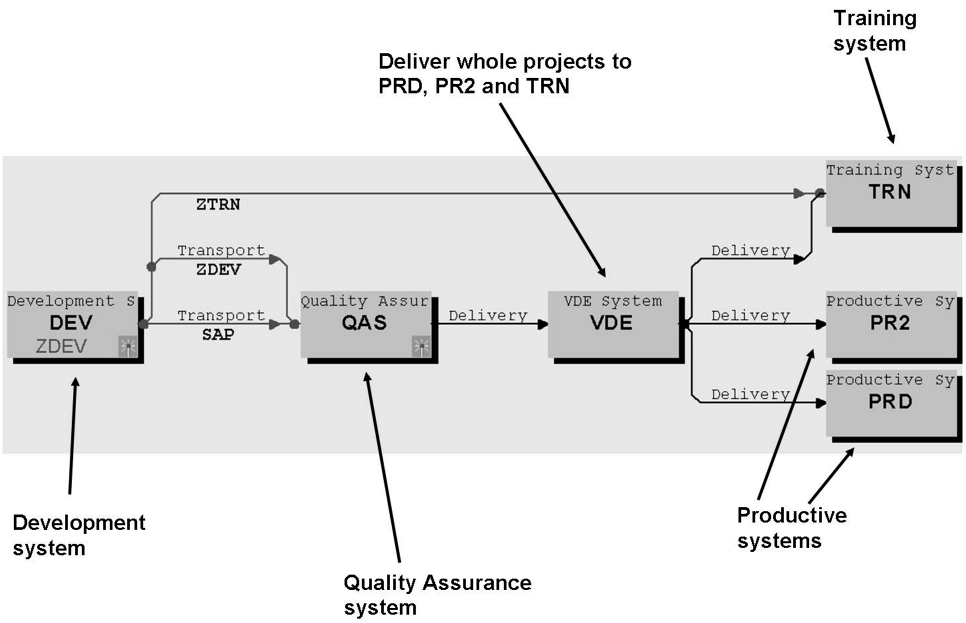


Figure 27: More complex system landscape

Distribution and Activation of Transport Routes

Changes in transport route configuration are not valid until distributed and activated. To distribute and activate the changes from the TMS editor of the transport domain controller, choose the *Distribute and Activate* icon and activate or follow *Configuration* → *Distribute and Activate*. For activation the RFC connections are used, that are created during the setup of the transport domain.

To avoid any possible inconsistencies, especially if there are any released transport requests waiting in any import queues, TMS performs a number of consistency checks on the version of the transport routes being activated, for example, whether the settings of the transport control program *tp* are consistent with the transport route configuration. The activation also triggers checks to verify that new transport routes do not affect existing transport requests. If any problems exist, the configuration will not be activated and you will receive an error message indicating the nature of the error.

Note: Once you have configured the transport routes correctly, the only reason that changes would need to be made would relate to a change in your system landscape, for example, adding a new system or new clients to the landscape.

Configuring QA Approval Procedure

When a transport request is released and exported from the development system (DEV), the consolidation system's (QAS) import buffer is populated. The import buffer is a list of transport requests waiting for import. Once the transport request is successfully imported into the consolidation system, the import buffer for all delivery systems (PRD) is populated.

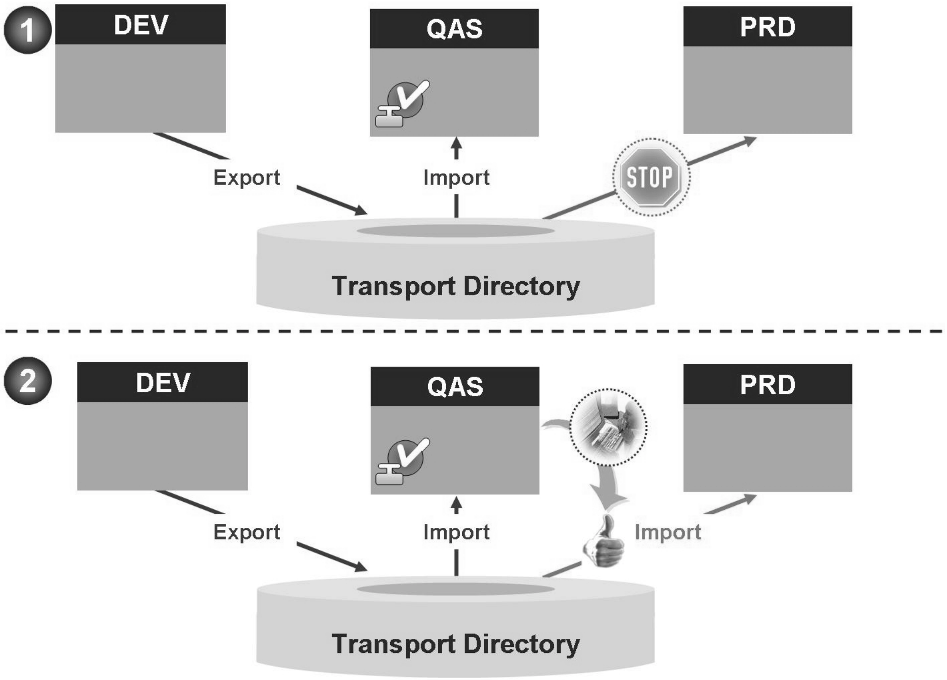
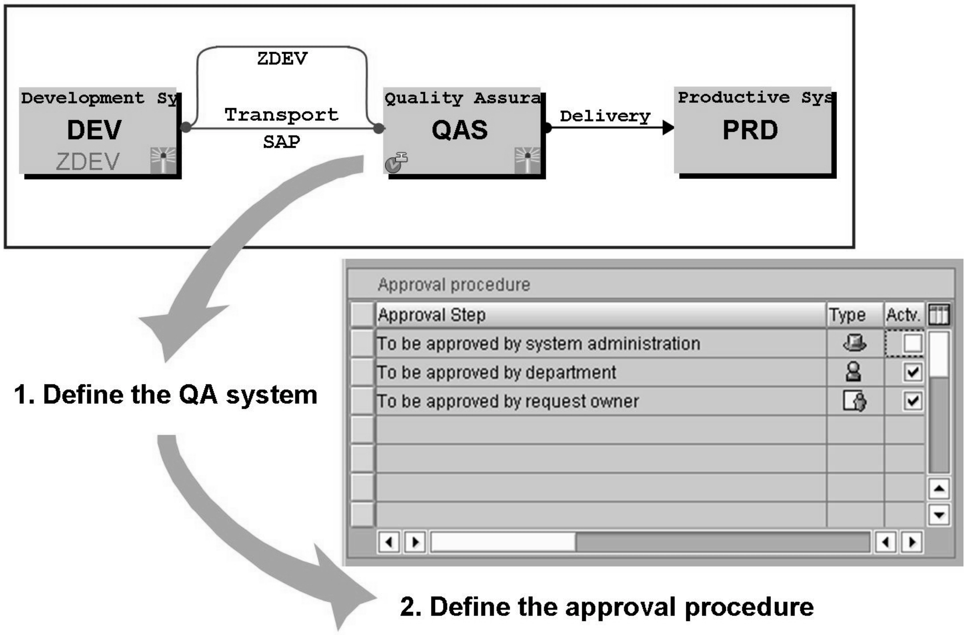


Figure 28: QA Approval Procedure

With the QA Approval Procedure , the buffer of the delivery systems is populated but the entries are flagged as inactive. In other words, the transport requests cannot be imported until the responsible person(s) flags the transport request as approved, which results in activating the entries in the delivery systems' buffers.

TMS Quality Assurance increases the quality and the availability of the production systems by letting you check transport requests in the QA system before they are delivered to subsequent systems. The system for which the QA approval procedure is activated is called the QA system. When the QA approval procedure is activated, a transport request will only be imported into the delivery system(s) if all the QA approval steps are processed in the QA system and the request has been approved. When you configure the QA system, you determine how many QA approval steps have to be processed for each request.



Approval procedure

Approval Step	Type	Actv.
To be approved by system administration		□
To be approved by department		☑
To be approved by request owner		☑

Figure 29: Configuring the QA Approval Procedure

The figure above shows you how to activate the QA approval process from the graphical editor and how to define the approval steps.

Verifying the TMS Setup

Once you have configured the TMS, there are some tests that you can perform to verify that there are no technical problems with the TMS configuration:

- RFC connection test
- Transport directory check
- Transport control program (tp) check

Because the domain controller communicates with the other systems in the domain using RFCs, you should check the RFC connections between the systems in the domain. To do this, from the *Systems* screen of the STMS, select a system from the list and choose *SAP system → Check → Connection Test*. This verifies communications in both directions – to and from the domain controller.

Since exports physically write files to the operating system (OS) level and imports read and update these files, you should verify that the transport directory is available and has the appropriate OS level permissions. To do this, from the *Systems* screen of the STMS, choose *SAP system → Check → Transport Directory*. This action creates, reads, and deletes a file to each of the transport directory subdirectories and returns a report on the success or failure of all subdirectories. If you receive any errors, the appropriate corrections would need to be made at the operating system level.

Because exports and imports are physically executed by the operating system level program *tp*, you should verify that the tp program and its parameter configuration are consistent. To do this, from the *Systems* screen of the STMS, follow *SAP system → Check → Transport Tool*. This executes tests verifying the *tp* interface, the transport profile, the RFC destinations, and connection to the databases and returns a report of the results.

Exercise 1: Setting Up the Transport Management System (TMS) (verification)

Exercise Objectives

After completing this exercise, you will be able to:

- Review the transport domain configuration
- Review transport routes
- Verify the technical setup of the Transport Management System (TMS)

Business Example

As an SAP administrator you want to change the configuration of your transport domain. Before doing this, it is necessary to verify the existing configuration.

Task 1: Transport Domain

Review the transport domain configuration.

1. Determine the name of the transport domain that includes the development system DEV.

   ➡ **Note:** Log on with your user in client 100 of system DEV.

2. Which systems are included in the domain?
3. Which system is the domain controller? Is there a backup domain controller configured? Are there any virtual systems configured?
4. Do all systems belong to the same transport group?

Task 2: Transport Routes

Review transport domain and transport routes.

1. View the current transport domain and its transport routes.
2. What transport layers and transport routes have been defined for this system landscape? List the system(s) that are considered consolidation system(s)? Which system(s) are considered delivery system(s)?

Continued on next page

Task 3: QA Approval Procedure

Verify the configuration of the *QA approval procedure*.

1. Is the *QA approval procedure* activated? Which system is configured as the QA system? Who is responsible for approving/accepting change requests?

Result

The approval steps *To be approved by request owner* and *To be approved by department* are activated.

> **Note:** To be identified as system administration, you need the authorization for the value *TADM* for the authorization object *S_CTS_ADMI* (*CTS_ADMFCT* field). By default, this is contained in the authorization *S_CTS_ADMIN*.

Task 4: Checking TMS configuration

Verify the technical configuration of the TMS.

1. The domain controller uses RFCs to communicate with the systems in the domain. Verify the RFC communications between QAS and the domain controller.
2. Exports physically write files to the file system level directories and imports read and update these files. Verify that the directory is available to the DEV and QAS systems and has the appropriate file system level permissions.
3. Exports and imports are physically executed by the operating system level program *tp*. Verify that the *tp* program and its parameter configuration is consistent and that it can access the DEV and QAS databases.

Solution 1: Setting Up the Transport Management System (TMS) (verification)

Task 1: Transport Domain

Review the transport domain configuration.

1. Determine the name of the transport domain that includes the development system DEV.

   Note: Log on with your user in client 100 of system DEV.

   a) Execute transaction STMS.

      This screen displays:

      - The name of the system you are logged on to, in this case DEV
      - The name of the transport domain: for example: DOMAIN_DEV

2. Which systems are included in the domain?

   a) In transaction STMS choose the *System Overview* icon or choose *Overview → System*.

      The systems DEV, QAS, VDE, PRD, PR2 and TRN.

3. Which system is the domain controller? Is there a backup domain controller configured? Are there any virtual systems configured?

   a) Examine the column displaying the system type (Column heading *Typ* or *TMS System Type*).

   b) Either highlight the icons or select the *Legend* icon and compare the icon status.

      The development system DEV is defined as the transport domain controller. The quality assurance system (QAS) is, if defined, the backup domain controller. The systems VDE, PRD, PR2 and TRN are virtual systems.

Continued on next page

4. Do all systems belong to the same transport group?

   a) From the *System Overview* screen of transaction STMS, choose *Goto → Transport Groups.*

      This displays a hierarchical list of the TMS configuration sorted by the SID within the group within the domain. There is only one domain, for example DOMAIN_DEV.

      There is at least one group, for example GROUP_DEV, which consists of the systems DEV, QAS, PRD, and, if defined, other systems.

Task 2: Transport Routes

Review transport domain and transport routes.

1. View the current transport domain and its transport routes.

   a) Execute transaction STMS.

   b) Choose the *Transport Routes* icon or choose *Overview → Transport Routes*.

      Hint: If the display is difficult to read, choose the *Zoom In* icon.

Continued on next page

2. What transport layers and transport routes have been defined for this system landscape? List the system(s) that are considered consolidation system(s)? Which system(s) are considered delivery system(s)?

   a) On the graphical display, the lines connecting the systems represent the transport routes.

      - The standard transport layer ZDEV exists as a consolidation route from DEV to QAS for the transport of customer objects.

        Hint: You know ZDEV is the standard transport layer because it displays in the DEV system box.

      - The transport layer SAP exists as a consolidation route from DEV to QAS for the transport of SAP-delivered objects.
      - The transport layer ZTRN exists as a consolidation route from DEV to TRN for the transport of special customer objects that should not follow the standard transport layer.
      - There are delivery routes from QAS to VDE, from VDE to PRD, from VDE to PR2 and from VDE to TRN.

      QAS and TRN are considered consolidation systems, because they are target systems of a consolidation route.

      VDE, PRD, PR2 and TRN are considered delivery systems, because they are target system of a delivery route.

Continued on next page

Task 3: QA Approval Procedure

Verify the configuration of the *QA approval procedure*.

1. Is the *QA approval procedure* activated? Which system is configured as the QA system? Who is responsible for approving/accepting change requests?

   a) Execute transaction STMS. Choose the *Transport Routes* icon or choose *Overview → Transport Routes.*

   b) Notice the *Stamp of Approval* icon in the lower left corner of the QAS system. This identifies that the QA approval procedure is activated and QAS is the QA system.

   c) Double click the QAS system. Notice that the *Delivery after confirmation* check box is activated.

   d) Choose *Procedure* to display the configured approval steps.

Result

The approval steps *To be approved by request owner* and *To be approved by department* are activated.

Note: To be identified as system administration, you need the authorization for the value *TADM* for the authorization object *S_CTS_ADMI* (*CTS_ADMFCT* field). By default, this is contained in the authorization *S_CTS_ADMIN*.

Continued on next page

Task 4: Checking TMS configuration

Verify the technical configuration of the TMS.

1. The domain controller uses RFCs to communicate with the systems in the domain. Verify the RFC communications between QAS and the domain controller.

   a) Execute transaction STMS. Choose the *Systems Overview* icon or choose *Overview → Systems.*

   b) Choose the QAS system. Choose the *Connection Test* icon or follow *SAP System → Check → Connection Test*.

   If the systems can communicate using RFCs, you will receive the message *Connection to System QAS (host <hostname>) OK*.

   Hint: If you marked the whole column in the *System Overview* screen, the test would be performed on every real system of the transport domain.

2. Exports physically write files to the file system level directories and imports read and update these files. Verify that the directory is available to the DEV and QAS systems and has the appropriate file system level permissions.

   a) Execute transaction STMS. Choose the *Systems Overview* icon or choose *Overview → Systems.*

   b) Choose the DEV system. Choose *SAP System → Check → Transport Directory*.

   The system attempts to write a file to each subdirectory of the transport directory, read the file, and delete the file. The returned screen displays the results of this test. A green check mark indicates that the test executed successfully. Expand the display list to see the individual tests.

   c) Repeat the test for the QAS system.

   Hint: If you marked the whole column in the *System Overview* screen, the test would be performed on every real system of the transport domain.

Continued on next page

3. Exports and imports are physically executed by the operating system level program *tp*. Verify that the *tp* program and its parameter configuration is consistent and that it can access the DEV and QAS databases.

   a) Execute transaction STMS. Choose the *Systems Overview* icon or choose *Overview → Systems.*

   b) Choose the DEV system. Choose *SAP System → Check → Transport tool*.

      The system runs some consistency and connection tests on the transport program *tp*. The returned screen displays the results of this test. A green check mark indicates that the test executed successfully. Expand the display list to see the individual tests.

   c) Repeat the test for the QAS system.

      Hint: Again, if you marked the whole column in the *System Overview* screen, the test would be performed on every real system of the transport domain.

Lesson Summary

You should now be able to:

- explain the steps for setting up the transport management system (TMS)
- configure transport routes

Related Information

You find additional information concerning the setup of the TMS in the online documentation for SAP NetWeaver 7.0. Choose *SAP NetWeaver → Solution Life Cycle Management → Software Change Management → Change and Transport System → Transport Management System (BC-CTS-TMS)*.

The following SAP notes might be helpful:

- SAP Note 556734: FAQ Transport: Setup and further information
- SAP Note 28781: Central transport directory NT/UNIX
- SAP Note 83327: Setting up transport system in heterogeneous system group

Lesson: Extended Transport Control

Lesson Overview

In this lesson, you will learn about extended transport control. This covers the use of both transport target groups and client-dependent transport routes. Transports between transport groups and between different transport domains are also discussed.

Lesson Objectives

After completing this lesson, you will be able to:

- explain the use of transport target groups and client-dependent transport routes
- list the necessary steps to perform transports between different transport groups and transport domains

Business Example

As a system administrator you are asked to ensure that certain development objects or customizing objects can be imported into more than one subsequent system or into more than one specific client of a Quality Assurance (QA) system.

Extended Transport Control

Using SAP standard transport control it is not possible to transport a transport request from one development system into multiple QA systems in an easy way. The reason for that is that it is not possible to create two transport routes with the same transport layer from one SAP system. As every repository object is assigned (via a package) to a certain transport layer, it follows that a transport request can only have one target system.

Transport Target Groups

The solution for the problem stated above is to create a transport target group, which may lead to one or to more SAP system(s).

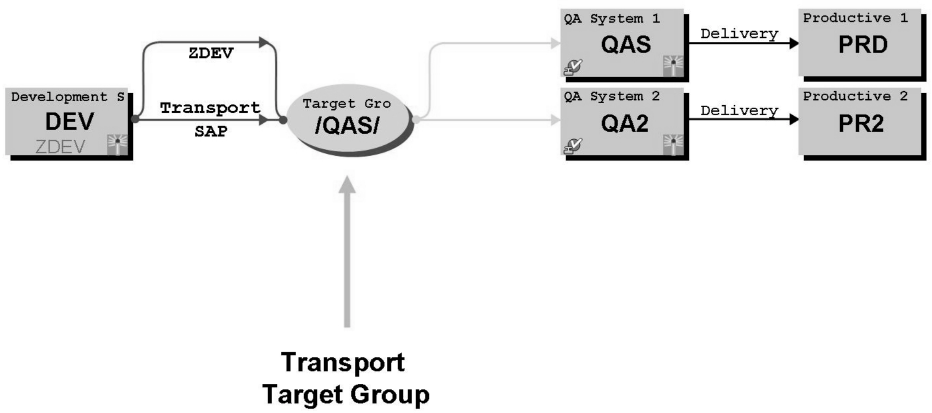


Figure 30: Transport Target Groups

To create a transport target group, use transaction STMS and choose *Overview → Transport Routes* and from there choose (using the change mode) *Edit → Transport Target Group → Create*.

Note: The name of the transport target group must start and end with “/”.

When a transport request, which has a transport target group as target, is released, it will fill the import queue(s) of all systems in this transport target group. In the figure above, all released transport requests that contain objects pointing to the transport layer ZDEV are ready for import into both the QAS system and the QA2 system.

Client-Specific Transport Control

Because some landscapes contain multiple clients in the development and quality assurance systems, it is a challenge for the TMS administrator to maintain consistent (client-dependent) customizing across the landscape. Different systems and different clients within a system may need to receive changes at different times, depending on quality assurance approval and acceptance procedures. Communication errors between customizing project leaders and TMS administrators can inadvertently cause inconsistencies in the configuration settings of certain clients.

Because the majority of customizing is client-specific, during the scheduling of an import process, the import scheduler prompts the TMS administrator for a target client. The administrator would need to manually schedule the imports for the different clients, possibly based on instructions from customizing project leaders, and would also have to keep track of which change requests had and had not been imported into which clients.

TMS offers the Extended Transport Control (also known as client-dependent transport control, CTC) whereby the administrator can automate the process by:

- Client-specific transport target groups.
- Client-specific consolidation routes.
- Client-specific delivery routes.

Client-specific transport targets: the transport targets of consolidation and delivery routes do not just specify a system, they also specify a client. Client-specific transport targets are entered in the form: <SID>.<client> (for example, QAS.100). Transport target groups combine several client-specific transport targets under a symbolic name. You can specify transport target groups when you define consolidation and delivery routes. To differentiate them from traditional transport targets, you must start and end the names with "/" (for example, /EUROPE01/).

Note: In the context of client-independent transport routes, transport target groups were already discussed in the previous subsection. The concept of transport target groups and client dependent transport routes can easily be combined.

Client-specific consolidation routes: For each transport layer, the consolidation routes determine where changes made in the system are transported to after the transport request has been released. If you have activated extended transport control, then the transport target can be a specific client in a target system or a transport target group. If you do not activate extended transport control, the transport administrator has to specify the correct target client at the time of import.

Client-specific delivery routes: Delivery routes determine whether transport requests are to be flagged for import into subsequent systems/clients, after they have been imported into a system. If you have activated extended transport control, then you can set the delivery routes as client-specific. This makes it possible to supply several clients in one system in sequence. You can also specify a target group as the target of a delivery route.

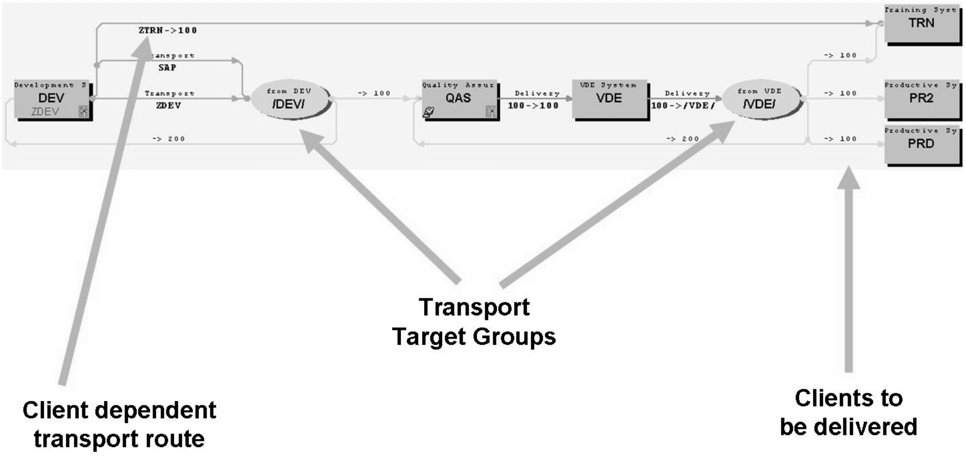


Figure 31: Extended Transport Control using Client Dependent Transport Routes

In the figure above, objects pointing to the transport layer ZDEV will be imported (via transport target group /DEV/) into QAS client 100 and DEV client 200. Imagine that client 200 in DEV is a client for functional tests. After QA approval in the QAS system (client 100), the objects are forwarded to VDE client 100. From here the approved transport requests are forwarded (via the transport target group /VDE/) to PRD client 100, PR2 client 100, TRN client 100 and QAS client 200.

Extended transport control makes daily transport tasks easier and increases security. Extended transport control also reduces the need for communication between project leaders and system administrators, because the transport routes can now be configured completely. No additional details about the target client need to be given at the time of import.

To take advantage of this function, in the transport program profile, you must explicitly set the (tp) parameter *CTC* (client dependent transport control) to TRUE (value of 1). The default value is FALSE (value of 0), which deactivates the extended transport control. This is done automatically, when choosing a client-specific transport target group.

Caution: After activating the CTC, you can use the normal system-to-system transport or client-specific transport routes, but not a mixture of both types of connections in the same system landscape. When using client-specific transport routes, you must specify the clients when defining the source and target systems.

The standard transport layer (Z<SID of the development system>) determines the default transport target of the requests. When you use Extended Transport Control, you can set a different standard transport layer for individual clients from that set in the SAP system. This means that you can forward customizing requests from different clients into different transport targets. The client-specific standard transport layer is also the default transport layer for new packages that have been created in a client. If you accept this default, then the cross-client objects that have been created in cross-client customizing are transported along the same route as the corresponding client-specific customizing.

Note: This scenario would contradict SAP's recommendation that all customizing/development changes originate in one single client. Client-specific transport layers should only be used in circumstances, for example, as shown in the figure above.

Transport between Transport Groups and Transport Domains

If you configure different transport groups in your system landscape, there will be the need to perform transports between these different transport groups. If there is more than one transport domain in your company, there might be the need to perform a transport from one transport domain to another domain.

Transports between different Transport Groups

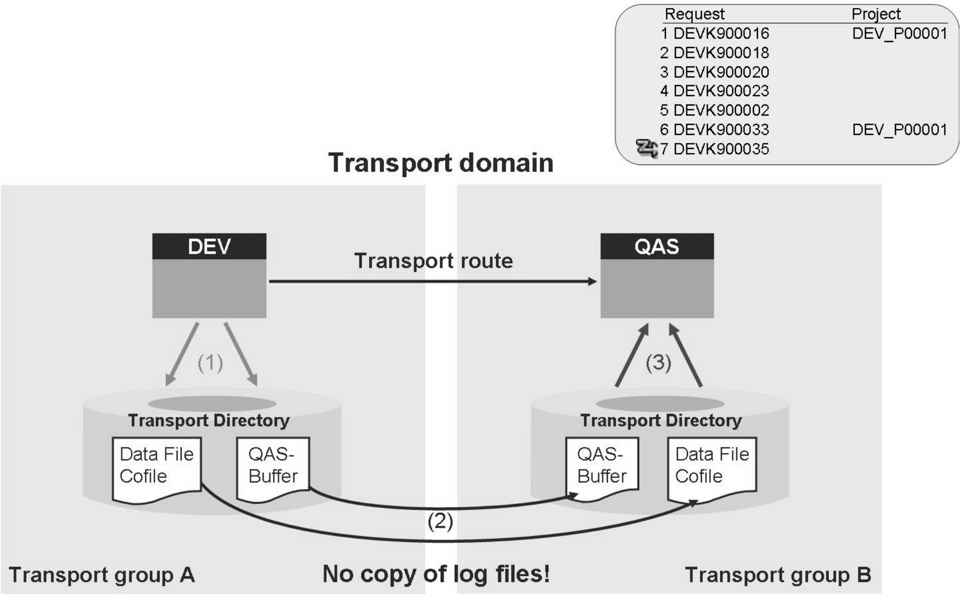
Usually all the SAP systems in a transport domain share the same common transport directory. There are situations, however, in which separate transport directories are set up for different parts of the same domain.

Multiple common transport directories are used, for example, when:

- an SAP system is connected to the domain through a slow or expensive network connection,
- strict security measures prevent allowing direct file system access from other systems.
- dissimilar hardware platforms prevent the use of a common transport directory.

Within a transport domain, SAP systems that share a common transport directory form a transport group. The TMS supports transports between transport groups. The following figure outlines the procedure on how to transport change requests between different transport groups.

(1)	**DEV**, Transport Organizer:	Export of transport request
(2)	**QAS**, Import Queue:	Extras → Other Requests → Find in Other Groups
(3)	**QAS**, Import Queue:	Import of transport request



Request	Project
1 DEVK900016	DEV_P00001
2 DEVK900018	
3 DEVK900020	
4 DEVK900023	
5 DEVK900002	
6 DEVK900033	DEV_P00001
⚡7 DEVK900035	

Figure 32: Transport between Transport Groups

After a transport request has been released from DEV, the request is marked for import into the target system. Be aware that this happens in the transport directory of the DEV system (step 1). If the source and target systems are in different transport groups, however, the import queue of the target system must be adjusted from the *Import Queue* screen in the target system group choosing *Extras* → *Other requests* → *Find in Other Groups* (this corresponds to step 2 in the figure above). TMS searches for requests for the selected SAP system in the import buffers (on operating system level) of all transport groups in the transport domain, and transfers the data files and cofiles belonging to the requests. Before the transfer of the data file, the transport request is marked in the import queue with a “lightning” icon that disappears after adjusting the import queue of the target system.

Note: There are limitations in regard to transporting between different transport groups:

- The transport logs displayed are specific to the transport group of the SAP system you are using, i.e. they are not copied to the other transport directory.
- Transports displayed in the Change and Transport Organizers are also specific to the transport group of the SAP system you are using.

Transports between different Transport Domains

If you have configured multiple transport domains and want to make transports between systems in different domains, you can use domain links to link the two domains or external systems to transport between different transport domains.

- Domain link (needs a permanent network connection)
- External systems

For a **domain link** there must be a permanent network connection between the systems in the two domains, similar to the connection between systems within the same domain. The transport domain controller systems in both domains must have SAP_BASIS 4.6C, or higher.

Linking two domains with a domain link involves two steps: First one transport domain controller has to request a link between the two domains. Then the transport domain controller of the other transport domain has to confirm the link between the two domains.

To request a domain link, enter transaction STMS and choose *Overview → Systems*. In the system overview choose *SAP System → Create → Domain Link.* Enter the system name, host name and instance number of the transport domain controller you want to link to, and then confirm your entries. The SAP system generates the required RFC destinations automatically and sends the address data of the local transport domain controller to the transport domain controller of the other domain.

To confirm a link between two transport domains, enter transaction STMS and choose *Overview → Systems*. In the *System Overview* position the cursor on the domain controller where you requested the domain link and choose *SAP System → Approve*. Confirm the prompt and distribute the configuration.

After you have established a domain link,

- you can make transports between systems in different domains in the same way as you make transports between systems in different transport groups; RFC is used to transfer transport files between the transport directories involved.
- you can display transport logs of systems in the other domain.

Note: Transport domains are independent administrative units. You can only make changes (such as transport route changes) in the transport domain controller of the domain. Transport routes also are not distributed between domains. You can, however, configure a route between systems in different domains, but must configure it twice, once in each domain. The domains that are linked by a domain link must have different names.

If you cannot operate a permanent RFC connection between systems in the two domains, you can use **external systems** to make transports between the two domains.

External systems are like virtual systems. However, for this type of system, an separate transport directory is also defined. This directory can either be located on a disk partition that can be accessed by an SAP system in the other domain, or it is on a replaceable data medium, for example a DVD or portable hard disk.

The idea behind external systems is to reduce the problem of transporting between **different** transport domains to the problem of transporting between different transport groups in the **same** transport domain.

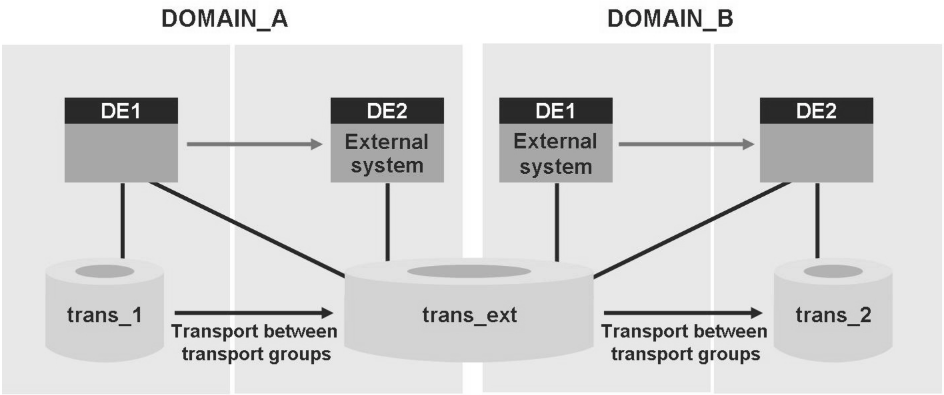


Figure 33: Transport between Transport Domains using External Systems

In the figure above, you want to transport between the systems DE1 in *DOMAIN_A* and DE2 in *DOMAIN_B*. In transport domain *DOMAIN_A*, you create an external system called DE2 (pointing to the transport directory *trans_ext*). In transport domain *DOMAIN_B*, you create an external system called DE1 (pointing to the transport directory *trans_ext* as well). In addition in both transport domains you need to define a transport route between these two systems. You must also configure the SAP servers of the system DE1 so that they can access the directory *trans_ext*. In the same way, the system DE2 in the domain *DOMAIN_B* must be able to access the directory *trans_ext*.

After you export a transport request from DE1 to *trans_1*, you perform a transport between transport groups in *DOMAIN_A*. As a result, data files, cofiles and the buffer entries for DE2 exist in *trans_ext*. Next, you perform a transport between transport groups in *DOMAIN_B* so that all necessary data is sent to the transport directory *trans_2*. Then you can perform the import into DE2.

To configure an external system, log on to the SAP system that functions as the transport domain controller and call transaction STMS. In the system overview that appears choose *SAP System → Create → External System*. Enter the path and a short description of the transport directory. You also need to enter the transport directory path for the specific platform and relative to the communications system. The transport domain controller is proposed as the communications system. Save your entries and confirm the distribution prompt.

Lesson Summary

You should now be able to:

- explain the use of transport target groups and client-dependent transport routes
- list the necessary steps to perform transports between different transport groups and transport domains

Related Information

You find additional information concerning the setup of the TMS in the online documentation for SAP NetWeaver 7.0. Choose *SAP NetWeaver* → *Solution Life Cycle Management* → *Software Change Management* → *Change and Transport System* → *Transport Management System (BC-CTS-TMS)*.

Lesson: Configuration of Change and Transport System For Transporting Non-ABAP Objects

Lesson Overview

As well as ABAP objects, the Change and Transport System (CTS) also lets you transport Java objects (J2EE, JEE) and SAP-specific non-ABAP technologies (such as Web Dynpro Java or SAP NetWeaver Portal) in your landscape. The enhanced CTS functions are available as of Support Package Stack (SPS) 12 of SAP NetWeaver 7.0.

This lesson describes how to configure the Change and Transport System for transporting non-ABAP objects. First the prerequisites for the CTS and the runtime systems are mentioned. Then the configuration steps that are needed to set up the transport domain controller accordingly are listed. Next the creation of non-ABAP systems and the definition of transport routes between them are discussed. This lesson ends with an appendix about the configuration of various runtime systems for the use with the enhanced CTS.

Note: Please note that this lesson uses terms and transactions that have not yet been explained in former lessons of this class but that are well known from preceding classes (SAPTEC, ADM100). The instructor will not have the time to explain all these terms and transactions in this class.

Lesson Objectives

After completing this lesson, you will be able to:

- list the prerequisites for CTS domain controller system, source system and target systems that are needed to perform transports of non-ABAP objects
- list the configuration steps to set up a transport landscape for transporting non-ABAP objects
- create non-ABAP systems in transaction STMS and define transport routes between them.

Business Example

Your company uses the Change and Transport System (CTS) to perform transports in transport landscapes based on SAP systems with AS ABAP. In addition your company is performing some portal development and some development on Exchange Infrastructure (XI) systems.

As a member of the transport administration team you want to know if you are able to transport portal content files (EPA files) and XI transport files (TPZ files) with the help of CTS as well.

Introduction

In an SAP system you have the possibility to develop in ABAP, in the J2EE/JEE standard, or to use SAP-specific non-ABAP technology, such as Web Dynpro Java, or the SAP NetWeaver Portal. This is not a fight one technology versus the other but different approaches to a solution for business needs. You choose from these options based on your preferences, knowledge that is already available in one of the technologies, or with respect to specific advantages of one technology in certain areas. Therefore, in any bigger landscape you will find many or even all of these objects. When you, for example, use the SAP NetWeaver Portal to provide users with role-based access to a new functionality in your ERP backend system, you have to synchronize the update of both the Portal and the ERP runtime, which even might be connected to other systems using the SAP NetWeaver Exchange Infrastructure.

This was simplified a lot when you use the SAP NetWeaver Development Infrastructure (NWDI for short) with its transport capabilities, the easiest way, however, is to manage such transports centrally in one system.

So there are tools for ABAP & JAVA transports available, but

- there was no automatic synchronization for mixed applications (such as XI) so you needed to use different tools to transport parts of the same application
- there was no central control of Portal transports (in addition only rudimentary export/import tools were available that were neither integrated into NWDI nor into CTS)
- there was no central control of all transports into productive systems

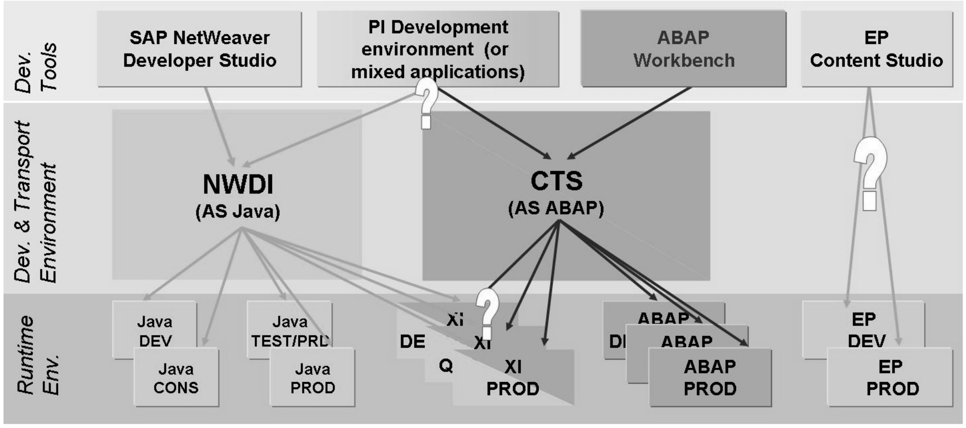


Figure 34: The Present Transport Scenario – Different Tools for ABAP and Java Transports

With the enhancements of the SAP NetWeaver Change and Transport System (CTS for short, or enhanced CTS respectively, if specifically the enhanced capabilities are meant), available with SP Stack 12 of SAP NetWeaver 7.0 and further enhanced with SP Stack 13, the CTS has been enhanced to handle any type of object that can be transported in an SAP system landscape.

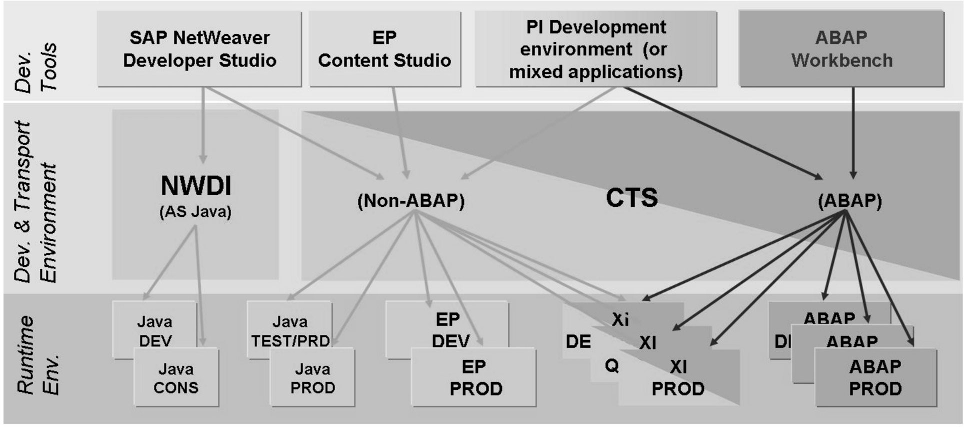


Figure 35: The New Option – Using CTS Combined with NWDI

Therefore, an administrator of your ABAP system can now also manage the transports for the non-ABAP parts of your system landscape. By combining & synchronizing transports in a comprehensive approach, the enhanced Change and Transport System therefore solves transport issues.

Architecture

The following figure shows an example of a system landscape for the use with the enhanced Change and Transport System.

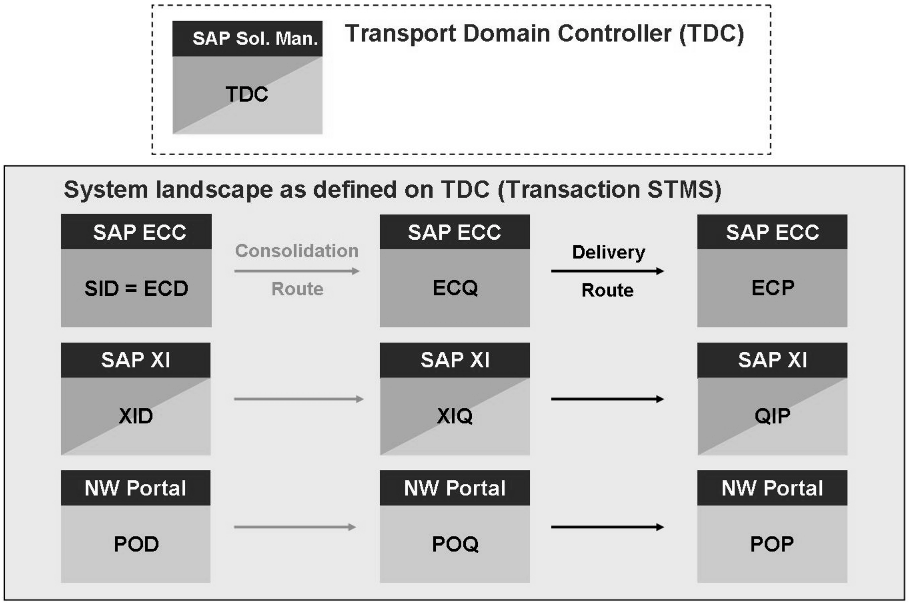


Figure 36: Example for a System Landscape with the Enhanced CTS

In this example, the Transport Domain Controller (TDC) is located on an Solution Manager System which is an ABAP+Java System. In addition, on this system, the CTS Deploy Web Service is configured.

Hint: A different option is to use an AS ABAP as TDC and to connect this system to a different AS Java (different SID) on which the CTS Deploy Web Service is configured.

Within transaction STMS on the TDC you can define both ABAP and non-ABAP systems. The system landscape shown here contains of a 3 system landscape for SAP ECC systems. In addition it contains a 3 system landscape of SAP NetWeaver Portals (Usage Type EP) and a 3 system landscape for SAP Exchange Infrastructure systems (Usage type PI).

Note: In the figure above, we assume that the XI systems are based on SAP NetWeaver 7.0, SP Stack 12 or higher. Otherwise the SAP XI landscape would consist of a three system landscape for the AS ABAP part and a different 3 system landscape for the AS Java part.

System Requirements

In order to use the enhancements of the Change and Transport System, both the Transport Domain Controller / the systems based on AS ABAP + Java on which the CTS Deploy Web Service is running and the runtime systems must fulfil certain requirements.

Note: In this lesson, the term CTS-System refers to SAP Systems based on AS ABAP + Java on which the CTS Deploy Web Service is configured and will be used.

Transport Domain Controller System / Systems with CTS Deploy Web Service

This list contains the system requirements which must be fulfilled for the Transport Domain Controller (TDC) system / the systems based on AS ABAP + Java on which a CTS Deploy Web Service is running (CTS-System).

- AS ABAP part
  - At least AS ABAP 7.00 with SP Stack 12 (SP stack 13 recommended)
  - Latest tp / R3trans version
  - Connection to CTS Deploy Web Service must be configured
  - System must be configured as Transport Domain Controller in Transaction STMS
  - If applicable: Non-ABAP-Systems and transport routes between these systems must be defined
- AS Java part
  - AS Java 7.00 with SP Stack 12 or higher (SP Stack 13 recommended)
  - CTS Deploy Web Service must be configured and running

If the TDC is an AS ABAP+Java system (as shown on the figure before), all requirements must be fulfilled on this system. If the TDC is an AS ABAP only, the AS Java part must be fulfilled on a connected AS Java system (with different SID).

Hint: Note that the CTS Deploy Web Service belongs to the software component *DI_CMS*, so in addition this software component needs to be deployed on the AS Java part of every CTS system. It is intended however to switch this Web Service to a software component that belongs to the AS Java itself with a future enhancement package.

Runtime Systems

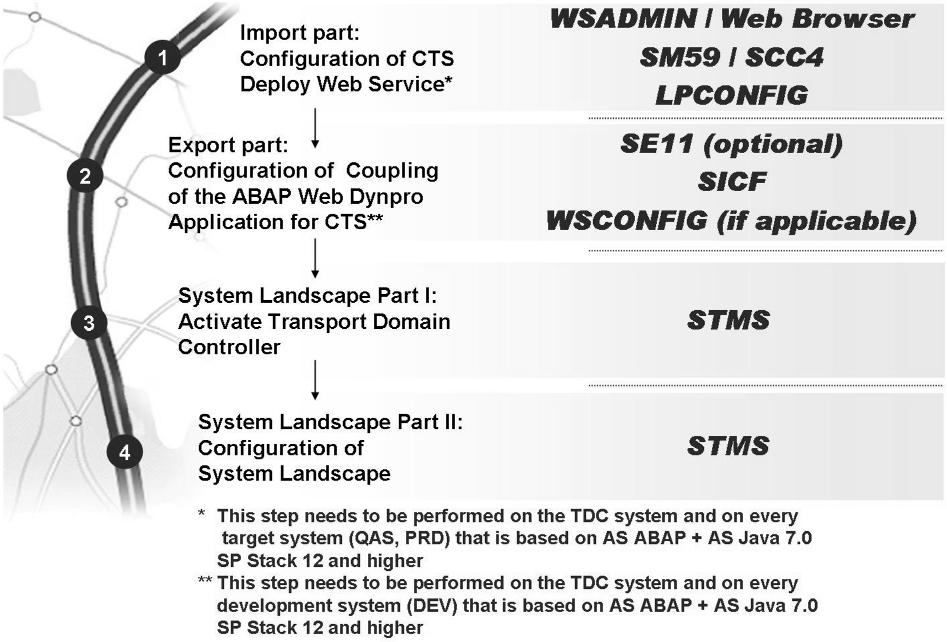
In addition to the CTS systems,, the runtime systems (that means the systems that are included in the STMS system landscapes) must fulfil certain requirements. These requirements are as follows:

- SAP Systems based on AS ABAP
  - No specific release is required (CTS compatibility is sufficient)
- SAP Systems with Usage Type EP
  - Minimum release SAP NetWeaver '04 SP Stack 09
  - Configuration / Transport Scenario depends on Release / SP Stack (changed with SAP NetWeaver 7.0 SP Stack 13 and higher)
- SAP Systems with Usage Type PI
  - Minimum release SAP NetWeaver '04 SP Stack 13
  - Configuration / Transport Scenario depends on Release / SP Stack (simplified with SAP NetWeaver 7.0 SP Stack 12 and higher)
- SAP Systems with Usage Type DI (NWDI Systems)
  - No specific release required
  - Simplified with SAP NetWeaver 7.0 SP Stack 12 or higher

Note: The configuration of the runtime systems with usage type PI and the corresponding transport scenario was simplified as of SAP NetWeaver 7.0 SP Stack 12. Also, the transport scenario for runtime systems with usage type EP changed was simplified as of SAP NetWeaver 7.0 SP Stack 13 provided that the TDC system also is at least on this SP stack level.

Configuration of the Transport Domain Controller System / Systems with CTS Deploy Web Service

The following overview lists the main configuration steps that are necessary on the Transport Domain Controller (TDC) system / systems based on AS ABAP + Java with CTS Deploy Web Service (CTS system).



1 Import part:
Configuration of CTS
Deploy Web Service*

WSADMIN / Web Browser
SM59 / SCC4
LPCONFIG

2 Export part:
Configuration of Coupling
of the ABAP Web Dynpro
Application for CTS**

SE11 (optional)
SICF
WSCONFIG (if applicable)

3 System Landscape Part I:
Activate Transport Domain
Controller

STMS

4 System Landscape Part II:
Configuration of
System Landscape

STMS

* This step needs to be performed on the TDC system and on every target system (QAS, PRD) that is based on AS ABAP + AS Java 7.0 SP Stack 12 and higher

** This step needs to be performed on the TDC system and on every development system (DEV) that is based on AS ABAP + AS Java 7.0 SP Stack 12 and higher

Figure 37: Configuration of the CTS Systems– Main Steps

Note that the CTS Deploy Web Service runs on the AS Java part of the CTS system but it must be configured on the AS ABAP part of the CTS system because the AS ABAP part needs to know how to connect to this service.

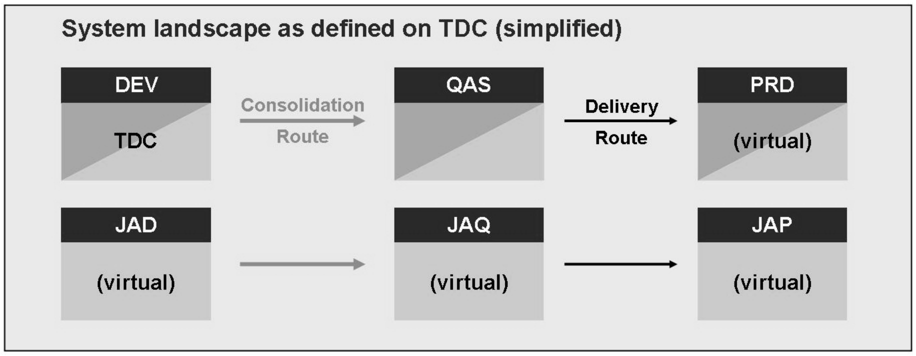
Hint: This CTS Deploy Service needs to be configured on every system on which the tp program triggers the SDM deployment. For target systems based on AS Java (only) this will be the Transport Domain Controller system. For target systems based on AS ABAP + Java 7.0 SP Stack 12 or higher however, tp can be called on the system itself.

All these four main steps will be discussed in more detail in the following subsections.

Note: In step 2, transaction WSCONFIG will only be used for the *SODIS_CORE_WBO* ABAP Web Dynpro application. This transaction is no longer needed for CTS systems as of SP Stack 13 for SAP NetWeaver 7.0 (in case of Usage Type DI: as of SP Stack 14 for SAP NetWeaver 7.0).

The System Landscape in this Class as an Example

The following figure points out the system landscape (simplified) in this class.



System ID	Configuration of Export Part	Configuration of Import Part	Remarks
DEV	✓	✓	Communication system for JAD, JAQ, JAP
QAS		✓	
PRD			Virtual system
JAD, JAQ, JAP			Systems based on AS Java only

Figure 38: Example for System Landscape in this Class (simplified)

In this scenario, there is a three system landscape DEV – QAS – PRD of systems based on AS ABAP + Java 7.0 SP Stack 12 or higher. The DEV system plays the role of the development system and the transport domain controller. The PRD system up to now is a virtual system. And there is a three system landscape JAD – JAQ – JAP based on (virtual) AS Java only. These three systems will have the DEV system as a communication system (in transaction STMS, area *System Overview*).

In this case, the import part needs to be configured on the QAS system as there will be imports later into the QAS system. In addition the import part needs to be configured on the DEV system as well because DEV plays the role of the communication system for the JAQ and JAP system and the import into JAQ for example will therefore use tp and the CTS Deploy Web Service on the DEV system. As PRD is a virtual system, nothing needs to be configured on this system so far.

The export part needs to be configured on the DEV system only because this system acts both as development system for the DEV – QAS – PRD system landscape and as communication system for the JAD system so that transport request for JAD (transport requests JADK9<number>) will be created with the help of the Web UI that is located on the DEV system.

Import part: Configuration of CTS Deploy Service

The following roadmap shows the necessary steps to configure the CTS Deploy Web Service on the CTS system. The CTS Deploy Service connects the ABAP transport tools to the deployment tools in the non-ABAP applications.

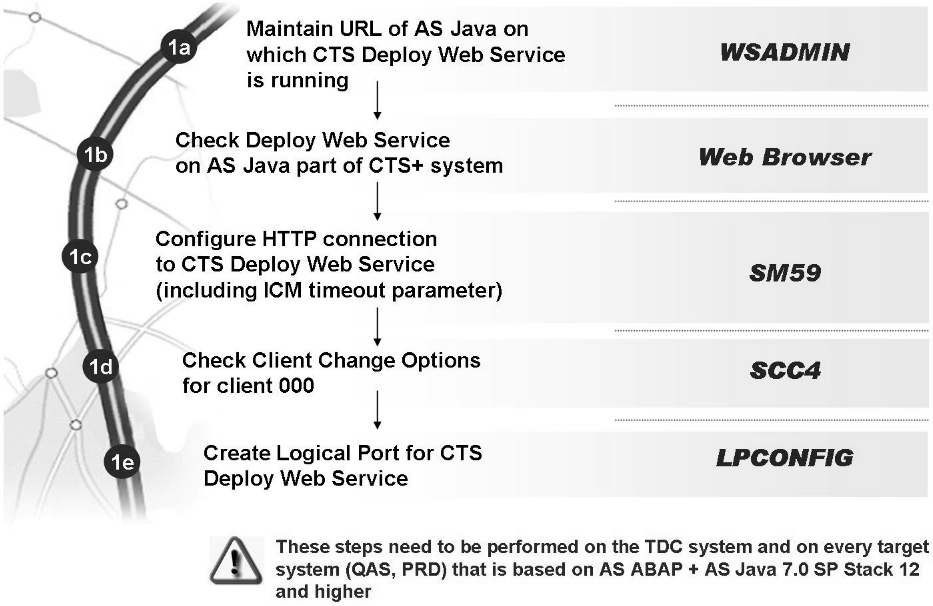

1a
Maintain URL of AS Java on which CTS Deploy Web Service is running
WSADMIN
1b
Check Deploy Web Service on AS Java part of CTS+ system
Web Browser
1c
Configure HTTP connection to CTS Deploy Web Service (including ICM timeout parameter)
SM59
1d
Check Client Change Options for client 000
SCC4
1e
Create Logical Port for CTS Deploy Web Service
LPCONFIG
These steps need to be performed on the TDC system and on every target system (QAS, PRD) that is based on AS ABAP + AS Java 7.0 SP Stack 12 and higher

Figure 39: Import part: Roadmap to Configure the CTS Deploy Service

The CTS Deploy Web Service is part of SP Stack 12 of SAP NetWeaver Application Server 7.0. In order to configure this Web Service on the CTS system, you first need to define the connection from the AS ABAP part of the CTS system to the Deploy Web Service, which in turn runs on the AS Java part of the CTS system. This is done in transaction WSADMIN in client 000 of the CTS system (AS ABAP part). Here choose *Go to → Administration Settings*. Enter the URL (http://<host of deploy web service>:<5$$00> where `<host of deploy web service>` and `$$` represents the instance number [SAP system number]) for the Java stack of your SAP system on which the CTS Deploy Web Service is running).

Note: Example: For the XI quality assurance system and XI productive system for example, the URL may point to the quality assurance system (or the productive system respectively) itself if the CTS Deploy Service is configured and running on this system.

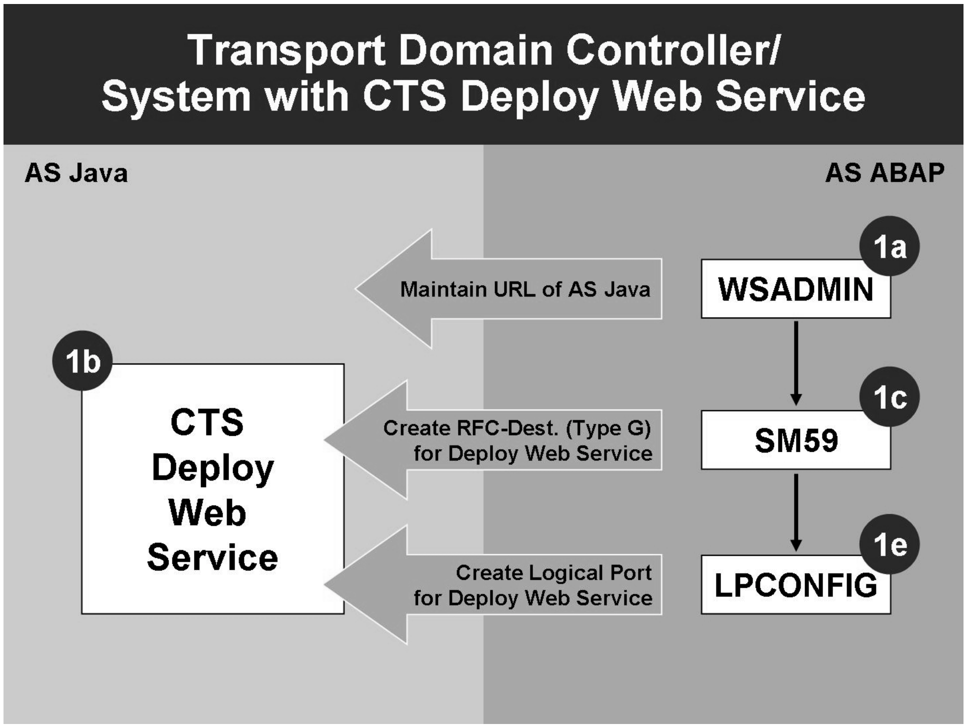

Transport Domain Controller/ System with CTS Deploy Web Service
AS Java
AS ABAP
1a
WSADMIN
Maintain URL of AS Java
1b
CTS Deploy Web Service
1c
SM59
Create RFC-Dest. (Type G) for Deploy Web Service
1e
LPCONFIG
Create Logical Port for Deploy Web Service

Figure 40: Configuration of CTS Deploy Web Service

Next, you should check the CTS Deploy Web Service on the AS Java part of the CTS system: To do this, open a browser and enter the URL for the AS Java start page on the CTS system and navigate to the *Web Service Navigator*. Check that the WebService *DeployProxy* is available in the list of deployed Web Services on that server.

⚠ **Caution:** As of December 2007 this Web Service belongs to the Software Component (SC) *DI_CMS*. This means that this Deploy Web Service is **not** part of every SAP system based on AS ABAP+Java. As a consequence, the CTS system needs to have the SC *DI_CMS* installed. It is planned however, that in the future this Deploy Web Service will be part of the AS Java itself.

Next, you need to create a new HTTP connection with the name *CTSDEPLOY* as RFC destination of connection type *G* in transaction SM59. The name of this RFC Destination will be needed for the logical port of the CTS Deploy Web Service later. On the *Technical Settings* tab enter the host name and the port of the AS Java part of the CTS system (that is host name and port that you have entered in transaction WSADMIN already). Choose */DeployProxy/default?style=document* as Path Prefix. On the tab *Logon & Security*, choose `Basic Authentication` as logon procedure. As logon user enter as user ID of the AS Java part to where the CTS Deploy Web Service is deployed. This user must have the authorization to use the CTS Deploy Web Service.

Large objects can cause timeouts in deployments. Therefore you may need to set a timeout parameter for the Internet Communication Manager (ICM) on the CTS system (AS ABAP part) – and on target systems based on AS ABAP + Java – to be longer than the longest expected deployment time. This can be done during the definition of the RFC destination (transaction SM59, tab *Special Options*) as well.

Then you need to configure a logical port for the CTS Deploy Web Service. As logical ports are customizing objects, you must enable customizing changes for client 000. This can be done in transaction SCC4. Make sure that changes and transports for client-specific objects are allowed in this client.

After verifying that changes to customizing objects are allowed in client 000, you now need to create a logical port for the CTS Deploy Web Service. This needs to be done in transaction LPCONFIG (client 000 in the AS ABAP part of the CTS system). In transaction LPCONFIG, select the Proxy Class *CO_TFLDEPLOY_PROXY_VI_DOCUMENT* from the list (F4). Enter the Logical Port `CTSDEPLOY` and Create. Enter a description for that Logical Port and check the checkbox *Default Port*. On the *Call Parameters* tab choose the HTTP Destination *CTSDEPLOY* that you have created before. Save and activate your changes within a new transport request.

Hint: After this step, reset the client change options of client 000 in transaction SCC4 to *No changes allowed*.

Export part: Configuration of Coupling of the ABAP Web Dynpro Application for CTS

The following roadmap shows the necessary steps to configure the Transport Organizer ABAP Web Dynpro Application for the Transport Domain Controller system and for all development systems that are based on AS ABAP + Java 7.0 SP Stack 12 or higher.

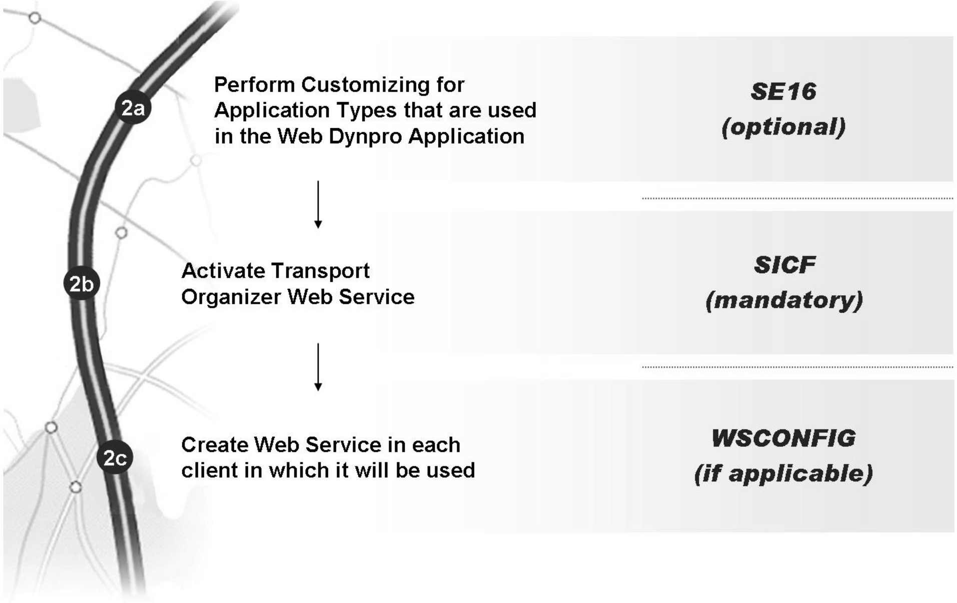

2a
Perform Customizing for Application Types that are used in the Web Dynpro Application
SE16 (optional)
2b
Activate Transport Organizer Web Service
SICF (mandatory)
2c
Create Web Service in each client in which it will be used
WSCONFIG (if applicable)

Figure 41: Export Part: Configuration the Coupling

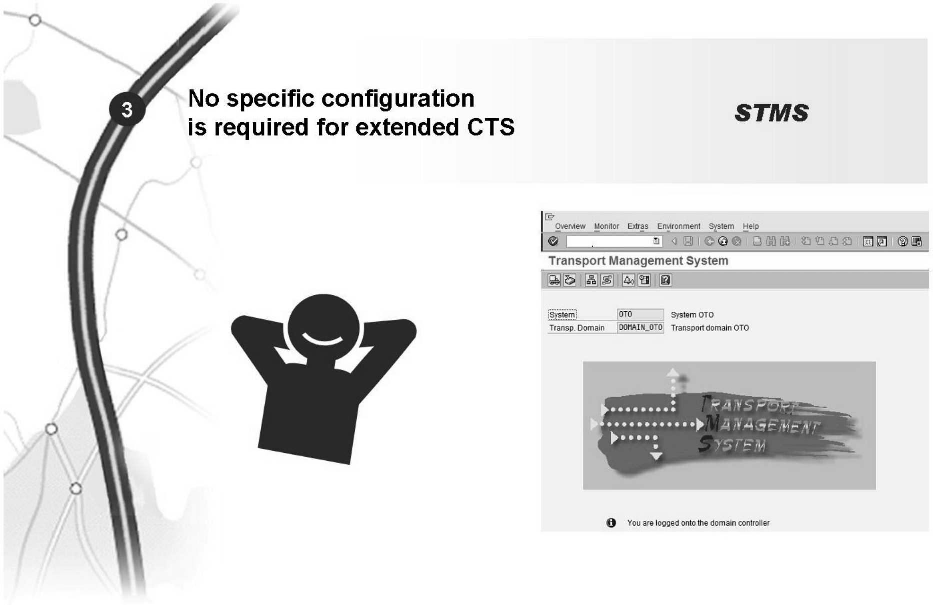
SP Stack 13 of SAP NetWeaver 7.0 provides a new ABAP Web Dynpro application that helps you to create transport orders and attach objects. You have to carry out certain configuration steps on the CTS system (AS ABAP part) to run and use this application. The application type for the file objects must be available in the *SI_RQ_APPL_NAME* database table. For example, for you to use the enhanced CTS in a portal environment, the entry **EP** must be available. The table entries are part of the delivery of SP Stack 12 (SAP NetWeaver 7.0). To check the entries, you may call transaction SE16.

Before you can use the Transport Organizer Web Service, you need to activate the *CTS_BROWSER* service in transaction SICF on the CTS system (AS ABAP part).

If you use the Transport Organizer Web service for other clients than 000, you have to create the service in each client in which you want to use it. To create the web service in a specific client, call transaction WSCONFIG and enter *SI_CORE_WBO_API* both as the service definition and as the variant.

Note: If your CTS system is based on SAP NetWeaver 7.0 SP Stack13 (SP Stack 14 for NWDI integration) this step is no longer necessary.

System Landscape Part I: Activate Transport Domain Controller


3
No specific configuration
is required for extended CTS
STMS
Overview Monitor Extras Environment System Help
Transport Management System
TRANSPORT MANAGEMENT SYSTEM

System	OTO	System OTO
Transp. Domain	DOMAIN_OTO	Transport domain OTO

You are logged onto the domain controller

Figure 42: Activate Transport Domain Controller

Before you can work with the Transport Management System (TMS), you must configure it in all SAP Systems in your system landscape.

The TMS configuration includes the configuration of the transport domain. In this step, you define which SAP Systems in your system landscape form a transport domain, and which SAP System is to be the transport domain controller. Choose a CTS system to be the transport domain controller.

Note: The procedure to activate the TMS is the same as in the “normal” ABAP transports, therefore in this step there is no specific configuration required for the enhanced functionality of CTS..

System Landscape Part II: Configuration of System Landscape

This roadmap shows the necessary steps to configure the System Landscape on the Transport Domain Controller system.

Note: The configuration steps differ with respect to both the SP Stack level and the usage type (PI versus EP versus DI) of the non-ABAP Systems.

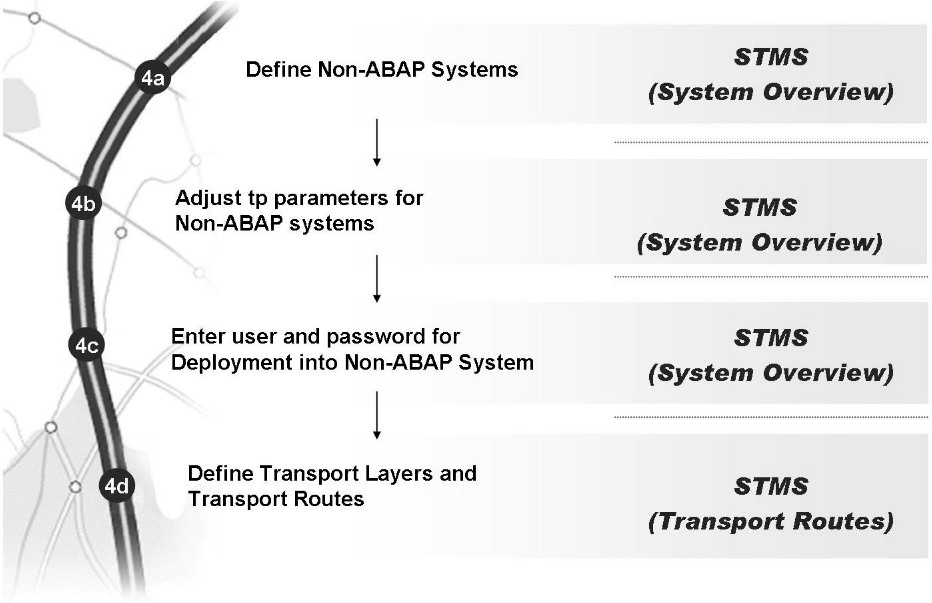


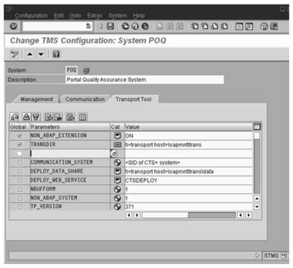
Figure 43: Configuration of the System Landscape

Depending on the Usage Type and SP Stack level of the CTS system, you need to define the following Non-ABAP systems in transaction STMS on the Transport Domain Controller System:

- Transport between systems based on AS ABAP + Java (Usage type PI for example): If the CTS system is based on SAP NetWeaver 7.0 SP Stack13 or higher however, you do not need to define any non-ABAP systems in transaction STMS. It is sufficient to include the XI development system, the XI Quality Assurance system and the XI productive system to the transport domain as you usually do it for ABAP based systems. You therefore in total need 3 systems in transaction STMS (in addition to the CTS system if the CTS system is different from one of the XI systems in question).
- For transports between systems based on AS ABAP (only), you do not need any non-ABAP systems in transaction STMS.
- For transports between systems based on AS Java only (Usage type EP for example), you need to define 3 non-ABAP systems in transaction STMS representing your Java Development system, the Java Quality Assurance System and the Java Production system. Together with the CTS system you therefore need 4 systems in transaction STMS.

Note: For any non-ABAP-System choose the Transport Domain Controller system (TDC system) to be the communication system if this system fulfils the requirements listed above.

Define the following tp parameters on the CTS+ System for the (Non-ABAP) systems in question:



Change TMS Configuration: System POQ

System POQ
Description Portal Quality Assurance System

Global	Parameters	Cat	Value
☑	NON_ABAP_EXTENSION		ON
☑	TRANSDIR		\\<transport host>\sapmnt\trans
□			
□	COMMUNICATION_SYSTEM		<SID of CTS+ system>
□	DEPLOY_DATA_SHARE		\\<transport host>\sapmnt\trans\data
□	DEPLOY_WEB_SERVICE		CTSDEPLOY
□	NBUFFORM		1
□	NON_ABAP_SYSTEM		1
□	TP_VERSION		371

	tp parameters
Source System **(Development system (AS Java part) or Transport Domain Controller)**	NON_ABAP_WBO_INBOX NON_ABAP_WBO_CLIENT NON_ABAP_SYSTEM+ WBO_GET_REQ_STRATEGY++ WBO_REL_REQ_STRATEGY++
Target system **(Quality Assurance System and Production System (AS Java part))**	DEPLOY_DATA_SHARE DEPLOY_WEB_SERVICE DEPLOY_URL (or DEPLOY_XI_URL or DEPLOY_SLD_URL) NON_ABAP_SYSTEM+

+ needed for systems without AS ABAP
++ optional; needed in case of "strong coupling" only

Figure 44: tp parameters for the enhanced functionality of the CTS

The figure above shows the necessary tp parameter that need to be maintained on the transport domain controller system for the non-ABAP systems in question.

The necessary tp parameters differ between the source system on the one hand and the target systems on the other hand.

Note: It depends on the SP Stack level of the non-ABAP system which system plays the role of the source system. For non-ABAP systems based on SAP NetWeaver 7.0 SP Stack 13 and higher (XI systems: SP Stack 12 and higher), the non-ABAP development system plays the role of the source system. For lower patch level or releases, the Transport Domain Controller system plays the role of the development system

To maintain these parameter, choose transaction STMS of the Transport Domain Controller system.

The parameter *NON_ABAP_WBO_INBOX* points to the directory in which the transport organizer looks for files that should be transported. *NON_ABAP_WBO_CLIENT* specifies the client (on the CTS system or the development system – if it contains an AS ABAP stack) in which transport requests can be created and released. This parameter is needed to enable the transport organizer Web UI.

If you want to deploy from Software Deployment Manager (SDM) in the target systems, you need to specify the tp parameter DEPLOY_URL. This parameter should point to URL of the SDM in the target system. By default this is http://<host name of target system>:5$$18 where $$ represents the system number for the central instance of the target system. Note that for the deployment into XI systems or into the SLD, this parameter is called *DEPLOY_XI_URL* or *DEPLOY_SLD_URL* instead.

The parameters *DEPLOY_DATA_SHARE* and *DEPLOY_WEB_SERVICE* as well as the parameter *NON_ABAP_SYSTEM* (value = 1) are automatically generated. The value of parameter *DEPLOY_WEB_SERVICE* must be **CTSDEPLOY**. This has been configured as the name for the RFC destination of type *G* either on the CTS system (for target systems based on AS Java only) or on the target system itself (for target systems based on AS ABAP + Java according to the system landscape in transaction STMS) as described before. The parameter *DEPLOY_DATA_SHARE* points to the directory where tp stores the data and the deploy web service takes it for deployment to the runtime systems. For systems based on AS ABAP (or AS ABAP + Java) the value of parameter *DEPLOY_DATA_SHARE* should point to the data directory beneath *DIR_TRANS* on the system in question. You can check the value of *DIR_TRANS* by calling transaction AL11 in the system. For systems based on AS Java (only) the value of this parameter may point to the data directory beneath *DIR_TRANS* on the CTS system.

The parameter *WBO_GET_REQ_STRATEGY* (values: **smart** or **tagged**) defines the automatic creation of transport requests during the export of an object in a Portal system or in an XI system. The parameter *WBO_REL_REQ_STRATEGY* (values **auto** or **manual**) defines the automatic release of transport requests in a portal system or in a XI system. Both parameters are optional and are used in the strong coupling scenario (direct integration of the transport organizer Web UI into the corresponding Java runtime systems / application) only.

The import of Java applications into subsequent systems will be performed by the ABAP transport control program tp. tp calls the CTS Deploy Web Service according to the tp parameters. The CTS Deploy Web Service in turn calls the Software Deployment Manager (SDM). You need to specify a user with appropriate permissions to perform the import. This used must be named on the transport domain controller system of the transport domain to which the system belong. In the system overview area of transaction STMS the transport domain controller, double click on the Non-ABAP system in question and switch to the *Transport Tool* tab.

If your system is an XI system, choose *Goto → XI User / password* and enter the user / password who will start the deployment. This user must exist on the XI system in question and the PFCG role *SAP_XI_CMS_SERV_USER* must be assigned to this user.

If your system is based on an AS Java only (for example an enterprise portal system), the deployment will be performed by the SDM. You therefore need to choose *Goto → SDM user / password* and enter the SDM user and password of the Java System in question (this user has been created during the installation of that system).

For System Landscape Directory (SLD) systems, the deployment procedure is a little bit different however. You therefore need to specify a different user on the SLD application.

Note: You need to perform this step for every system in which you want to import non-ABAP objects.

Finally you need to define transport layers and transport routes between the non-ABAP systems in question. The transport layers and the transport routes that are needed depend on the usage types of the systems in question and on the SP stack level of the CTS system. The definition of transport layers and transport routes is done on the Transport Domain Controller system (TDC system) in transaction STMS (*Transport Routes* pushbutton).

If the CTS system is based on SAP NetWeaver 7.0 SP Stack 13 or higher, the development system itself can be used as integration system (that is as source system for a consolidation route). Therefore all transport requests that are created are of type *<SID of development system>K9<number>*.

Appendix: Configuration of Runtime Systems

In addition to the configuration of the Transport Domain Controller, you also may need to configure your runtime systems (that means the systems between which the export and import take place).

The necessary steps for the configuration of runtime systems differ with respect to the usage type of these systems.

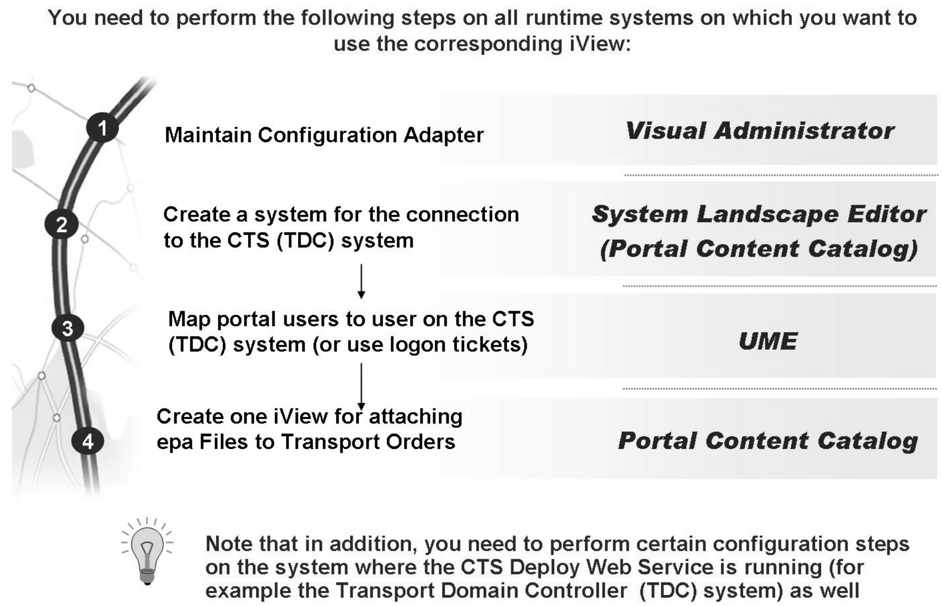
For target systems based on AS ABAP only, no specific preparation steps are required for the use of enhanced CTS. You simply configure your systems for the use with CTS “as usual”. In addition no specific release of the systems based on AS ABAP only is required. The CTS compatibility is sufficient in this case.

If you want to transport portal content with the help of the enhanced CTS however, you need to configure the portal systems accordingly. The main steps are listed on the roadmap below.

In this case, you need to create an iView in the portal system. This iView allows the attachment of epa files to SE09 transport orders. This iView uses the ABAP Web Dynpro application *CTS_BROWSER* in the CTS system. As a prerequisite on the CTS system, this ABAP Web Dynpro application needs to be activated in transaction

SICF. The portal system needs to connect to this Web Dynpro Application. Therefore a system needs to be defined. This system points to the integrated ITS of the CTS system.

Note: Instead of mapping portal users to users on the CTS system, it is also possible to use logon tickets.



You need to perform the following steps on all runtime systems on which you want to use the corresponding iView:

1 Maintain Configuration Adapter — ***Visual Administrator***

2 Create a system for the connection to the CTS (TDC) system — ***System Landscape Editor (Portal Content Catalog)***

3 Map portal users to user on the CTS (TDC) system (or use logon tickets) — ***UME***

4 Create one iView for attaching epa Files to Transport Orders — ***Portal Content Catalog***

Note that in addition, you need to perform certain configuration steps on the system where the CTS Deploy Web Service is running (for example the Transport Domain Controller (TDC) system) as well

Figure 45: Configuration of Portal Systems as Runtime Systems

In order to enable strong coupling for SAP NetWeaver Portal Systems based on AS Java 7.0 SP Stack 13 or higher (or for NWDI systems based on AS Java SP Stack 12 or higher) , check in non-ABAP objects from a file server, you need to maintain the configuration adaptor. This is done on the Visual Administrator on the SAP NetWeaver Portal server (or NWDI server). Navigate to the Configuration Adapter (*Server* → *Services* → *Configuration Adapter*). Within the tab *Runtime* choose *Configurations* → *apps* → *sap.com* → *tc~com.sap.tmw.clients~_ctsa_manager* → *appcfg* → *Propertysheet application.global.properties*. Enter or change the values for the properties *target.url* and *tms.system.*

To create a system for the connection to the CTS server in your Portal System, go to *System Administration → System Configuration → System Landscape*. Navigate through the *Portal Content Catalogue* to the folder where you want to create the system. Click with the secondary mouse button on this folder and choose *New System (from template)*. The system creation wizard helps you to create this system.

Hint: With SP Stack 14 of SAP NetWeaver 7.0, the *Configuration Adapter* will be replaced by the *Destination Service* in the Visual Administrator.

If in your portal system user mapping is used as logon method, you need to map your portal user to an existing backend user in the CTS system (for the client that corresponds to the tp parameter *NON_ABAP_WBO_INBOX* that is for the client in which transport requests will be created and released). The User Mapping needs to be done for every user.

Hint: User Mapping is only necessary if logon tickets are not available. If it is possible to use Logon Tickets or Certificates you should always choose this option. User mapping is done in the *User Management Engine (UME)* console of your portal system (area *User Administration → Identity Management*). In the details section, click on the tab *User Mapping for system access*.

Note: In XI runtime systems you need to create the Service-User **`NWDI_CTSADMIN`** and to assign the role *SAP_XI_CMS_SERV_USER* to this user. The user that you have created will be needed in the TDC system in the TMS system overview. There you need to choose *Goto → XI User/Password* from the menu and enter the user/password that you maintained in the XI system.

Lesson Summary

You should now be able to:

- list the prerequisites for CTS domain controller system, source system and target systems that are needed to perform transports of non-ABAP objects
- list the configuration steps to set up a transport landscape for transporting non-ABAP objects
- create non-ABAP systems in transaction STMS and define transport routes between them.

Related Information

Additional information for the transport of non-ABAP-objects with the help of the Change and Transport Systems can be found in the online documentation of SAP NetWeaver 7.0, area *SAP Library → SAP NetWeaver Library → SAP NetWeaver by Key Capability → Solution Life Cycle Management by Key Capability → Software Life Cycle Management → Software Logistics → Change and Transport System → Change and Transport System - Overview (BC-CTS) → Non-ABAP Transports in Change and Transport System (BC-CTS)*.

In addition there are some articles in SAP Developer Network (SDN) regarding the enhanced functionality of the CTS. These are available on SDN in the area *SDN Community → SAP NetWeaver Capabilities → Lifecycle Management → Software Logistics*. From here follow the link *Change and Transport System* (in the area *Knowledge Center → Software Change Management*).

Unit Summary

You should now be able to:

- explain the steps for setting up the transport management system (TMS)
- configure transport routes
- explain the use of transport target groups and client-dependent transport routes
- list the necessary steps to perform transports between different transport groups and transport domains
- list the prerequisites for CTS domain controller system, source system and target systems that are needed to perform transports of non-ABAP objects
- list the configuration steps to set up a transport landscape for transporting non-ABAP objects
- create non-ABAP systems in transaction STMS and define transport routes between them.

Test Your Knowledge

1. Which statements concerning transport routes are correct

   Choose the correct answer(s).

   □ A SAP distinguishes between consolidation routes and delivery routes

   □ B SAP distinguishes between export routes and import routes

   □ C If you want to export transport requests from an SAP system, there needs to be at least one consolidation route starting from this system.

   □ D If you want to export transport requests from an SAP system, there needs to be at least one export route starting from this system

   □ E If you want to export change requests from an SAP system, there needs to be at least one delivery route starting from this system

2. Which statements are correct

   Choose the correct answer(s).

   □ A A transport domain always consists of SAP systems sharing the same transport directory.

   □ B A transport group consists of SAP systems sharing the same transport directory.

   □ C A system landscape always consists of SAP systems sharing the same transport directory.

   □ D You can add an additional system to a transport domain by logging on to this system and pressing the *approve* button in transaction STMS.

   □ E Additional systems can be approved for the domain only from the transport domain controller system.

3. How can you achieve that a transport request from the DEV system will be imported into several clients of the QAS system?

   Choose the correct answer(s).

   □ A By using a client-specific transport target group

   □ B By defining different consolidation routes pointing from the DEV system to the QAS system

   □ C It is impossible to achieve this with TMS.

4. Assume that you want to transport non-ABAP obejcts with the help of the ABAP Change and Transport System. Which statements are correct?

   Choose the correct answer(s).

   □ A The Transport Domain Controller must be based on SAP NetWeaver AS 7.00 SP Stack 12 or higher.

   □ B If you want to transport Portal Content with CTS, the portal runtime systems need to based on AS Java 7.00 SP Stack 12 or higher.

   □ C You need to define non ABAP Systems in Transaction STMS and you need to maintain the tp parameters for these systems correctly

   □ D You need to define transport routes between the non-ABAP systems in question.

   □ E The CTS deploy web service must be configured on every non-ABAP system that is defined in transaction STMS.

Answers

1. Which statements concerning transport routes are correct

   Answer: A, C

   SAP distinguishes between consolidation routes and delivery routes. A consolidation route is used as an export / import route, i.e. you export transport requests from the source system and import them into the following system. A delivery route means "another import" i.e. there is no export from the source system of the delivery route.

2. Which statements are correct

   Answer: B, E

   A transport domain consists of SAP systems that are administered centrally with respect to TMS. Certain administrative tasks (as approving other systems in the domain) can only be done in the transport domain controller system. A system landscape consists of SAP systems that are linked by transport routes. A transport group consists of SAP systems sharing the same transport directory.

3. How can you achieve that a transport request from the DEV system will be imported into several clients of the QAS system?

   Answer: A

   You can not address more then one consolidation route with one transport request. If you use a client-specific transport target group, however, you can achieve that the transport request will be ready for import into different clients of the QAS system.

4. Assume that you want to transport non-ABAP obejcts with the help of the ABAP Change and Transport System. Which statements are correct?

   Answer: A, C, D

   You can transport portal content with the enhanced CTS even between portal runtime systems based on AS Java 6.40. It is only necessary that the Transport Domain Controller (TDC) is based on SAP NetWeaver AS 7.00 SP stack 12 or higher. Concerning the CTS Deploy Web Service, it is sufficient to configure and use this Web Service on the TDC system only.

Unit 3

Creating and Exporting Transport Requests

Unit Overview

In this unit, you will learn the general concepts of creation and management of transport requests. Transport requests can result from customizing or development activity.

Unit Objectives

After completing this unit, you will be able to:

- Explain what Customizing is, from the business and the technical perspective
- Explain how the SAP Implementation Guide (IMG) and IMG Projects can be used for Customizing
- Explain how the SAP Solution Manager can be used for implementation projects
- Create transport requests and tasks
- List the tools used for managing transport requests
- Describe how to customize
- Follow steps in procedure for customizing an SAP system implementation
- Define cross-client customizing
- Plan change management for development
- Identify various implementation tools used for Customizing and development
- Describe the function of the Transport Organizer
- create a simple program
- understand the principles of the version management
- check and adjust the Transport Organizer settings
- Modify SAP Repository objects
- Describe the difference between customer development and modifications
- Describe how the Modification Assistant assists developers

Unit Contents

Lesson: Customizing and Customizing Projects

Lesson Overview

In this lesson, you will get an overview on customizing of SAP systems to meet individual business requirements. You will also learn how the SAP Implementation Guide (IMG) and the SAP Solution Manager can be used in the Customizing process.

Lesson Objectives

After completing this lesson, you will be able to:

- Explain what Customizing is, from the business and the technical perspective
- Explain how the SAP Implementation Guide (IMG) and IMG Projects can be used for Customizing
- Explain how the SAP Solution Manager can be used for implementation projects

Business Example

Customizing settings adapt SAP systems to meet a company's business requirements by defining the logic paths that the application programs follow. As a TMS administrator, you want learn about the roles and responsibilities of the team that will do the Customizing. This is necessary for supporting the Customizing teams during the complete life cycle of the project. With that knowledge, you are also able to recommend the right tools for the efficient realization of the Customizing project.

Definition of Customizing

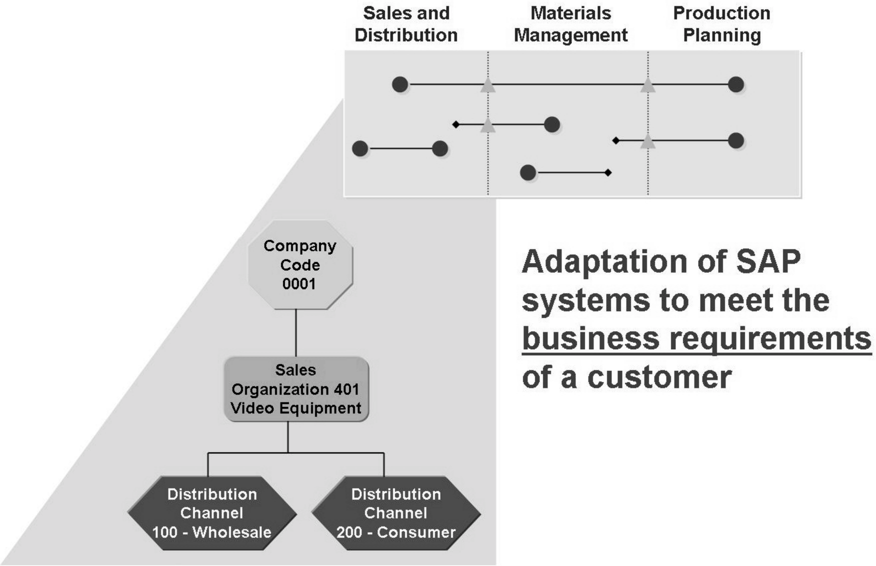

Sales and Distribution
Materials Management
Production Planning
Company Code 0001
Sales Organization 401 Video Equipment
Distribution Channel 100 - Wholesale
Distribution Channel 200 - Consumer
Adaptation of SAP systems to meet the business requirements of a customer

Figure 46: What is Customizing?

Customizing adapts the SAP software to meet a company's individual business requirements by setting up the business transactions the company requires in the system.

To illustrate the kinds of business transactions, an enterprise may need to customize in an SAP system, consider the example of an international company that sells video equipment wholesale. To enable customers at a particular outlet to place a wholesale order for video equipment,, the company must customize the system to recognize the company's structure. This could encompass:

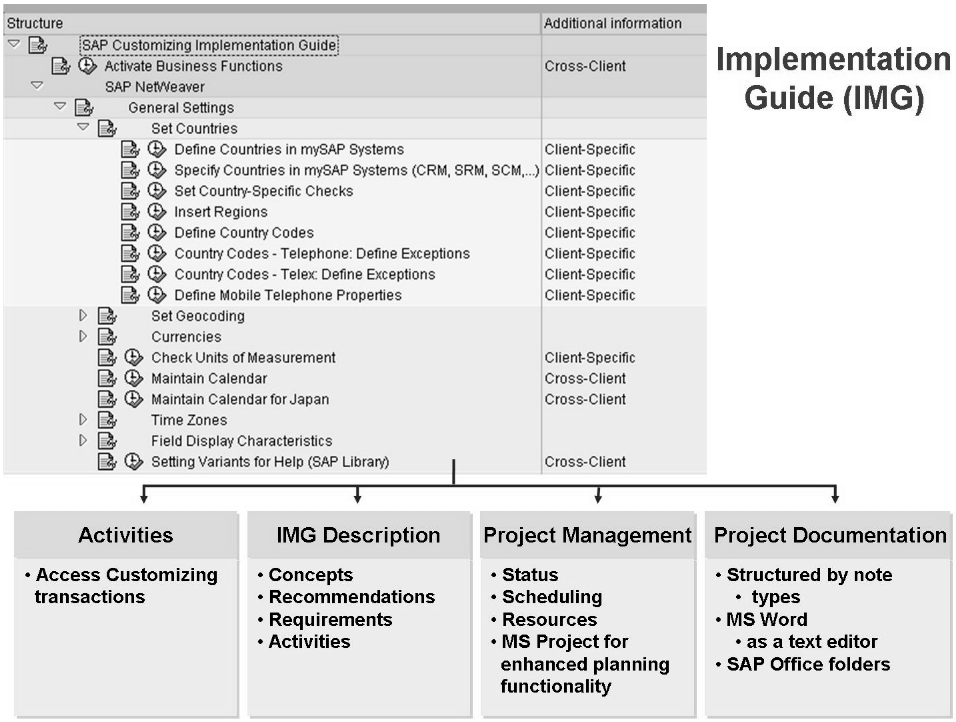
- Defining a sales organization
- Defining distribution channels
- Assigning distribution channels to the sales organization
- Defining pricing conditions and tax determination rules
- Assigning general ledger accounts

In the example the definition of a sales organization and the distribution channels are done. This Customizing is a example for the definition of the individual organizational structure of a company. This kind of Customizing is typically stored in client-dependent Customizing tables. In general this Customizing is also known as a client-dependent Customizing. Client-dependent Customizing affects only one particular client.

Another type of Customizing is Customizing which adjusts the execution of programs such as the pricing conditions or the tax determination rules. This Customizing is mostly also client-dependent, but some customizing activities insert entries in client-independent tables or generate some Repository objects (tables or programs). In general this Customizing is known as client independent Customizing. Client-independent Customizing affects all clients in the system.

Customizing Using the SAP Implementation Guide

An SAP System provides a huge variety of different business processes that can adapted to the individual needs of a customer by customizing. An SAP ECC 6.0 system for example contains round about 20000 customizing tables. The customer is guided thru this customizing with the help of the IMG. An SAP system is delivered with a complete IMG for all processes available in the system. This IMG is known as the SAP Reference Implementation Guide (IMG). The IMG can be accessed via transaction SPRO.



Structure	Additional information
SAP Customizing Implementation Guide	
Activate Business Functions	Cross-Client
SAP NetWeaver	
General Settings	
Set Countries	
Define Countries in mySAP Systems	Client-Specific
Specify Countries in mySAP Systems (CRM, SRM, SCM,...)	Client-Specific
Set Country-Specific Checks	Client-Specific
Insert Regions	Client-Specific
Define Country Codes	Client-Specific
Country Codes - Telephone: Define Exceptions	Client-Specific
Country Codes - Telex: Define Exceptions	Client-Specific
Define Mobile Telephone Properties	Client-Specific
Set Geocoding	
Currencies	
Check Units of Measurement	Client-Specific
Maintain Calendar	Cross-Client
Maintain Calendar for Japan	Cross-Client
Time Zones	
Field Display Characteristics	
Setting Variants for Help (SAP Library)	Cross-Client

Implementation Guide (IMG)

Activities
- Access Customizing transactions

IMG Description
- Concepts
- Recommendations
- Requirements
- Activities

Project Management
- Status
- Scheduling
- Resources
- MS Project for enhanced planning functionality

Project Documentation
- Structured by note
  - types
- MS Word
  - as a text editor
- SAP Office folders

Figure 47: Using the IMG for Customizing (SAP ECC 6.0)

The IMG provides a hierarchical list with all possible customizing activities in that system. This Customizing is grouped by modules and ordered in the sequence in which the individual activity has to be performed.

The IMG does not only show and group all customizing activities. The IMG also provides documentation on each customizing activity. This documentation explains the customizing activity that can be executed in this IMG node. Also each customizing activity is classified into different areas. For example, if you are interested in which customizing activity is client independent, you can use the SAP menu: transaction SPRO *Additional Information →Technical Data →Client-Dependence*. There is also a classification on language dependency and on transport type that can be displayed.

Beside these tools for accessing customizing activities, displaying customizing documentation, displaying customizing classification the IMG also provides tools for: project management and project documentation. These tools are supporting the project administration and project control out of the IMG.

The SAP Reference IMG is split up into different areas.

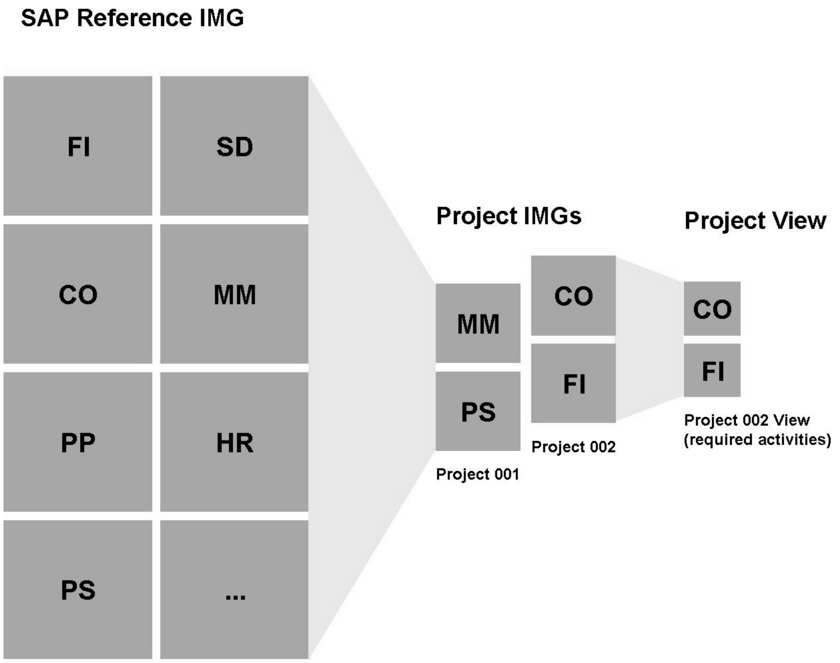


Figure 48: Customizing Projects as part of the IMG (SAP ECC 6.0)

If a customer is setting up a new project for adoption of the SAP system, only some areas are in the scope of this adoption project. The customer can use a project IMG. A project IMG is the implementation guide for a specific customizing project. It contains all the customizing activities which have to be performed in that project. You can determine the project scope of a project IMG by countries, components, or customizing activities. A project IMG is based on the SAP reference IMG, which can, for example, be reduced to the project scope by selecting countries and components.

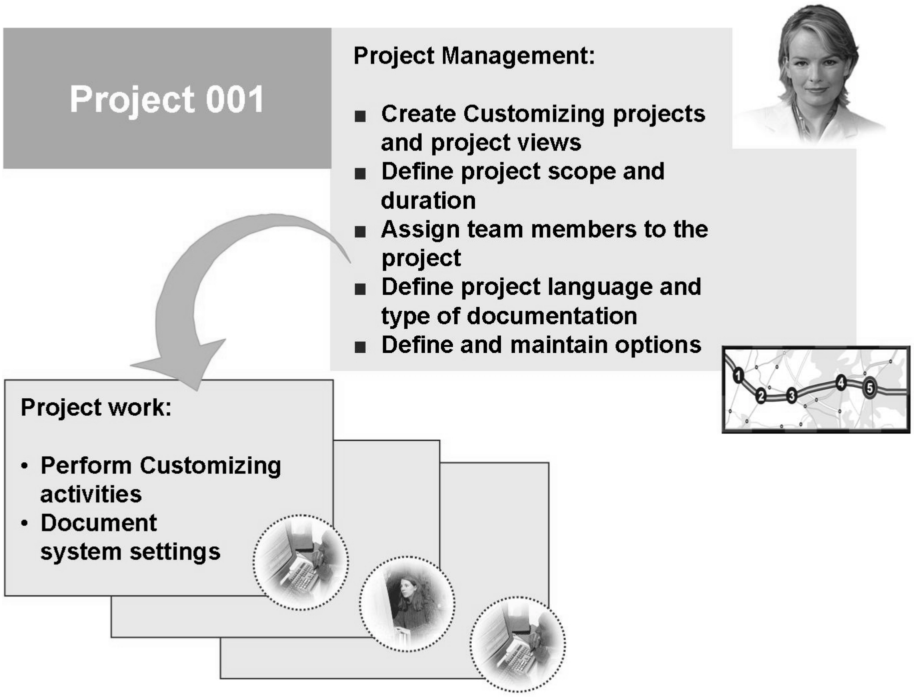
When defining a customizing project, customers select the functions they require in the different SAP areas and components. These functions are used to generate the project IMG. All relevant customizing transactions, project-specific or cross-project documentation, and project-management information are then located in the subsets of the SAP reference IMG, known as project IMGs. All customizing activities should be done directly within the project IMG.

The project IMGs can be changed at any time through regeneration. The respective previous IMG is then overwritten. However, status information and project documentation is not lost.

Within a project IMG it is possible to generate project views that restrict the extent of Customizing, for example, to different levels of necessary activities. So it is possible to show within an IMG project view only the activities that are classified as mandatory and in a second IMG project view the activities that are classified as optional.

The project IMGs are cross-client, that means the project IMG is visible in all clients.

To access the IMG project definition you can use the transaction code SPRO_ADMIN.



Project 001

Project Management:

- Create Customizing projects and project views
- Define project scope and duration
- Assign team members to the project
- Define project language and type of documentation
- Define and maintain options

Project work:

- Perform Customizing activities
- Document system settings

Figure 49: Projects in Customizing

Depending on the scope of a customizing project, a lot of different customizing activities have to be done. These customizing activities are normally done by different project members that are working on the implementation of a specific area. To make sure that everyone is working efficiently together, customizing projects need a clear organizational structure and responsibilities.

The customizing and transport tools support the task sharing between the project leader and the project members. The customizing process is based on this to clearly-defined user roles. Each role has its specific areas of responsibility. This structures and organizes the implementation activities.

The project leader is responsible for defining and managing the project. This includes if IMG Projects are used:

- creating the customizing IMG projects and the project views
- define the project scope and duration of the IMG project
- assign team members to the IMG project (allow who is permitted to do changes)
- define the IMG project language and type of project documentation
- define and maintain IMG project options

As you can see with the use of IMG projects, a lot of the general project administration tasks can be done inside the SAP system.

The customizing team members are doing their project work:

- performing customizing activities within the project IMG
- document the system setting inside the project IMG

Any number of project team members can be assigned to each project. The project leader makes this assignment. The assigned project team members and new project team member assignments can be managed in transaction SPRO_ADMIN.

SAP Solution Manager in Implementation

As you have seen in this lesson so far, the IMG offers a lot of tools and documentations that can be used for supporting the implementation process. But if you set up and plan an implementation project, you will have to manage all the different project phases which are:

- Project preparation
- Business blueprint
- Realization
- Final preparation
- Go-live support

For the coverage of these project phases, SAP offers the *Accelerated SAP / ValueSAP* method.

The process of customizing the system can be controlled and accelerated using the *Accelerated SAP / ValueSAP* method. ValueSAP represents a framework of tools, content and methodology to support the entire *live cycle* of a solution. Since the SAP Solution Manager 2.2 these ValueSAP tools are transitioning to the SAP Solution Manager. The SAP Solution Manger is the portal for implementation, operation and continuous business improvement for SAP applications. The AcceleratedSAP/ ValueSAP CDs that were provided in the past with that methodologies and tools are now replaced and extended with the implementation content inside the SAP Solution Manager. Now it is possible to run project with this tools inside the SAP Solution Manager (see note 576877). Consequently, the SAP Solution Manager is now the platform for the entire project team.

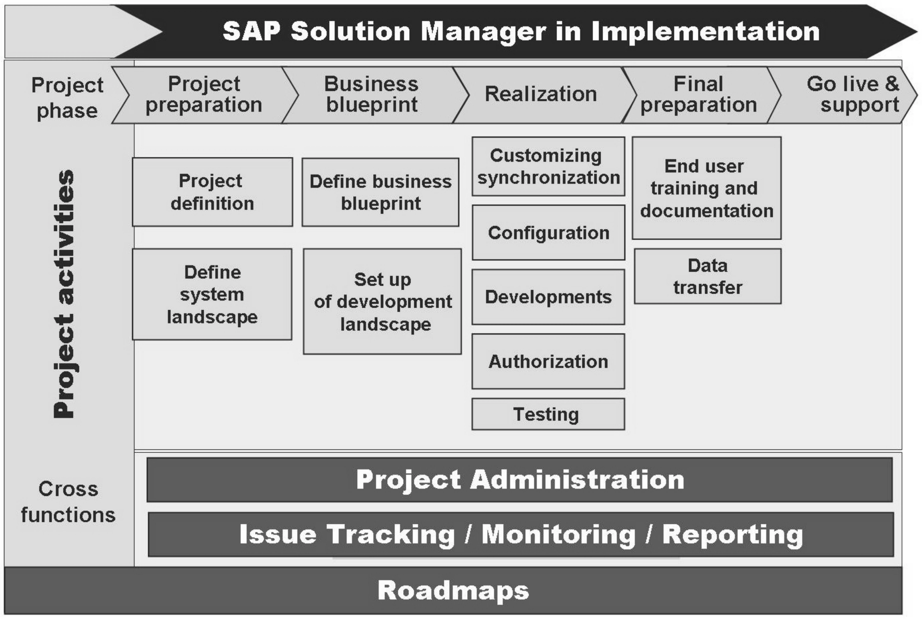


Figure 50: SAP Solution Manager Implementation Area

The SAP Solution Manager in Implementation provides the following features:

- Project Administration/Project Definition supports you initially in setting up your project during project preparation and enables you to carry out major administrative tasks such as the definition of project standards during the entire project.
- In the SAP Solution Manager, you can centrally define and manage your system landscape centrally, relevant for your implementation or template projects, which allows the interaction and navigation into a related system landscape; for example, during configuration and testing.
- The definition and documentation of your project scope in business blueprint is accelerated through the Business Process Repository providing latest available implementation content. The purpose of the business blueprint is to document in detail the scope of business scenarios, business processes, and process steps of your implementation project from a business and technical perspective.
- During configuration, you have to configure the business requirements specified in the business blueprint phase in the related system landscape. The SAP Solution Manager as central platform for your project team provides the access to the related project implementation guides (IMG's) and integrated use of other customizing technologies such as Business Configuration Sets (BC Sets) and customizing distribution.
- The customizing distribution used with the SAP Solution Manager helps to centrally control customizing changes across the related system landscape of your implementation project.
- Testing functions support test coordinators to centrally create test cases and test plans reflecting sequence and integration tests. Testers can centrally execute their test packages, also leveraging existing test technologies for automated testing such as eCATT.
- The latest available roadmaps representing the standard SAP implementation methodologies are provided through the SAP Solution Manager. With the SAP Solution Manager, you can track status, issues and/or upload your project-specific accelerators along the work packages and tasks of your implementation project.

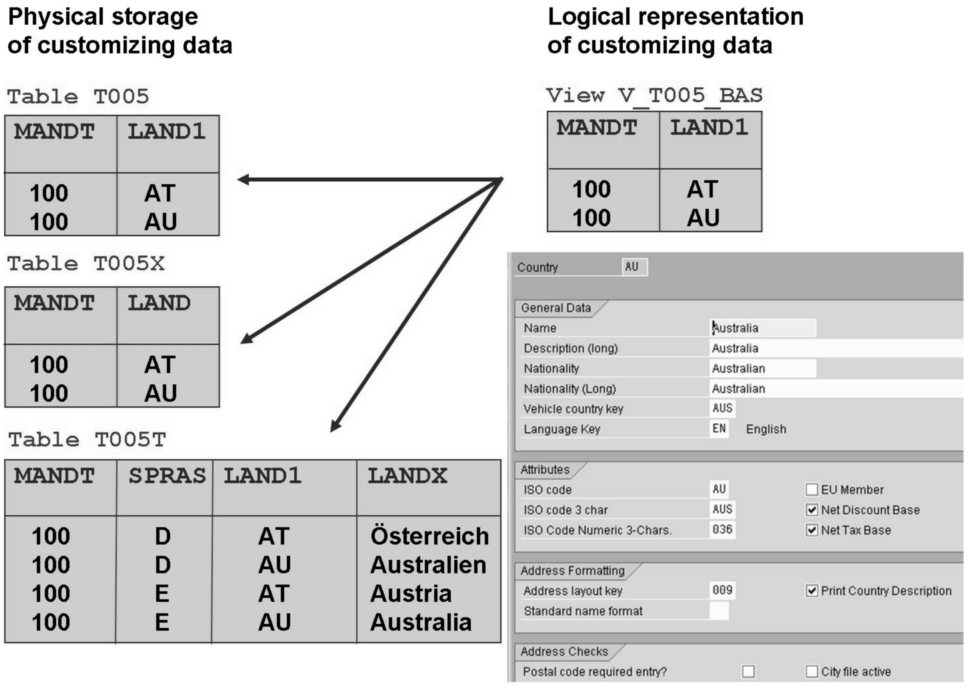
Implementation projects in an SAP system landscape today normally affect more than one system (e.g. CRM and ECC). During that kind of projects Customizing in more than one systems has to be done. The SAP Solution Manager in Implementation supports project maintenance and administration in a system landscape. Implementation projects that affect a special system landscape can be defined centrally for this system landscape and then the project IMGs in the different systems are generated.

The SAP Solution Manager for Implementation provides the tools for supporting customizing projects in a system landscape of multiply systems. In the future this function will be extended, e.g. with the *Change manager*, a collection of tools that provide support for:

- Implementation projects
- Ad hoc development
- Maintenance tasks
- Hot-fix tasks

including the transport tracking and approval tasks that are necessary in a productive system landscape with on-going implementation project and maintenance tasks.

The following figure shows the technical representation of Customizing in the system.



Physical storage of customizing data

Table T005

MANDT	LAND1
100	AT
100	AU

Table T005X

MANDT	LAND
100	AT
100	AU

Table T005T

MANDT	SPRAS	LAND1	LANDX
100	D	AT	Österreich
100	D	AU	Australien
100	E	AT	Austria
100	E	AU	Australia

Logical representation of customizing data

View V_T005_BAS

MANDT	LAND1
100	AT
100	AU

Country AU

General Data

Name	Australia
Description (long)	Australia
Nationality	Australian
Nationality (Long)	Australian
Vehicle country key	AUS
Language Key	EN English

Attributes

ISO code	AU	□ EU Member
ISO code 3 char	AUS	☑ Net Discount Base
ISO Code Numeric 3-Chars.	036	☑ Net Tax Base

Address Formatting

Address layout key	009	☑ Print Country Description
Standard name format		

Address Checks

Postal code required entry?	□	□ City file active

Figure 51: Customizing from a Technical Perspective

Customizing activities typically create or change entries in multiple client-specific tables. This is done through table views.

A table view is a virtual table, which presents data that is physically stored in multiple tables. You can compare a table view with a view that is used in an SQL database. The different customizing tables of a table view are related to each other by foreign keys.

The table entries created with the help of table views are identified by the key fields of the respective table.

In the picture above the table view for maintenance *Global Parameters of Countries* is shown. The country parameters are maintained with the help of view *V_T005_BAS*. All the different attributes for a country, e.g. name, nationality, ISO code, date format and decimal notation are maintained. These parameters are stored physically in different tables. For example, general country values in table *T005*, the country names in different languages in table *T005T* and the decimal point and date format in table *T005X*.

The customizer does not need to care about the technical storage of the Customizing, he or she only has to maintain the information in the table view.

The table maintenance can be accessed directly with the IMG or even with a generic table maintenance transaction as transaction SM30.

Lesson Summary

You should now be able to:

- Explain what Customizing is, from the business and the technical perspective
- Explain how the SAP Implementation Guide (IMG) and IMG Projects can be used for Customizing
- Explain how the SAP Solution Manager can be used for implementation projects

Lesson: Transport Requests for Customizing

Lesson Overview

This lesson introduces the tools SAP delivers to assist in the management of implementing and maintaining customizing changes to an SAP system landscape. This lesson will explain the tools for creating and working with transport requests and tasks.

Lesson Objectives

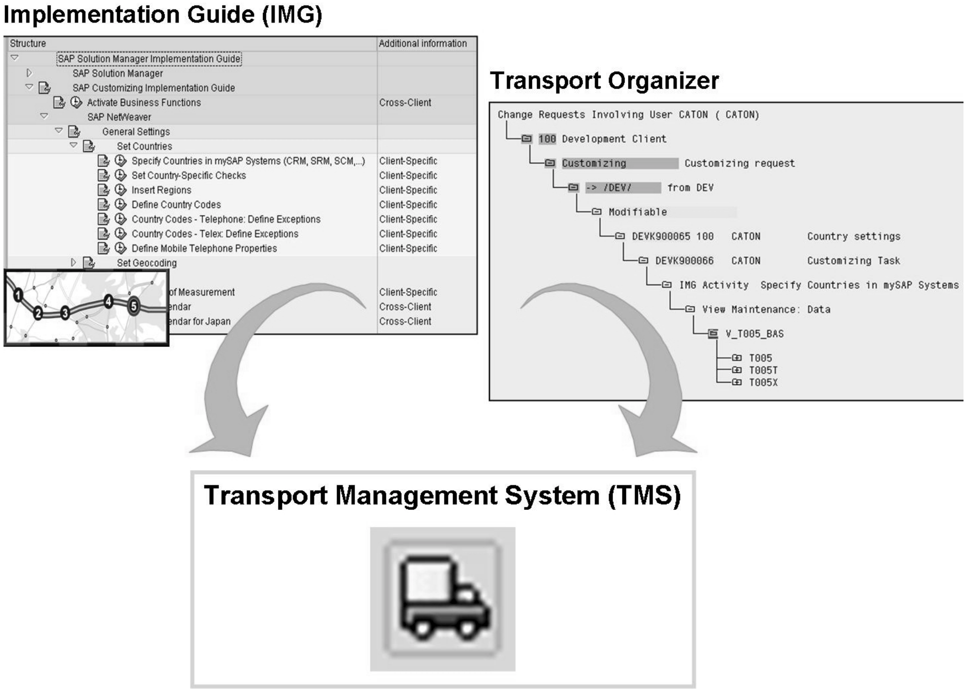
After completing this lesson, you will be able to:

- Create transport requests and tasks
- List the tools used for managing transport requests

Business Example

Customizing activities are performed in the development system. The Customizing changes performed must be recorded for transferring these changes to the quality assurance and production system. For the recording of customizing changes, transport requests must be created. In addition, it is necessary to assign tasks for the different customizers in the transport request.

Tools for Managing Transport Requests



Structure	Additional information
SAP Solution Manager Implementation Guide	
SAP Solution Manager	
SAP Customizing Implementation Guide	
Activate Business Functions	Cross-Client
SAP NetWeaver	
General Settings	
Set Countries	
Specify Countries in mySAP Systems (CRM, SRM, SCM,...)	Client-Specific
Set Country-Specific Checks	Client-Specific
Insert Regions	Client-Specific
Define Country Codes	Client-Specific
Country Codes - Telephone: Define Exceptions	Client-Specific
Country Codes - Telex: Define Exceptions	Client-Specific
Define Mobile Telephone Properties	Client-Specific
Set Geocoding	
of Measurement	Client-Specific
endar	Cross-Client
endar for Japan	Cross-Client

```
Change Requests Involving User CATON ( CATON)
 └─ 100 Development Client
     └─ Customizing          Customizing request
         └─ -> /DEV/          from DEV
             └─ Modifiable
                 └─ DEVK900065 100   CATON       Country settings
                     └─ DEVK900066   CATON       Customizing Task
                         └─ IMG Activity  Specify Countries in mySAP Systems
                             └─ View Maintenance: Data
                                 └─ V_T005_BAS
                                     ├─ T005
                                     ├─ T005T
                                     └─ T005X
```

Figure 52: Customizing Tools

SAP provides the following implementation tools for Customizing:

- Implementation Guide (IMG)
- Transport Organizer

The Implementation Guide (IMG) is the main customizing tool. It represents the central entry point for all project members for carrying out the customizing activities. Customizing takes place out of the created IMG project, which contains all the customizing activities for a specific project scope in a hierarchical list. If the Customizing is changed, these changes must be recorded. The Transport Organizer (transaction SE09 /SE10) records the customizing changes in transport requests.

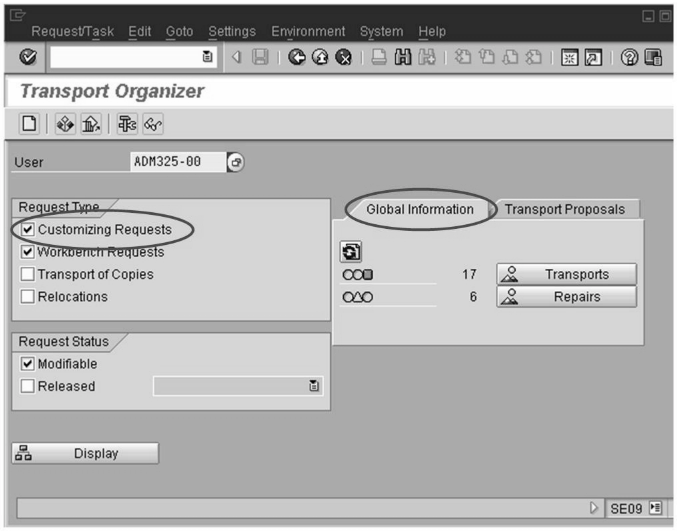
It is an information source in the Transport Organizer that records all changes and modifications made to Repository objects and customizing settings during a development project.

If the Customizing is finished, the transport request can be released to the transport system for transferring the changes to other systems in the system landscape.

The function for recording the changes made during Customizing, is integrated in the customizing and table maintenance transactions. The Transport Organizer is called from within these customizing transactions for recording the changes into transport requests. The Transport Organizer is fully integrated in the Transport Management System (TMS).

The Transport Organizer provides information about:

- **Transport requests for Customizing**
- **Transport requests for Repository objects**
- **Global transport information**



User ADM325-00

Request Type
☑ Customizing Requests
☑ Workbench Requests
□ Transport of Copies
□ Relocations

Request Status
☑ Modifiable
□ Released

17 Transports
6 Repairs

Figure 53: Transport Organizer used for Customizing

The Transport Organizer is used to create, manage, release, and analyze transport requests that record Customizing or Repository changes. To access the Transport Organizer you can use the transaction code SE09.

All customizing requests that belong to a specific user are displayed according to a set of standard selection criteria in the Transport Organizer. The selection options include user, transport request type, transport request status and date. An example of these selection criteria is shown in the picture above. Also in the Transport Organizer the screen area *Global Information* is displayed. With the help of this area you can get a graphical cross- system overview on all transport requests. Within this area you can track, if a transport was imported for example in the consolidation system.

Creating Transport Requests and Tasks

Customizing mainly consists of client-specific table entries and are saved to transport requests of type Customizing. A transport request belongs only to one user who owns the transport request. It is possible, that other users are also storing their objects in the same transport request. The owner of a transport request is able to administer and release the request. For example, the owner can assign other users to the transport requests, release the transport request and so on.

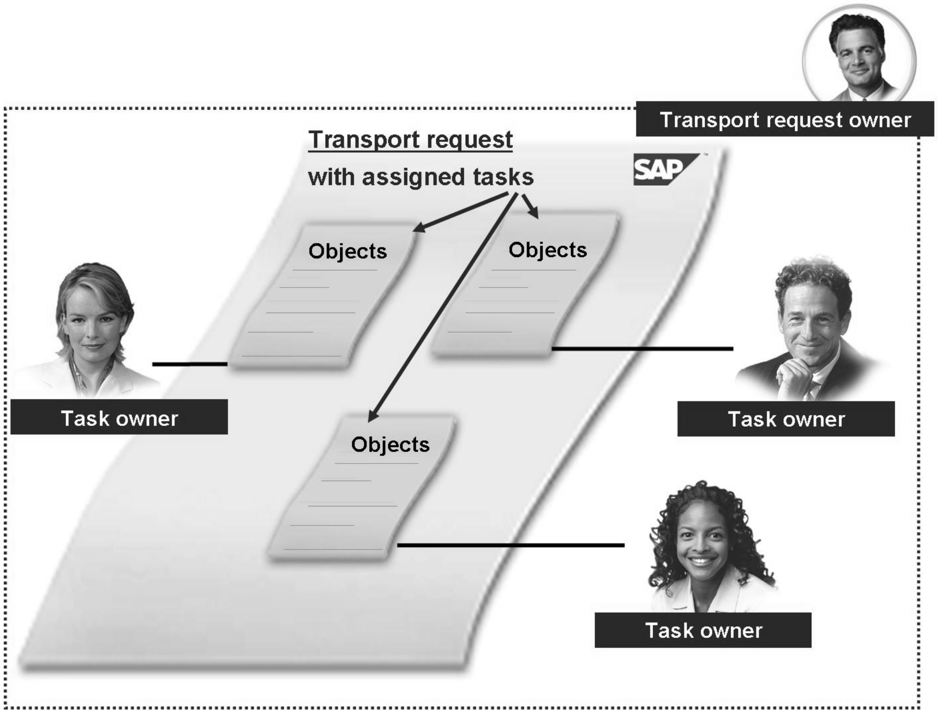


Figure 54: Transport Request and Tasks

The Transport Organizer allows you to work with a team on a development or customizing project, and allows the changes made to the SAP system to be recorded.

A transport request is used for transferring different changes together into another system. You can compare a transport request with a folder which contains different documents (the customizing objects) and subfolders (the tasks). A transport request initially contains one task per member assigned to the transport request.

Tasks are smaller units that are used by team members or customizers for recording the objects they want to transport. The changes a project member has done are stored in the tasks that the project member owns.

In general a customizing object can be listed in several tasks belonging to the same transport request. Likewise a team member can own several tasks belonging to a transport request.

The recorded objects are only transported in the context of the entire transport request. This function is used to manage and control the distribution of changes within a system landscape.

SAP recommends the following use of the transport function:

- The project manager or responsible person creates a transport request.
- The project members are assigned to the transport request by creating a task inside the transport request for each project member.
- The project members are recording their objects to their own tasks.

This leads to the following advantages for the management of transport requests:

- The project manager has the control of all changes that are recorded in his transport request.
- Objects in tasks cannot be transported individually. They are only transported with the whole transport request.
- With the release of the transport request the project leader is able to control when the objects are to be transferred to the other systems in the system landscape.

Managing Transport Requests in Projects

In this lesson you have seen so far how customizing changes are recorded to tasks and transport requests and how this is done technically. You also now have an idea of which organizational procedures during the creation and recording of transport request can help you get an overview of when the changes are transported.

Now you see which functions the IMG projects provide for the assignment of transport requests to an IMG project.

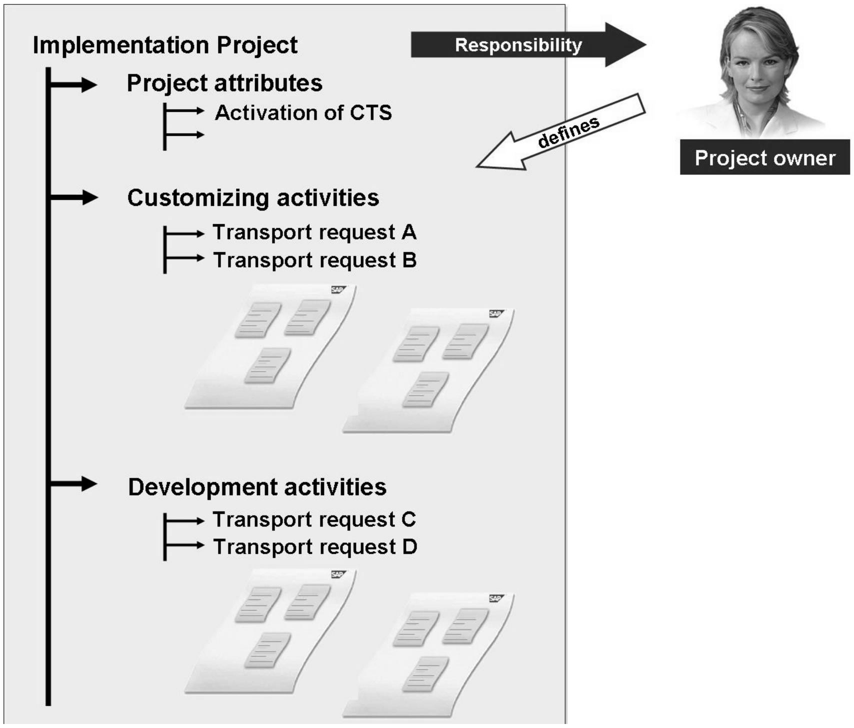
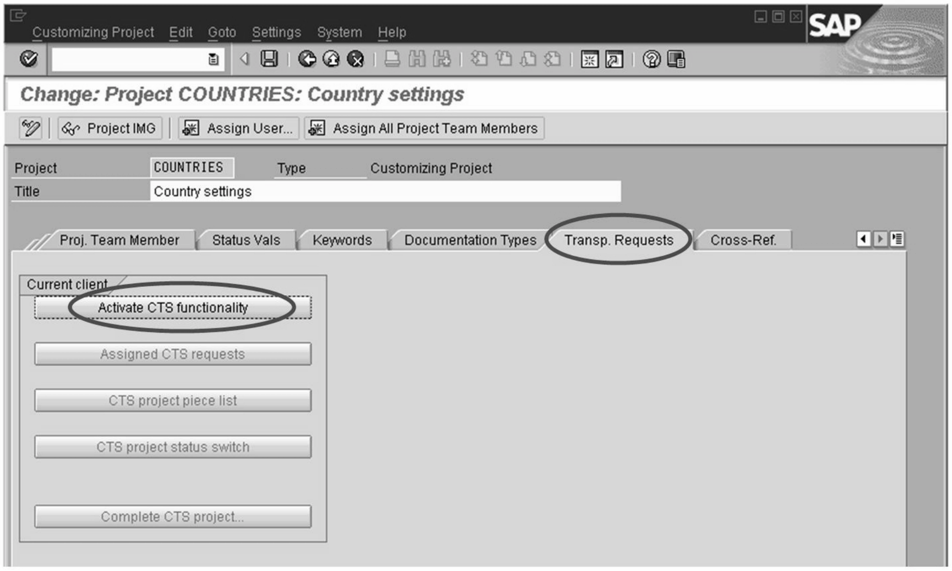


Figure 55: Managing Transports in Projects

As of SAP R/3 4.6A project IMGs may be linked to the Change and Transport System (CTS). The transport requests created and assigned by the project leader contain smaller units called tasks that have been allocated to the project team members. In these tasks, the changes made by the team members are recorded. A team member can also own several tasks belonging to a transport request.

With the CTS functionality in the IMG projects it is now possible to link the transport projects to the project IMG in which the changes are done. This is possible by activating the project function inside the IMG project. The activation can be done inside the IMG project administration.

If you want to use the CTS functionality in the IMG projects you have to activate the functionality in the IMG project first. IMG projects are client-independent. As a result, the CTS function may be separately enabled in different clients for the same IMG project. The following screen shot shows how to activate an IMG project. This activation can be done by changing the IMG project in the transaction SPRO_ADMIN.



Change: Project COUNTRIES: Country settings

Project COUNTRIES Type Customizing Project

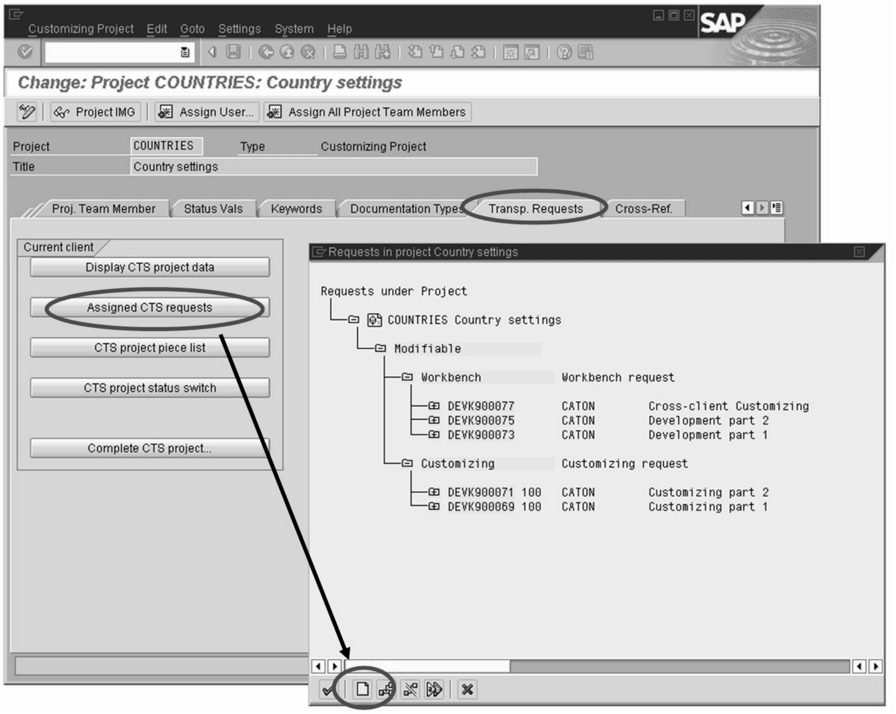
Title Country settings

Figure 56: Activation of CTS Project Functionality

When you activate the CTS project functions for an IMG project, you change the way the system reacts as follows:

- You can assign transport requests to the project.
- You can see this assignment in the request overview.
- When you save any changes that you have made to customizing settings in the project IMG, you are only proposed the requests that are assigned to the IMG project.

The assignment of transport requests to an IMG project can be done in two different ways. The first way is to assign the transport requests from the project administration (transaction SPRO_ADMIN). This is shown in the screen shot below. The second way is to assign a transport request to an IMG project and to use the Transport Organizer and to put the IMG Project into the properties of the transport request. This can be done by calling transaction SE09 and double-clicking the modifiable transport request and then choosing *Properties*.



Project COUNTRIES Type Customizing Project
Title Country settings

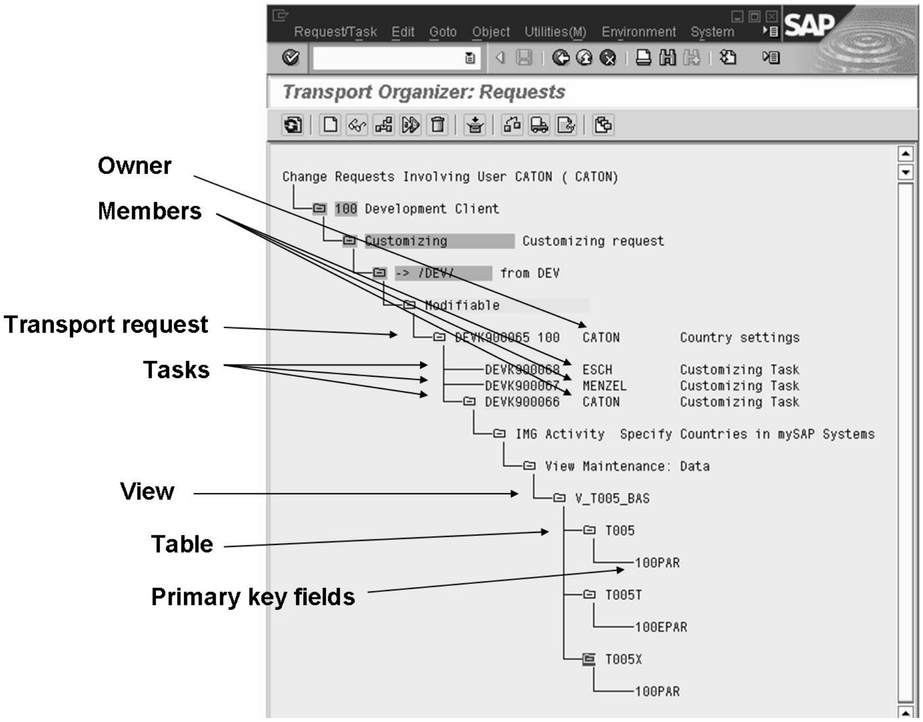
```
Requests under Project
  COUNTRIES Country settings
    Modifiable
      Workbench              Workbench request
        DEVK900077          CATON       Cross-client Customizing
        DEVK900075          CATON       Development part 2
        DEVK900073          CATON       Development part 1
      Customizing            Customizing request
        DEVK900071 100      CATON       Customizing part 2
        DEVK900069 100      CATON       Customizing part 1
```

Figure 57: Assignment of Transports to CTS Projects

In the import overview of the TMS you can sort requests according to their assignment to projects, and import them. For more information on CTS project functions, see application help under *SAP NetWeaver -> Solution Life Cycle Management -> Software Change Management -> Change and Transport System -> Change and Transport System*.

Using this function it is possible to transport projects separately as a unit containing several transport requests. In general, customizing projects should not overlap. However, during the release of transport requests, dependencies of requests belonging to different projects should be recorded, and projects checked for overlapping.

Transporting Customizing Changes


Request/Task Edit Goto Object Utilities(M) Environment System
SAP
Transport Organizer: Requests
Owner
Members
Transport request
Tasks
View
Table
Primary key fields

```
Change Requests Involving User CATON ( CATON)
  100 Development Client
    Customizing              Customizing request
      -> /DEV/            from DEV
        Modifiable
          DEVK900065 100   CATON          Country settings
            DEVK900068     ESCH           Customizing Task
            DEVK900067     MENZEL         Customizing Task
            DEVK900066     CATON          Customizing Task
              IMG Activity  Specify Countries in mySAP Systems
                View Maintenance: Data
                  V_T005_BAS
                    T005
                      100PAR
                    T005T
                      100EPAR
                    T005X
                      100PAR
```

Figure 58: Transporting Customizing Changes

Customizing transport requests usually contain changes or additions to Customizing. At the time the change is made in the system, the keys that are identifying the table entry are stored in the transport request's task. At the time when the transport request is exported, this transport request will extract the relevant table entries from the database of the SAP system and copies them to the transport directory. The content of the transport request always represents the customizing entries at the time when the request was released.

Because of the dynamic access to table entries during Customizing, both client-specific and cross-client customizing are not protected from being overwritten. Table entries are locked while the customizing transaction is being used (by the enqueue work process), but they are unlocked as soon as the changes are completed and saved to a transport request. For additional information see SAP Note 1916 in the SAP Service Marketplace.

Customizing changes can be logged. This is very often used for documentation such as who has changed a value of a customizing table. As you know, transport requests only record the table keys at the time when a customizing entry is changed. From a customizing request you can only obtain information on who has made the changes but no information about the state before and after the change. This information can be stored by activating the logging of customizing tables.

To activate the logging of changes to customizing tables, use IMG logging through transaction code SCU3. Prerequisite for *IMG Logging* is:

- In *Technical settings* in ABAP Dictionary maintenance (transaction SE11), the function *Log data changes* has been selected for the respective customizing table or view.
- The profile parameter *rec/client* is set for the appropriate clients.

Generally, all objects are transported to the target system in the state in which they exist in the source system. Objects transported from the source system overwrite objects in the target system that have the same names. Objects are deleted in the target system if they do not exist in the source system. For productive systems there are also some legal requirements in some countries that it make necessary to activate customizing table logging. Some legal requirements for activating the table logging can be found in SAP Note 112388.

If not otherwise specified in the TMS configuration, the customizing settings are transported using the standard transport layer.

Exercise 2: Transport Requests for Customizing

Exercise Objectives

After completing this exercise, you will be able to:

- activate the CTS functions for your project

Business Example

Customizing is ideally organized within projects. For each group, a project named *P_GROUP_##* is already defined. However, to use the Change and Transport System (CTS), the project leader must first activate the CTS functions.

Task 1:

1. Log on to the *DEV* system, client *100* and review your project *P_GROUP_##* (whereby ## is your group number) that was created in the system. Check the scope of your project.

Task 2:

1. Activate the CTS functions for your project. Assign some transport requests of type customizing to your project.

Task 3:

1. Are the project status switches for your project correctly maintained so that the project can be transported into the QAS system?

Solution 2: Transport Requests for Customizing

Task 1:

1. Log on to the *DEV* system, client *100* and review your project *P_GROUP_##* (whereby ## is your group number) that was created in the system. Check the scope of your project.

   a)

      1. To review the definition of your project *P_GROUP_##* choose transaction SPRO_ADMIN.
      2. Mark your project *P_GROUP_##* and choose *Display Project*. In order to check the scope of the project, choose the push-button *(Display) Project-IMG.*

Task 2:

1. Activate the CTS functions for your project. Assign some transport requests of type customizing to your project.

   a)

      1. Enter your project in change mode.
      2. Select the *Transport Request* tab and choose *Activate CTS functionality*.
      3. Confirm the information dialog box related to CTS activation. To assign change requests of type customizing, choose *Assigned CTS requests*.
      4. In the next screen, choose *Create Request*. In the selection dialog box, choose *Customizing request*. To create the customizing request, enter a short description and notice that the CTS project is already filled in. Enter all project members, that should use this transport request. Save your work. Notice that the system generates a transport request name and task names in the format of *<SID_P>K9<5 digit number>*.
      5. Check that a new transport request has been created with one task for each project member. .

         Repeat this step three times, in order to create four customizing requests in total for your project.

         Save your work

Continued on next page

Task 3:

1. Are the project status switches for your project correctly maintained so that the project can be transported into the QAS system?

   a)

      1. From the *Transport Request* tab, in the *Change: Project* screen, choose the *CTS project status switches* for all configured transport targets. Choose *Enter*. Set your project so that:

         1) Requests can be created and released on system DEV in client 100 and

         2) Requests can be imported on system QAS.

         Note: You should not need to change the status switches, because they should already be set correctly.

Lesson Summary

You should now be able to:

- Create transport requests and tasks
- List the tools used for managing transport requests

Lesson: Customizing Procedure

Lesson Overview

In this lesson, you will learn how customize procedures should be set up. You will also identify how to test customizing transport requests. In addition, you will get an idea about cross-client Customizing.

Lesson Objectives

After completing this lesson, you will be able to:

- Describe how to customize
- Follow steps in procedure for customizing an SAP system implementation
- Define cross-client customizing

Business Example

Customizing changes require testing and approval before being promoted to production. These implementations require communication between project leaders, who create transport requests, and the TMS administrator who is responsible for importing the transport requests. Procedures need to be defined to keep the movement of transports through the landscape consistent with the transport strategy.

Steps in Customizing Procedure

In an SAP implementation, where customizing and development changes are integral to the system being available, projects must be executed in a structured environment using defined procedures, to minimize the threat of downtime caused by bugs.

The goal of your project organization must be to divide the large number of activities among the project team so the team members do not interfere with each other's work. You must make sure that work that logically belongs together but is being performed by different team members is still connected. This is carried out by dividing the tasks in a customizing project among three roles, each of which is responsible for performing certain tasks.

Project Manager

The project manager creates the transport requests and assigns the appropriate members to them. When a memeber is assigned to a transport request, the system creates a task; the settings for each member are recorded in his or her task. For the transport to other systems, the project manager can release transport requests that he or she created.

Customizer/Developer

The customizers or developer perform their Customizing from the IMG or their development and allocate their settings to a transport request and thus, to their individual task. The customizers can copy their settings to the *TEST* client to test them. This, of course, only makes sense for client-dependent data. They are authorized to release their own tasks in a transport request but are not allowed to release the transport request.

TMS Administrator

The TMS administrator uses the TMS to transport released transport requests to subsequent systems in the landscape using the predefined paths.

Normally, application consultants and employees in the department handle the roles of project manager and the customizers and developers. The project manager decides which customizing settings to perform and how to divide them among the project team members, who execute the customizing transactions. The TMS administrator is responsible for the transport between the systems along the transport path, after the transport requests are released to the TMS.

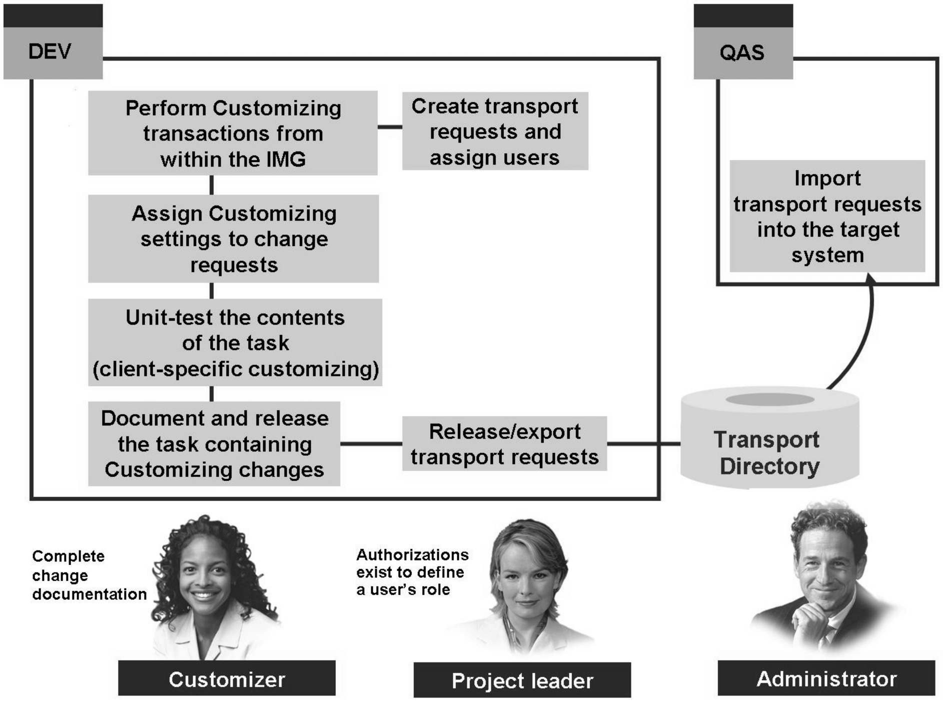

DEV
Perform Customizing transactions from within the IMG
Create transport requests and assign users
Assign Customizing settings to change requests
Unit-test the contents of the task (client-specific customizing)
Document and release the task containing Customizing changes
Release/export transport requests
QAS
Import transport requests into the target system
Transport Directory
Complete change documentation
Authorizations exist to define a user's role
Customizer
Project leader
Administrator

Figure 59: Customizing Procedure

The Transport Organizer and the Transport Management System are designed for supporting this task split. The sequence of the different customizing steps are shown in the picture above. During normal customizing work, the sequence of the steps is as follows:

1. The project leader first assigns a transport request to a project and assigns the subsidiary tasks to the users involved. These users perform customizing changes that are recorded in the transport request.
2. After Customizing is completed, users must release their tasks so that the transport request can then be released from the source system for export to the operating system.
3. The transport request can now be imported into the subsequent systems.

Experience has shown that by utilizing these three, clearly defined roles and instituting strict guidelines for the procedures and documentation of Customizing, the overall customizing process is easier to manage and the chance for errors in production is greatly reduced.

Note: The authorization (security) administrator enforces these roles by assigning the appropriate authorizations to each user ID. For example, the customizer must be able to execute the assigned customizing transactions and be able to release his or her own tasks, but not be able to create or release transport requests. SAP delivers standard roles for the team leaders, team members, and TMS administrators. For more information on authorization and role management, refer to SAP course *ADM940*.

So far you have seen how the creation and assignment of transport requests can be done. On the following pages you will see how the other steps are working technically.

Recording of Customizing Changes

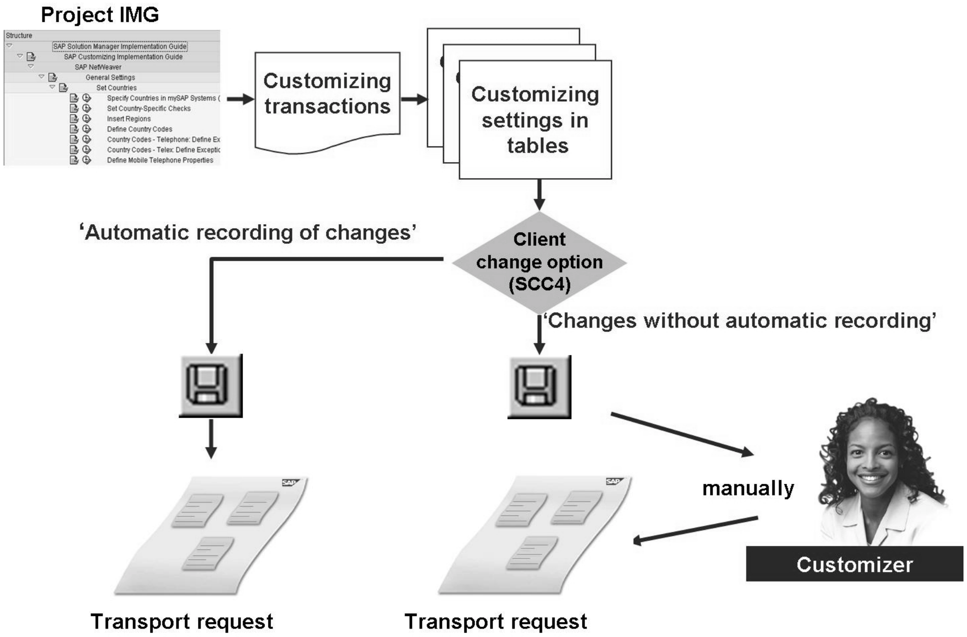

Project IMG
Structure
SAP Solution Manager Implementation Guide
SAP Customizing Implementation Guide
SAP NetWeaver
General Settings
Set Countries
Specify Countries in mySAP Systems (
Set Country-Specific Checks
Insert Regions
Define Country Codes
Country Codes - Telephone: Define Ex
Country Codes - Telex: Define Exceptic
Define Mobile Telephone Properties
Customizing transactions
Customizing settings in tables
Client change option (SCC4)
'Automatic recording of changes'
'Changes without automatic recording'
manually
Customizer
Transport request
Transport request

Figure 60: Recording of Customizing Changes

A customizing transaction is a transaction for setting customizing table entries. To use a customizing transaction, you do not have to know about the technical aspects of where and how a business object is maintained, or which transactions are used to access and change certain fields in specific tables.

After the customizing settings have been changed using a customizing transaction, the settings should be saved as follows:

- When a client is configured to automatically record changes, the settings are automatically saved to a transport request managed by the Transport Organizer.
- If the client is not configured for automatic recording of changes, the settings are saved but not recorded in a transport request. However, they can be included in a transport request manually. This can be done from the specific customizing transaction or from within the Transport Organizer.

All customizing transactions in the IMG also allow entries to be manually saved to a transport request .

Note: Some transactions are classified as manual transports. These transactions must be manually added to a transport request for transport to their target system. In addition, some customizing transactions have transport steps which differ for those indicated in the above graphic.

To view transport dependencies for items within the IMG: from the initial screen of the SAP system, choose transaction SPRO, *Additional Information → Technical Data → Transport Type*.

SAP recommends all customizing changes originate in one client only and all changes be saved to transport requests. This control is put in place by the system administrator via use of the Client Change Options.

Testing of Customizing

Testing of changes in the development is very important. Only tested and error-free transport requests could be released in the development system. The advantage of these procedure is that the amount of transport requests (only a few correction transports) and error corrections between the DEV and the QAS system is dramatically reduced, while the most errors are detected early before releasing a request and not lately during testing in the QAS System.

- Before releasing a customizing transport request, perform a unit test to:
  - Test the functionality of the customizing within the transport request
  - Verify that the content of the transport request is complete
- Maintaining a separate client for testing allows:
  - True unit testing
  - Maintenance of test data without the risk of creating customizing-dependent data
- Transaction SCC1 copies changes from one client to another based on:
  - A task
  - A transport request
  - A transport request and its tasks

All customizing must be tested prior to production in two ways: unit testing and system testing. Perform unit testing first.

Unit testing is the testing of individual customizing settings. System testing is the testing of all customizing settings together. Unit testing should be performed by the customizer and completed before the release of the transport requests.

Unit testing typically requires application data. Because many customers find it advantageous to keep their customizing client free of application data, another client is created with the necessary application data for unit testing.

Thus, prior to releasing a transport request, copy the recorded changes to a separate client for unit testing. To copy the contents of a transport request or one of the constituent tasks to another client within the same SAP system, use transaction SCC1.

After unit testing the tasks, a transport request can be released. Unit testing alone, however, is not sufficient for transporting customizing changes to production. After unit testing, the change needs to be tested with all other customizing settings in system testing. This is done to ensure that all system settings function properly together.

In the SAP landscape, all customizing occurs on the *CUST* client but no application data exists in the *CUST* client. Unit testing occurs in the *TEST* client where application data does exist. The team member is responsible for copying their work to the TEST client and must do so using the SCC1 client. To do this, the customizer must also have a user ID on the *TEST* client and use the following procedure.

Transaction *SCC1*:
Copy based on a transport request
(released or not released)

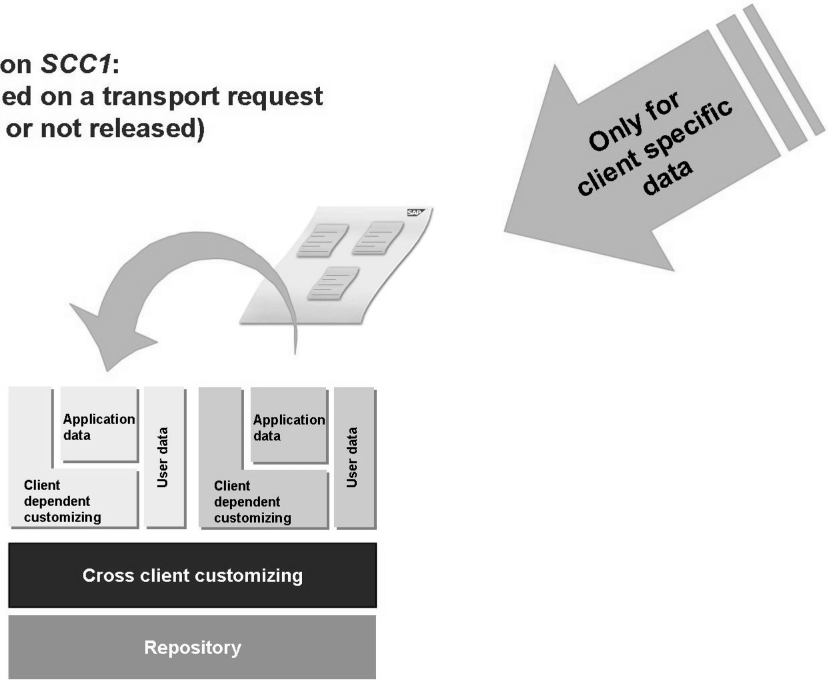


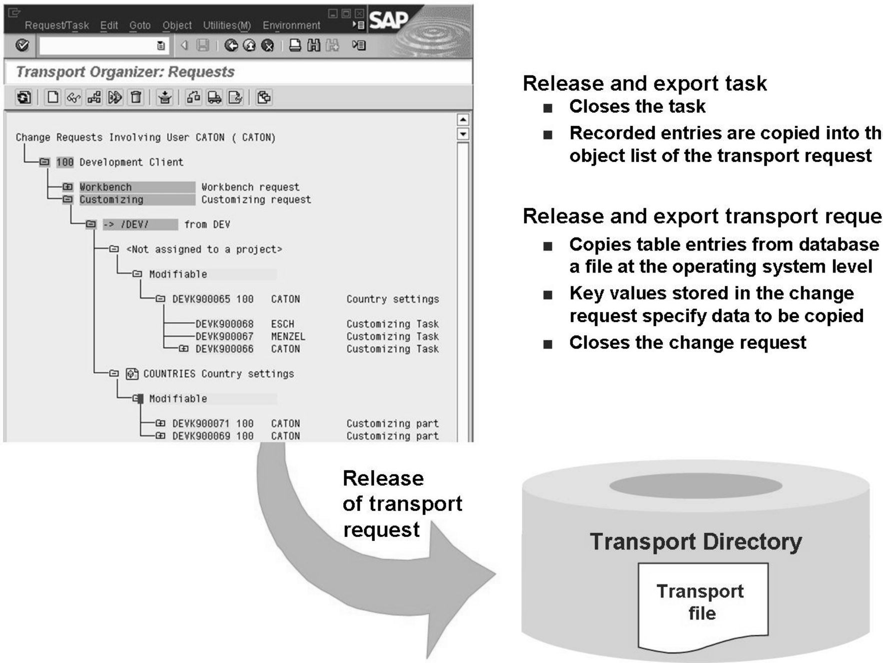
Figure 61: Testing Customizing Change Requests

Note: When copying the contents of a task to the unit test client, the task does not have to be released! Not releasing the task allows any errors identified during the unit test process to be corrected and assigned to the same task. Once a task is released, no further changes can be recorded to it.

To copy the contents of a transport request from one client to another client, use Transaction SCC1. Log on to the target client, that is, the unit test client. Enter the source client and the transport request to be copied.

Note: If a transport request contains cross-client objects, these objects are not copied.

Releasing Customizing Change Requests


Request/Task Edit Goto Object Utilities(M) Environment
SAP
Transport Organizer: Requests

```
Change Requests Involving User CATON ( CATON)
  100 Development Client
    Workbench               Workbench request
    Customizing             Customizing request
      -> /DEV/           from DEV
        <Not assigned to a project>
          Modifiable
            DEVK900065 100   CATON       Country settings
              DEVK900068     ESCH        Customizing Task
              DEVK900067     MENZEL      Customizing Task
              DEVK900066     CATON       Customizing Task
        COUNTRIES Country settings
          Modifiable
            DEVK900071 100   CATON       Customizing part
            DEVK900069 100   CATON       Customizing part
```

Release and export task

- Closes the task
- Recorded entries are copied into the object list of the transport request

Release and export transport request

- Copies table entries from database to a file at the operating system level
- Key values stored in the change request specify data to be copied
- Closes the change request

Figure 62: Releasing Customizing Change Requests

Promoting changes recorded in a transport request begins with releasing the relevant tasks. Releasing a task indicates that the owner of the task has completed Customizing or development work, that the unit testing was successful, and that the appropriate documentation is complete.

- The task contains a recorded object (change).
- The task has been documented.
- The task is owned by the person releasing it or the person releasing it has the appropriate system authorizations.

To release a task, execute the following procedure.

- Enter the Transport Organizer by executing transaction SE09.
- Choose *Display*. The request overview displays.
- To view all tasks in a specified transport request, expand the tree structure.
- Position the cursor on the task you wish to release and choose the Release icon.

By releasing a transport request, you indicate that it has sufficient documentation, the changes recorded in it have been tested, and the changes are ready to be transported using the TMS transport routes. During the export process triggered by the release, the objects recorded in the transport request are copied from the SAP database to operating system level files in the transport directory. In addition, a record of the transport request is automatically added to the appropriate import queues of the systems defined in the TMS.

Releasing and exporting a single transport request generates export logs and import logs. Testing in the quality assurance system and QA approval sign off are necessary before import into the production system. To support the validation process and limit the technical and administrative overhead, SAP recommends to transport changes that belong to the same project together in a limited number of transport requests. This is ideally done by assignment of tasks within the transport requests by the project leader or the person responsible for the transport.

If you discover that transport requests should be merged during the project work or before releasing the transport request, you can combine multiple transport requests into one single transport request. This is usually the case when the transport requests are related and need to be tested together. Merging requests can be done explicitly in the Transport Organizer by choosing *Utilities →Reorganize →Merge Requests*.

Cross Client Customizing

As you have learned in this chapter before, in the SAP system not only client-specific customizing exists. There are also client-independent customizing and repository objects.

- Client-specific customizing

  These are typically customizing tables where the client field is on the first position in the table key. A example of this is the customizing table *T005* with the country customizing. The entries in that table only affect the client that is specified in the key field of the entry.

- Client-independent customizing

  These are typically customizing tables without a client field in the key. An example of a client-independent customizing table is the table *T000* in which all clients that exist in a system are stored.

- Repository objects

  Repository objects are for example table definitions in the ABAP dictionary or programs and reports.

The most difficult task in the distribution of customizing changes is distinguishing between cross-client and client-specific customizing and knowing which client-specific customizing activities affect master data.

To determine the client dependency of IMG transactions in a project IMG, from the IMG menu, choose *View → Additional information → Technical Data -> Client Dependence*.

Cross-client customizing affects one of the following:

- Cross-client customizing objects, which are repository objects generated by customizing demands. To ensure proper transport, assign these repository objects to a customer package. Examples of these objects are search helps, condition tables, and hierarchies.
- Global customizing settings are the standard system settings and configurations in various tables whose key value does not contain a client number. Examples of these settings are calendars, printer settings, communication settings, and schedules.

While client-specific customizing changes are recorded to customizing transport requests, cross-client changes must be saved to Workbench transport requests. Therefore, changes to cross-client customizing objects, global settings and Repository objects require a Workbench transport request.

Customizing Request Versus Workbench Requests

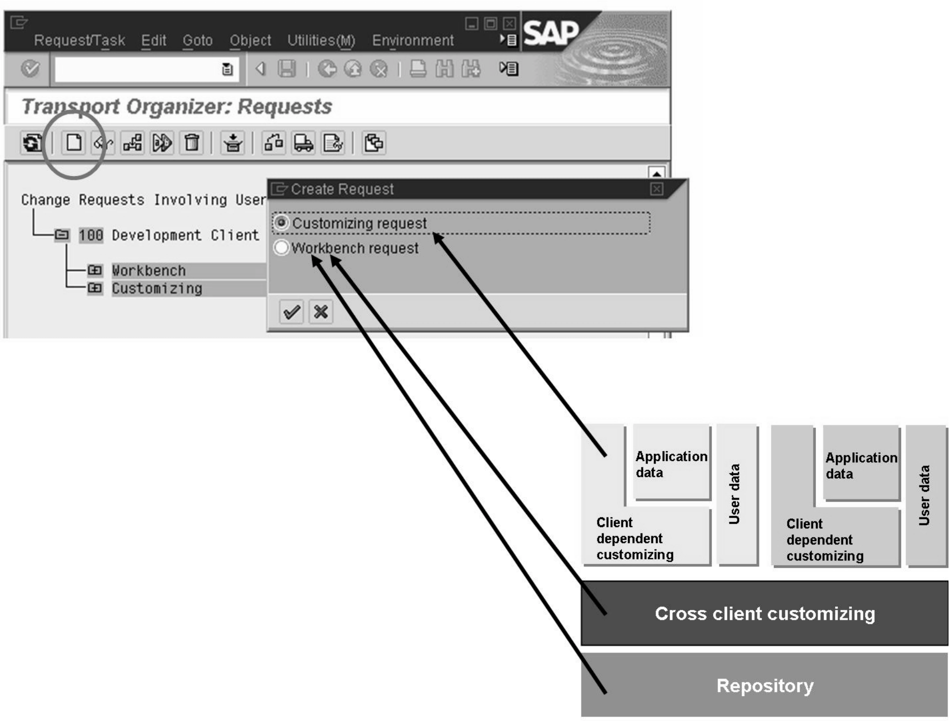


Figure 63: Customizing And Workbench Request

As we have mentioned before, there are different types of transport requests with special properties.

The first type is the transport request of type customizing. This is a transport request for copying and transporting settings from client-specific tables. Client-independent Customizing or workbench objects cannot be recorded into this kind of transport request.

Another type of transport request is of type workbench. This is a transport request for transferring and transporting Repository objects and changed system settings from client-independent tables.

With this split between client-specific data that can be recorded in transport requests of type customizing and the transport request of type workbench, for example a project leader or a person responsible for the transport can control who is allowed to transfer client-independent changes by assigning tasks inside the workbench requests.

Planning Customizing Change Management

Now you know the different customizing tools and the technical settings that are required for the usage of these tools. As a summary of all that, we will now discuss what is important for planning the customizing change management.

From a process point of view, it is essential to establish policies on how customizing and development will be carried out and by whom.

Define Customizing policies:

- **Define policies for creating the project IMGs and views within a project, and the use of project status and documentation.**
- **Establish a single source client for all Customizing.**

Restrict access to transport request management tasks:

- **Determine change management roles**
- **Assign appropriate user authorizations**

Establish project teams:

- **Establish project team leaders to define projects, and to assign tasks within transport requests to team members.**
- **Provide training on the use of change request management tools to all project teams.**

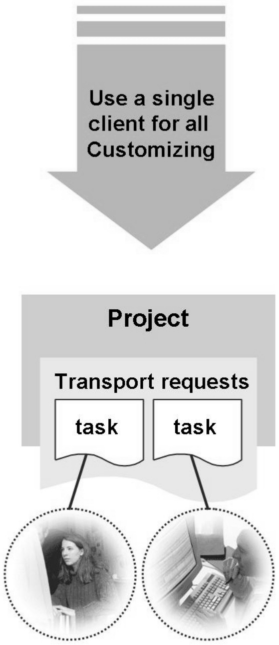


Figure 64: Planning Customizing Change Management

Customizing is the standard way of adapting the functions of the delivered SAP components to the requirements of a company. Project control tools allow easy documentation and monitoring of each phase of the SAP implementation and Customizing.

Customizing is required for implementations of SAP systems. Because customizing is a highly integrated process, both between functional modules and within the modules themselves, all customizing activities must be performed in a single client, usually known as the development client. Development tasks should also be restricted to this client, thus ensuring that there is a single environment for all implementation efforts.

The project leader is responsible for creating transport requests for all project team members at the start of a development or customizing project.

After project team members complete their work, they save their changes in the tasks that comprise the transport request, and document the changes in the task. When all project team members have released their tasks, the project leader releases the transport request for transport to other SAP Systems within the system landscape.

The project managers or team leaders are responsible for creating the appropriate project IMGs for the specific business areas and assigning who is responsible for executing the specific customizing transactions. The project management function enables you to assign tasks in the Transport Organizer. The project leader should activate the CTS functions of the project, create transport requests for recording and transporting the customizing settings, and add users to the transport requests.

The project managers or team leaders are also responsible for training the team members on the CTO tools and processes, as well as establishing and enforcing documentation and unit testing standards. Since SAP recommends that transport requests should contain testable units of work, it is best to have a minimum number of transport requests with many tasks. This reduces the number of transport request moving through the transport system and makes it easier to troubleshoot problems when they arise.

Exercise 3: Customizing Procedure

Exercise Objectives

After completing this exercise, you will be able to:

- Save client-specific customizing settings to a customizing transport request
- Review the content of a transport request and release the task
- Release and export a transport request

Business Example

Regardless of the application area, the customizing procedures to record, copy, and test changes are the same.

Task 1:

A part of the implementation project involves setting up country definitions. Perform the customizing activity in the SAP system to accomplish this task. Execute the required customizing transaction and save your work to a transport request assigned to your project.

1. Execute the transaction to access your project and perform the necessary customizing activities for the implementation.

Task 2:

Before transporting your customizing request to the quality assurance system (QAS), perform a unit test in client 200 of the DEV system.

1. Log on to the test client 200 of the DEV system and copy your customizing settings.

Task 3:

Change to client 100. View the content of the transport request and release the task that contains the customizing performed in the previous exercise tasks.

1. From the Transport Organizer, review the content of the task of your customizing transport request and release the task.

Task 4:

Release and export the transport request.

1. Release and export the entire transport request to the transport system.

Solution 3: Customizing Procedure

Task 1:

A part of the implementation project involves setting up country definitions. Perform the customizing activity in the SAP system to accomplish this task. Execute the required customizing transaction and save your work to a transport request assigned to your project.

1. Execute the transaction to access your project and perform the necessary customizing activities for the implementation.

   a) Call transaction SPRO_ADMIN. The *Customizing: Project Administration* screen appears.

   b) Double-click your project and select the *Display Project IMG* button.

   c) From the *Change: Project View* screen, choose *SAP Solution Manager Implementation Guide SAP → SAP Customizing Implementation Guide → NetWeaver → General setting → Set Countries → Specify Countries in mySAP Systems*. Confirm the information popup, that the same customizing settings are maintained in other projects.

   d) Choose *New Entries* to add your customizing entry.

   e) In the *Country* field, enter the first character of the country in which you are taking the course and your group number. For example: `U` for the United States and `01` for group 01. In the *Name* field, enter the name of the country. Also choose a date format.

   f) Choose *Save*. The *Prompt for Customizing Request* dialog box displays. Select a customizing request assigned to your project and choose *Enter*.

   g) Choose *Back* three times to return to the *Customizing: Project Administration* screen.

Continued on next page

Task 2:

Before transporting your customizing request to the quality assurance system (QAS), perform a unit test in client 200 of the DEV system.

1. Log on to the test client 200 of the DEV system and copy your customizing settings.

   a) Copy your customizing settings with transaction SCC1 (Copy by Transport Request): choose your transport request and check the check box *Including Request Subtasks*, because your tasks are not released to the transport request until now.

   b) Choose *Start Immediately*.

   c) After the copying is finished, perform a unit test by looking at your country settings. You can do this with your project IMG.

Task 3:

Change to client 100. View the content of the transport request and release the task that contains the customizing performed in the previous exercise tasks.

1. From the Transport Organizer, review the content of the task of your customizing transport request and release the task.

   a) Choose transaction SE09. In the initial screen of the Transport Organizer, select both request types *Customizing* and *Workbench*. Choose *Display*. The transport requests assigned to your project are displayed.

   b) Select the folder in front of your customizing transport request to display all task associated with the request.

   c) Select the folder in front of your task. A folder that contains your customizing entries is displayed.

   d) Continue to drill down on the folders until you see the view and the tables of the view that contains your customizing data. Select the folder of the first table displayed. You should see the primary key of the customizing data that you entered, the client and the country. The data behind this key will be transported to the subsequent SAP systems in the landscape.

   e) Release the task in the customizing request to the transport request by selecting the task and selecting the *Release directly* icon. A check mark now displays behind the task number indicating that the task has been released.

Continued on next page

Task 4:

Release and export the transport request.

1. Release and export the entire transport request to the transport system.

   a) From the *Transport Organizer: Request* screen, select the transport request and choose the *Release directly* icon.

   b) The transport request is being released to the transport system. Review the logs generated for the export.

Lesson Summary

You should now be able to:

- Describe how to customize
- Follow steps in procedure for customizing an SAP system implementation
- Define cross-client customizing

Lesson: Transport Requests for Development

Lesson Overview

In this lesson, you will learn about planning change management for development. You will identify the different implementation tools provided by SAP for Customizing and development. You will also identify the role of Transport Organizer in recording the transport requests.

Lesson Objectives

After completing this lesson, you will be able to:

- Plan change management for development
- Identify various implementation tools used for Customizing and development
- Describe the function of the Transport Organizer

Business Example

As the transport administrator for your company, you must ensure that development changes are distributed to all SAP systems in the system landscape. Distributing development changes to all systems in the system landscape will provide a consistent and stable implementation of business processes and customer development. To streamline the process of distributing changes, the transport administrator must be familiar with the Transport Organizer, which controls the distribution of changes throughout the landscape.

Planning Change Management for Development

Before starting the development in a system, you should plan how the change management for development is done.

The areas which have to be described in the change management procedure for development are the following:

- Restrict Repository object changes:
  - Create a single SAP system for all developments

    Note: Usually one per SAP system landscape

  - Ensure proper system and client change options
  - Assign appropriate user authorizations
- Define development standards
  - Use packages to group Repository objects
    - Establish standards for development and documentation
    - Maintain versions
- Establish project teams
  - Provide all project teams with training on transport request management tools.

    Assign team leaders to projects and assign tasks within transport requests to team members.

The tools that the SAP systems provides for change management are based on creating, documenting, and distributing transport requests. The customer must set up the infrastructure and procedures for the management, verification, and testing of these development changes.

Recommendations for development change management include the following:

- Perform development efforts in a single environment only: the development system. Accordingly set the system change options.
- Use packages to group functionally related Repository objects. The transport layer assigned when creating a package enables the same predefined transport route to be used for all objects in the package.
- When releasing a transport request, document the purpose and the status of the changes.
- To maintain security, use authorizations to control which users can create, modify, or release transport requests. SAP delivers sample authorization profiles that provide the system access required for various levels of responsibility in change management.

Most customers have to define development rules, customizing rules and transport rules which are describing how and where changes are made, how and where these changes are tested, how the quality assurance is done and how and who creates, release and imports transport requests in the system landscape.

Customizing versus Development

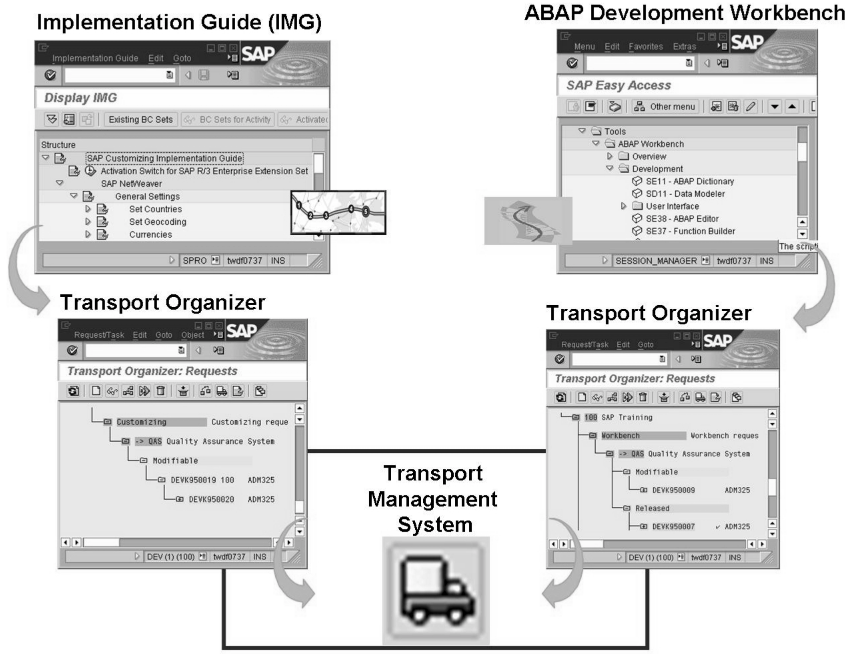

Implementation Guide (IMG)
Implementation Guide Edit Goto
Display IMG
Existing BC Sets
BC Sets for Activity
Activated
Structure
SAP Customizing Implementation Guide
Activation Switch for SAP R/3 Enterprise Extension Set
SAP NetWeaver
General Settings
Set Countries
Set Geocoding
Currencies
SPRO twdf0737 INS
ABAP Development Workbench
Menu Edit Favorites Extras
SAP Easy Access
Other menu
Tools
ABAP Workbench
Overview
Development
SE11 - ABAP Dictionary
SD11 - Data Modeler
User Interface
SE38 - ABAP Editor
SE37 - Function Builder
The script
SESSION_MANAGER twdf0737 INS
Transport Organizer
Request/Task Edit Goto Object
Transport Organizer: Requests
Customizing Customizing reque
-> QAS Quality Assurance System
Modifiable
DEVK950019 100 ADM325
DEVK950020 ADM325
DEV (1) (100) twdf0737 INS
Transport Organizer
Request/Task Edit Goto
Transport Organizer: Requests
100 SAP Training
Workbench Workbench reques
-> QAS Quality Assurance System
Modifiable
DEVK950009 ADM325
Released
DEVK950007 ✓ ADM325
DEV (1) (100) twdf0737 INS
Transport Management System

Figure 65: Customizing Tools versus Development Tools

SAP provides various implementation tools for Customizing and development.

- For Customizing:

  The Implementation Guide (IMG) is the main customizing tool. Once you decide which business functions you require, the IMG automatically generates a hierarchical list of steps or customizing transactions for Customizing.

  The Transport Organizer (transaction SE09) records customizing changes in transport requests, which can be released to the transport system for export to other systems in the system landscape.

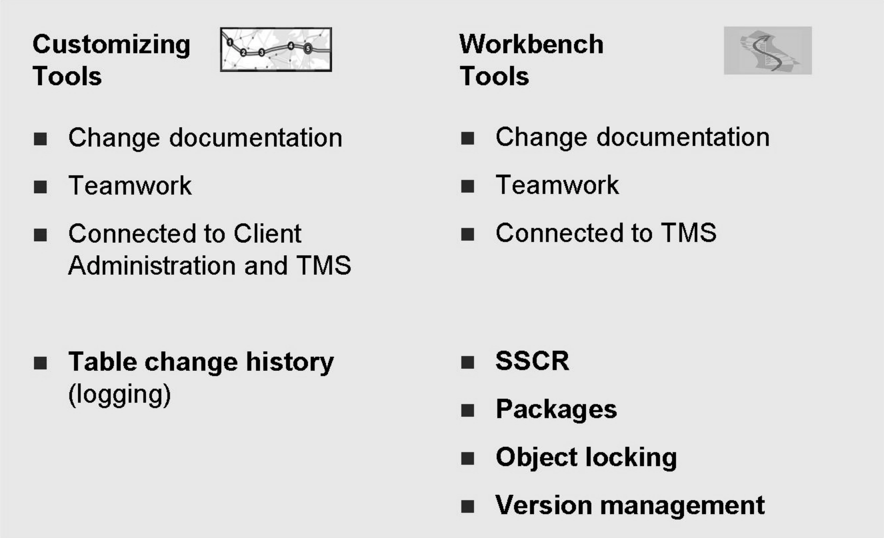
- For development:

  The ABAP Development Workbench provides access to development tools, which cover the entire software development cycle. You can use these tools for customer-specific development and for SAP enhancements of business.

  The Transport Organizer (transaction SE09) also records ABAP Development Workbench changes in transport requests, which can then be released to the transport system for export to other systems in the system landscape.

  The Transport Organizer is completely integrated with the Transport Management System (TMS).

The difference between these tools are shown in the following figure:



Customizing Tools

- Change documentation
- Teamwork
- Connected to Client Administration and TMS
- **Table change history** (logging)

Workbench Tools

- Change documentation
- Teamwork
- Connected to TMS
- **SSCR**
- **Packages**
- **Object locking**
- **Version management**

Figure 66: Transport Organizer for Customizing and Development

The Transport Organizer records the customizing changes and ABAP Development Workbench changes in two types of transport requests.

Customizing changes are saved to customizing transport request. Development changes are saved to Workbench transport requests.

Customizing requests possess the following characteristics:

- Change documentation
- Teamwork
- Table change history (logging)
- Connected to Client Administration and Transport Management System

Customizing changes consist of table entries

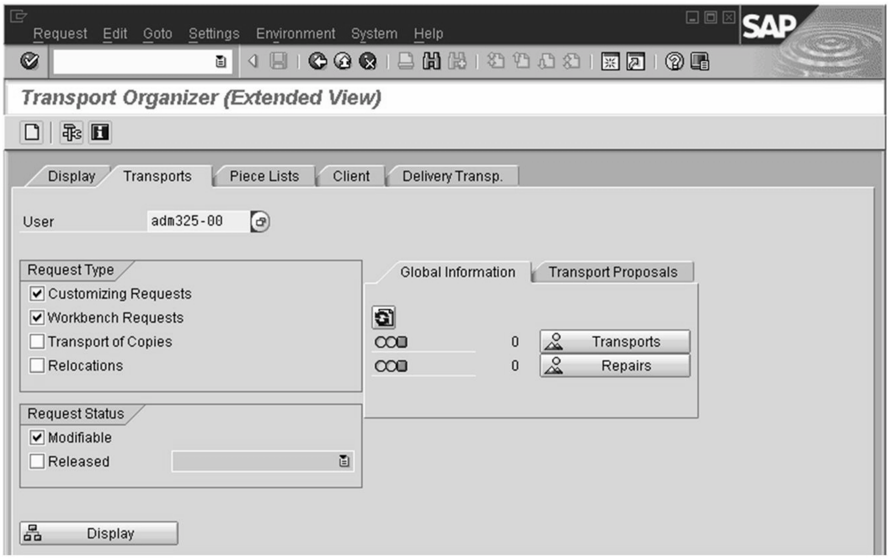
Workbench requests have the following characteristics:

- Change documentation
- Teamwork
- SSCR
- Packages
- Object locking
- Version management
- Connected to Transport Management System

Workbench transport requests concern changes to Repository objects, such as ABAP programs, screens, data dictionary objects and global documentation.

Compared to Customizing, change management for development provides additional functionality. These are the following:

- Registration of developers within SAP Software Change Registration (SSCR)
- Packages for Repository objects
- Locking of Repository objects during development
- Versioning of Repository objects



User adm325-00

Request Type
☑ Customizing Requests
☑ Workbench Requests
□ Transport of Copies
□ Relocations

Request Status
☑ Modifiable
□ Released

0 Transports
0 Repairs

- **Creates and displays transport requests for Customizing and development**
- **Displays global transport information**
- **Provides access to special tools**

Figure 67: Transport Organizer (Extended View)

The Transport Organizer can be used for Customizing and workbench transport requests. It does the following:

- Displays transport requests
- Displays global transport information
- Provides access to special tools

The Transport Organizer creates, manages, releases, and analyzes transport requests for development.

To access the Transport Organizer, use transaction SE09 or SE01 for extended functionality.

All Workbench transport requests that belong to a specific user are displayed according to a set of standard selection criteria. Selection options include user (owner), request type, request status, and date.

The screen area *Global Information*, provides a quick graphical overview of the status of transported transport requests and repairs.

SSCR Registration Concepts

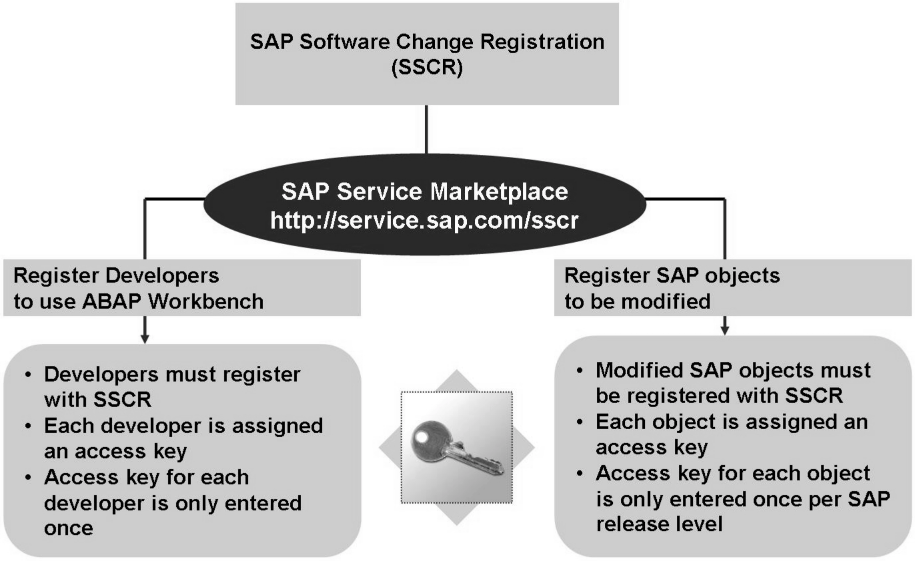

SAP Software Change Registration (SSCR)
SAP Service Marketplace http://service.sap.com/sscr
Register Developers to use ABAP Workbench
Register SAP objects to be modified
• Developers must register with SSCR
• Each developer is assigned an access key
• Access key for each developer is only entered once
• Modified SAP objects must be registered with SSCR
• Each object is assigned an access key
• Access key for each object is only entered once per SAP release level

Figure 68: Concepts of SSCR Registration

SAP Software Change Registration (SSCR) process has certain key concepts:

Any user in an SAP system who wishes to use the ABAP Workbench to create, modify, or delete Repository objects, including customer objects, must be registered using the SAP SSCR process. Such users are often referred to as development users or developers.

As a result of the registration process, each developer is assigned an access key. The access key is entered and saved on the developer's system in the table *DEVACCESS*.

The access key is associated with the developer's logon ID and the SAP systems license number. The developer is prompted for the access key during the initial attempt to create or change a Repository object.

You must register developers and all SAP Repository objects (not customer objects) that are to be modified. When registering an object, you must supply the object program ID, object type, object name, the SAP system's license number, and the release of your system. After registering an SAP object in an SAP system and applying the access key, the key is stored in the database table *ADIRACCESS*. This will ensure that further changes to the object do not require another key. Registered object keys become invalid after a release upgrade.

The Registration can be done via the SAP Service Marketplace via logon to the SAP Service Marketplace *http://service.sap.com/sscr* .

SSCR provides development reliability, rapid error correction, and high system availability. This can be done by limiting the access to the development and object keys. Most customers are doing the registration of objects and developers centrally.

Repository objects and Attributes

The naming conventions for SAP Repository objects are listed in the picture below.

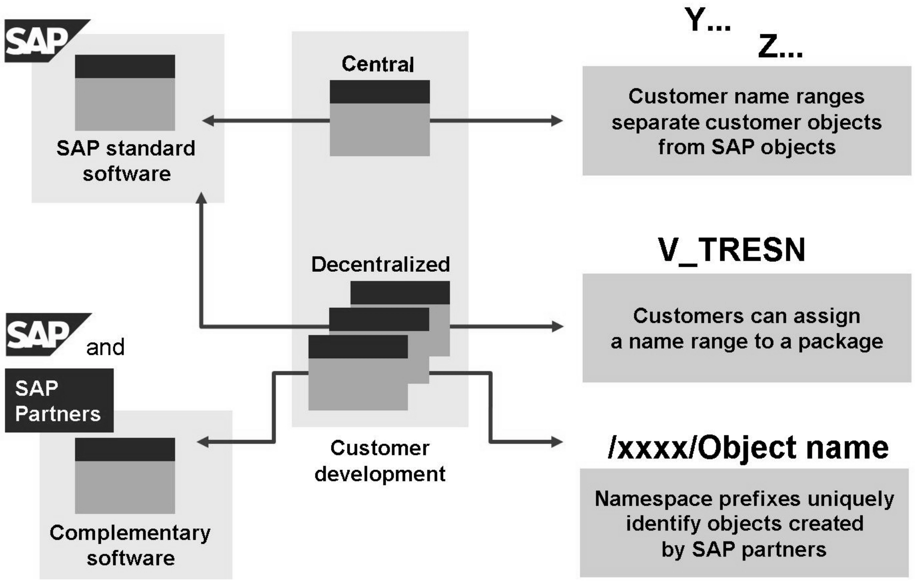

SAP
SAP standard software
Central
Y...
Z...
Customer name ranges separate customer objects from SAP objects
Decentralized
V_TRESN
Customers can assign a name range to a package
SAP and
SAP Partners
Complementary software
Customer development
/xxxx/Object name
Namespace prefixes uniquely identify objects created by SAP partners

Figure 69: Correctly Naming SAP Repository Objects

To prevent conflicts with object names when creating Repository objects, developers must follow naming conventions. Extended name lengths allow you to use descriptive names. Object names should clearly describe the function of the Repository object.

Name ranges differentiate between SAP Repository objects and customer Repository objects. Customer Repository object names begin with a *Y* or *Z*.

Hint: There are also other naming conventions: for example customer fields in an SAP table definition must beginn with *ZZ*, **not** with *Z*. Or: if the object name is a number, the customer name range begins with *9*.

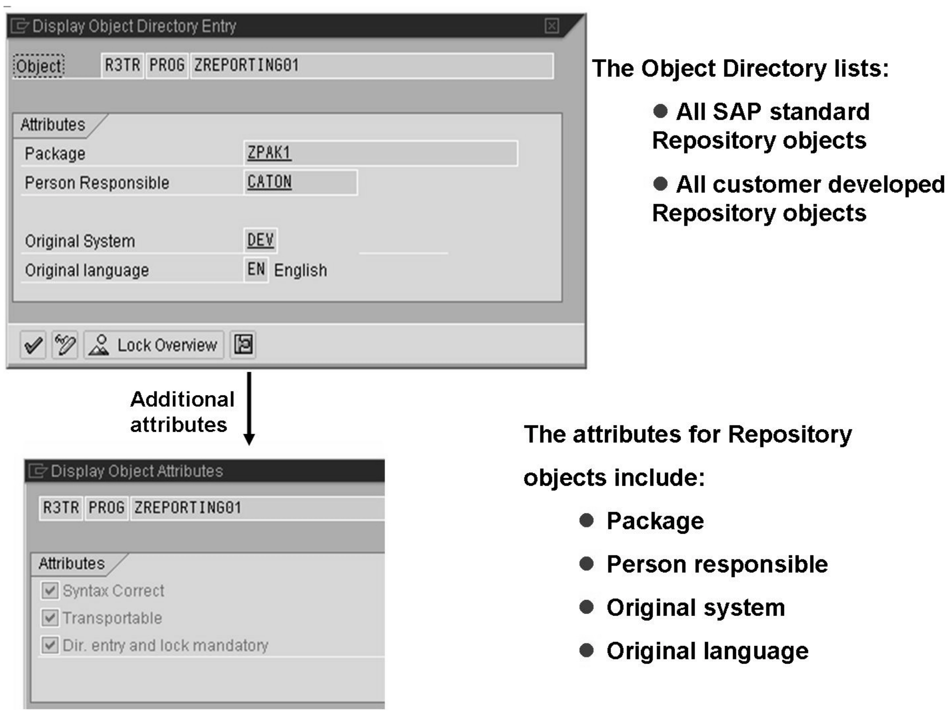
For an overview of all of the current naming conventions for Repository objects, see SAP Note 16466 in the SAP Service Marketplace.

Using customer name ranges correctly prevents object name conflicts between customer objects and SAP objects. If a customer has a more complex system landscape with different system strands and more than one development system, he or she requires additional naming conventions to prevent object name conflicts.

Customer name ranges can be reserved using the view *V_TRESN*, which enables developers to assign a specific name range to a package. For example: you are developing in the SAP system *DE1* for a project within the package *ZPROJECT1*. You have chosen the naming convention *ZPROJ1* for this project and maintained it in all your development systems. A developer in the SAP system *DE2*, working on another project within another package, now tries to create a program with the name *ZPROJ1PROGRAM*. The developer is told that this name is reserved for the package *ZPROJECT1*. The developer is prevented from assigning the object to another package. The developer has to choose a different name to assign the object to his package. This avoids naming conflicts right from the start when objects are created.

Reserving name spaces is the only way to prevent naming conflicts that occur with objects created in complementary software from SAP partners. SAP partners and customer companies can apply to SAP for a name space prefix through SAP Service Marketplace. For more information concerning name spaces and reservation use SAP Note 84282.

SAP Repository objects are listed in the object directory. The object directory is a catalog of all Repository objects in the SAP system, including all standard Repository objects that are delivered with the SAP system, and all Repository objects created by the customer using the ABAP Workbench.



Display Object Directory Entry

Object R3TR PROG ZREPORTING01

Attributes

Package ZPAK1

Person Responsible CATON

Original System DEV

Original language EN English

Display Object Attributes

R3TR PROG ZREPORTING01

Attributes

☑ Syntax Correct

☑ Transportable

☑ Dir. entry and lock mandatory

The Object Directory lists:

- All SAP standard Repository objects
- All customer developed Repository objects

The attributes for Repository objects include:

- Package
- Person responsible
- Original system
- Original language

Figure 70: SAP Repository Object Attributes

The Object Directory lists:

- All SAP standard Repository objects
- All customer-developed Repository objects

  The attributes for Repository objects include:

  - Package
  - Person responsible
  - Original system
  - Original language

Attributes for each Repository object are assigned by the system. The object directory is stored in table *TADIR*. This table is very central to your system's consistency. To change entries in *TADIR*, use only the standard functions SAP provides.

With the appropriate authorization, you can modify the package and person responsible for the object. To modify object directory entries from the Workbench Organizer, choose transaction SE09 and choose *Goto -> Transport Organizer Tools*. Alternatively, use transaction SE03.

Some Repository objects may be generated automatically by the system as a result of customizing activities. In the object directory, these system-created objects are flagged as “generated”.

Modifications to SAP objects or to objects that are not in their original system, such as objects being modified in the quality assurance system after having been created in the development system, are termed **repairs**. In the object directory, these objects are flagged as **repaired**.

For each entry in the object directory, the primary key is comprised of the following fields: program identification (PGMID), object type, and object name. The program identification is usually R3TR. Examples of object types are PROG (ABAP program), DEVC (package), TABL (table definition).

Packages

Repository objects are organized by packages. All Repository objects are assigned to a package, which is formerly known as development class. The package/development class:

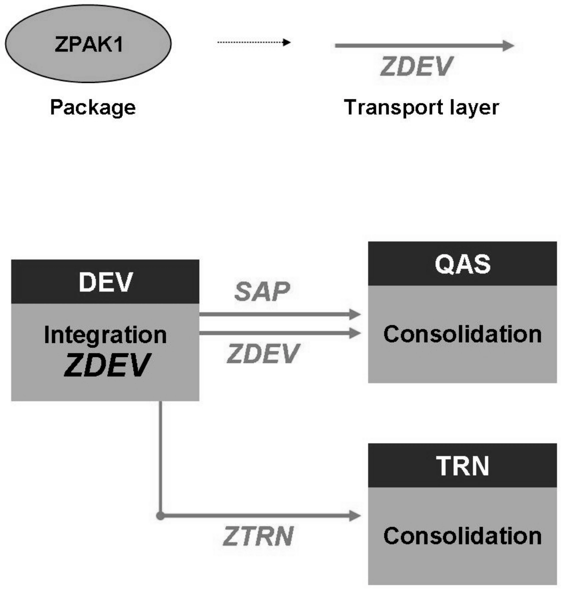
- Provides a logical grouping of objects for coordination of development efforts
- Defines a Repository object’s transport layer
- Can control naming objects

Examples of package definitions are shown in the picture below.

Assigning SAP Repository objects packages to transport layers:

- A package is assigned to a transport layer. If the transport layer is a valid consolidation route, the objects of that package are transported to the consolidation system.
- SAP objects modified in the customer integration system follow the SAP consolidation route.

- **A package is assigned to a transport layer**
- **If the transport layer is a valid consolidation route, the objects of that package are transported to the consolidation system**
- **SAP objects modified in the customer integration system follow the SAP consolidation route**



- **Consolidation routes**

Figure 71: Packages and Transport Layers

When a transport layer is assigned to a package, all objects belonging to that package have the same predefined consolidation route, normally pointing from development to quality assurance system.

In the diagram, both the SAP transport layer and the standard transport layer *ZDEV* follow the same consolidation route from DEV to QAS, that is, from the development system to the quality assurance system. All Repository objects assigned to a package whose transport layer is *ZDEV*are transportable. In the diagram these would be objects, that are assigned to package *ZPAK1*, for example.

In this example, the transport layer *ZTRN* follows a consolidation route from DEV to TRN , from the development system to the training system. All Repository objects assigned to a package whose transport layer is *ZTRN* are transportable.

SAP objects belong to the pre-installed SAP transport layer. In the diagram, SAP objects that are modified in DEV will follow the transport layer *SAP* and be transported to QAS. There is only one *SAP* transport layer in a system landscape.

In this example, you can transport only Repository objects whose package has the transport layer *SAP*, *ZDEV*, or *ZTRN* using the Transport Organizer. If an object is assigned to a transport layer that is either blank or for which there is no transport route, you cannot transport the object using the Transport Organizer.

Packages/development classes are objects of the ABAP Workbench and can be created by the Repository Browser (transaction SE80). When creating a package, the standard transport layer is assigned by default to the package. Users may also choose an alternative transport layer. The transport layer will be used by the TMS tools for determine the consolidation route for the objects inside the package.

Customer created packages may begin with the following letters:

- *Y* or *Z* indicates that the package is for objects that are to be transported.
- *$* indicates a package for temporary objects that are not to be transported and therefore do not require a transport layer.
- *TEST* indicates a package for local objects which are linked to the Transport Organizer, which provides version management and object locking.

The package class *$TMP* is used when a Repository object is saved as a local object and is not recorded to a transport request.

You can use view V_TRESN to specify which packages can be assigned for a name range. For example, program names starting with the string ZABC are prevented from being assigned to a package other than the one defined for this name range in V_TRESN.

The view *V_TDEVC* holds all packages in the SAP system, including all SAP packages.

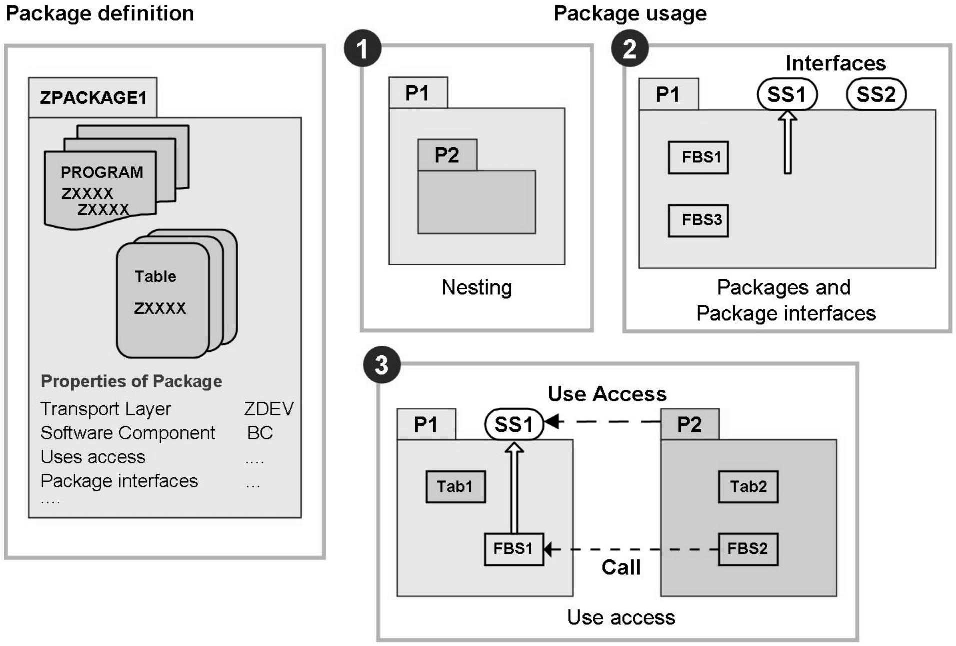


Figure 72: Package usage for Development

Since the SAP Web Application Server 6.10 the concept of packages was extended. Existing development classes are simply containers for development objects with a transport layer that determines how the objects will be transported. Packages extend the concept of development classes with the addition of new attributes: nesting, interfaces, visibility, and use accesses.

1. Nesting of packages, defining the package hierarchy

   Nesting allows you to split the larger units of the SAP system into a hierarchical structure. The combination of interfaces and use accesses allows developers to hide package elements, and protect them from unauthorized use.

2. Definition of package interface

   Packages use interfaces and visibility to make their services known to other packages. All the visible elements in a package can, potentially, be used by other packages; invisible elements cannot. This allows the package to encapsulate its contents and protect its elements from being used by unspecified external packages.

3. Definition of use access

   Use access is the right of one package to use the visible elements in the interface of a second package (but not the other way round).

The package concept also offers the option of splitting and encapsulating the SAP system into technical units (the packages), reducing high levels of dependency and thus decoupling the system on a large and small scale.

Exercise 4: Transport Requests for Development

Exercise Objectives

After completing this exercise, you will be able to:

- Create a workbench transport request and assign it to a project.
- Create a package

Business Example

Before starting any development efforts, the project team leader must create a transport request for the project. Team members are assigned to tasks in the transport request where their efforts are then recorded. Project team members have been assigned a role in the project to create a development object and save it to the transport request.

Repository objects are grouped by packages. To group your own development, create one or more packages to assign your Repository objects to them.

Task 1:

Create five workbench transport requests for development efforts within your project:

1. Log on to the development client as the project leader.
2. Create a workbench transport request and assign it to the project *P_GROUP_##*.
3. How many tasks are associated with your workbench transport request?
4. Repeat these steps.
5. What type of transport request was created? What is the difference between a local transport request and a workbench transport request?

Task 2:

Create a package for your customer developments, assign it to one of your transport requests. Then export this transport request.

1. Create a package *ZPACKAGE_##* using the Object Navigator. ## represents your group number.
2. In order to transport your newly created package to the subsequent systems of your system landscape, export the transport request containing your package.

Solution 4: Transport Requests for Development

Task 1:

Create five workbench transport requests for development efforts within your project:

1. Log on to the development client as the project leader.

   a) Log on to the DEV system, client 100 with your user.

   Only the project team leader *ADM325-##* has the authority to create transport requests.

2. Create a workbench transport request and assign it to the project *P_GROUP_##*.

   a) To create a transport request and assign it to your project, choose transaction SE09.

   From the *Transport Organizer* initial screen, choose *Request/Task → Create*. The *Create request* dialog box appears. Select *Workbench request* and choose *Copy (Enter)*.

   Provide the request with short text. To assign the transport request to your group's project, select your project *P_GROUP_##* in the *Project* field . ADM325-## is added to the list of users assigned to the transport request by default. You can add other project members to the transport request. Choose *Save*.

   Note: Alternatively you could also create the workbench transport request from within your project. To do this, choose transaction SPRO_ADMIN, select your project, and add a workbench request. Your project is assigned automatically, when creating the transport request from within the project.

3. How many tasks are associated with your workbench transport request?

   a) You should have as many tasks assigned to the transport request, as there are users assigned.

   Note: If you have saved the transport request and you want to add additional users, from the *Display request* screen select the transport request and choose *Request/task → Request → Add user*. The *Add user* dialog box displays. Enter additional team members and *Save*.

Continued on next page

4. Repeat these steps.

   a) Repeat these steps four times in order to create five workbench transport requests.

5. What type of transport request was created? What is the difference between a local transport request and a workbench transport request?

   a) The newly created request is a workbench transport request.

      Workbench transport requests can be distributed/transported to target systems. Local transport requests are local to the SAP system that it was created in and can not be distributed/transported to target systems.

Task 2:

Create a package for your customer developments, assign it to one of your transport requests. Then export this transport request.

1. Create a package *ZPACKAGE_##* using the Object Navigator. ## represents your group number.

   a) Call transaction SE80. In the initial screen of the Object Navigator, choose *Repository Browser* and choose the Option *Package* in the drop-down list.

   b) Assign the name **`ZPACKAGE_##`** to your package.

      Because your package does not exist, the system asks you if you want to create it after you choose *Enter*. Confirm this with *Yes*.

   c) Assign the necessary attributes to the package: enter a meaningful short description, software component **`HOME`**, and the standard transport layer **`ZDEV`**. Choose package type **`Not a Main Package`**. Choose *Save* to create your package.

   d) The system prompts you to assign your newly created Repository object to a workbench request. Select *Own Requests*, choose one of the transport requests that you created for your project, and confirm the selection.

   e) Check the assignment to your request in the Transport Organizer (transaction SE09). Display all transport requests that you created but are not yet released by choosing *Display*.

      There is a task described as *Development/Correction* under the transport request, you assigned the newly created package.

Continued on next page

2. In order to transport your newly created package to the subsequent systems of your system landscape, export the transport request containing your package.

   a) In the Transport Organizer (SE09): Select the task of the transport request containing your package. Release the task. Then select the transport request itself and release it. The transport request with your package is now being exported. Check, that the export terminated without errors.

Lesson Summary

You should now be able to:

- Plan change management for development
- Identify various implementation tools used for Customizing and development
- Describe the function of the Transport Organizer

Lesson: Customer Development

Lesson Overview

In this lesson, you will learn the concepts of customer development.

Lesson Objectives

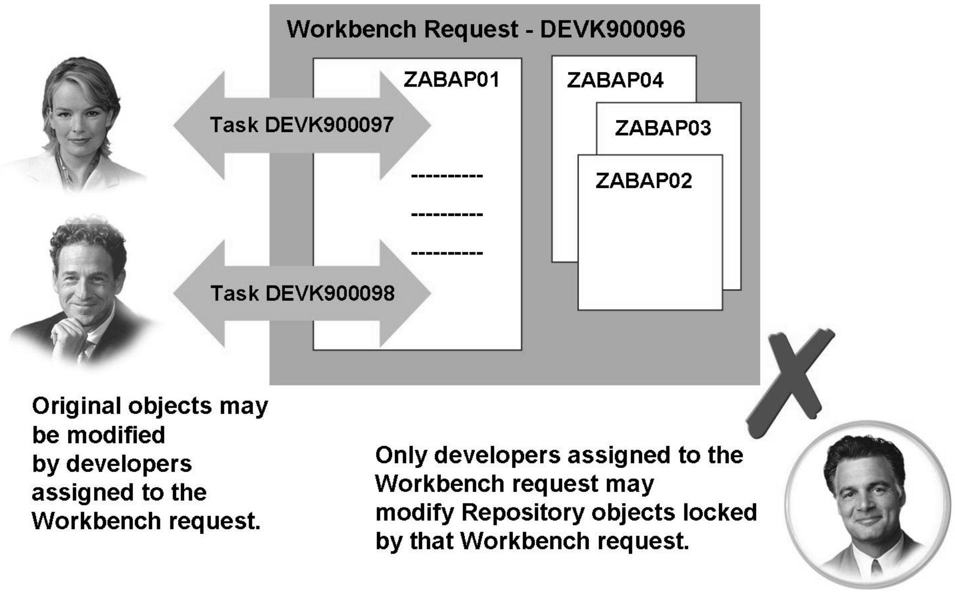
After completing this lesson, you will be able to:

- create a simple program
- understand the principles of the version management
- check and adjust the Transport Organizer settings

Business Example

As the transport administrator, you must work with the development teams to ensure that the appropriate system settings are configured to meet the requirements for the development projects.

Object Locking


Workbench Request - DEVK900096
ZABAP01
ZABAP04
ZABAP03
ZABAP02
Task DEVK900097

Task DEVK900098
Original objects may be modified by developers assigned to the Workbench request.
Only developers assigned to the Workbench request may modify Repository objects locked by that Workbench request.

Figure 73: SAP Repository Object Locking

There are two locking mechanisms when Repository objects are being modified:

- The editor program, which works with the enqueue work process to ensure that only one user at a time can modify an object in the system
- The workbench transport request, which ensures that the developer modifying the object is assigned to a valid task within the Workbench transport request.

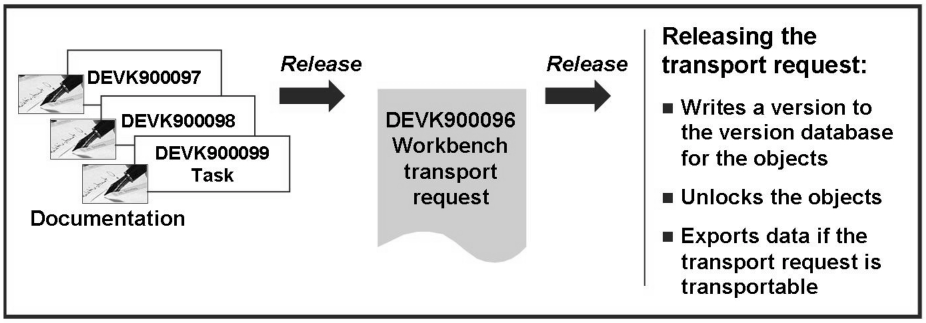
When a Repository object is assigned to a task within a Workbench transport request, that object can only be modified by the developers associated with that transport request. This prevents users outside the development team from making changes to any of the objects in the transport request before the development project is completed and the associated transport requests are released.

An object list is associated with each task. Each user working on an object has a corresponding entry in the object list of their task. The object list records which users have actually edited the object.

Objects may be manually entered in the object list of a task or request. Go to the display view of the Transport Organizer: transaction SE09 -> *Display*. You can manually add objects, for example, by double-clicking the transport request and adding the desired objects. Or you can use the function *Request/Task -> Object List -> Include Objects*.

These objects are not automatically locked if they are entered manually. To manually lock the object, in the Transport Organizer initial screen (transaction SE09), choose *Display → Request/Task → Object List → Lock Objects*.

Releasing Transport Requests



DEVK900097

DEVK900098

DEVK900099
Task

Documentation

Release

DEVK900096
Workbench
transport
request

Release

Releasing the transport request:

- Writes a version to the version database for the objects
- Unlocks the objects
- Exports data if the transport request is transportable

Releasing a task requires:

- Content in the task

Releasing a transport request requires:

- Release of all tasks in the transport request
- Documentation of all tasks (if switched on)

Figure 74: Releasing Transport Requests

After the development tasks are completed, a transport request is released so that the content can be transported to another SAP system.

Releasing a task or transport request requires the to have the necessary authorization.

Releasing a transport request requires that all tasks are documented and released.

A released task can no longer be modified, but additional tasks can be created to modify the objects in the released task, as long as the transport request is not released. A released task cannot be deleted. However, if the task is empty, it is deleted when the transport request is released.

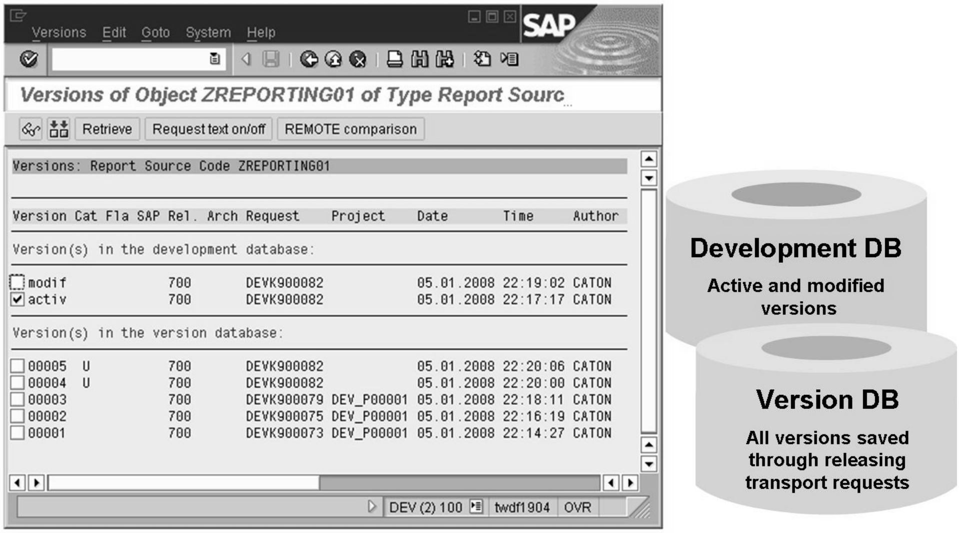
Releasing a transportable transport request records a version of all the objects included in the transport request (that are included in the version management), and then exports the objects, that is, copies the objects from the database to an operating system file in the transport directory.

Releasing a local transport request records a version of the objects in the transport request (that are included in the version management), but does not export them to a file at the operating system level.

Version Management

Versions of SAP Repository objects can be:

- **Displayed**
- **Compared**
- **Restored**


Versions
Edit
Goto
System
Help
SAP
Versions of Object ZREPORTING01 of Type Report Sourc
Retrieve
Request text on/off
REMOTE comparison
DEV (2) 100
twdf1904
OVR
Development DB
Active and modified versions
Version DB
All versions saved through releasing transport requests

```
Versions: Report Source Code ZREPORTING01

Version Cat Fla SAP Rel. Arch Request    Project    Date       Time     Author

Version(s) in the development database:

□ modif           700       DEVK900082            05.01.2008 22:19:02 CATON
☑ activ           700       DEVK900082            05.01.2008 22:17:17 CATON

Version(s) in the version database:

□ 00005  U        700       DEVK900082            05.01.2008 22:20:06 CATON
□ 00004  U        700       DEVK900082            05.01.2008 22:20:00 CATON
□ 00003           700       DEVK900079 DEV_P00001 05.01.2008 22:18:11 CATON
□ 00002           700       DEVK900075 DEV_P00001 05.01.2008 22:16:19 CATON
□ 00001           700       DEVK900073 DEV_P00001 05.01.2008 22:14:27 CATON
```

Figure 75: Version Management

Versions of SAP Repository objects can be compared or restored.

When a Workbench transport request is released, a new version of each Repository object in the transport request is written to the version database, which thus contains a complete change history of all Repository objects.

In addition to the versions created automatically by the release of transport requests, users can also create temporary versions at any time. To do this, in the maintenance transaction for the Repository object, for example transaction SE38, choose *Utilities(M) -> Versions -> Version Management -> Generate version*.

You can access version management *Utilities(M) -> Versions -> Version Management* from:

- Object Navigator / Repository Browser (transaction SE80)
- Transport Organizer (transaction SE09)
- Display and maintenance transactions for all Repository objects, for example SE38, SE11, SE37

In the *Version Overview*, the active and modified versions are displayed in the development database, and versions saved as a result of released transport requests are displayed in the version database.

The version database resides in the development system. Versions cannot be transported between SAP systems. If the development system is removed from the system landscape, all versions in the version database are lost.

Transport Organizer Settings

Components that can be analyzed are (e.g.):

- **Objects**
- **Objects in transport requests**
- **Object Directory**
- **Transport request / tasks**
- **Administration**

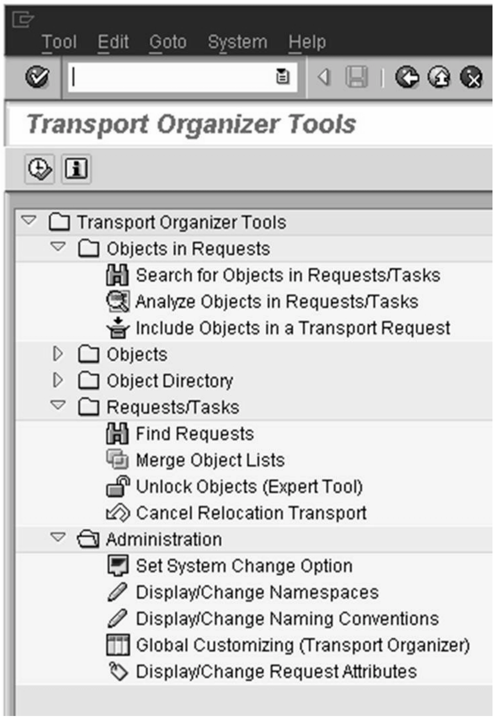

Tool
Edit
Goto
System
Help
Transport Organizer Tools
Transport Organizer Tools
Objects in Requests
Search for Objects in Requests/Tasks
Analyze Objects in Requests/Tasks
Include Objects in a Transport Request
Objects
Object Directory
Requests/Tasks
Find Requests
Merge Object Lists
Unlock Objects (Expert Tool)
Cancel Relocation Transport
Administration
Set System Change Option
Display/Change Namespaces
Display/Change Naming Conventions
Global Customizing (Transport Organizer)
Display/Change Request Attributes

Figure 76: Transport Organizer Tools (SE03)

The Transport Organizer tools are a collection of tools that support the work with the change and transport system. These tools can be reached either by the corresponding button in the Transport Organizer (transaction SE09) or by starting transaction SE03 directly.

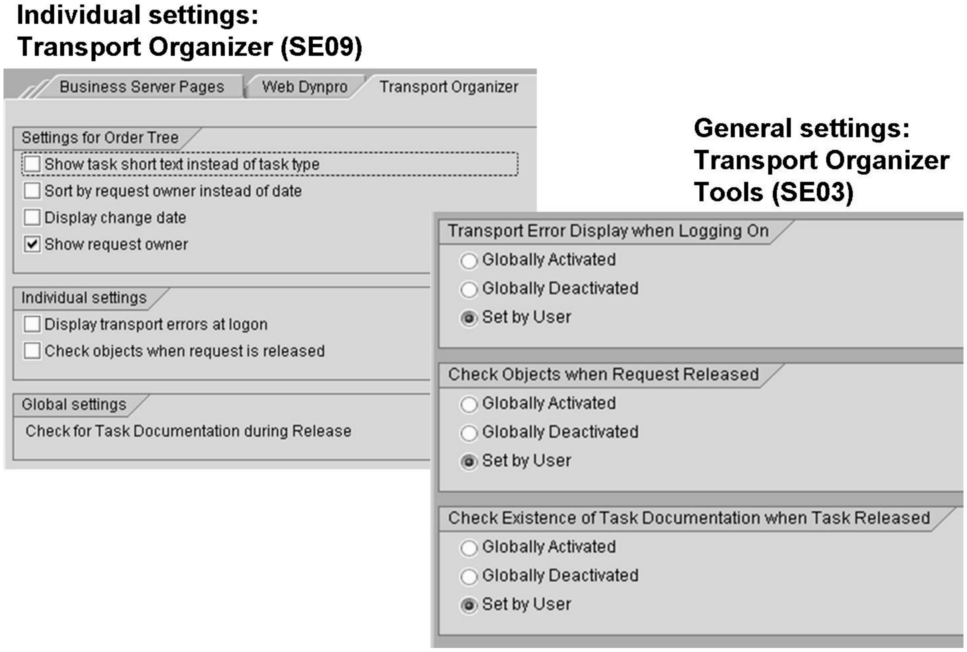
You can find documentation of the functions by using the *Documentation* button, shown in the figure above. A documentation of the tools can also be reached, when right-clicking them.

Transport Organizer settings provide customers with several options that can be used for:

1. Displaying transport errors at logon
2. Applying objects checks to objects contained in transport requests before the transport request is exported to a file in the transport directory

To cause transport errors to be displayed to transport request owners when they log on, select display transport errors at logon.

Object checks identify and display errors found in Workbench transport requests. Errors, such as program syntax errors, are identified and displayed when a transport request is released, but not yet exported. More detailed information about the error can be found by simply double-clicking the error on the displayed report.



Individual settings: Transport Organizer (SE09)

Settings for Order Tree

- ☐ Show task short text instead of task type
- ☐ Sort by request owner instead of date
- ☐ Display change date
- ☑ Show request owner

Individual settings

- ☐ Display transport errors at logon
- ☐ Check objects when request is released

Global settings

Check for Task Documentation during Release

General settings: Transport Organizer Tools (SE03)

Transport Error Display when Logging On

- ○ Globally Activated
- ○ Globally Deactivated
- ● Set by User

Check Objects when Request Released

- ○ Globally Activated
- ○ Globally Deactivated
- ● Set by User

Check Existence of Task Documentation when Task Released

- ○ Globally Activated
- ○ Globally Deactivated
- ● Set by User

Figure 77: Transport Organizer Settings

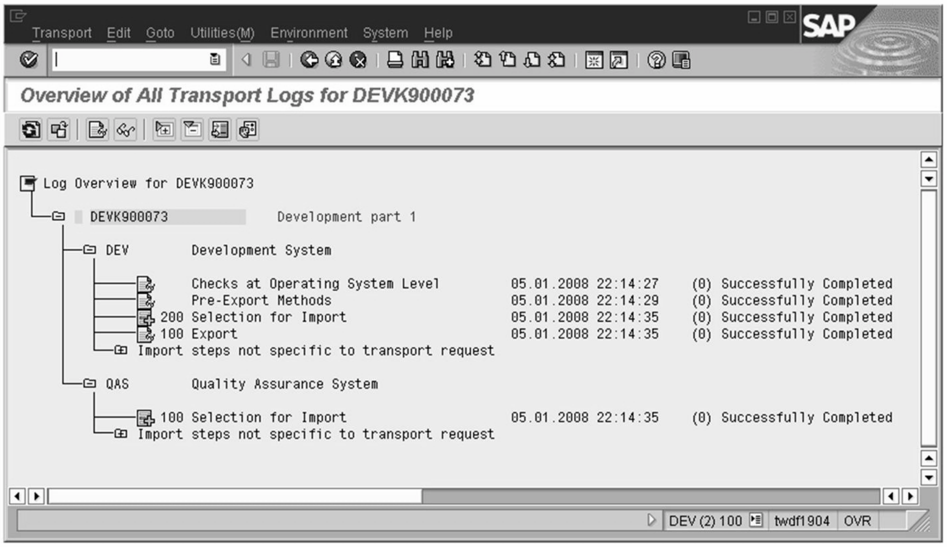
To activate or deactivate the display of transport errors at logon and object checks:

Globally, choose transaction SE03 *Administration → Global Customizing Transport Organizer.* Select *Globally activated*.

For a specific user, choose transaction SE03 *Administration → Global Customizing (Transport Organizer)*. Select the option *Set by User*.

For each logged-on user, choose transaction SE09 *Display → Utilities(M) → Settings*. Change to the *Transport Organizer* tab. The options *Display transport errors at logon* and *Check objects when request is released* can be set by the user that is logged on.

You can use the Transport Organizer tools for many different tasks concerning objects in tasks and requests. These tools also allow access to several administrative transactions, which can also be accessed in other ways.



Overview of All Transport Logs for DEVK900073

```
Log Overview for DEVK900073
  DEVK900073                     Development part 1
    DEV        Development System
                Checks at Operating System Level       05.01.2008 22:14:27   (0) Successfully Completed
                Pre-Export Methods                     05.01.2008 22:14:29   (0) Successfully Completed
            200 Selection for Import                   05.01.2008 22:14:35   (0) Successfully Completed
            100 Export                                 05.01.2008 22:14:35   (0) Successfully Completed
          Import steps not specific to transport request
    QAS        Quality Assurance System
            100 Selection for Import                   05.01.2008 22:14:35   (0) Successfully Completed
          Import steps not specific to transport request
```

Return code 0 means "Everything OK"

Figure 78: Transport Logs

After releasing a transportable transport request, review the transport logs to ensure that the export was successful.

To access the transport logs from the initial screen of the Transport Organizer (transaction code SE09), choose *Display* → *Goto* → *Transport Logs*.

In the transport logs, the following return codes indicate the success or failure of the export:

- 0: The export was successful.
- 4: A warning was issued but all objects were transported successfully.
- 8: A warning was issued and at least one object could not be transported successfully.
- 12 or higher: A critical error had occurred, generally not caused by the objects in the request.

Authorizations for Change Management

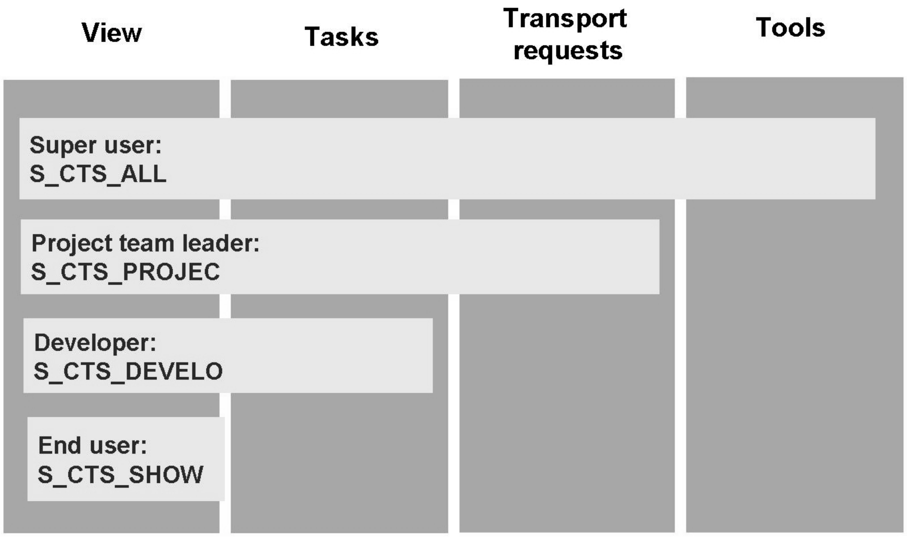


Figure 79: Authorizations for Change Management

As you have seen in this unit, different tools and concepts are existing for managing Customizing and workbench transport requests in a controlled way. How these tools are used and the responsibility for changing Customizing and repository in a system are normally defined in the customizing and development roles for that system. But in real life most of these roles are only observed if they are disposed in user authorizations and authorization roles.

In the most cases, the authorization split up can be done as shown in the picture above. These are the following tasks:

- Super user (e.g. system administrator)

  Has all authorizations related to transport requests and tasks.
- Project leader

  Can create and manage transport requests and tasks.
- Developer

  Can only use existing transport requests.
- End user

  Only has display authorization.

Exercise 5: Customer Development

Exercise Objectives

After completing this exercise, you will be able to:

- name the steps for creating repository objects
- work with the version management

Business Example

During the implementation process, you may be required to create Repository objects.

Task 1:

Create a new ABAP program.

1. Create the ABAP program *ZPGM_##* where ## is your group number.
2. If the team member has not yet been registered in the system as a developer, the system prompts you for your developer's access key.
3. In the *ABAP: Program Attributes* popup, enter a title, choose the type *Executable program*, and the status *Test program*.

   Choose *Save*.
4. Since this is a Repository object, the system displays the *Create Object Directory Entry* screen. Assign your new ABAP program to the package you created for your project and choose *Save*.
5. The *Prompt for transportable workbench request* popup appears. Select one of the transport requests you created for your project.
6. From the *ABAP Editor* screen, on a new line below the report name, enter the following command: **`Write 'Hi!  I am an ADM325 test program!'.`**. Make sure you surround the text with single quotes and end the command with a period.

   Verify that there are no syntax errors in your program. Save, activate, and execute your program.

Task 2:

Release your workbench task and transport request.

1. As the developer, release the task containing your ABAP program.

Continued on next page

How can you tell if the task has been released?

2. As the project leader, release and export the transport request.

Task 3:

Change the program, you created. Then export the corresponding transport request.

1. Change the ABAP program *ZPGM_##* by adding a new line to the program. Add the following command: `Write 'The end.'.`
2. As the developer, release the task. As the project leader, release the transport request.

Task 4:

Create a local object and later reassign it to a package for transporting.

1. Create an ABAP program called *ZCREATEDLOCAL_##*. Enter a short title for your program. Enter *Execute program* for type, and *Test program* for status.
2. Save this new program as a local object.
3. On a new line of the ABAP Editor, enter the command:

   ```
   Write 'I was born as a local object...'.
   ```

   Save your change. Check this program for syntax errors, then activate it.
4. Is there a transport request number associated with this program? Why or why not?
5. Is there a package associated with this program? Why or why not?
6. Is this object transportable? Why? Which change is necessary to make this local object a transportable object? Reassign your program to the package, you created. What happens?
7. Assign the former local program to one of the transport requests of your project. Release the transport request.

Task 5:

Restoring a Repository object from a previous version.

1. How many versions of your ABAP program, *ZPGM_##* exist? Compare the first version with the last version of your program. What does the comparison indicate?

Continued on next page

2. Retrieve the first saved version, restoring your program back to its former status. Why does the system ask for a transport request?
3. As the developer, release the task. As the project leader, release the transport request.

Task 6:

Analyze the attributes of Repository objects using the Transport Organizer Tools.

1. Search for change requests and tasks using the relevant tools in the Transport Organizer. Locate all transport request and tasks containing your *ZPGM_##* program.
2. Display all of the attributes associated with the objects in the transport request.

Solution 5: Customer Development

Task 1:

Create a new ABAP program.

1. Create the ABAP program *ZPGM_##* where ## is your group number.

   a) Call transaction SE38. Enter the name of the program that you wish to create, **`ZPGM_##`**, and choose *Create*.

2. If the team member has not yet been registered in the system as a developer, the system prompts you for your developer's access key.

   a) For the purposes of this class, you will not be required to provide an access key.

3. In the *ABAP: Program Attributes* popup, enter a title, choose the type *Executable program*, and the status *Test program*.

   Choose *Save*.

   a) Enter the title, type, and status.

4. Since this is a Repository object, the system displays the *Create Object Directory Entry* screen. Assign your new ABAP program to the package you created for your project and choose *Save*.

   a) When the *Create Object Directory Entry* popup displays, enter the package assigned to your project and save.

   Hint: If a package with an invalid transport layer is assigned to an object, the object cannot be transported to the target systems.

5. The *Prompt for transportable workbench request* popup appears. Select one of the transport requests you created for your project.

   a) To display your transport requests, use the possible entries arrow beside the *Request* field. Choose one of the transport requests created for your project. The transport request must be of type *Workbench request*. Choose *Enter*.

6. From the *ABAP Editor* screen, on a new line below the report name, enter the following command: **`Write 'Hi!  I am an ADM325 test program!'.`**. Make sure you surround the text with single quotes and end the command with a period.

Continued on next page

Verify that there are no syntax errors in your program. Save, activate, and execute your program.

a) When entering the command, make sure that you end the command with a period, otherwise a syntax error will result. Enter **`Write 'Hi!  I am an ADM325 test program!'.`**.

   To check your program for errors choose *Program → Check → Syntax.* If there are no syntax errors in the program, activate the program. Choose *Program → Activate*.

b) Choose the *Back arrow* icon to return to the *ABAP Editor* initial screen and press the processing button to run your program.

Task 2:

Release your workbench task and transport request.

1. As the developer, release the task containing your ABAP program.

   How can you tell if the task has been released?

   a) Display the transport requests and tasks assigned to your user, *ADM325-##*, from the initial screen of the Transport Organizer. Transaction SE09, *Display*.

      To release a task, select the task number and choose *Request/Task → Release → Directly*. When a task is released, a check mark appears to the right of the task.

2. As the project leader, release and export the transport request.

   a) Call transaction SE09. Select your transport request and choose *Request/Task → Release → Directly*.

      If there are tasks in the transport request that have not been released, you cannot release the transport request unless the tasks are empty. The system deletes empty tasks when a transport request is released.

Continued on next page

Task 3:

Change the program, you created. Then export the corresponding transport request.

1. Change the ABAP program *ZPGM_##* by adding a new line to the program. Add the following command: **`Write 'The end.'.`**

   a) To edit a program, choose transaction SE38. Enter the name of the program *ZPGM_##* and select **Change**. Add the following command: **`Write 'The end.'`**.

      Save and activate the program. When prompted, create a new transport request for the changes you just made.

2. As the developer, release the task. As the project leader, release the transport request.

   a) With transaction SE09: select the task of the transport request to be released and choose *Request/Task → Release → Directly*. Then select the transport request and choose *Request/Task → Release → Directly*.

Task 4:

Create a local object and later reassign it to a package for transporting.

1. Create an ABAP program called *ZCREATEDLOCAL_##*. Enter a short title for your program. Enter *Execute program* for type, and *Test program* for status.

   a) To call the ABAP Editor, choose transaction SE38.

   b) Enter the name of the program you wish to create, **`ZCREATEDLOCAL_##`** and choose *Create*. Then enter a description, the type **`Executable program`** and status **`Test program`** and choose *Save*.

2. Save this new program as a local object.

   a) When the *Package* popup displays, select *Local object*.

3. On a new line of the ABAP Editor, enter the command:

   `Write 'I was born as a local object...'.`

   Save your change. Check this program for syntax errors, then activate it.

   a) Make sure that you end the statement **`Write 'I was born as a local object...'.`** with a period, otherwise a syntax error results.

      To verify that there are no syntax errors, check your program by choosing *Program → Check → Syntax.* Save and activate your program.

Continued on next page

4. Is there a transport request number associated with this program? Why or why not?

   a) Go to the Transport Organizer screen by choosing transaction SE09.

      There is no transport request associated with this program. Local objects are not linked to the Transport Organizer.

5. Is there a package associated with this program? Why or why not?

   a) Yes. The package is *$TMP*.

      All repository objects, including local objects, are listed in the Object Directory (table *TADIR*). To search the Object Directory, choose transaction SE38. Enter the name of the program, select *Attributes* and choose *Display*. The package for this object is *$TMP*.

      Hint: You can also find out the attributes in the object catalog by choosing the Workbench Organizer Tools, transaction SE03

6. Is this object transportable? Why? Which change is necessary to make this local object a transportable object? Reassign your program to the package, you created. What happens?

   a) No, the object is not transportable. The package is *$TMP*, which does not have a valid transport layer. Also, local objects are not linked to the Transport Organizer.

      The object must be reassigned to a package that contains a valid transport layer. Call the ABAP editor (transaction SE38). In the *Program name* field, enter your program, **`ZCREATEDLOCAL_##`**. Choose *Goto → Object Directory* entry. In the *Display Object Directory Entry* screen, choose *Change* and enter the package that you created, **`ZPACKAGE_##`** (the package you created). Choose *Save*.

      Now the *Change request* popup displays because the object is now assigned to a package with a transport layer, it is linked to the Transport Organizer.

Continued on next page

7. Assign the former local program to one of the transport requests of your project. Release the transport request.

   a) In the *Change request* dialog box, select one of your transport requests. the transport request must be of the type *Workbench*.

      To verify that the transport request is transportable, go to the Transport Organizer (transaction SE09) and view the transport request.

   b) In the Transport Organizer, select the task, and choose *Release directly*. Afterwards, select the *Change request* and choose *Release directly*.

Task 5:

Restoring a Repository object from a previous version.

1. How many versions of your ABAP program, *ZPGM_##* exist? Compare the first version with the last version of your program. What does the comparison indicate?

   a) To go to the ABAP editor, choose transaction SE38. Enter your program name **`ZPGM_##`**. To view all saved versions of the program, choose *Utilities → Versions → Version management.* To compare two versions, select the two versions that you wish to compare, then choose *Versions → Compare*.

      There should be two versions of your program. One was created when your first transport request was released, and the second one was created when your second transport request was released.

      Comparing these two versions shows that a new command line was added to the program.

2. Retrieve the first saved version, restoring your program back to its former status. Why does the system ask for a transport request?

   a) In version management, to bring back the first version of your program, mark this first version in the list and choose *Retrieve*.

      This is a change on the program (just as a manual change), and in order to be recorded, it must be assigned to a transport request.

      Save and activate the program.

Continued on next page

3. As the developer, release the task. As the project leader, release the transport request.

   a) With transaction SE09: select the task of the transport request to be released and choose *Request/Task → Release → Directly*. Then select the transport request and choose *Request/Task → Release → Directly*.

Task 6:

Analyze the attributes of Repository objects using the Transport Organizer Tools.

1. Search for change requests and tasks using the relevant tools in the Transport Organizer. Locate all transport request and tasks containing your *ZPGM_##* program.

   a) From the initial screen of the Transport Organizer (transaction SE09), choose *Goto→ Transport Organizer Tools*. A list of Transport Organizer tools displays. Select *Search for objects in requests/tasks* and choose *Execute*. The *Objects search* screen displays.

   On the left side of the screen, mark the check box beside *R3TR PROG*, and enter the name of your program. To display all released transport requests containing this ABAP program, select *Released* in selection of *Requests/Tasks* at the bottom of the screen and choose *Execute*.

   Note: Check *Also Search for Subobjects* to also find transport requests containing for example *LIMU REPS* entries of the program.

   Hint: You can also use transaction SE03 to call the Transport Organizer Tools directly.

2. Display all of the attributes associated with the objects in the transport request.

   a) From the Transport Organizer Tools initial screen (transaction SE03), choose *Analyze Objects in change Requests/Tasks*. Enter the transport request number and choose *Execute*. A list of all objects contained in the transport request displays. To display the attributes associated with an object, select one of the objects and select the *Display object directory* button.

   Note: The attributes are stored in the object directory (table *TADIR*).

Lesson Summary

You should now be able to:

- create a simple program
- understand the principles of the version management
- check and adjust the Transport Organizer settings

Lesson: Modifying SAP Objects

Lesson Overview

In this lesson, you will learn about modifying Repository objects and the factors that have to be taken care of while making changes to these objects.

Lesson Objectives

After completing this lesson, you will be able to:

- Modify SAP Repository objects
- Describe the difference between customer development and modifications
- Describe how the Modification Assistant assists developers

Business Example

The purpose of your development work is to introduce new or improved functions into the production system. This may require the creation of new objects and the modification of existing objects in the SAP system. Developers can use the Modification Assistant in the ABAP editor to make changes to Repository objects, such as ABAP programs. This reduces the amount of effort required when applying Support Packages and when performing an upgrade of an SAP system.

Creating and Modifying Objects

Repository objects should only be changed in the SAP system where the system identifies them as original objects. An object is the original in only one system, the system where it was created. All other systems may only contain copies of the object. This ensures that changes are only made to Repository objects in the development system.

When a Repository object is transported to a downstream system, it exists there as a copy. Changes to copies, which are referred to as **repairs**, are made only in exceptional cases. In all customer systems, including the customer's development system, SAP-delivered Repository objects are copies. The original of those objects exists at SAP in the Development System. Repairs to SAP objects are referred to as **modifications**.

The system change options selected when setting up the Transport Organizer (transaction SE06) control which Repository objects can be modified in the SAP system. Only in the development system, the system change option should be set to permit changes to customer-developed objects and, if necessary, SAP objects.

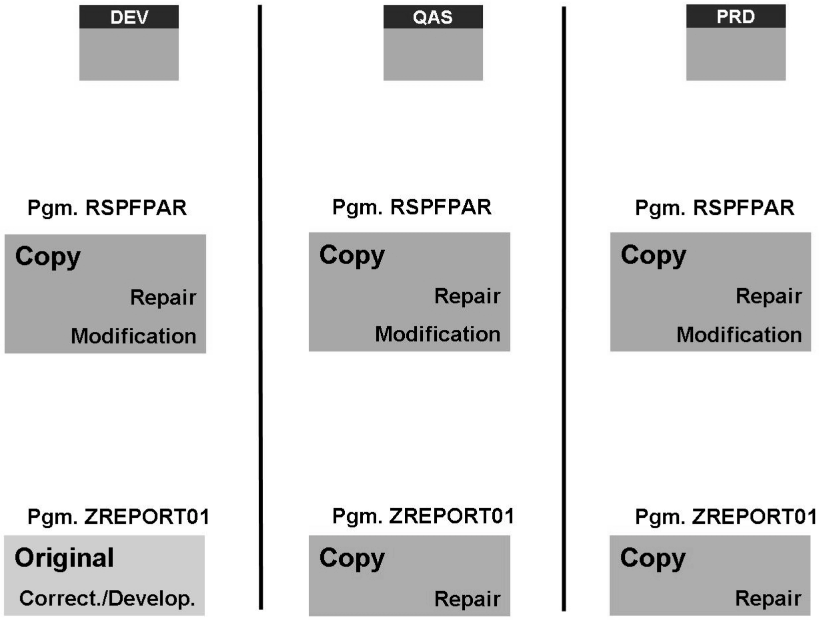


Figure 80: Repository Objects: Originals and Copies

When changing an object that is not an original, the system prompts the user to specify the object for a repair. For the object specified, a repair flag is set to prevent the object from being overwritten by imports into the system. Thus, the object is locked exclusively for the task of type repair.

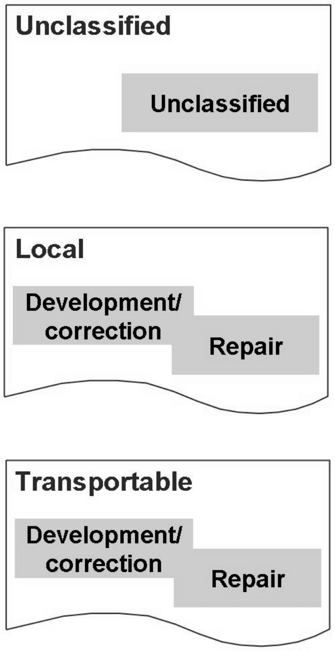
When a task of type repair is released, the repair flag is deleted automatically.

Classification of Tasks and Transport Requests

A distinction is made between two types of development tasks: development/correction, that is changes in the original system, and repair, that is changes in a system other than the original system.

Objects saved as local objects cannot be transported. The only way to transport such an object is to assign it to another package, one that has an existing transport layer assigned.

The package of the Repository object determines whether the transport request type is transportable or local. If the assigned package has a valid transport layer, a request of type transportable is used. Otherwise, a local transport request is used.



- **Transport requests or tasks of type unclassified have empty object lists.**
- **Changes to original objects are saved to tasks of type Development / Correction.**
- **Changes to copies are saved to tasks of type Repair.**
- **The package of a Repository object determines if a change request of type Transportable or Local is required.**

Figure 81: Classifying Tasks and Transport Requests

There are three types of transport requests or development tasks: unclassified, local, and transportable.

- Transport requests or tasks of type *unclassified* have empty object lists.
- Changes to original objects are saved to tasks of type *development/correction*.
- Changes to copies are saved to tasks of type *repair*.

The package of a Repository object determines if a transport request of type *transportable* or *local* is required.

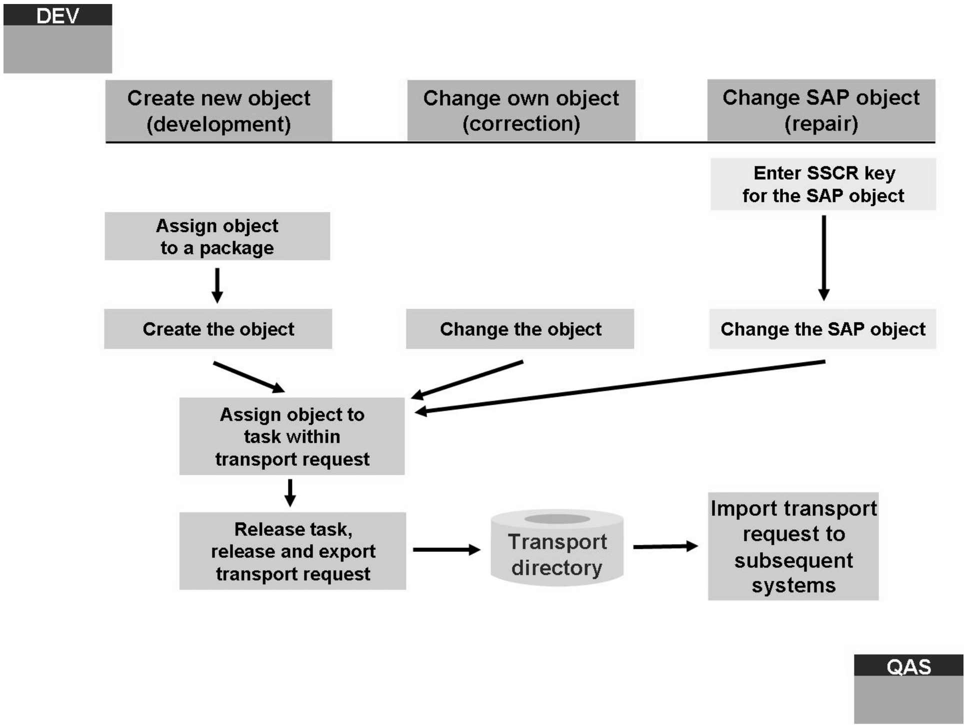

DEV
Create new object (development)
Change own object (correction)
Change SAP object (repair)
Enter SSCR key for the SAP object
Assign object to a package
Create the object
Change the object
Change the SAP object
Assign object to task within transport request
Release task, release and export transport request
Transport directory
Import transport request to subsequent systems
QAS

Figure 82: Changing Customer and SAP Objects

Before starting a modification, you should think about if it is really necessary to modify the standard programs or if the enhancement can also be done by means of SAP extension techniques (e.g. BADI, appends, user-exits). Please keep in mind that the modification may have to be adjusted if SAP delivers this objects in a newer version; for example, in an Upgrade or a Support Package. That can cause additional effort for maintenances of the modification.

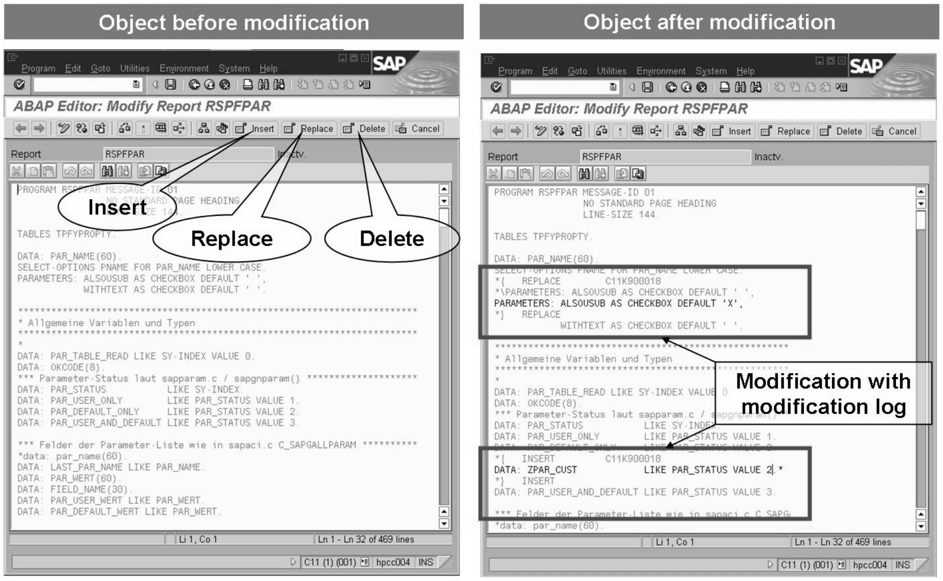
Before changing SAP-delivered Repository objects (performing a modification), you have to be registered as a developer in SAP Software Change Registration (SSCR), and you have to register each SAP object you intend to change as well. Registering the object gives you an access key which you apply only once to the object.

When performing a modification, the Transport Organizer prompts you for a transport request in the same way that it does when you make changes to customer-owned objects. Since the object is owned by SAP and not the system itself, the change will be saved to a task of type repair.

All SAP objects have SAP-defined packages. All new customer Repository objects must be assigned to a customer-created package. Packages are used to group objects in the project and transport them along the same transport route.

After the repair task or transportable task is complete, the developer releases the task, and the lock on the object is transferred to the transport request.

At the end of the project, the transport request is released. This releases all locks and records a version of the changed objects, both SAP and customer objects, in the version database.


Object before modification
Program Edit Goto Utilities Environment System Help
ABAP Editor: Modify Report RSPFPAR
Insert Replace Delete Cancel
Report RSPFPAR Inactv.
Insert
Replace
Delete
Li 1, Co 1
Ln 1 - Ln 32 of 469 lines
C11 (1) (001) hpcc004 INS
Object after modification
Program Edit Goto Utilities Environment System Help
ABAP Editor: Modify Report RSPFPAR
Insert Replace Delete Cancel
Report RSPFPAR Inactv.
Modification with modification log
Li 1, Co 1
Ln 1 - Ln 32 of 469 lines
C11 (1) (001) hpcc004 INS

```
PROGRAM RSPFPAR MESSAGE-ID 01
                NO STANDARD PAGE HEADING
                LINE-SIZE 144.

TABLES TPFYPROPTY.

DATA: PAR_NAME(60).
SELECT-OPTIONS PNAME FOR PAR_NAME LOWER CASE.
PARAMETERS: ALSOUSUB AS CHECKBOX DEFAULT ' ',
            WITHTEXT AS CHECKBOX DEFAULT ' '.

************************************************************************
* Allgemeine Variablen und Typen
************************************************************************
*
DATA: PAR_TABLE_READ LIKE SY-INDEX VALUE 0.
DATA: OKCODE(8).
*** Parameter-Status laut sapparam.c / sapgnparam() ********************
DATA: PAR_STATUS            LIKE SY-INDEX.
DATA: PAR_USER_ONLY         LIKE PAR_STATUS VALUE 1.
DATA: PAR_DEFAULT_ONLY      LIKE PAR_STATUS VALUE 2.
DATA: PAR_USER_AND_DEFAULT LIKE PAR_STATUS VALUE 3.

*** Felder der Parameter-Liste wie in sapaci.c C_SAPGALLPARAM **********
*data: par_name(60).
DATA: LAST_PAR_NAME LIKE PAR_NAME.
DATA: PAR_WERT(60).
DATA: FIELD_NAME(30).
DATA: PAR_USER_WERT LIKE PAR_WERT.
DATA: PAR_DEFAULT_WERT LIKE PAR_WERT.
```

```
PROGRAM RSPFPAR MESSAGE-ID 01
                NO STANDARD PAGE HEADING
                LINE-SIZE 144.

TABLES TPFYPROPTY.

DATA: PAR_NAME(60).
SELECT-OPTIONS PNAME FOR PAR_NAME LOWER CASE.
*{   REPLACE        C11K900018
*\PARAMETERS: ALSOUSUB AS CHECKBOX DEFAULT ' ',
PARAMETERS: ALSOUSUB AS CHECKBOX DEFAULT 'X',
*}   REPLACE
            WITHTEXT AS CHECKBOX DEFAULT ' '.

************************************************************************
* Allgemeine Variablen und Typen
************************************************************************
*
DATA: PAR_TABLE_READ LIKE SY-INDEX VALUE 0.
DATA: OKCODE(8).
*** Parameter-Status laut sapparam.c / sapgnparam() *
DATA: PAR_STATUS            LIKE SY-INDEX.
DATA: PAR_USER_ONLY         LIKE PAR_STATUS VALUE 1.
DATA: PAR_DEFAULT_ONLY      LIKE PAR_STATUS VALUE 2.
*{   INSERT         C11K900018
DATA: ZPAR_CUST             LIKE PAR_STATUS VALUE 2. *
*}   INSERT
DATA: PAR_USER_AND_DEFAULT LIKE PAR_STATUS VALUE 3.

*** Felder der Parameter-Liste wie in sapaci.c C_SAPG
*data: par_name(60).
```

Figure 83: Modification Assistant

To support the different levels of modification, SAP provides tools such as the IMG, Business Add-Ins, transaction CMOD, and the ABAP development environment.

The Modification Assistant helps you modify SAP standard delivered objects and simplify the upgrade process. When you change SAP objects in an ABAP Workbench editor, you branch into a special modification mode. Originals are initially protected in this mode and are only changed when you click the corresponding buttons.

The Modification Assistant logs all modifications that are made in the system. The log is easy to read and dramatically reduces the amount of effort needed to apply Support Packages or to upgrade your system.

In addition, the Modification Assistant works with the most commonly used ABAP development tools such as ABAP Editor, Screen Painter, Menu Painter and text element maintenance.

The Modification Assistant is not used for changes made to objects such as global interfaces and classes and their components. These objects are adjusted in the system during applying of Support Packages or during upgrade in the conventional manner.

A list of all modified objects in the system can be shown with the Modification Browser (transaction SE95).

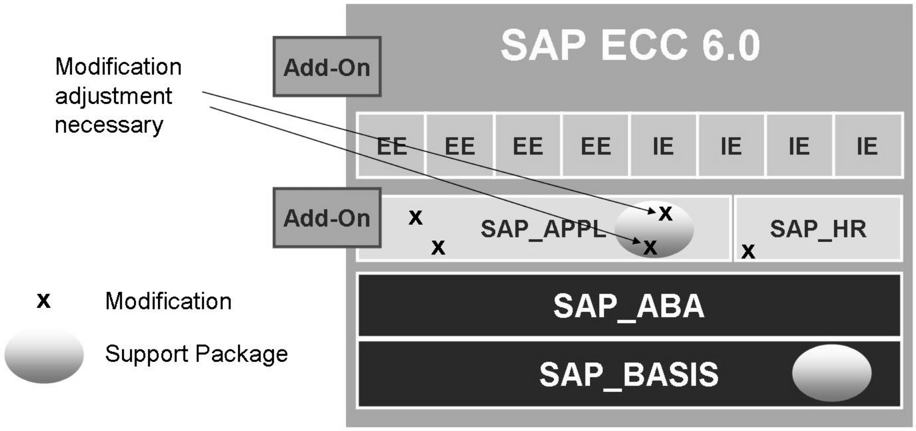


Figure 84: Modification Adjustment

If a new version of an SAP object is imported into the customer system, as part of a Support Package or an Upgrade, adjustments must be made to the repaired SAP object. Modifications are also considered repairs. This may involve making the same customer changes to the new SAP object.

Before applying a Support Package or performing an Upgrade, all open repairs must be confirmed and released. This check is performed by the upgrade check routines of the Support Package/Upgrade tool.

To adjust the ABAP Dictionary Repository objects during an upgrade, use transaction SPDD. To merge all other Repository objects, use transaction SPAU.

Adoption of modification is neither automatic nor trivial. Customers must decide which development efforts and modifications they wish to retain, bearing in mind that the new SAP functions may quite possibly make retaining the modification unnecessary.

Adjusting modified objects is simplified by the Modification Assistant.

Exercise 6: Modifying SAP Objects

Exercise Objectives

After completing this exercise, you will be able to:

- know, how to modify SAP Repository objects

Business Example

During the implementation process, you may be required to make changes to standard SAP Repository objects in order to fulfill the requirements of your business. The Modification Assistant is a tool delivered with the SAP system to aid you in making the modifications as well as helping with a future modification adjustment.

Task 1:

Modify an SAP Repository object.

1. You will be modifying the ABAP program *ADM325X##*, where ## is your group number. Which package does this object belong to? Which system is the original system?
2. Modify the source code of *ADM325X##*. Because this object is owned by SAP, normally you would be required to enter a registration key for the object. For the purposes of this class, you will not be prompted for a registration key.
3.

Hint: Either modify the program as described here or do any modification you want. For example, just add a **`write 'hello!'.`** line. The main thing is, you modify the program at all!

The program ADM325X## currently sorts transport request by the transport request number. You would like to see the transport requests sorted by author. Modify line 38 of the source code, which reads:

```
order by strkorr trkorr
```

Modify it so that it now reads:

```
order by strkorr as4user trkorr
```

Continued on next page

For all modifications, use the *Replace* function of the Modification Assistant. Save and activate your work to one of your transport requests. What type of change was created? Execute the program ADM325X##.

View the repair flag.

4. Release your task.

Task 2:

Modifying an ABAP Dictionary object.

1. You need to add an additional field to the table *ADM325X##* (where ## is your group number), review the structure of this table using the ABAP Dictionary.
2. Add the field *ZZBIRTHPLACE* to table *ADM325X##*. Assign data element *GBORT* to the field and choose *Enter*.
3. Activate and save the changes to the table *ADM325X##* to a new task within the same transport request used in the previous step of this exercise. The system automatically prompts you for the new task.
4. Add some new entries to the table *ADM325X##*.
5. Which TMS configuration setting ensures that modifications to SAP objects are transportable?
6. Release your task.
7. Now release the transport request.

Solution 6: Modifying SAP Objects

Task 1:

Modify an SAP Repository object.

1. You will be modifying the ABAP program *ADM325X##*, where ## is your group number. Which package does this object belong to? Which system is the original system?

   a) To access the ABAP Editor, choose transaction SE38. Enter the name of the SAP program, **`ADM325X##`**. Select *Attributes* and choose *Display*. The package for this object is the package *ADM325*.

   b) To find out the original system (one of the other attributes of this object), choose *Goto → Object Directory Entry.*

   > **Hint:** You can also find out the attributes in the object catalog, by choosing transaction SE03 and choose *Change objects directory entries*.
   >
   > Check the box in front of *R3TR PROG*, enter the name of the SAP program in the *Program name* field, **`ADM325X##`** and choose the *Execute* icon. On the next screen, select the program name. The original system is SAP.

2. Modify the source code of *ADM325X##*. Because this object is owned by SAP, normally you would be required to enter a registration key for the object. For the purposes of this class, you will not be prompted for a registration key.

   a) In the ABAP Editor (transaction SE38), enter the name of the SAP program, **`ADM325X##`**, select *Source Code*, and choose *Change*. Read the information in the next window about the usage of the modification assistant and then choose enter.

3.

   > **Hint:** Either modify the program as described here or do any modification you want. For example, just add a **`write 'hello!'.`** line. The main thing is, you modify the program at all!

Continued on next page

The program ADM325X## currently sorts transport request by the transport request number. You would like to see the transport requests sorted by author. Modify line 38 of the source code, which reads:

```
order by strkorr trkorr
```

Modify it so that it now reads:

```
order by strkorr as4user trkorr
```

For all modifications, use the *Replace* function of the Modification Assistant. Save and activate your work to one of your transport requests. What type of change was created? Execute the program ADM325X##.

View the repair flag.

a) Since the original system for this program is SAP, this type of change is termed a repair so that the Modification Assistant can control the changes performed to the object.

To view the repair flag, call transaction SE38. Enter the program name and choose *Goto* → *Object directory entry*. The most important object directory entries display. To view all attributes including the repair flag, choose the *Object attributes* icon.

4. Release your task.

a) To release your task, choose transaction SE09. Display the transport requests and tasks assigned to your user, ADM325-##. Select your task number and choose *Release directly*. **Do not** release the transport request yet.

Task 2:

Modifying an ABAP Dictionary object.

1. You need to add an additional field to the table *ADM325X##* (where ## is your group number), review the structure of this table using the ABAP Dictionary.

a) To display the structure of this table *ADM325X##*, call transaction SE11, the ABAP Dictionary.

b) Enter table name **`ADM325X##`**, choose the *Change* option.

Continued on next page

2. Add the field *ZZBIRTHPLACE* to table *ADM325X##*. Assign data element *GBORT* to the field and choose *Enter*.

   a) When in change mode in the ABAP Dictionary, to add a new field, place your cursor on a blank line and enter the field name **ZZBIRTHPLACE**.

   b) In the *Data element* field, enter **GBORT**.

3. Activate and save the changes to the table *ADM325X##* to a new task within the same transport request used in the previous step of this exercise. The system automatically prompts you for the new task.

   a) To activate the table, choose *Table → Activate*. Once activated, the status active displays for the table.

   b) Save the change to a new task within the same transport request you used for the SAP program ADM325X##.

4. Add some new entries to the table *ADM325X##*.

   a) To add additional table entries, choose *Utilities → Table Contents → Create Entries* or use transaction SE16.

5. Which TMS configuration setting ensures that modifications to SAP objects are transportable?

   a) All SAP objects are assigned a package with the transport layer SAP. In your system landscape, a valid consolidation route is assigned to the transport layer SAP. The consolidation route is from the development system, DEV, to the quality assurance system, QAS (or, in this case: via the transport target group to QAS).

6. Release your task.

   a) To release your task, go to the Transport Organizer (transaction SE09). Choose *Display* to display the transport requests and tasks assigned to your user, *ADM325-##*. Select your tasks and choose *Release directly*.

7. Now release the transport request.

   a) Select your transport request and choose *Release directly*.

Lesson Summary

You should now be able to:

- Modify SAP Repository objects
- Describe the difference between customer development and modifications
- Describe how the Modification Assistant assists developers

Unit Summary

You should now be able to:

- Explain what Customizing is, from the business and the technical perspective
- Explain how the SAP Implementation Guide (IMG) and IMG Projects can be used for Customizing
- Explain how the SAP Solution Manager can be used for implementation projects
- Create transport requests and tasks
- List the tools used for managing transport requests
- Describe how to customize
- Follow steps in procedure for customizing an SAP system implementation
- Define cross-client customizing
- Plan change management for development
- Identify various implementation tools used for Customizing and development
- Describe the function of the Transport Organizer
- create a simple program
- understand the principles of the version management
- check and adjust the Transport Organizer settings
- Modify SAP Repository objects
- Describe the difference between customer development and modifications
- Describe how the Modification Assistant assists developers

Test Your Knowledge

1. Select the correct descriptions of what the IMG is or does:

   Choose the correct answer(s).

   □ A Provides customizing transactions.

   □ B Is where all Customizing activities must be performed.

   □ C Gives you offline access to all the information required to Customize an SAP system.

   □ D Offers project management tools.

2. It is possible to extend an existing project IMG.

   Determine whether this statement is true or false.

   □ True

   □ False

3. What is a transport request ?

   ____

   ____

   ____

   ____

4. SAP recommends that:

   Choose the correct answer(s).

   □ A Project manager or person responsible for the transport (a limited number of users) are creating the transport requests

   □ B Project members should always create their transport request by there own

   □ C The project members are assigned to the transport request by creation of tasks for each project member inside the transport request

   □ D Project members are recording changes in their own task (assigned by the project manager or person responsible for the transport)

5. What should be done before releasing a transport request in the development system?

   Choose the correct answer(s).

   □ A Unit testing

   □ B Release of all tasks inside the transport request

   □ C Deletion of tasks inside the transport request

   □ D Information of system administration about the released tasks

6. Client-independent customizing objects must be saved to transport requests of type ___________________.

   Fill in the blanks to complete the sentence.

7. The change management solution in SAP systems is based on creating, documenting and distributing transport requests.

   Determine whether this statement is true or false.

   □ True

   □ False

8. What is the function of Transport Organizer?

   ____

   ____

   ____

   ____

9. All development in a system landscape should be done in a single SAP System.

   Determine whether this statement is true or false.

   □ True

   □ False

10. When are versions being generated?

   ____

   ____

   ____

   ____

11. How is the changing of an SAP standard Repository object termed?

12. What is the function of Modification Assistant?

Answers

1. Select the correct descriptions of what the IMG is or does:

   Answer: A, D

   The IMG gives you online access to all the information required to customize an SAP system. Customizing can also be performed without the IMG, of course.

2. It is possible to extend an existing project IMG.

   Answer: True

   The project IMG can be extended with the help of transaction SPRO_ADMIN at any time. After the change of the IMG, a regeneration has to be done.

3. What is a transport request ?

   Answer: A transport request is a information source in the Transport Organizer that can record all changes and modifications made to repository objects and customizing settings during the adoption of the system.

4. SAP recommends that:

   Answer: A, C, D

   These are SAP recommendations for projects.

5. What should be done before releasing a transport request in the development system?

   Answer: A, B

   Before a transport request in the development system is released, first the unit testing in the development system has to be successfully finished. Second, all tasks that are contained in the transport request have to be released by the customizers. Otherwise it is not possible to release the transport request by the project leader.

6. Client-independent customizing objects must be saved to transport requests of type workbench.

   Answer: workbench

7. The change management solution in SAP systems is based on creating, documenting and distributing transport requests.

   Answer: True

   The change management solution in SAP systems is based on creating, documenting, and distributing transport requests.

8. What is the function of Transport Organizer?

   Answer: Transport Organizer records customizing changes and ABAP Development Workbench changes in transport requests for Customizing and development.

9. All development in a system landscape should be done in a single SAP System.

   Answer: True

   All development in an SAP system landscape has to be done in the same source system, the development system.

10. When are versions being generated?

    Answer: Versions are generated when exporting a transport request. According to the tp parameter settings also when importing a transport request (not by default). A temporary version can be generated manually at any time by the developer.

11. How is the changing of an SAP standard Repository object termed?

    Answer: An SAP standard Repository object is delivered as a copy in all customer systems. Changing it is classified as a repair. Repairing an SAP standard Repository object is also called modification.

12. What is the function of Modification Assistant?

    Answer: The Modification Assistant helps you modify SAP standard delivered objects and simplify a possible Modification adjustment during applying a Support Package or performing an Upgrade.

Unit 4

Importing Transport Requests

Unit Overview

In this unit, you will learn about the transport process. You will learn how to perform imports using the Transport Management System (TMS) and also monitor and troubleshoot the import process using TMS. You will learn about the different transport management strategies.

In addition, this unit will help you identify the file naming conventions of the transport directory. You will also be able to clean up the transport directory.

Unit Objectives

After completing this unit, you will be able to:

- Describe the transport flow within a three-system landscape
- Outline the transport process
- Identify the tools used to perform imports using the Transport Management System (TMS)
- Explain the relationship between import queues and import buffers
- Follow the procedure for setting up the date and time for an import
- Decide whether to import single transport requests or to import the entire import queue
- List the different transport strategies
- Define Quality Assurance (QA) procedure
- Describe a situation in which using special transport workflow would be practical
- Explain why the transport proposal is necessary
- Identify the steps of the import process
- Identify the relationship between tp, R3trans and RDDIMPDP
- Identify the tools for monitoring transport activity
- Identify the way to ensure a stable development environment
- Follow the steps for cleaning up a transport directory

- Outline the process of transporting non-ABAP objects with the help of ABAP Change and Transport System

Unit Contents

Lesson: The Transport Process

Lesson Overview

In this lesson, you will learn about the Transport Management System (TMS) within the SAP system and learn what's involved in the transport process.

Lesson Objectives

After completing this lesson, you will be able to:

- Describe the transport flow within a three-system landscape
- Outline the transport process

Business Example

As the transport administrator, you should ensure the correct movement of changes throughout the system landscape by managing the flow of the transport process.

Steps in Transport Process

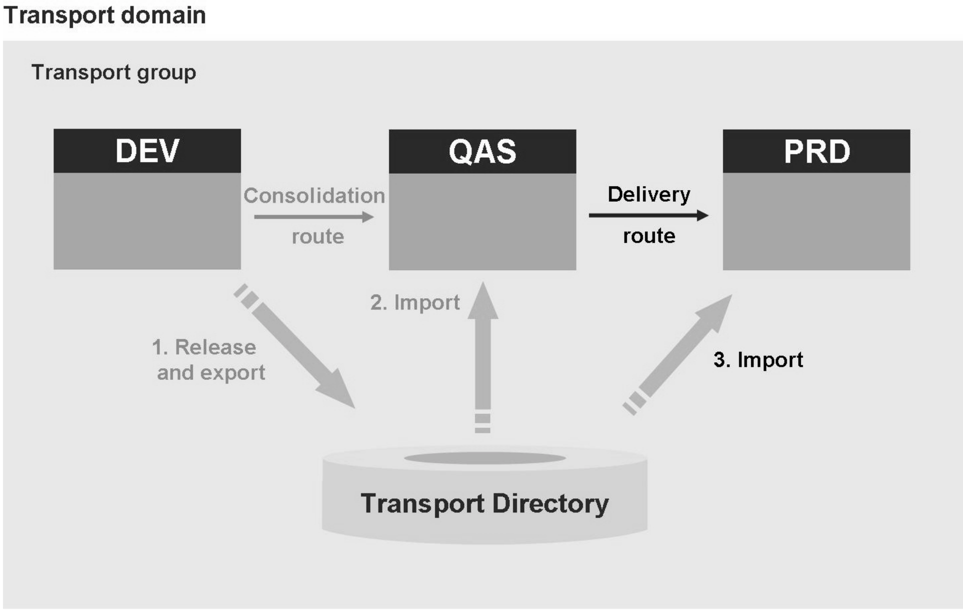


Figure 85: Overview: TMS Transport Strategy

Before release 3.1H, transports were performed at the operating system level. With the introduction of the Transport Management System (TMS), transports are performed from within the SAP system.

In the TMS, transport requests are transported along predefined transport routes. You can define multiple consolidation and delivery routes. The import procedure can be performed by any authorized user from within the SAP system. Nevertheless nearly all functions in transaction STMS are executed by tp commands at operating system level, which someone with technical knowledge can also perform manually.

Change requests to be imported are displayed in the import queue of the target system in the SAP system.

Using TMS you can import a complete import queue, that is, all transport requests that were exported from the development system. This ensures that no import errors occur due to missing objects and that newer versions of an object do not get overwritten by older versions.

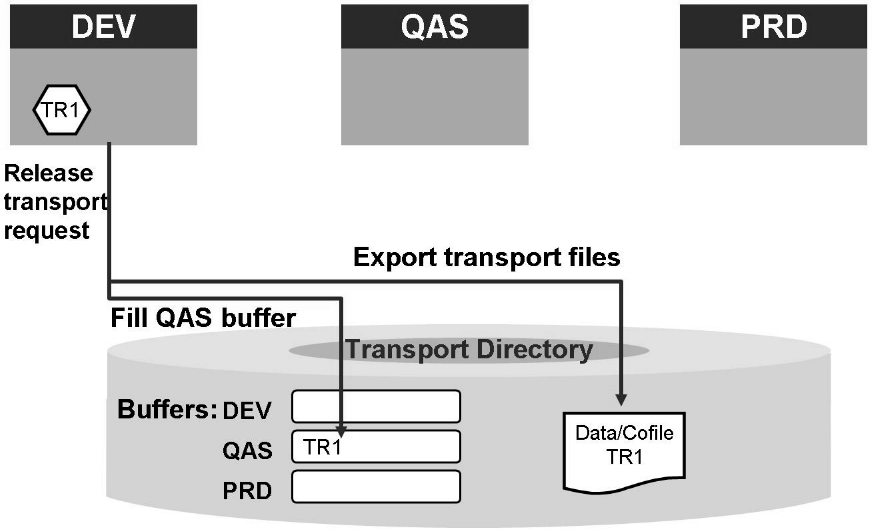


Figure 86: Transport Process: Release and Export

The first step in the transport process is to release a transport request and thereby export all the associated objects from the database of the development system (DEV) to files in the common transport directory at the operating system level. For each request released, the data is exported to a data file in the subdirectory *data* and a control file is written to the subdirectory *cofiles*.

During export, the entries required for the subsequent import are created in the import buffer of the target system, and a test import can be performed.

In the directory *buffer*, at the operating system level, there is an import buffer file for each SAP system in the transport domain. The file is named after the corresponding system ID and contains control information regarding which transport requests are to be imported and the order of import.

Several transport control commands can be used to manage the import buffer files at the operating system level. The control information in the import buffer files is read and represented in import queues accessed from within the SAP system. An import queue shows all transport requests that are listed in the corresponding buffer.

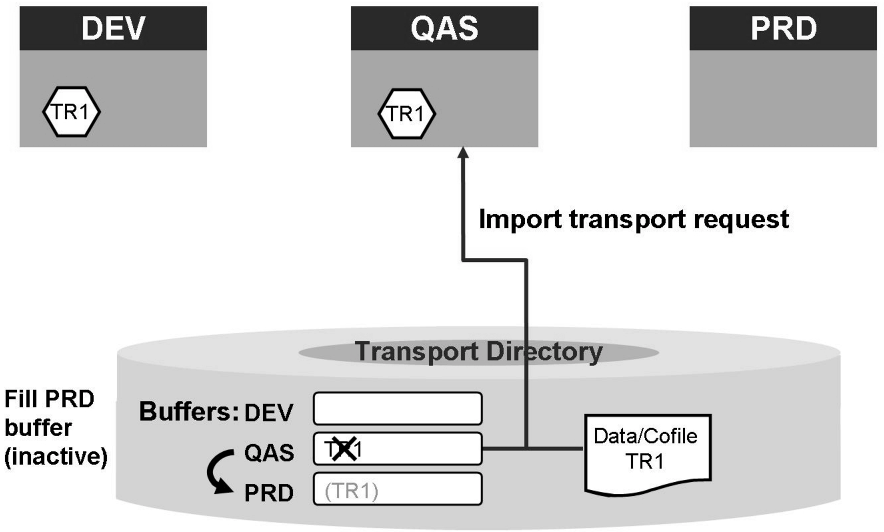


Figure 87: Transport Process: Import into Quality Assurance

Using TMS from within the SAP system, the second step in the transport process is: importing all requests listed in the import queue of the quality assurance system (QAS). TMS performs the import by starting the transport control program tp at the operating system level.

After the successful import into the quality assurance system, the requests are placed inactively in the import buffer and import queue of the production system (PRD) and any other delivery system.

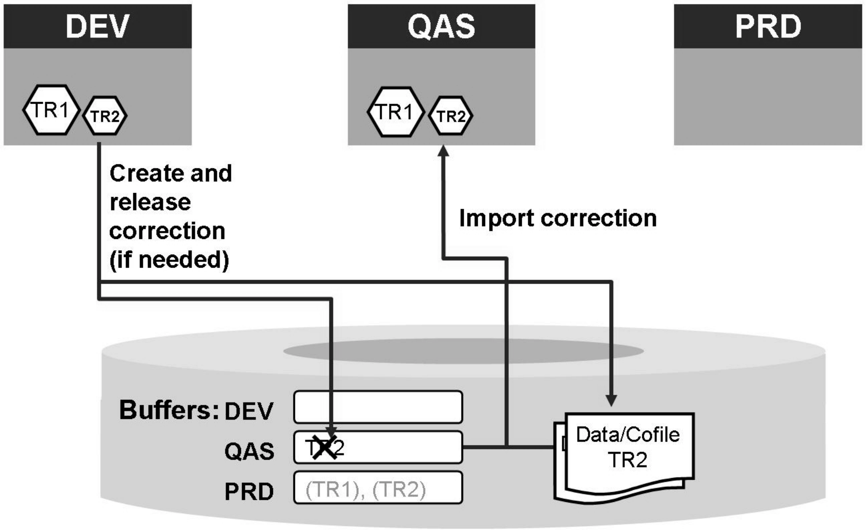


Figure 88: Transport Process: Quality Assurance

After import to the quality assurance system, e.g. QAS, the objects need to be tested for possible errors. Errors should be corrected in the development system, e.g. DEV, and the changes again imported into QAS. During import into QAS, additional transport requests are added to the buffer of the production system PRD.

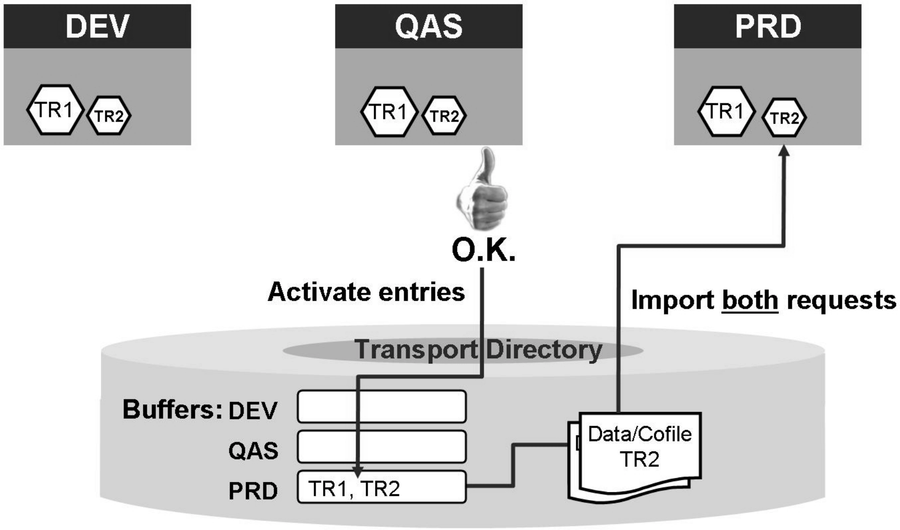


Figure 89: Transport Process: Import into Production

After all the transport requests that were imported into QAS have been thoroughly tested and verified, the transport requests must be approved. The status of the entries then changes from inactive to active and the transport requests are ready for import into the PRD.

Using TMS, you can import all transport requests, or simply a first set of verified transport requests, listed in the production system import queue in the given, not manipulated, sequence.

To ensure that there is no negative effect on production activities in PRD, ensure that errors and their corrections are imported in the correct order.

Lesson Summary

You should now be able to:

- Describe the transport flow within a three-system landscape
- Outline the transport process

Lesson: Imports Using TMS

Lesson Overview

In this lesson, you will learn about the tools for performing imports using the Transport Management System (TMS). You will learn how import queues and import buffers are related and about importing the requests in the queue. You will also learn how to set up the date and time for carrying out an import. The different transport strategies will be introduced.

Lesson Objectives

After completing this lesson, you will be able to:

- Identify the tools used to perform imports using the Transport Management System (TMS)
- Explain the relationship between import queues and import buffers
- Follow the procedure for setting up the date and time for an import
- Decide whether to import single transport requests or to import the entire import queue
- List the different transport strategies

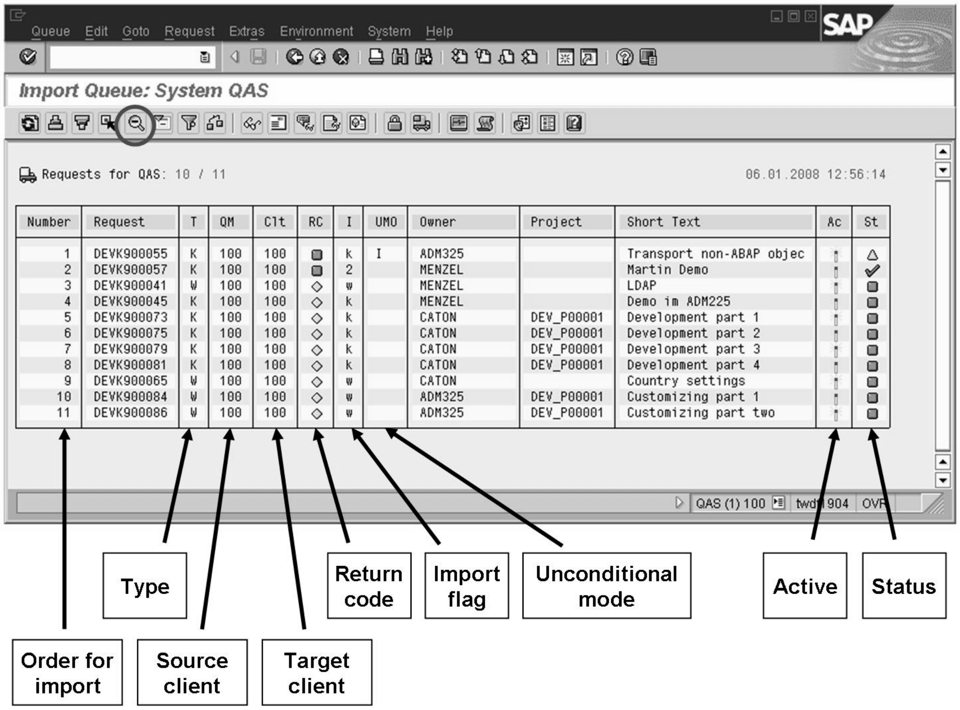
Business Example

After the project team leader has confirmed and released all required transport requests from the development system, the transport administrator must import the transport requests into the target systems. The transport administrator must follow the guidelines established for the transport strategy and the transport schedule. In this way, the transport administrator can ensure that changes are consistently distributed to all systems in the landscape.

Importing Change Requests Using TMS

The most important tools used to perform imports using TMS are the import queues that reflect the system-specific import buffers at the operating system level. The import queues display the requests that are to be imported, in the correct order. The import queues of all systems are displayed in each system of the transport domain. You can perform imports to all the systems from any SAP system in the domain.

To access the TMS import overview, use transaction STMS and choose *Overview → Imports*. The import overview shows the current status of the import queue of each SAP system of the transport domain. If you jump in an import queue of one system, you can see all to be imported transport requests.



Import Queue: System QAS

Requests for QAS: 10 / 11 06.01.2008 12:56:14

Number	Request	T	QM	Clt	RC	I	UMO	Owner	Project	Short Text	Ac	St
1	DEVK900055	K	100	100	■	k	I	ADM325		Transport non-ABAP objec	▮	△
2	DEVK900057	K	100	100	■	2		MENZEL		Martin Demo	▮	✓
3	DEVK900041	W	100	100	◇	w		MENZEL		LDAP	▮	■
4	DEVK900045	K	100	100	◇	k		MENZEL		Demo im ADM225	▮	■
5	DEVK900073	K	100	100	◇	k		CATON	DEV_P00001	Development part 1	▮	■
6	DEVK900075	K	100	100	◇	k		CATON	DEV_P00001	Development part 2	▮	■
7	DEVK900079	K	100	100	◇	k		CATON	DEV_P00001	Development part 3	▮	■
8	DEVK900081	K	100	100	◇	k		CATON	DEV_P00001	Development part 4	▮	■
9	DEVK900065	W	100	100	◇	w		CATON		Country settings	▮	■
10	DEVK900084	W	100	100	◇	w		ADM325	DEV_P00001	Customizing part 1	▮	■
11	DEVK900086	W	100	100	◇	w		ADM325	DEV_P00001	Customizing part two	▮	■

Figure 90: Import Queue Information

With the extended transport control you could have multiple target clients for one target system. With SAP Web AS 6.40 there is a new alternative display format available for this. You can choose either you want to see one row for each target client or only one row per transport request..

To improve performance, data is read from the transport directory only the first time you start TMS. After that, the information shown is buffered in the database. The time stamp in the import overview indicates how recent the data is. The internal buffers of TMS become invalid at midnight. To refresh the data, choose *Edit → Refresh*. It may be more convenient to have the refresh performed periodically in the background. To do this, choose *STMS → Overview → Imports → Extras → Update All Import Queues*. SAP recommends running this refresh hourly using **RSTMSCOL**.

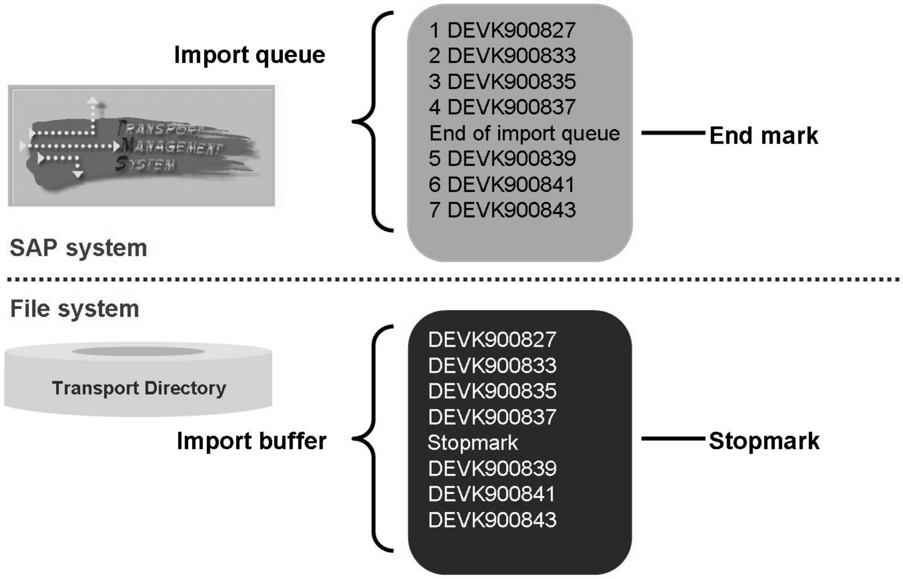


Figure 91: Import Queues and Import Buffers

The terms import buffer and import queues are related. The import queue in the SAP system represents the import buffer located in the transport directory. The import queue highlights the requests that will be imported during the next start import (import all). Because of end marks, there may be more requests in the import buffer than those highlighted in the import queue.

The end mark and stopmark are the corresponding markers in the import queues and buffers respectively. These indicate that only the requests before the mark will be imported. Regardless of how the end mark or stopmark is created, the mark is set in both the import buffer and the import queue. An end mark is indicated in an import queue through the statement “End of import queue.” In an import buffer, the term **stopmark** is visible. There can only be one end mark or stopmark in each import queue or import buffer.

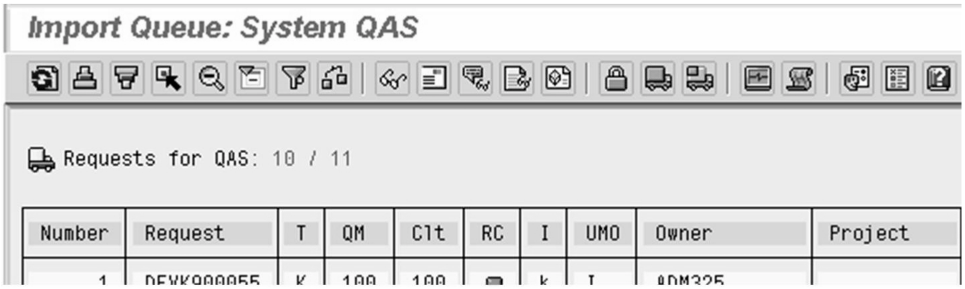
To set an end mark/stopmark by closing an import queue, in the import queue choose *Queue* → *Close*. This is analogous to the operating system command **tp setstopmark**. The status bar shows the action performed.

To remove an end mark/stopmark by opening an import queue manually, normally not needed, in the import queue screen choose *Queue* → *Open*. Opening an import queue is analogous to the operating system command **tp delstopmark**.

Using TMS, you can move end marks to any position in the import queue in front of a transport request (*Queue* → *Move End Mark*). This is analogous to the operating system command **tp mvstopmark**.

Use the import queue to:

- **View the status of requests and access:**
  - Object lists
  - Documentation
  - Transport logs
- **Close and open the queue, move the end mark**
- **Import:**
  - All requests
  - Complete Projects
  - Preliminary requests
  - Requests according to filter settings
- **Add, delete and forward requests**


Import Queue: System QAS

Requests for QAS: 10 / 11

Number	Request	T	QM	Clt	RC	I	UMO	Owner	Project
1	DEVK900055	K	100	100	▬	k	I	ADM325	

Figure 92: Import Queue

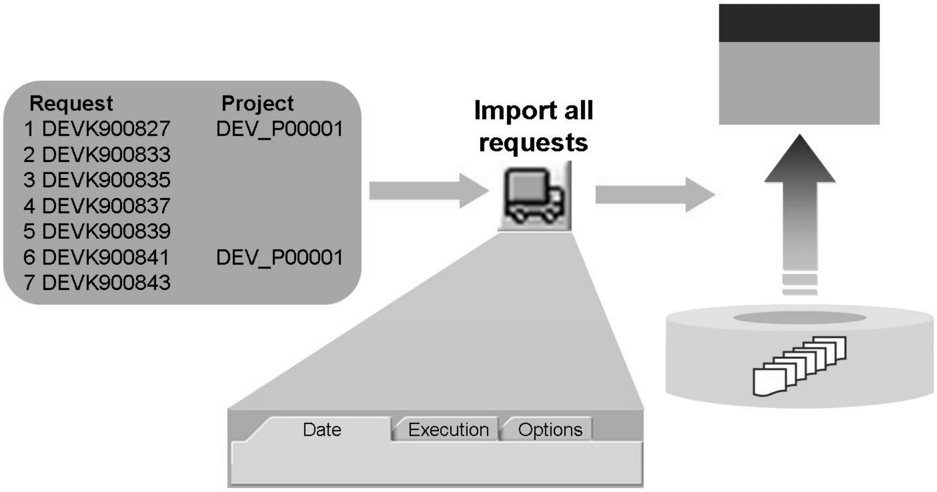
Use the import queue to:

- View the status of requests
- Access object lists, documentation, and transport logs
- Close and open the queue, and move the end mark
- Import all requests, completed projects, preliminary requests, and selected requests according to filter settings
- Add, delete and forward requests

To keep target SAP systems consistent, you need deadlines to coordinate the release of transport requests by developers. To prevent transport requests released after the deadline from being imported, the import queue can be closed. These transport requests are positioned after an end mark in the queue for the next import. Only the requests before the end mark are imported in the next import.

In exceptional cases, you can forward a transport request to another SAP system before being imported into the defined target system. For example, before being imported into the quality assurance system, a request may need to be rushed to a training system. To import to a target system outside the predefined transport routes, in the *Import Queue* screen, choose *Request → Forward → System*.

You can also delete transport requests from or add them to an import queue. Object dependencies may cause inconsistencies in the target system after the next import. For example, if you delete a request containing a new data element, all other requests containing tables that depend on that data element will fail. To avoid these inconsistencies, you are strongly advised not to delete individual transport requests. Make your corrections in the development system and release a new transport request.



Request	Project
1 DEVK900827	DEV_P00001
2 DEVK900833	
3 DEVK900835	
4 DEVK900837	
5 DEVK900839	
6 DEVK900841	DEV_P00001
7 DEVK900843	

Figure 93: Import all Transport Requests

To import all requests in the queue (thereby performing what is known as an **import all**), choose the *Import All Requests* button (the “big truck”). The *Start Import* popup appears.

If you have configured the Extended Transport Control, the target client is fixed. Otherwise you can either choose a target client or keep the default. The default target client is the source client. (This is one reason, why you should name all “major” clients in all systems alike.)

Hint: If you start the import from another SAP system in the transport domain, a logon window from the target system is displayed.

In the *Start Import* popup you have several options to control the import:

- In the *Date* tab you can schedule the import.
- In the *Execution* tab you can select whether TMS starts tp synchronously or asynchronously. Selecting *asynchronously* means tp works in the background so that your user session is not blocked for the duration of the import.
- In the *Options* tab you can select “expert options”, the so-called unconditional modes. The options and their defaults vary according to the selected import method and the configured transport strategy.

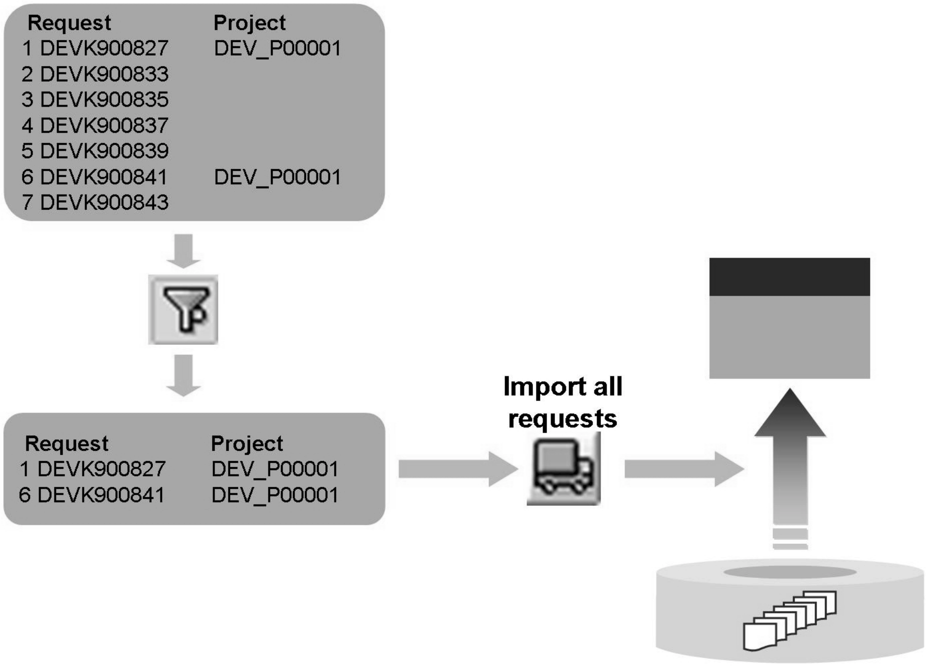
, The *Import Overview* screen indicates whether the import is running. After the import, the end mark is removed and the queue is opened again automatically. After transport requests have been imported successfully, they are automatically added to the import queue of further target systems. When using the QA Approval Procedure function, all requests in the import queue of the subsequent systems are set to inactive. The configured transport route specifies which transport requests are automatically forwarded to which target systems. If an import containing one or more inactive requests is started, the TMS will not perform the import.

Note: You can only import all the requests into the delivery systems if all the requests ready for import have been checked for all the applicable approval steps (approved or rejected).

If all the requests for one project have been approved, you can import them into the delivery system even if there are still unprocessed or rejected requests in the work list.

If you perform an import through an *import all*, objects are imported in the correct sequence as they are listed in the import buffer file on operating system. This means that if transport requests near the start of the list and those near the end of the list affect the same objects, the final versions of the objects after import will represent the latest changes. As a result, the incorrect objects do not affect your productive environment.

You can deactivate the ability to perform a complete import (import all) for each system using the TMS parameter **`NO_IMPORT_ALL`**.



Request	Project
1 DEVK900827	DEV_P00001
2 DEVK900833	
3 DEVK900835	
4 DEVK900837	
5 DEVK900839	
6 DEVK900841	DEV_P00001
7 DEVK900843	

Request	Project
1 DEVK900827	DEV_P00001
6 DEVK900841	DEV_P00001

Figure 94: Import Complete Project

Before performing the import, SAP recommends setting the end mark to close the import queue. This avoids unintentionally importing other transport requests that may appear in the import queue.

You can set a filter on the import queue to limit the displayed transport requests to transport requests with specific properties so that you can see only those transport requests that belong to a specific project. To set a filter, place the cursor in a row of the import queue and press the *Filters* button.

To prevent transport requests being imported from unapproved projects, use the filter to import only the transport requests belonging to approved projects.

As a new feature with SAP Web AS 6.40 it is possible with one tp process to import the requests of multiple projects into multiple target clients.

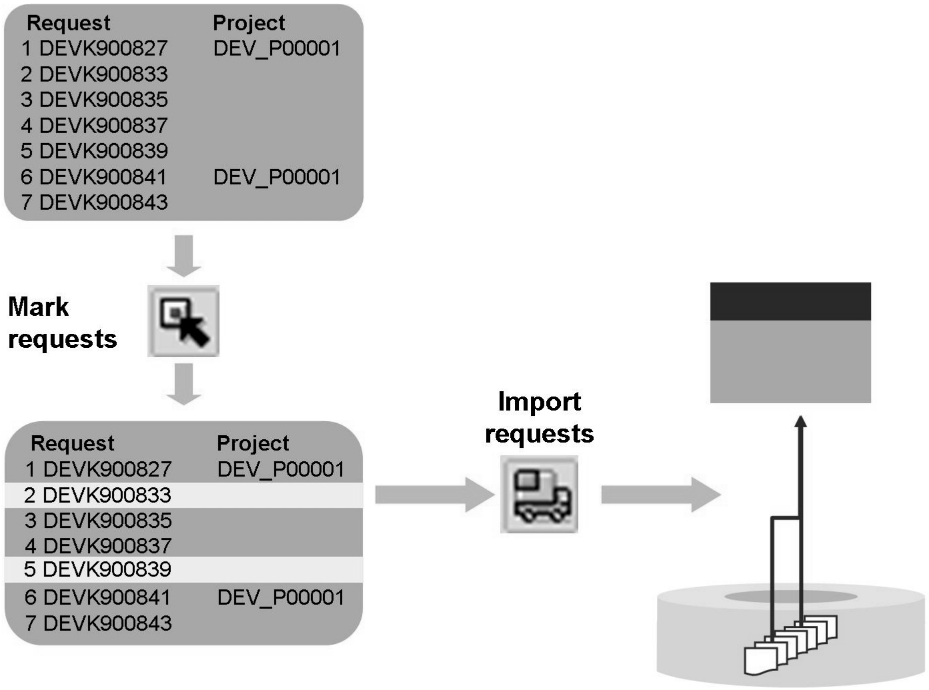


Figure 95: Import Single Transport Requests (Preliminary Import)

In contrast to standard imports, preliminary imports are imports of single requests. SAP strongly recommends only using standard imports because of object dependencies and the risk of inconsistencies when importing individual requests. For example, an ABAP report in one request can be successfully imported, but the table that it refers to can be in another request that has not yet been imported. Until the table is imported, executing the report generates short dumps. Use preliminary imports only in exceptional cases.

To minimize the risks associated with preliminary imports, the request remains in the import queue after the import and is re-imported the next time the entire import queue is imported. This guarantees that export and import sequences are always the same. This is defined by the option `Leave Transport request in queue for later import` , which is marked, depending on the transport strategy.

By default, the TMS checks if requests in the import queue depend on transport requests in other projects. The import is only possible, if the predecessor's relationships will not be damaged. To change this default setting, choose for example, *Request → Import* and select *Continue import*. In particular circumstances, it may be necessary to specify additional options when performing a preliminary import:

- Ignore that the transport request has already been imported
- Overwrite originals
- Overwrite objects in unconfirmed repairs
- Ignore invalid transport type

The objects from the transport requests marked for import will be imported as follows:

- all objects of all selected transport requests are merged together
- first of all the objects are sorted according their level (e.g. table definitions before programs)
- in case of an object in more than one transport requests, only the version in the last transport request is imported (according to the sequence in the import queue)
- this is independent of the import method (import all, import project, import single)

Note: In a three system landscape, the import queue of QAS reflects the order of export from DEV. The import queue of PRD reflects the order of import into QAS. This is not identically in all cases. But it is the correct sequence.

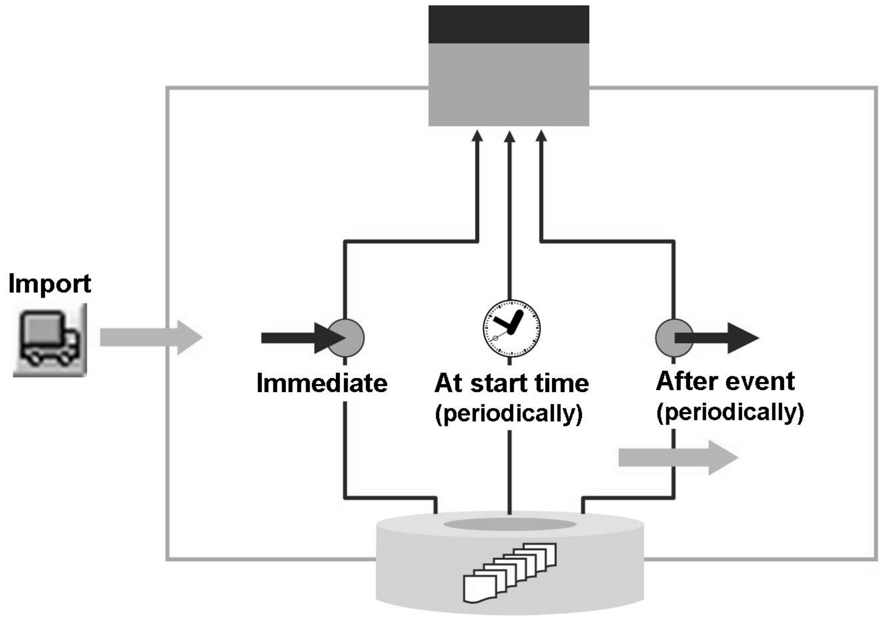


Figure 96: Time and Date of Import

Depending on if you choose to import by project, individual import, import all, or special transport workflow, and especially on the SAP release and Support Package level, the available options may vary. When starting an import, you can select the following options in the *Date* tab:

Immediate

Selecting this option immediately starts the import in a dialog work process.

At Start Time

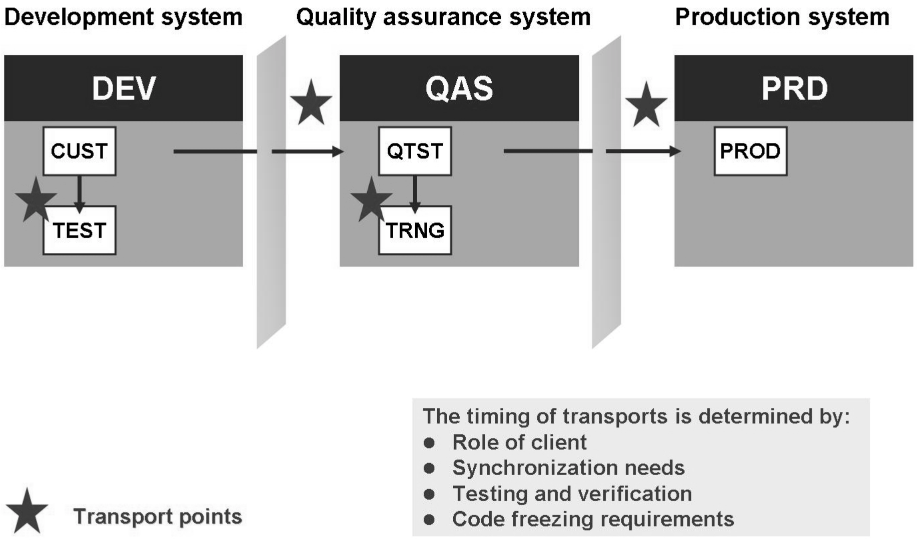
Selecting this option starts the import at the specified time. The import is scheduled as a background job in the target system. If you also enter a date and time in the *No Start After* field, the import is started in the time frame between the times entered in *Planned Start* and *No Start After*. If there is no background process available during this time, an import does not occur. If you want the import to be performed regularly, you must enter a time in the *Period* field.

After Event

Selecting this option starts the import only after a specified event is triggered. If you select *Execute Import Periodically*, the import is started each time the specified event is triggered. Otherwise, the import is started only the first time the event occurs.

Note: Note that the *Period* field does not exist for individual transports and the special transport workflow.

From the import queue of each SAP system, you can monitor and maintain all planned imports by choosing *Goto → Job Monitor*



The timing of transports is determined by:

- Role of client
- Synchronization needs
- Testing and verification
- Code freezing requirements

Figure 97: Timing Transports

After export, a request is not automatically imported, but must be manually imported. When planning imports, include enough time to accommodate post-import tasks such as the quality assurance testing. SAP recommends planning imports at regular intervals, such as monthly, weekly, or daily, using *Import All* into the target system. Frequent importing is not recommended.

The following actions have to be considered:

- Releases of transport requests
- Copy into a client in the same system using transaction SCC1 (unit testing)
- Imports into clients in downstream systems

The timing of transports is based on the following factors:

- Clients and their roles in your system landscape
- Synchronization requirements, that is, when changes are required in different systems, prior backups
- Code freezing requirements

Transport strategies

There are three different transport strategies available:

- queue-controlled mass transports
- queue-controlled single transports
- workflow-controlled transports

Hint: The transport strategy is set by default to queue-controlled mass transports.

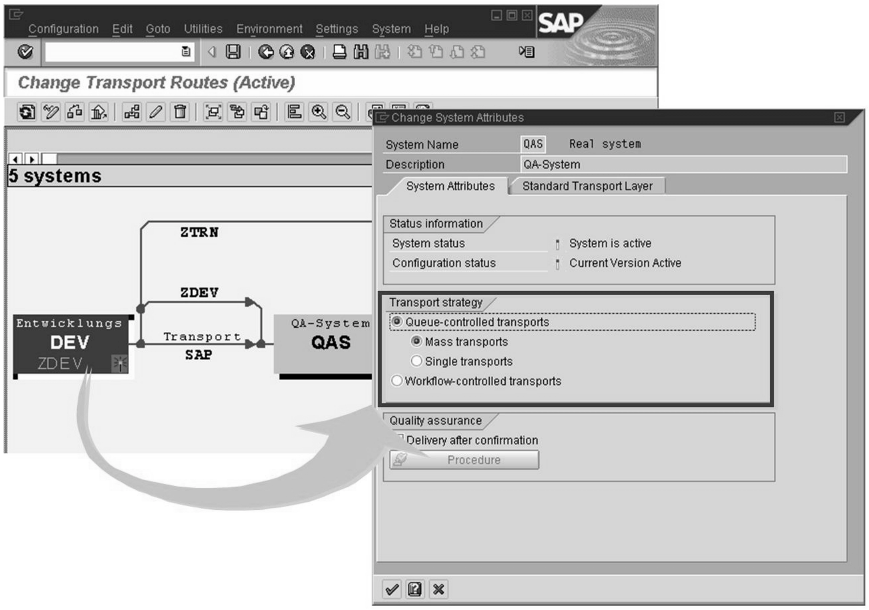
Mass transports are a good solution if you have a large number of transports to administrate and want to automate the process as much as possible. The continuous use of mass transports is the most secure way of keeping your systems synchronized and consistent. Before you import mass imports into your production systems, you must check all requests in the quality assurance system and confirm their transport into other systems. Use the TMS Quality Assurance procedure when doing this. You also need to define mass imports as the import method for the relevant systems. To do this, choose the transport strategy Queue-controlled mass imports. The administrator can schedule the imports periodically in TMS, or start each import manually. Only import single transport requests (single imports) before others in the queue in special cases. Change requests that are imported in advance by the TMS are imported again in the regular import. You can use the transport workflow to import single imports in advance.

If you want to maintain a production system with specific transports, it is best to **import single** requests rather than importing all changes waiting for import. Use single transports if you have fewer changes to transport and your organization prevents you from having a fixed transport schedule. This method usually entails extra work for the administrators compared to periodic imports. Developers need to pay extra attention to the consistency of their transport requests. IIf a small number of developers are working on a project, or if the developers work very closely with the administrator, they often make their own single transports.

You want to make specific single transports into your systems, but would rather have this done by the system administrator, we recommend that you use the **transport workflow**. This method automatically triggers a workflow when you release a transport request. The workflow ensures close communication between development and administration. As prerequisite you have configured the transport workflow for your system (see Lesson *QA Approval Procedure and Transport Proposals*).

Maintain transport strategy

If you want to work with queue controlled single transports or workflow-controlled transports instead, change the configuration as described here.


Configuration Edit Goto Utilities Environment Settings System Help
SAP
Change Transport Routes (Active)
5 systems
ZTRN
ZDEV
Entwicklungs
DEV
ZDEV
Transport
SAP
QA-System
QAS
Change System Attributes

System Name QAS Real system

Description QA-System

System Attributes | Standard Transport Layer

Status information

System status: System is active

Configuration status: Current Version Active

Transport strategy

● Queue-controlled transports

● Mass transports

○ Single transports

○ Workflow-controlled transports

Quality assurance

□ Delivery after confirmation

Procedure

Figure 98: Transaction STMS: Overview -> Transport Routes

1. Start transaction *STMS* and choose *Overview* → *Transport Routes*. The screen *Display Transport Routes* appears displaying the existing transport routes in the transport domain.
2. Switch to the change mode by choosing *Configuration* → *Display <-> Change*.
3. Position the cursor on the SAP system for which you want to choose a transport strategy.
4. Choose *Edit* → *System* → *Change*. The popup *Change System Attributes* appears.
5. Select the tab *System Attributes* and choose your transport strategy.

   Note: Within a transport group you must set all systems either to *Queue-controlled transports* or *Workflow-controlled transports*.

6. Choose *Copy*.
7. Choose *Configuration* → *Distribute and activate*

Settings in TMS depending on the transport strategy

Queue controlled mass transports

By default, the import option *Leave transport request in queue for later import* is activated, when doing an *import single*.

Hint: The import option *Leave transport request in queue for later import* causes requests that have been imported as single transports to be imported in the correct order in the next import of all requests. This option is useful if you have to make preliminary imports for individual requests. This prevents older objects from being imported at the next regular import of all the requests (overshooter problem).

Single transports

- By default, the import option *Leave transport request in queue for later import* is deactivated.

  Hint: If you supply a system only with single transports, this option is not useful since the requests remain in the queue after the import and have to be manually deleted from the import queue.

- The push-button bar in the import queue changes according to the requirements of the single import strategy (for example, it contains a select function).
- In the *Import All Requests* function ("big truck") is only available if you have selected one or more projects using the filter function. This prevents accidentally importing all the requests in the queue.

Workflow-Controlled Transports

- Transport proposals are created automatically when requests are exported.
- The import options correspond to those for single transports.
- Imports become linked to the processing of transport proposals in the TMS worklist.
- A warning appears in the import queue if you try and make transports without using the transport workflow

  .

The following parameters for the transport control program and the Change and Transport System (CTS) are set according to the chosen transport strategy:

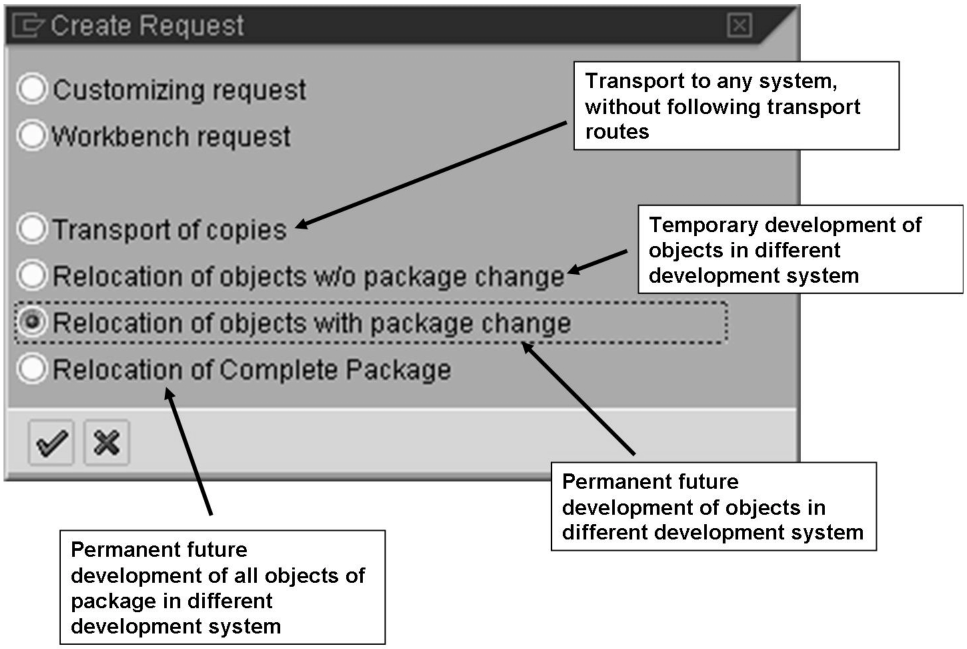
tp parameters for transport strategies

Parameter	Value for Mass Transports	Value for Single Transports	Value for Workflow-Controlled Transports
IMPORT_SINGLE_ONLY	0	1	1
NO_IMPORT_ALL	0	1	1
IMPORT_SINGLE_STRATEGY	0	1	0
WORKFLOW_STRATEGY	0	0	1
REPEATONERROR	9	8	8
STOPONERROR	9	9	9

The tp parameter *STOPONERROR* defines, starting from which return code the import stops. *REPEATONERROR* defines, from which return code on, the import is not classified as successful and has to be repeated. For example, with *Single Transports*, return code 8 is classified as an non successful import and has to be repeated. With *Mass Transports*, the same return code 8 would be a successful import.

Hint: Do not manually change the parameters that are relevant to the transport strategy. TMS generates these parameters each time the transport route configuration is changed.

Transport of Copies and Relocation



Create Request

○ Customizing request

○ Workbench request

○ Transport of copies

○ Relocation of objects w/o package change

● Relocation of objects with package change

○ Relocation of Complete Package

Transport to any system, without following transport routes

Temporary development of objects in different development system

Permanent future development of objects in different development system

Permanent future development of all objects of package in different development system

Figure 99: Transport of Copies and Relocation

Use *Transport of copies* to transport objects to another SAP system of your choice. Use this function especially, if the target system is not a consolidation system from the point of view of the current SAP system. The objects are transported with the version they have in the current SAP system. The original location of the objects is not changed. The transport request will not be added to the import queue of a subsequent system. This function replaces T-type transports as of release 3.x.

Use *Relocations of objects w/o package change* if development work on objects is to take place in another SAP system temporarily. Special developments may be carried out in a separate SAP system, for example, so as not to interfere with the development process. This request type allows you to move the original location of objects to the target system. This function replaces "Transport of Originals" or C-type transports as of Release 3.x.

Use *Relocations of objects with package change* when the development system of individual objects is to be changed on a permanent basis. This request type allows you to change the original location of objects to the target system and change the package of the objects at the same time. Due to the package being changed automatically, the objects have the desired transport attributes immediately after being imported into the target system of the request.

Use *Relocations of Complete Package* when the development system of a complete package is to be changed on a permanent basis.

Exercise 7: Imports Using TMS

Exercise Objectives

After completing this exercise, you will be able to:

- Use the tools for performing imports using the Transport Management System (TMS)

Business Example

The project team leader has confirmed and released all required transport requests from the development system. The transport administrator can now import the transport requests into the target systems. The transport administrator has to follow the guidelines established for the transport strategy and the transport schedule. This way the transport administrator can ensure that changes are distributed consistently to all systems in the landscape.

Task 1:

Hint: Before proceeding, please check, if all transport requests are released from the development system (DEV). If not, release them.

Reviewing the import queues for all SAP systems

1. Are there any request waiting to be imported?
2. Are there any requests to be imported into the production system?
3. Review the current status of the import queues of all SAP systems. Is the import queue of QAS open or closed?

Task 2:

Review the import queue of the QAS

1. Display the requests that are to be imported into the quality assurance system (QAS). Which SAP users released the request?
2. Display the objects to be imported with a transport request that was released with your account.

Continued on next page

Task 3:

Importing all transport requests of a project into the quality assurance system (QAS)

Hint: For importing transport requests, you should log on to the QAS system with your user. If you would start the imports from within the DEV system, you would be prompted for a logon, when you start the import.

Note: Use the option *Leave transport request in queue for later import* (tabstrip *Options* on the *Import Transport Request* popup), if it exists. The existence of this option depends on the tp parameter *CTC* (client dependent transport routes) and is release dependent!

1. Display more information about the transport requests now waiting in the import queue for QAS.
2. Set a filter so that the import queue displays only the transport requests belonging to your project. Import all transport requests into the quality assurance system by performing a project import.

Solution 7: Imports Using TMS

Task 1:

Hint: Before proceeding, please check, if all transport requests are released from the development system (DEV). If not, release them.

Reviewing the import queues for all SAP systems

1. Are there any request waiting to be imported?

   a) Use transaction STMS to access the initial TMS menu. To see the import overview of all SAP systems in the transport domain from the TMS menu, choose *Overview* → *Imports*. On the *Import Overview* screen, the *Requests* column displays how many requests are ready to be imported into a specific SAP system. At this point, there are requests ready to be imported into the quality assurance system (QAS).

   Note: The number of requests ready to be imported for a particular SAP system may differ from the actual number of requests in the import queue. This number can be different, for example, if there are non-standard transport requests in the queue or if there are additional requests after an end mark.

2. Are there any requests to be imported into the production system?

   a) The only requests to be imported into the production system PRD are those requests that have successfully been imported into the quality assurance system (QAS) (in this setup: and are forwarded to the virtual VDE system).

3. Review the current status of the import queues of all SAP systems. Is the import queue of QAS open or closed?

   a) From the Import overview screen, the *Status* column displays the current state of each system's import queue. To see the meaning of the icons associated with the SAP systems, display the legend/key using *Extras* → *Legend* .

Continued on next page

Task 2:

Review the import queue of the QAS

1. Display the requests that are to be imported into the quality assurance system (QAS). Which SAP users released the request?

   a) Use transaction STMS to access the initial TMS menu. From the TMS menu, choose *Overview → Imports* to see the import overview of all SAP systems in the transport domain. Choose the QAS import queue. Choose the *Refresh* button. All requests to be imported into the QAS are listed. The *Owner* column gives information about the user who released the specific request.

2. Display the objects to be imported with a transport request that was released with your account.

   a) To display the object list of requests owned by your user, from the *Import Queues: System QAS* screen, either select the request number or highlight the request and choose *Request → Display → Object List*. The object list can be expanded to see all objects.

Continued on next page

Task 3:

Importing all transport requests of a project into the quality assurance system (QAS)

Hint: For importing transport requests, you should log on to the QAS system with your user. If you would start the imports from within the DEV system, you would be prompted for a logon, when you start the import.

Note: Use the option *Leave transport request in queue for later import* (tabstrip *Options* on the *Import Transport Request* popup), if it exists. The existence of this option depends on the tp parameter *CTC* (client dependent transport routes) and is release dependent!

1. Display more information about the transport requests now waiting in the import queue for QAS.

   a) To see more technical information for the requests in the queue of QAS, from the *Import Queues: System QAS screen*, choose *Edit → Display More*. This displays more columns, including *T* and *UMO*.

      The column *T* displays the type of transport request either **W** for customizing or **K** for a workbench requests.

      The column *UMO* indicates what unconditional modes are associated with the requests. All preliminary imports should have the unconditional mode **I**. This indicates the request can be reimported into the target system without producing an error.

2. Set a filter so that the import queue displays only the transport requests belonging to your project. Import all transport requests into the quality assurance system by performing a project import.

   a) Use transaction STMS to access the initial TMS menu. From the TMS menu, choose *Overview → Imports* to see the import overview of all SAP systems in the transport domain. Select the SAP system *QAS*. All requests that are waiting to be imported into QAS are listed.

      Mark the *Project* column and choose *Edit → Filters*. Enter your project name in the next screen. To perform a project import, choose *Queue → Start Import*. If possible, choose the option *Leave transport request in queue for later import* (tabstrip *Options*). Choose *Continue*.

      Before the import starts, you are required to provide authorization for the target system, if you are not already logged on to it. Provide this information for the quality assurance system, client *100* on *QAS* as *ADM325-##* as required, to enable the import to take place.

Lesson Summary

You should now be able to:

- Identify the tools used to perform imports using the Transport Management System (TMS)
- Explain the relationship between import queues and import buffers
- Follow the procedure for setting up the date and time for an import
- Decide whether to import single transport requests or to import the entire import queue
- List the different transport strategies

Lesson: QA Approval Procedure and Transport Proposals

Lesson Overview

In this lesson, you will learn the use of the Quality Assurance (QA) procedure and the QA history. You will also learn the use of the special transport workflow and its configuration procedure. In addition, you will see why creating transport proposals can be necessary.

Lesson Objectives

After completing this lesson, you will be able to:

- Define Quality Assurance (QA) procedure
- Describe a situation in which using special transport workflow would be practical
- Explain why the transport proposal is necessary

Business Example

Transport requests are to be imported only in to the productive system, after they are approved in the quality assurance system. Therefor you need a defined QA approval procedure.

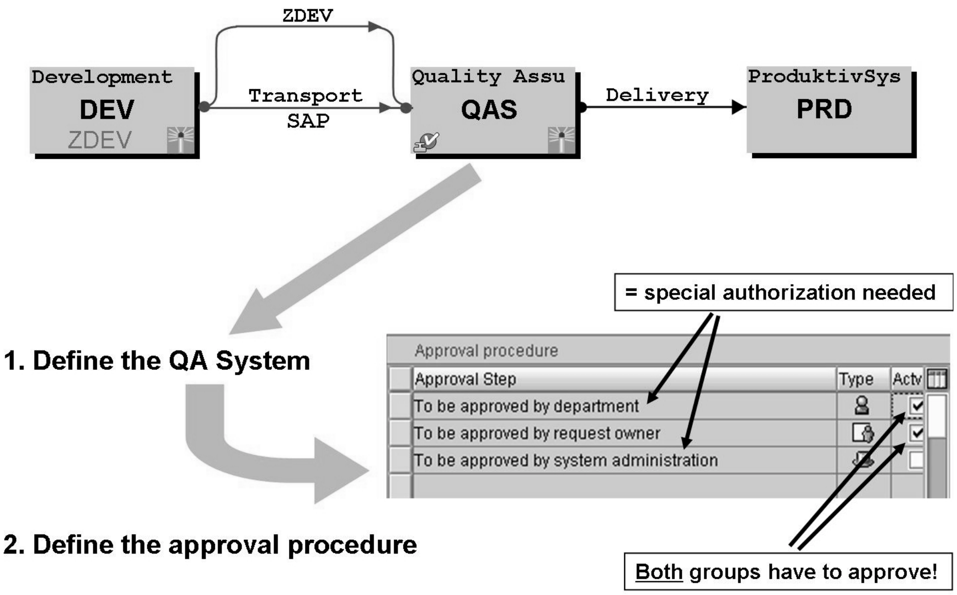
In some situations, it may be necessary to transport an urgent correction into the production system directly, bypassing the configured transport route. You can use the special transport workflow to perform this type of transport.

TMS Quality Assurance

The TMS Quality Assurance (QA) approval procedure increases the quality and the availability of the production systems by letting you check transport requests in the quality assurance system before delivering them to subsequent systems.

When you activate the QA approval procedure, transport requests are only forwarded to the delivery systems if all the QA approval steps are processed for the request in the QA system and the request has been approved. (When you configure the QA system, you determine how many QA approval steps have to be processed for each request.) If one check for an approval step is not successful, the entire request cannot be approved. You can only import completely approved requests into the delivery systems.

Rejected requests are not imported into the delivery systems of the QA system.



Approval procedure

Approval Step	Type	Actv
To be approved by department		☑
To be approved by request owner		☑
To be approved by system administration		□

Figure 100: Configuring the QA Approval Procedure

Before you can process transport requests you should configure the QA approval procedure. Ensure that the system landscape and/or transport domain is set up so that there is at least one development, one quality assurance, and one production system. The system to be configured as the QA system must have the following attributes:

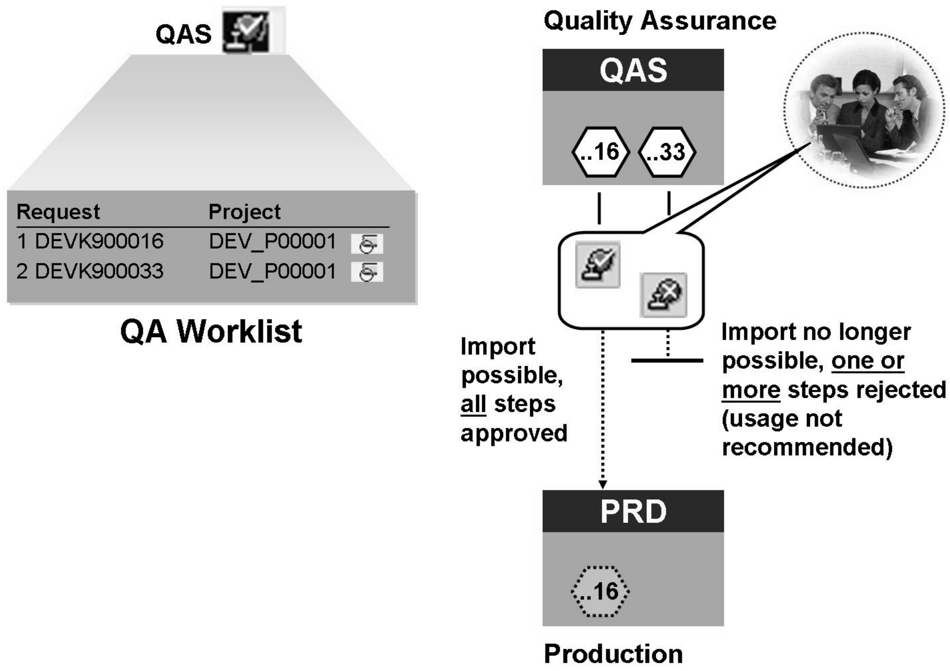
- It must be the target of at least one consolidation route (starting from a development system).
- It must deliver to at least one additional system.

In the system attributes for the chosen system, the option *delivery after confirmation* must be set. In the approval procedure, you can define which users must provide approval so that the transport can be imported into the delivery system.

After configuration, the QA worklist is automatically set up. All the requests imported into the QA system are included in the QA worklist.

Steps in QA Approval Procedure

To display the QA worklist, use transaction STMS and choose *Overview* → *Imports* → *Goto* → *QA worklist*. The date/time at the upper-right side of the screen indicates when the QA worklist was last updated; the upper-left side indicates how many requests still need to be processed. The list displays the transport requests corresponding to the selected approval steps. By default, the transport requests corresponding to all approval steps are shown. To select the approval step whose corresponding transport requests you wish to see, choose *Worklist* → *Select Approval Step*. By double-clicking various items in the table listing the transport requests, you obtain further information on those items.



Request	Project
1 DEVK900016	DEV_P00001
2 DEVK900033	DEV_P00001

QAS
QA Worklist
Quality Assurance
QAS
..16
..33
Import possible, all steps approved
Import no longer possible, one or more steps rejected (usage not recommended)
PRD
..16
Production

Figure 101: QA Approval

You should test the requests in a QA worklist before importing them in the delivery systems. The QA status *Rejected* means that one or more approval steps of a request were rejected by the person responsible. A request is approved only if all the approval steps have the status *Approved*.

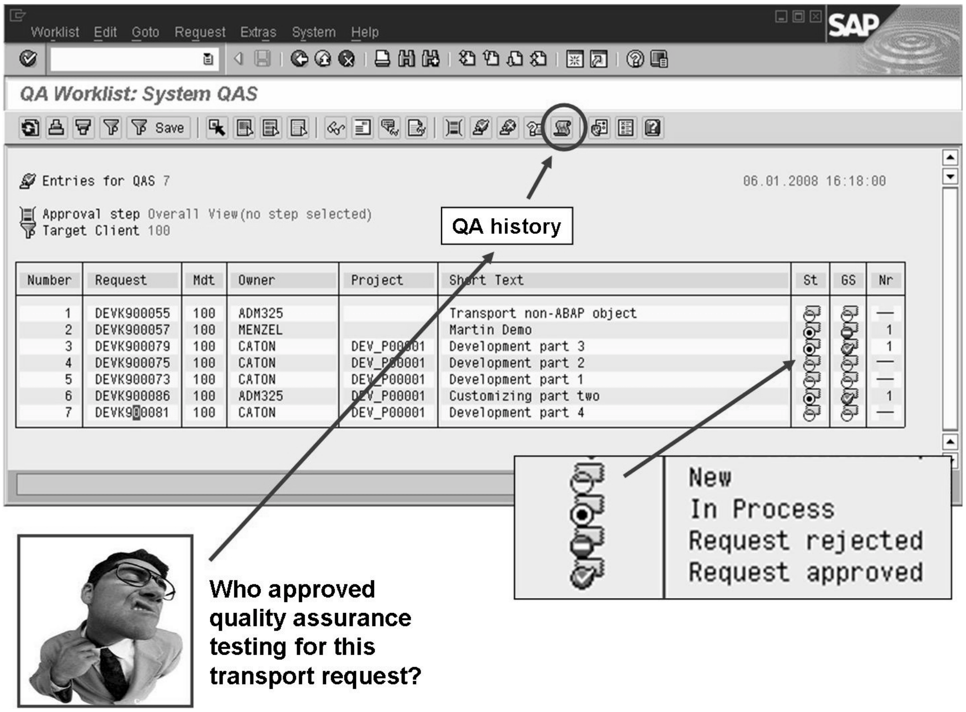
In the QA worklist, you can see:

- the QA status (St)
- the overall status (GS)
- the number of steps (Nr)

Requests can be imported into the delivery systems only if all the requests ready for import have received approval according to the various approval steps. If all the requests for one project have been approved, they can be imported into the delivery system if other projects still have unprocessed or rejected requests in the worklist. Requests with the QA status *Rejected* as well as unprocessed requests in the worklist are not imported into the delivery systems.

Starting with release 6.10 the approval or rejection of a single step can be changed, as long as not all approval steps were performed. As soon as all defined approval steps are performed, it is not possible to change the decision anymore.

SAP recommends **not** rejecting requests containing errors, but correcting the error using subsequent transport requests and approving the affected transport requests as an entire package.



QA Worklist: System QAS

Entries for QAS 7 06.01.2008 16:18:00

Approval step Overall View(no step selected)
Target Client 100

Number	Request	Mdt	Owner	Project	Short Text	St	GS	Nr
1	DEVK900055	100	ADM325		Transport non-ABAP object	○	○	—
2	DEVK900057	100	MENZEL		Martin Demo	◉	◒	1
3	DEVK900079	100	CATON	DEV_P00001	Development part 3	◉	✓	1
4	DEVK900075	100	CATON	DEV_P00001	Development part 2	○	○	—
5	DEVK900073	100	CATON	DEV_P00001	Development part 1	○	○	—
6	DEVK900086	100	ADM325	DEV_P00001	Customizing part two	◉	✓	1
7	DEVK900081	100	CATON	DEV_P00001	Development part 4	○	○	—

○	New
◉	In Process
◒	Request rejected
✓	Request approved

QA history

Who approved quality assurance testing for this transport request?

Figure 102: QA History

From the QA worklist screen, you can access the QA history by choosing *Goto → QA History*.

The QA history shows all the requests for a specific period that are no longer displayed in the QA worklist. Requests are no longer displayed in the QA worklist when they have been approved or deleted. The default period for the QA history is 30 days but this time period can be changed.

To determine who was responsible for approving a transport request, choose *Request → Display → QA status*.

Note: The QA history is stored in the database of the QA system (the QAS system). If you perform a database or system copy from PRD to QAS, the history is lost. To prevent the QA history from being deleted in this way, see SAP Note in the SAP Service Marketplace.

Special Transport Workflow

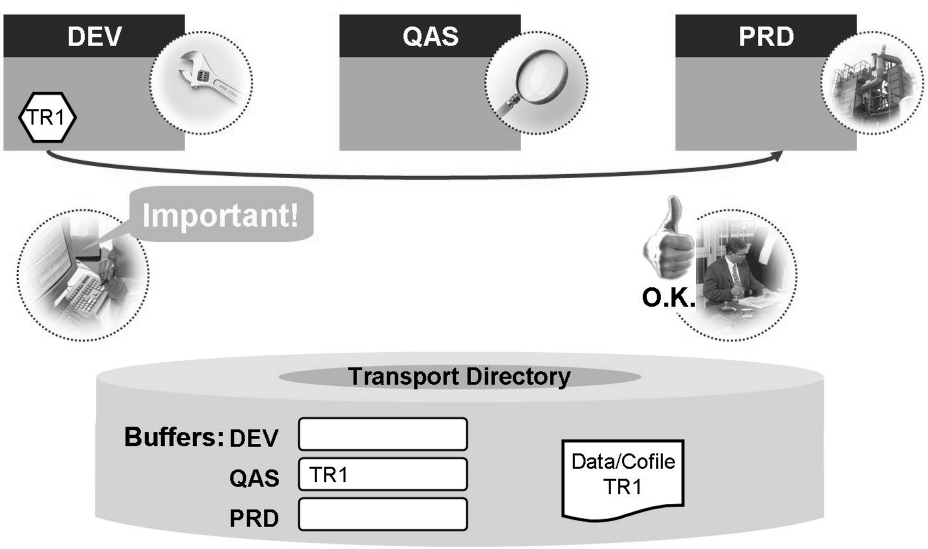


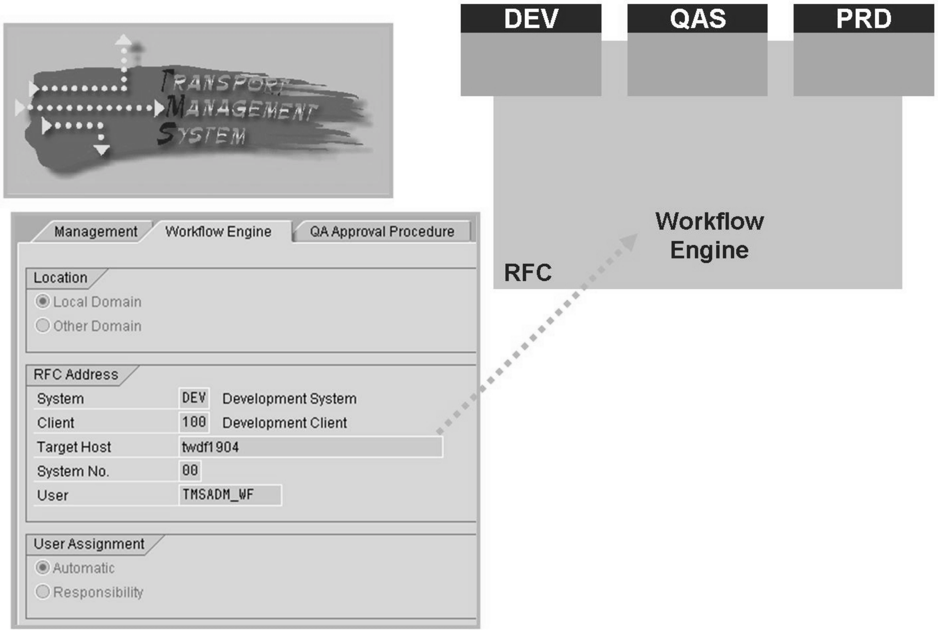
Figure 103: Special Transport Workflow

Use the **special transport workflow** if transport corrections are needed urgently, or if transports are required that do not follow the defined transport routes.

Before you can use the special transport workflow, you must configure one SAP system as the so-called Workflow Engine. This system must have the following criteria listed in order of importance:

- High availability
- Low to medium system load

These prerequisites are, for example, met by the Quality Assurance System (QAS). All the tasks performed in the special transport workflow are processed centrally on the Workflow Engine and the results communicated to the affected SAP systems.



Location

● Local Domain
○ Other Domain

RFC Address

System	DEV	Development System
Client	100	Development Client
Target Host	twdf1904	
System No.	00	
User	TMSADM_WF	

User Assignment

● Automatic
○ Responsibility

Figure 104: Configuring the Special Transport Workflow

To configure the special transport workflow, do the following:

Log on to the SAP system functioning as the transport domain controller.

Start transaction STMS and then choose *Overview* → *Systems* → *Goto* → *Transport Domain*.

Choose the *Workflow Engine* tab.

Switch to change mode by pressing the*Display* <-> *Change* button.

Enter the SAP system, the client, and the target host for your Workflow Engine. Choose *Save*. Respond affirmatively to the popup to distribute the configuration change. In the next screen, log on to the system that is to be configured as the Workflow Engine. The system automatically:

- Creates the user TMSADM_WF
- Generates the RFC destinations required for the Workflow Engine
- Sends the address data of the Workflow Engine to all systems in the transport domain
- Performs the workflow-related customizing on the Workflow Engine

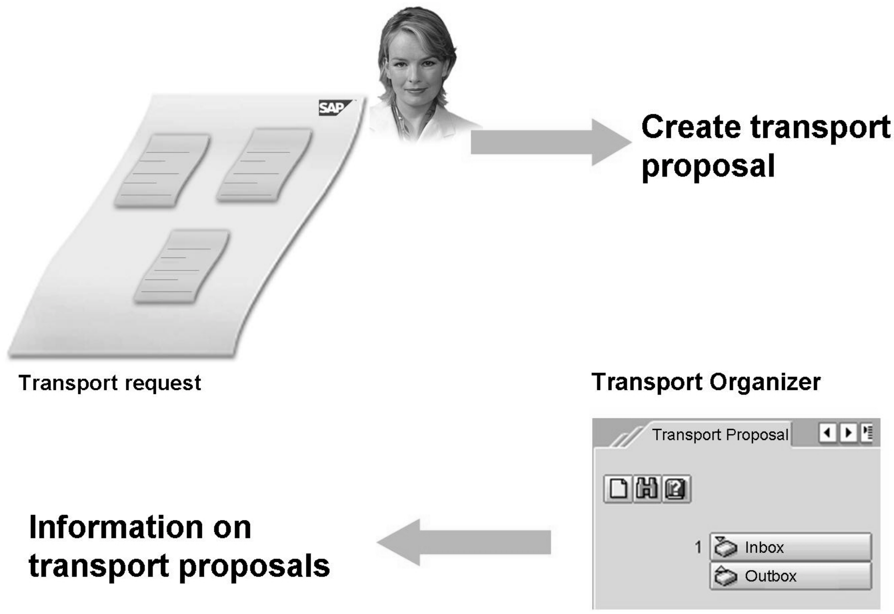


Figure 105: Creating Transport Proposals

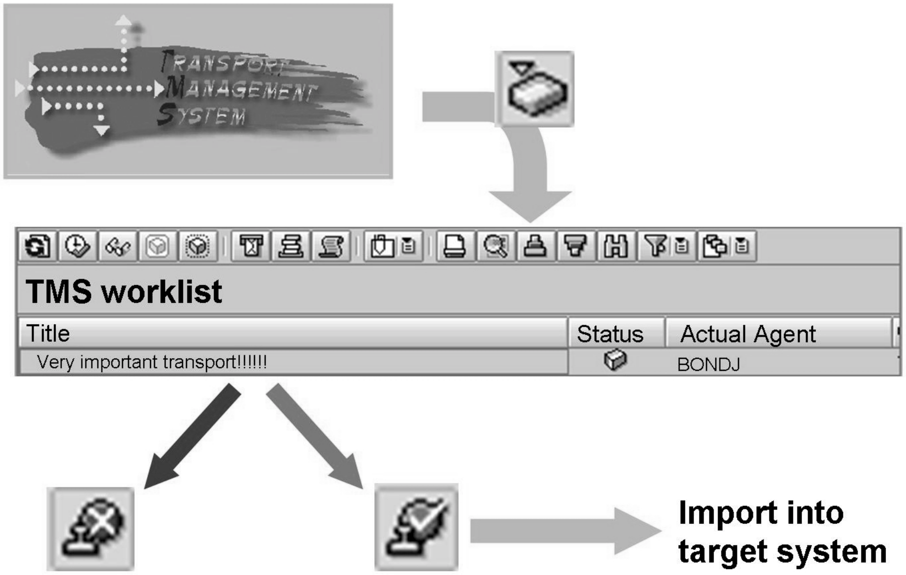
To use the special transport workflow, you must create a transport proposal. To do this, go to the Transport Organizer (Transaction SE09) and select released requests. Choose *Display*. Position the cursor on a request that you want to transport and choose *Utilities(M) → Create Transport Proposal*. The *Create Transport Proposal* popup appears. Enter a short text, the target system, and possibly other transport requests. The target systems apply to all requests that you enter.

Note that any requests that you place in the transport proposal must have been previously released.

If a transport proposal is created, the SAP system gives this transport proposal a proposal number, and then places it in the specific TMS work list for the transport administrator, as mentioned on the following pages.

If the transport administrator rejects your transport proposal, it reappears in your transport proposal inbox. You can cancel the transport proposal or revise it and send it back to the transport administrator.

After the transport administrator has approved your transport proposal, the import is started and the transport proposal reappears in your transport proposal inbox. Check that the transport requests are being imported correctly into the target systems, and confirm the transport proposal.



TMS worklist

Title	Status	Actual Agent
Very important transport!!!!!!		BONDJ

Figure 106: Transport Proposal Worklist

To approve or reject a transport proposal, start transaction STMS. To display the TMS work list, choose *Overview* → *Worklist*. Double-click the transport proposal you want to process. Check if the requests, the list of target systems, and the import times and import options for the transport proposal are correct. You can display the transport requests by choosing *Display* and the icon for the transport logs. Switch to change mode if you want to make changes to the transport requests, transport targets, import times, or import options.

You can create a message for a developer by choosing the *Manage Attachments* icon.

To process the transport proposal, choose the respective icon to approve or reject the transport proposal.

If you approved the transport proposal, the import into the specified systems is automatically started. The transport proposal is placed in the developer's transport proposal inbox and the developer confirms it.

If the transport proposal was rejected, it is placed in the developer's transport proposal inbox for modification.

Exercise 8: QA Approval Procedure and Transport Proposals

Exercise Objectives

After completing this exercise, you will be able to:

- Work with the Quality Assurance (QA) worklist and QA history
- Perform the Quality Assurance (QA) procedure
- Make use of the special transport workflow

Business Example

It is very important to import transport requests into the productive system only if they are tested in the quality assurance system. For the system administrator it is not easy to make sure that the objects from a transport request were tested or not. By using the quality assurance procedure, it is not only obvious if the transport request was tested or not. You can also see who took responsibility about it.

The transport workflow can be used to accommodate transport. It can be used in situations where it may be necessary to transport an urgent correction directly into the production system bypassing the configured transport route. Or for importing into the quality assurance system, although regular imports are not scheduled at the moment.

Task 1:

Perform quality assurance testing after successful imports into the quality assurance system.

1. After a successful import of transport requests, the functionality of the system must be extensively tested. How is this performed?
2. Verify that the customizing entries you transported exist in the quality assurance client.
3. Verify that the development objects you transported exist in the quality assurance system.

Continued on next page

Task 2:

Performing the QAS procedure that permits the import of transport request into the production system.

1. After transport requests imported into QAS, the requests are placed into the import queue of the subsequent system. Display the status of your transport requests in the import queue of the subsequent system. Is an import of this transport requests into the subsequent system now possible?

   Note: The subsequent system could be the productive system PRD. In this setup the subsequent system is VDE.

2. Perform the quality assurance procedure. Look at the QA worklist of the quality assurance system. Use the filter to display only the transport requests for your project. Approve all transport requests of your project.
3. Display the import queue of the subsequent system.

Task 3:

Optional: transport management: creating transport proposals for an important transport request.

1. You, in your role as a developer: Create a transport proposal. Specify the transport request that you created for the modification exercise. The target system should be QAS.

   Note: Prerequisites: the special workflow has been configured by the instructor.

2. You, in your role as a transport administrator: To check all transport proposals, call the worklist from the initial TMS screen. Mark your transport proposal and reject it.
3. Again, as a developer: Monitor the status of your transport proposal from the transport proposal inbox of the Transport Organizer. Find the rejected transport proposal and accept the rejection of the transport proposal.

   Note: As an alternative, you can send the transport proposal again, instead of accepting the rejection. Then you could - in your role as a transport administrator - approve this second try.

Solution 8: QA Approval Procedure and Transport Proposals

Task 1:

Perform quality assurance testing after successful imports into the quality assurance system.

1. After a successful import of transport requests, the functionality of the system must be extensively tested. How is this performed?

   a) Quality assurance testing requires that the desired functions are tested using a sample of “real-live” business data and existing functions are still working even after the import of new changes and additional functions.

      To perform quality assurance testing, there should be a dedicated quality assurance system with a well-defined set of test data. Using test scripts that may or may not be automated with a tool such as eCATT, business functions must be tested. The results must be compared with desired results. All inconsistencies must be noted and forwarded back to the customizers and developers to correct in the development system.

2. Verify that the customizing entries you transported exist in the quality assurance client.

   a) Log on to QAS in client *100* as *ADM325-##*.

      To verify customizing, use the SAP Reference IMG to see that some of your customizing objects exist in the client *100* on QAS. Just looking for the existence of changes is usually not enough. You will have to test the flow of processes as well, if working in a real-life scenario.

3. Verify that the development objects you transported exist in the quality assurance system.

   a) Log on to QAS in the quality assurance client *100* as *ADM325-##*. The most recently imported development changes can be tested by verifying that the objects such as the ABAP program *ZPGM_##* and package *ZPACKAGE_##* exist in QAS. Again just looking for the existence of objects is not true quality assurance testing. The object's function needs to be verified.

Continued on next page

Task 2:

Performing the QAS procedure that permits the import of transport request into the production system.

1. After transport requests imported into QAS, the requests are placed into the import queue of the subsequent system. Display the status of your transport requests in the import queue of the subsequent system. Is an import of this transport requests into the subsequent system now possible?

   Note: The subsequent system could be the productive system PRD. In this setup the subsequent system is VDE.

   a) Use Transaction STMS to access the initial TMS menu. From the TMS menu, choose *Overview → Imports* to see the *Import Overview* of all SAP systems in the transport domain. Select the subsequent system. The *Import Queues: System <SID>* screen appears. In the *Status* column, you can see that the transport requests imported into the QAS system have the status *Request must still be approved*. These requests cannot be imported into the subsequent system until they are confirmed as checked in the QA approval procedure.

   If a QA approval procedure is not defined, transport requests that are imported into the quality assurance system may technically directly be imported to any delivery system of this quality assurance system.

Continued on next page

2. Perform the quality assurance procedure. Look at the QA worklist of the quality assurance system. Use the filter to display only the transport requests for your project. Approve all transport requests of your project.

   a) Use transaction STMS to access the initial TMS menu. From the TMS menu, choose *Overview → Imports* to see the import overview of all SAP systems in the transport domain. Select the quality assurance system *QAS* and choose *Goto → QA Worklist*. The *QA Worklist: System QAS* screen appears.

      Imagine, you are here in the role of the transport request owner: select the corresponding approval step by choosing *Worklist → Select Approval Step*. Select *To be approved by request owner*. Then select the *Project* column and choose *Edit → Filters*. Enter your project name in the resulting screen. Mark your transport requests after they have been tested for your specified project and choose *Request → Approve*.

      Now imagine, you are here in the role of the department: select the corresponding approval step by choosing *Worklist → Select Approval Step*. Select *To be approved by department*. Then select the *Project* column and choose *Edit → Filters*. Enter your project name in the resulting screen. Mark your transport requests after they have been tested for your specified project and choose *Request → Approve*. Confirm the popup informing you, that after this approval step the decision can not be changed anymore.

3. Display the import queue of the subsequent system.

   a)

      Note: The status of your transport requests have changed.

      From the import overview of all SAP systems in TMS, select the subsequent system. The import queue appears. In the *Status* column, you can see that the transport requests that have been approved in the QAS system have the status *Request waiting to be imported*.

Continued on next page

Task 3:

Optional: transport management: creating transport proposals for an important transport request.

1. You, in your role as a developer: Create a transport proposal. Specify the transport request that you created for the modification exercise. The target system should be QAS.

   Note: Prerequisites: the special workflow has been configured by the instructor.

   a) Create a transport proposal in your inbox.

      1. Use the right side of the *Transport Organizer* (transaction SE09) to create a transport proposal. Select the *Transport Proposals* tab. Some new icons appear. Choose *Create transport proposal*. In the subsequent screen, enter a short text for your transport proposal. You may also add additional attachments for longer explanations.
      2. In the *Request for import* field, enter the transport request you created for the modification exercise. The target system should be *QAS* and the target client should be *100*. In the *Time* field, you can specify a given date and time for the import to be performed, otherwise, it will be done as soon as your transport proposal is accepted. In the *Options* field, you may add some unconditional modes.
      3. Choose *Create and Send Proposal* to save and send your transport proposal.

2. You, in your role as a transport administrator: To check all transport proposals, call the worklist from the initial TMS screen. Mark your transport proposal and reject it.

   a) Use transaction STMS to access the initial TMS screen. Choose *Overview* → *Worklist* to display the TMS worklist. A log on is performed on the workflow server.

      Mark your transport proposal and select *Execute*. Choose *Reject* to reject the request.

Continued on next page

3. Again, as a developer: Monitor the status of your transport proposal from the transport proposal inbox of the Transport Organizer. Find the rejected transport proposal and accept the rejection of the transport proposal.

   Note: As an alternative, you can send the transport proposal again, instead of accepting the rejection. Then you could - in your role as a transport administrator - approve this second try.

   a) From the initial screen in transaction SE09, in the tab *Transport Proposals* choose *Inbox* to display information on your transport proposals. You will see that your transport proposal is rejected.

      Double click your transport proposal. From this screen, you can either *Withdraw* or *Save and Send Back* your transport proposal.

Lesson Summary

You should now be able to:

- Define Quality Assurance (QA) procedure
- Describe a situation in which using special transport workflow would be practical
- Explain why the transport proposal is necessary

Lesson: Import Process

Lesson Overview

In this lesson you will learn about the technical steps involved in the import process.

Lesson Objectives

After completing this lesson, you will be able to:

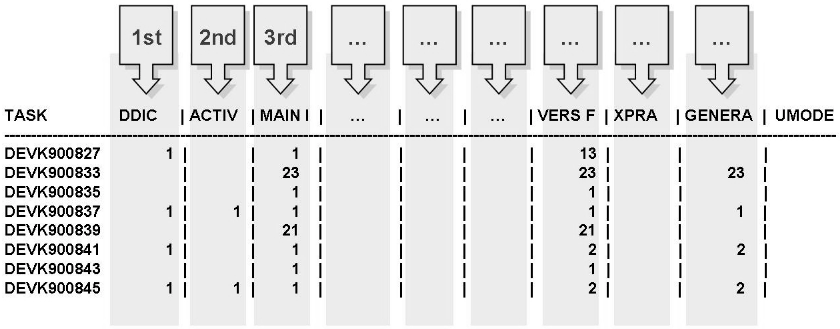
- Identify the steps of the import process
- Identify the relationship between tp, R3trans and RDDIMPDP

Business Example

To be able to analyze and resolve problems that may arise during imports, you must understand the steps in the import process.

Steps in Import Process

- **tp collectively processes each import step for all requests before proceeding with the next import step**
- **tp does NOT process all import steps for only a single request before proceeding to the the next request**



```
TASK         DDIC  | ACTIV  | MAIN I |   ...   |  ...  |  ...  | VERS F | XPRA   | GENERA | UMODE
--------------------------------------------------------------------------------------------------
DEVK900827      1  |        |    1   |         |       |       |    13  |        |        |
DEVK900833         |        |   23   |         |       |       |    23  |        |    23  |
DEVK900835         |        |    1   |         |       |       |     1  |        |        |
DEVK900837      1  |     1  |    1   |         |       |       |     1  |        |     1  |
DEVK900839         |        |   21   |         |       |       |    21  |        |        |
DEVK900841      1  |        |    1   |         |       |       |     2  |        |     2  |
DEVK900843         |        |    1   |         |       |       |     1  |        |        |
DEVK900845      1  |     1  |    1   |         |       |       |     2  |        |     2  |
```

Figure 107: tp Processing Sequence

tp processes each import steps collectively for all requests in the import queue. The required steps for each request are listed in the import buffer file. The contents of the buffer file are organized as a table in which each column represents an import step. The numbers in the columns indicate either if the import step is necessary, or the number of objects in the request that require the step.

As you see in the figure above, the first import step *DDIC* is processed for requests where it is required (DEVK900069 and DEVK900092); then the *ACTIV* step (for DEVK900069 and DEVK900092); then, the *MAIN I* step, for all requests; and so on.

The *tp import all* command does **not** process all steps for one request before moving on to the steps for the next request. For example, if you detect an error in an object that has already been released, you correct the object and release it again. 'Import all' then imports the whole buffer in the correct sequence and the faulty object is overwritten. Because the generation step is the last step and is performed only once for all objects in the requests, the faulty object does not affect your production system. Only the correct version of the report is activated and generated.

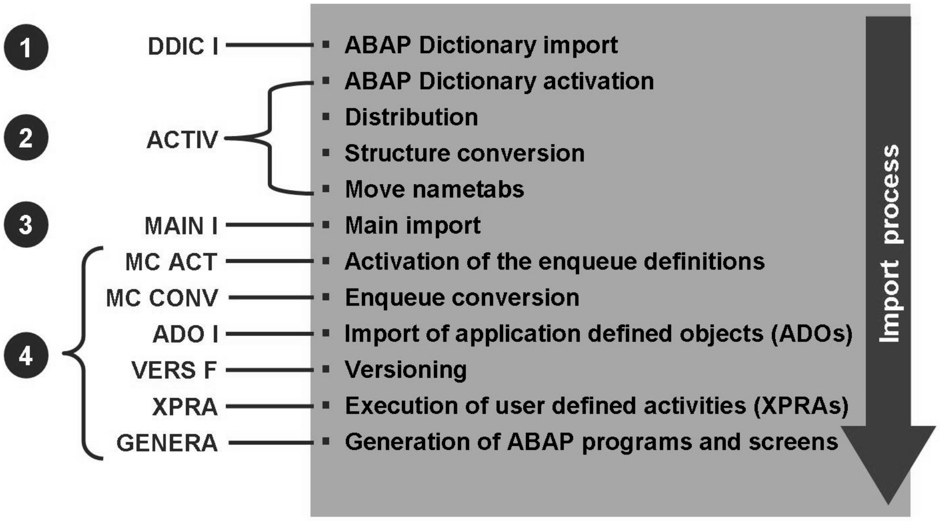

1
DDIC I
ABAP Dictionary import
2
ACTIV
ABAP Dictionary activation
Distribution
Structure conversion
Move nametabs
3
MAIN I
Main import
4
MC ACT
Activation of the enqueue definitions
MC CONV
Enqueue conversion
ADO I
Import of application defined objects (ADOs)
VERS F
Versioning
XPRA
Execution of user defined activities (XPRAs)
GENERA
Generation of ABAP programs and screens
Import process

Figure 108: Steps during Import

There are four major steps when importing transport requests:

1. Import dictionary objects
2. “Threatening” dictionary objects
3. Import “everything else”
4. “Threat” everything else

These steps in the import process are in detail:

DDIC I: ABAP Dictionary import

- In an import with R3trans, all ABAP Dictionary structural data is imported inactively, thus enabling you to import into an active SAP system.
- Log file: *<SourceSID>H9<number>.<targetSID>*
- transport tool: R3trans

ACTIV: ABAP Dictionary activation

- Runtime descriptions (nametabs) are written inactively.
- Log file: *<SourceSID>A9<number>.<targetSID>*
- transport tool: RDDMASGL

ACTIV: ABAP Dictionary distribution

- After activation and running logical checks for the new dictionary structures, the distribution program decides what actions are required to bring the new objects into the running system.
- Log file: *<SourceSID>S9<number>.<targetSID>*
- transport tool: RDDGENBB

ACTIV: Structure conversion

- If necessary, changes are made to table structures.
- Log file: *<SourceSID>N9<number>.<targetSID>*
- transport tool: RDDGENBB

ACTIV: Move nametabs

- The new ABAP runtime objects are put into the active runtime environment, database structures are adjusted if necessary.
- Log file: *<SourceSID>69<number>.<targetSID>*
- transport tool: pgmvntab

MAIN I: Main import

- All data is imported.
- Log file: *<SourceSID>I9<number>.<targetSID>*
- transport tool: R3trans

MCACT: Activation and conversion of enqueue objects

- Enqueue objects such as search helps that were not previously activated are activated in this separate step after the main import. They are used immediately in the running system.
- Logfile: *<SourceSID>M9<number>.<targetSID>*
- transport tool: RDDGENBB

ADO I: application-defined objects import

- Import of application-defined objects (*ADOs*): These objects include forms, styles, and printer definitions.
- Log file: *<SourceSID>D9<number>.<targetSID>*
- transport tool: RDDIC1L

VERSF: Versioning

- Versions of repository objects are only created if transport parameter vers_at_import is active. However, the import process does modify the object's version counter, which is incremented for all repository objects during import.
- Log file: *<SourceSID>V9<number>.<targetSID>*
- transport tool: RDDVERSL

XPRA: Execution of user-defined activities

- XPRAs are objects that can be used to start reports in the target system. The XPRA object has the same name as the report.
- Log file: *<SourceSID>R9<number>.<targetSID>*
- transport tool: RDDEXECL

GENERA: Generation of ABAP programs and screens

- During this step, you can resume normal business activities in the system.
- Log file: *<SourceSID>G9<number>.<targetSID>*
- transport tool: RDDIC03L

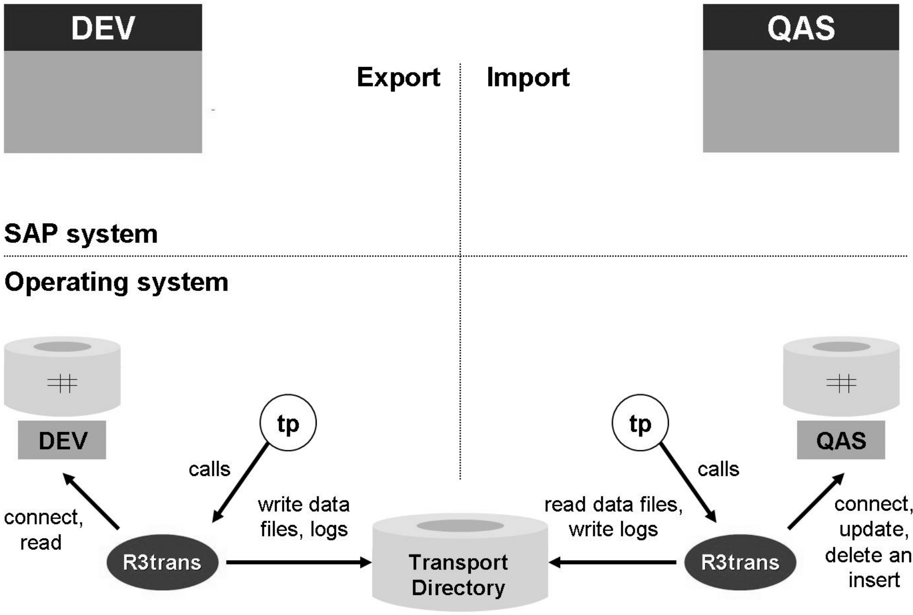


Figure 109: Introducing tp and R3trans

R3trans is a transport tool at the operating system level used to transport data between SAP systems. R3trans is usually called by other programs such as tp, the upgrade control program.

For transports between SAP systems, to access the database, tp indirectly calls R3trans – by causing UNIX to issue a fork(), Windows to issue a CreateProcess(), and AS/400 to issue a spawn(). During export, R3trans stores the object data extracted from the database in data files in the transport subdirectory data. The format of these data files, R3trans format, is independent of the platform. During import, R3trans reuses these data files.

Direct use of R3trans is not supported but may be required in exceptional cases. In case of transports, R3trans should always be used through tp. Import steps differ for the different object types. Further activities may be required in addition to R3trans activities. tp ensures that all export and import steps, including R3trans activities, are completed successfully.

Upward compatibility: R3trans writes data using a standard transport format. Thus you can export data with an old R3trans version and import data with a new version of R3trans. You can also transport between different databases or operating systems.

Note: Although exports and imports are independent of the R3trans version, the database platform, or the operating system, SAP does not support using tp or R3trans for transports between different SAP system releases.

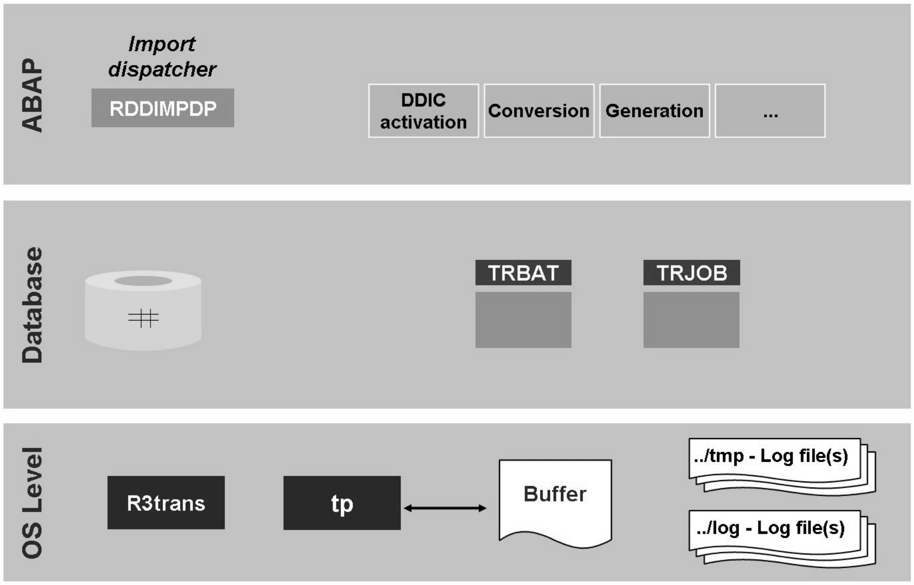


Figure 110: Import Process: tp and the Import Buffer

The first step of an import process is the tp call, triggered by starting an import through TMS or by a *tp import* command at operating system level.

During the whole import process, tp reads the buffer file that includes all the necessary steps for the specific request.

If you trigger an *tp import all*, tp has to ensure that at the beginning of the import process all transport requests stored at this time in the buffer are imported. This is done by executing command *tp setstopmark* each time a *tp import all* process is started. After the steps of the import process are completed, the command *tp delstopmark* is performed automatically, and a *tp cleanbuffer* deletes the transport requests from the import buffer.

After all involved tools have finished their work, tp exits to the operating system level and writes a return code to the appropriate log file for the activity. For example, *tp import* and *tp setstopmark* commands are recorded in the *ULOG* file.

Note: The command **tp import** is re-entrant. If an error occurs during import, after you eliminate the error condition and restart tp, tp finds the correct point to restart.

By default, tp aborts if one import phase receives a return code larger than 8. The transport profile parameter *stoponerror* defines what return code value should cause tp to abort.

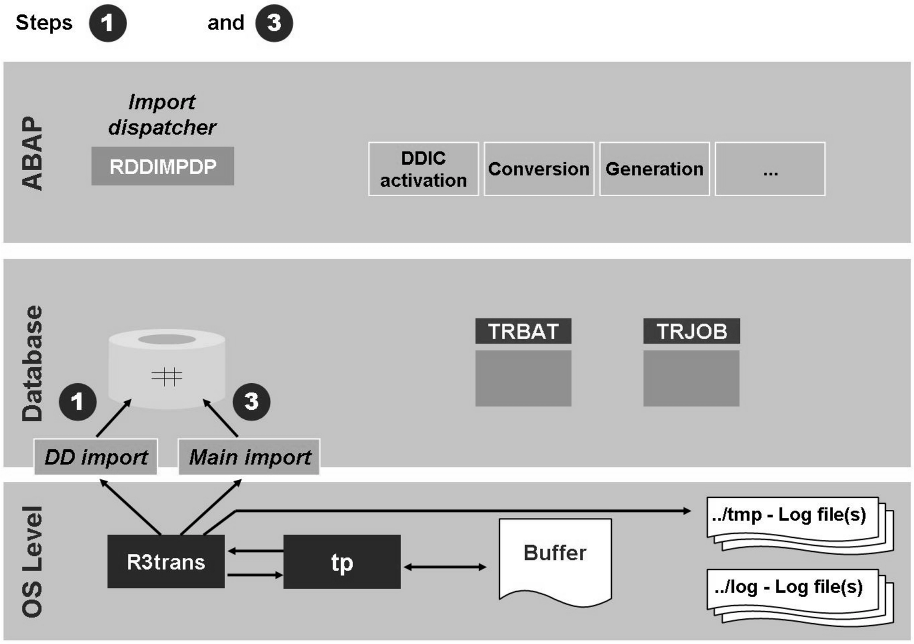


Figure 111: Import Process: R3trans

tp reads the buffer file that includes all the necessary steps for the specific request and calls R3trans at the operating system level.

For each import step, tp passes information from the buffer file to R3trans. R3trans reads the corresponding data files in the transport subdirectory data and connects directly to the database to perform inserts or updates to the included objects. After R3trans finishes performing inserts or updates, it passes an exit code to tp.

For each transport action, R3trans writes a log file in transport subdirectory tmp. After R3trans completes its work, tp moves these log files to the transport subdirectory log.

During the import process, R3trans is executed by the import steps ABAP dictionary import (for the import of ABAP dictionary definitions), and main import (for the import of table contents).

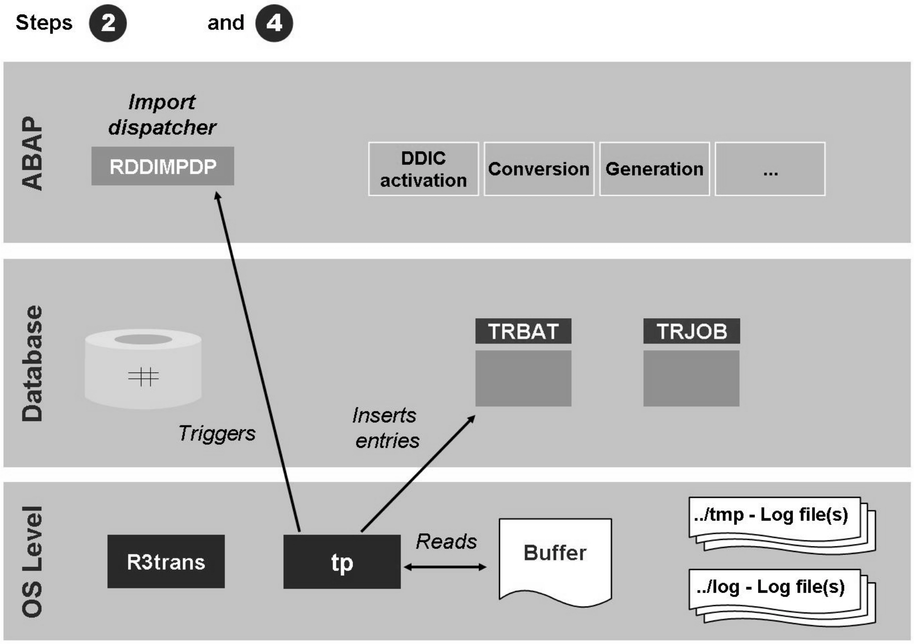


Figure 112: Import Process: tp / ABAP Communication (1)

In addition to the steps performed by R3trans, tp triggers the so-called import dispatcher. The import dispatcher is a background job with the name *RDDIMPDP* in the SAP system, which performs steps in the import process.

tp and *RDDIMPDP* communicate via table *TRBAT*. For every transport request, tp writes an entry in table *TRBAT*. The import function currently being performed for the request is represented by a character.

In the example below, there are three transport requests waiting for DDIC activation in the table *TRBAT*.

Request	Function	Return Code	Timestamp
DEVK904711	J	9999	00000001
DEVK904712	J	9999	00000002
DEVK904713	J	9999	00000003
HEADER	J	B	20083001143701

tp inserts a header entry to tell *RDDIMPDP* to start processing. Some activities that are independent of transport requests, such as distribution and structure conversion, only have a header entry in table *TRBAT*. Return code *9999* indicates that the step is waiting to be performed. For the header entry, tp inserts a *B* (for begin) as return code.

To trigger the transport daemon *RDDIMPDP* in the SAP System, tp uses the operating system level tool *sapevt*.

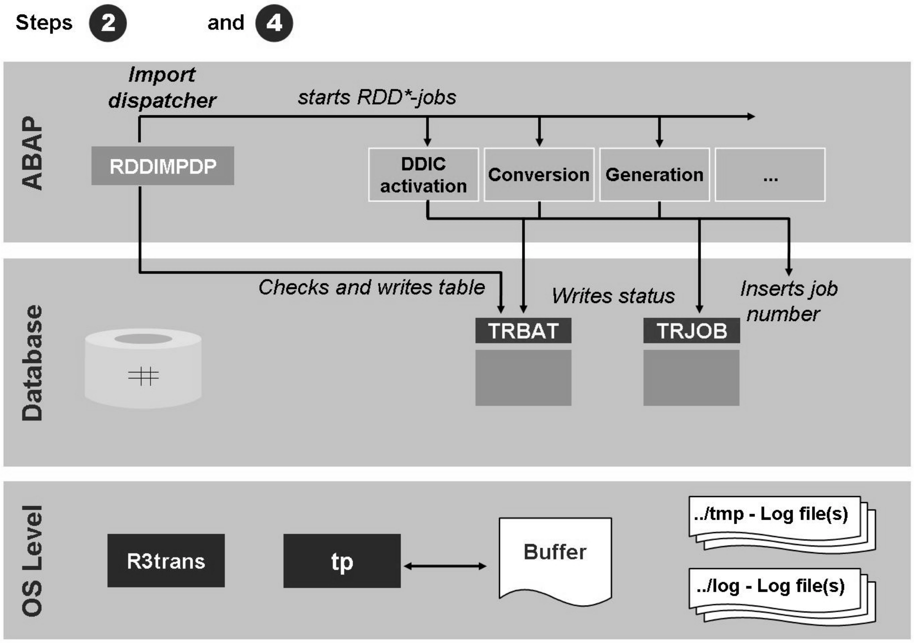


Figure 113: Import Process: tp / ABAP Communication (2)

When *RDDIMPDP* is started, it checks the table *TRBAT* to find out if there is an action to be performed for the transport requests such as mass activation, distribution, or table conversion. It sets the header entry to *R* (for run), and starts the appropriate *RDD** program as a background task, reschedules itself, and exits. The activated program (in this example, the mass activator) sets the status of the first entry in *TRBAT* to active (return code *8888*):

Request Function Return Code Timestamp

Request	Function	Return Code	Timestamp
DEVK904711	J	8888	00000001
DEVK904712	J	9999	00000002
DEVK904713	J	9999	00000003
HEADER	J	R	20083001143903

Each of the required background tasks receives a job number generated by background processing. This job number and the step ID are inserted into table *TRJOB* by the *RDD** jobs.

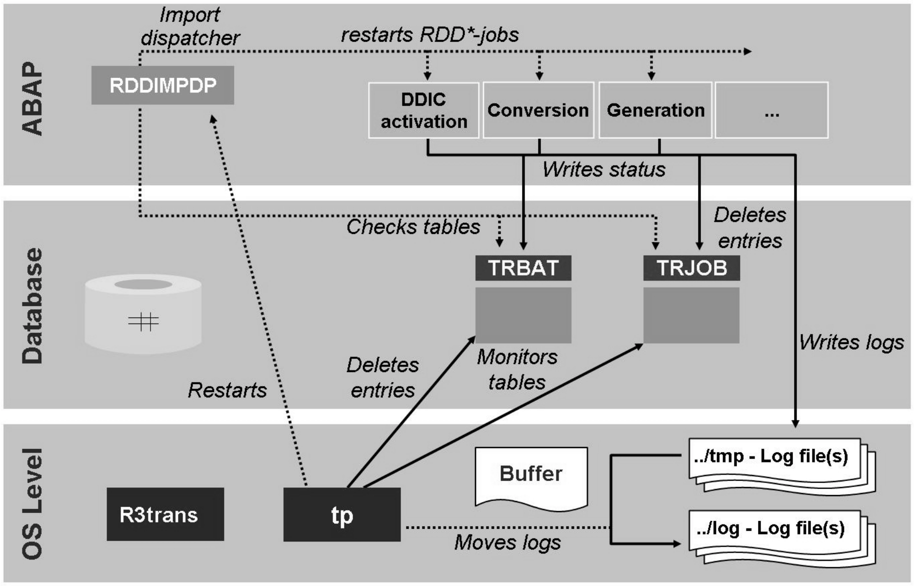


Figure 114: Import Process: tp /ABAP Communication (3)

The background tasks write their return codes in table *TRBAT* and delete the corresponding entry in table *TRJOB*. Return codes of *12* or less indicate that the step is finished. In *TRBAT*, the column *TIMESTAMP* contains the finishing time. When all the necessary actions are performed for all transport requests, the header entry is set to *F* (for finished) by the *RDD** jobs.

Request	Function	Return Code	Timestamp
DEVK904711	J	4	00000001
DEVK904712	J	0	00000002
DEVK904713	J	0	00000003
HEADER	J	F	20083001144512

All the background jobs log the performed steps either in the database or in transport subdirectory *tmp*.

tp monitors the entries in table *TRBAT* and table *TRJOB*. When the header entry in table *TRBAT* is set to *F* and table *TRJOB* is empty, tp copies the logs of the completed steps from directory tmp to log and deletes the corresponding entries in table *TRBAT*.

If problems are detected by tp when monitoring table *TRBAT* and table*TRJOB*, tp re-triggers *RDDIMPDP* through sapevt. *RDDIMPDP* automatically recognizes if a previous step is still active or was aborted by checking tables *TRJOB* and *TRBAT*. If a step was aborted, *RDDIMPDP* restarts this step.

tp uses the tables *TRBAT* and *TRJOB* to communicate with the various ABAP programs. These ABAP programs run as background processes, there must be at least two background work processes running in the SAP System.

tp and the background jobs needed for transports communicate using table *TRBAT*, which contains temporary data. tp inserts information into *TRBAT* specifying the steps to be performed for a specific request. tp then triggers *RDDIMPDP* to read *TRBAT* and execute the required programs as background jobs whose names start with *RDD*. For example, *RDDMASGL* is for mass activation, *RDDGENDB* for conversion, and *RDDVERSL* for versioning. Each *RDD** job receives a job number that is recorded in table *TRJOB*. The jobs report their current status back to *TRJOB*.

When imports are performed, tp triggers the import dispatcher *RDDIMPDP* by sending the event *SAP_TRIGGER_RDDIMPDP* with the help of the tool sapevt. In client *000* and each client that is to receive imports, user DDIC must schedule the job *RDDIMDP* with event-based scheduling.

Hint: *RDDIMPDP* can be scheduled by running the ABAP program *RDDNEWPP*.

Normally, *RDDIMPDP* is automatically scheduled after a client copy. To verify if the required import dispatchers are scheduled, check the job overview using transaction SM37. The correct name of the import dispatcher for each client is *RDDIMPDP_CLIENT_<###>*, where ### represents the client's name. The corresponding event is *SAP_TRIGGER_RDDIMPDP_CLIENT*.

Transport Commands

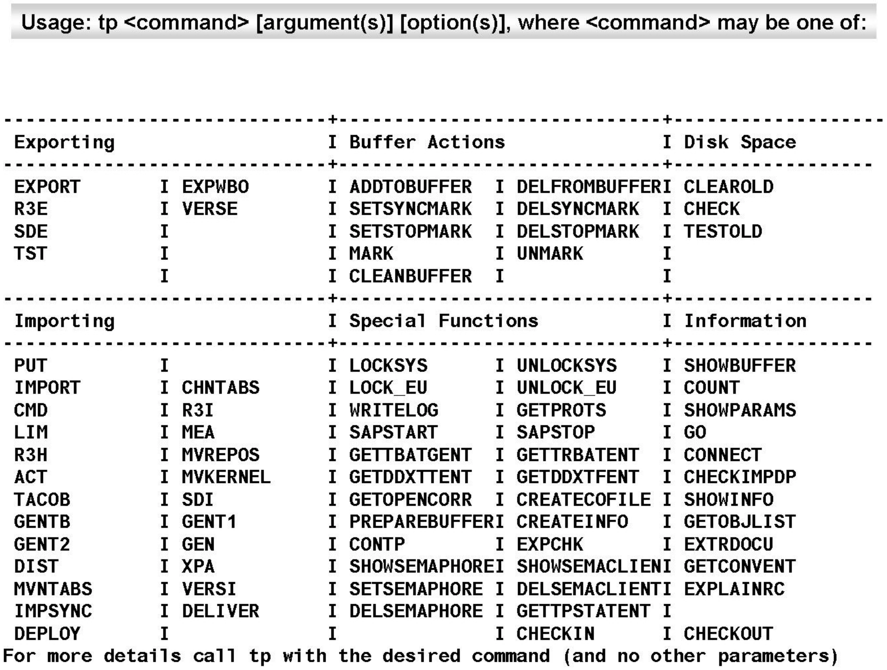
The transport control program **tp** is used for controlling transports between SAP systems and for performing release upgrades. The tp program tracks transports by controlling exports and imports of objects between SAP systems. The program also ensures that the steps in exporting and importing objects are performed in the correct order and that imports into a target system are done in the same order as the exports from the source systems.

tp is an operating system level program, which uses special programs, such as C programs, operating system commands, and ABAP programs in the SAP system.

tp performs exports and imports separately. In the export phase, the objects to be transported are extracted from the database of the source system and stored in files in the transport directory. In the import phase, the objects are added to the database of the target system. An export always takes place immediately after the transport request has been released so that the objects are freed for further modification. tp uses the program **R3trans** at operating system level to establish a connection to the database and to perform the necessary steps in the database.

There is no automatic mechanism that imports a request into a target system immediately after export.

Although most transport activities can be performed using TMS, you may find the need to use tp commands for exceptional cases. Because import buffers and import queues are identical, use of tp and TMS produces consistent results.



Usage: tp <command> [argument(s)] [option(s)], where <command> may be one of:

```
----------------------------+---------------------------+------------------
 Exporting                  I Buffer Actions            I Disk Space
----------------------------+---------------------------+-----------------
 EXPORT       I EXPWBO      I ADDTOBUFFER  I DELFROMBUFFERI CLEAROLD
 R3E          I VERSE       I SETSYNCMARK  I DELSYNCMARK  I CHECK
 SDE          I             I SETSTOPMARK  I DELSTOPMARK  I TESTOLD
 TST          I             I MARK         I UNMARK       I
              I             I CLEANBUFFER  I              I
----------------------------+---------------------------+-----------------
 Importing                  I Special Functions         I Information
----------------------------+---------------------------+-----------------
 PUT          I             I LOCKSYS      I UNLOCKSYS    I SHOWBUFFER
 IMPORT       I CHNTABS     I LOCK_EU      I UNLOCK_EU    I COUNT
 CMD          I R3I         I WRITELOG     I GETPROTS     I SHOWPARAMS
 LIM          I MEA         I SAPSTART     I SAPSTOP      I GO
 R3H          I MVREPOS     I GETTBATGENT  I GETTRBATENT  I CONNECT
 ACT          I MVKERNEL    I GETDDXTTENT  I GETDDXTFENT  I CHECKIMPDP
 TACOB        I SDI         I GETOPENCORR  I CREATECOFILE I SHOWINFO
 GENTB        I GENT1       I PREPAREBUFFERI CREATEINFO   I GETOBJLIST
 GENT2        I GEN         I CONTP        I EXPCHK       I EXTRDOCU
 DIST         I XPA         I SHOWSEMAPHOREI SHOWSEMACLIENI GETCONVENT
 MVNTABS      I VERSI       I SETSEMAPHORE I DELSEMACLIENTI EXPLAINRC
 IMPSYNC      I DELIVER     I DELSEMAPHORE I GETTPSTATENT I
 DEPLOY       I             I              I CHECKIN      I CHECKOUT
For more details call tp with the desired command (and no other parameters)
```

Figure 115: Overview of tp Commands (tp help)

tp commands are executed in the transport subdirectory **bin** by user *<SID>adm*. tp commands are called using syntax *tp <command> [argument(s)] [option(s)]*.

To display general information on the syntax of all tp calls, use *tp help*. To display a description of the syntax and function of a specific command, use *tp <command>* without any options.

To display a count of all transport requests registered for import into a specific system, use *tp count <SID>*.

To test whether a connection to a system's database was successful, use *tp connect <SID>*.

To display how the import dispatcher *RDDIMPDP* is scheduled for a specific SAP System, use *tp checkimpdp <SID>*.

To display all current settings of the parameter file, use *tp showparams <SID>*.

To display status information on the serialization of a specific system, use tp status <SID>.

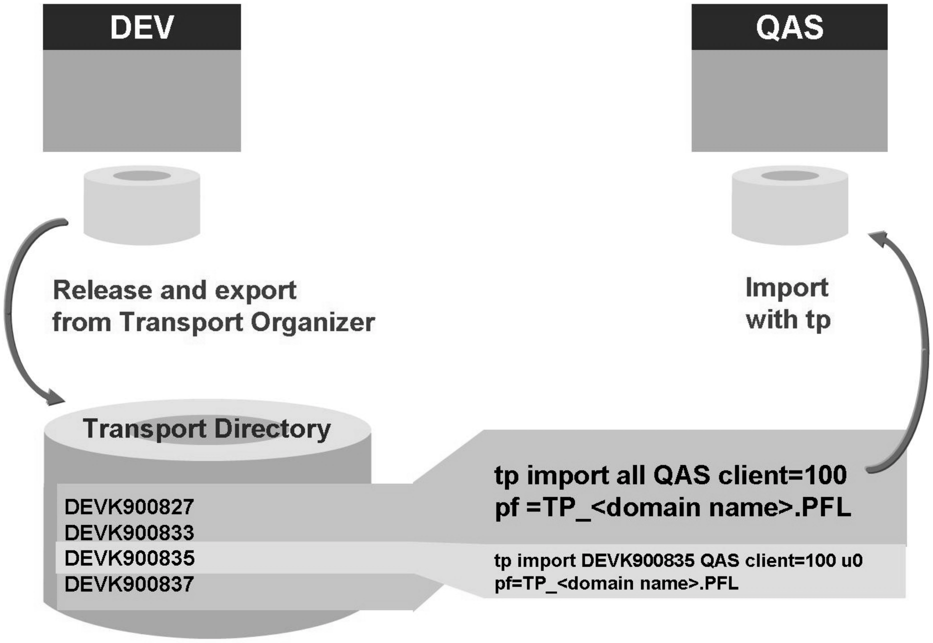


Figure 116: tp Import Command Line Option

Imports performed using tp at the operating system level should be restricted to exceptional cases, for example if the TMS is not available.

There are two commands to start an import using tp at the operating system level:

- **tp import all <target SID> client=<client number> pf=TP_<domain name>.PFL**

  Analogous to the default import of TMS, this command imports all waiting transport requests in the correct export sequence.

  Analogous to the TMS preliminary import, this command enables you to give individual transport requests priority.

- **tp import <transport request> <target SID> client=<client number> u0 pf=TP_<domain name>.PFL**

Importing individual requests is not recommended as the correct order of requests is not necessarily maintained and thus newer versions of objects may be overwritten by older versions through the regular import of all waiting requests. To ensure that the objects imported through an individual transport are not overwritten by an older version, only use unconditional mode 0, that is, the option u0, when starting individual imports. Using this mode is analogous to the TMS preliminary import and causes the transport request to remain in the list of requests to be imported. When the regular transport takes place, the request will automatically be imported again in the export sequence. Individual imports only should be performed in exceptional cases.

Processing imports out of order can result in severe inconsistencies in the target system. These are hard to diagnose. If you do not want to import single requests as preliminary import (NOT recommended), you would use *tp import <transport request> <target SID> client=<client number> pf=TP_<domain name>.PFL*, without the option u0.

Hint: Since release 6.10 you have to specify the full qualified path to the tp parameter file *TP_<domain name>.PFL* by using the parameter *pf=TP_<domain name>.PFL*.

The content of import buffer files in the transport subdirectory *buffer* are organized as a table. Each line contains information on a specific transport request. One column includes the unconditional mode that is linked to the transport request. Other columns specify import actions.

tp offers several commands for accessing buffers:

- **tp showbuffer <SID>**

  This command displays buffer entries.

- **tp addtobuffer <transport request> <target SID>**

  This command registers the request as the last request to be imported.

- **tp delfrombuffer <transport request> <target SID>**

  This command deletes a single request from the buffer.

- **tp cleanbuffer <SID>**

  This command removes successfully imported transport requests from the buffer. This function is included in the command *tp import all <target SID>*. This action can be started in transaction STMS by choosing *Overview → Imports → Import Queue → Display → Extras → Delete Imported Requests*.

- **tp setstopmark <SID>**

  This command places a stopmark at the end of a buffer so that subsequent tp import commands only process the requests that are located before the stopmark.

- **tp mvstopmark <TR> <SID>**

  This command moves the stopmark in front of the transport request in the buffer.

- **tp delstopmark <SID>**

  This command deletes the stopmark from a buffer.

The tp unconditional import modes are as follows:

- **0** Imported transport requests are not deleted from buffer.
- **1** Ignore that the transport request has already been imported
- **2** Overwrite originals
- **6** Overwrite objects in unconfirmed repairs.
- **8** Ignore restrictions based on table classification
- **9** Ignore the fact that the system is locked for this type of transport

Hint: The unconditional modes may vary in dependence of the SAP and tp release.

Unconditional modes are among the options that can be assigned to tp commands. They are used to cause specific rules of the Change and Transport System (CTS) to be ignored.

Each unconditional mode is represented by a digit. To use an unconditional mode, add a **u** to the tp command line and add the required digit(s). For example, if you want to import request DEVK900132 into the quality assurance system and overwrite originals, enter the following command: *tp import DEVK900132 QAS u2*

Use unconditional modes carefully. It is preferable to transport according to the rules of the Change and Transport System (CTS).

Some unconditional modes can be used directly in TMS.

Exercise 9: Import Process

Exercise Objectives

After completing this exercise, you will be able to:

- find out, if the import dispatcher RDDIMPDP is scheduled correctly
- monitor an import

Business Example

To enable the post import activities, the import dispatcher, *RDDIMPDP* has to be scheduled correctly in the SAP system.

Task 1:

Monitoring the import dispatcher *RDDIMPDP*.

1. Verify the scheduling of the import dispatcher RDDIMPDP in system QAS.
2. If the job RDDIMPDP or a client-specific job RDDIMPDP_CLIENT_### were not scheduled, how would you schedule them?

Task 2:

Monitor the 'import all'

1. After your instructor starts a long-running import of some transport requests, monitor the progress of the import from the QAS system. Start the different tools in parallel.

   Note: This exercise must be performed with the instructor's assistance. The instructor will import the transport requests into the quality assurance system (QAS).

Solution 9: Import Process

Task 1:

Monitoring the import dispatcher *RDDIMPDP*.

1. Verify the scheduling of the import dispatcher RDDIMPDP in system QAS.

   a) Log on to system QAS in client *100* as user *ADM325-##*. To view the scheduling of RDDIMPDP, use transaction SM37.

      - Choose *RDD** as job name and * as user name.
      - Deselect *Finished* in *Jobs status* box.
      - Enter * in the *or after event* field.
      - Choose *Execute*.

      In the *Job Overview* a list of all scheduled *RDD** jobs is displayed: *RDDIMPDP, RDDIMPDP_CLIENT_001, RDDIMPDP_CLIENT_100, ...*

      To get more details about the job, select a specific job and choose *Job → Job details*. After raising event *SAP_TRIGGER_RDDIMPDP* job RDDIMPDP is started. After raising event *SAP_TRIGGER_RDDIMPDP_CLIENT* all client-specific RDDIMPDP-jobs (RDDIMPDP_CLIENT_###) are started. The job frequency is event periodic.

2. If the job RDDIMPDP or a client-specific job RDDIMPDP_CLIENT_### were not scheduled, how would you schedule them?

   a) To schedule the job RDDIMPDP, log on to client *000* and run the program *RDDNEWPP*. To schedule the client-specific job RDDIMPDP_CLIENT_###, log on to the appropriate client ### and run the report *RDDNEWPP*. The step in the job is being scheduled under user *DDIC*, independent of the user, you logged on with.

Continued on next page

Task 2:

Monitor the 'import all'

1. After your instructor starts a long-running import of some transport requests, monitor the progress of the import from the QAS system. Start the different tools in parallel.

   Note: This exercise must be performed with the instructor's assistance. The instructor will import the transport requests into the quality assurance system (QAS).

   a) Log on to the QAS. Open six sessions:

      - Start transaction STMS, choose *STMS → Import Queue: System QAS*, choose *Goto → Import Monitor*. The TMS Import Monitor shows the current import process. Use *Refresh* to follow the progress of the import process.
      - Start transaction AL11, choose *DIR_TRANS → tmp*.
      - In transaction SM37, choose *Job name = RDD**, *User name* = *, **de**select *Finished* and select *start condition → after event* = *.
      - In transaction SE16, enter table *TRBAT*, enter table content, and choose *Execute*.
      - In transaction SE16, enter table *TRJOB*, enter table content and choose *Execute*.
      - In transaction SE06 choose *detailed analysis* and look for the CPU processes. Try to discover *tp* and *R3trans*.

      In the screen transport steps, you see the return code. During the import, entries are made in tables *TRBAT* and *TRJOB*. In transaction SM37, there are jobs visible, such as *RDDMASGL*, *RDDMNTAB*, and *RDDVERSL*.

Lesson Summary

You should now be able to:

- Identify the steps of the import process
- Identify the relationship between tp, R3trans and RDDIMPDP

Related Information

- help.sap.com: *SAP NetWeaver → Release '04 → SAP NetWeaver → Software Lifecycle Management→ Software Change Management → Change and Transport System → Transport Tools*

Lesson: Monitoring Tools

Lesson Overview

In this lesson, you will learn about the tools provided by the Transport Management System (TMS) for monitoring transport activity in your transport domain. You will also learn ways to ensure a stable test and development environment.

Lesson Objectives

After completing this lesson, you will be able to:

- Identify the tools for monitoring transport activity
- Identify the way to ensure a stable development environment

Business Example

As the Change and Transport System (CTS) administrator, you must be able to track the progress of the imports and make sure that only the required requests are being transported. TMS provides the administrator with a variety of tools to monitor all aspects of the import process and to ensure that the import is being performed with few or no errors.

Types of Monitoring Tools

TMS provides various tools with which you can monitor transport activities in your transport domain.

- Import monitor
- Import history
- tp system log
- Import queue consistency check
- Transport control program check
- Transport directory check
- RFC connection test

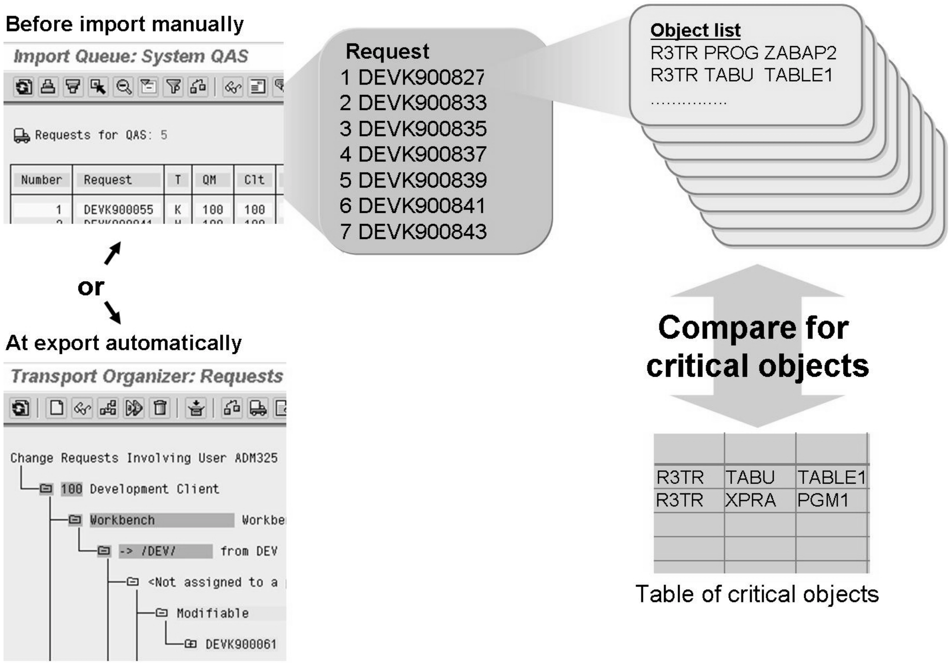
In transaction STMS choose *Overview → Imports*:

- Import monitor: To display tp status information from table *TPSTAT*, choose *Goto → Import Monitor*.
- Import monitor: To display the import history, choose *Goto → Import History*.
- tp system log: To display all tp calls and errors for the current SAP system in the file *SLOG*, choose *Goto → tp System Log*. To display the file *ALOG* containing the return codes of all individual transport steps for the current transport group, choose from the tp system log view *Goto → Transport Steps*.
- Import queue consistency check: To check if the data files and cofiles of transport requests in an import queue exist in buffers of the transport directory and can be read, choose *Queue → Check → Consistency*.

In transaction STMS choose *Overview → Systems* and mark one or more systems:

- Transport control program check: To check the transport control program tp for all or just one specific system within the domain, choose *SAP System → Check → Transport Tool*.
- Transport directory check: To verify the availability of the transport directories of all systems within the domain, choose *SAP System → Check → Transport Directory*.
- RFC connection test: To check RFC connections in both directions, choose *SAP System → Check → Connection test*.

Checking for Critical Objects



Number	Request	T	QM	Clt
1	DEVK900055	K	100	100

```
Change Requests Involving User ADM325
└─ 100 Development Client
   └─ Workbench                Workbe
      └─ -> /DEV/              from DEV
         └─ <Not assigned to a
            └─ Modifiable
               └─ DEVK900061
```

Request
1 DEVK900827
2 DEVK900833
3 DEVK900835
4 DEVK900837
5 DEVK900839
6 DEVK900841
7 DEVK900843

R3TR	TABU	TABLE1
R3TR	XPRA	PGM1

Figure 117: Checking for Critical Objects

There are two options to check for critical objects:

- **before the import of transport request into the target system**

  This option has to be performed manually and is only a display of the list of transport request that contain critical objects.

- **during the export of the transport request**

  This option has to be performed automatically, if tp parameter *CHK_CRIOBJ_AT_EXPORT* is set to *W* (warning) or *E* (error).

To check if the transport requests in an import queue contain critical objects that should not be imported into the target system, go to the *Import Overview* (*STMS → Overview → Imports*) and choose *Queue → Check → Critical Objects*. The object lists of the requests are checked to see if they contain critical transport objects. The results are displayed in a hierarchical list. Requests containing critical objects are marked with the appropriate icon. The critical objects are highlighted in color.

Before performing this check, you must have maintained, which objects are classified as critical. You must be logged on to the domain controller. The information about critical objects is distributed to the whole domain when you save.

To display or maintain the critical transport objects defined for the transport domain, call transaction STMS. Choose *Overview → Imports → Extras → Critical Transport Objects*.

Note: You can also check requests for critical objects in the QA worklist. This supports the decisions you make on approving or rejecting requests.

If you want to check the whole QA worklist for critical objects, choose from within the QA work list *Worklist → Check → Critical Objects*. Confirm the popup. An overview of the requests with critical objects appears. If you only want to check certain requests for critical objects, filter them in the display. Choose *Worklist → Check → Critical objects*. Confirm the dialog box. An overview of the requests with critical objects appears.

Only objects of the form object type *R3TR* are checked as transport objects. Transport objects that start with *LIMU* are subobjects of a Repository object with an object directory entry. To check these, you must find out the object directory entry, and enter it in the table of critical objects.

Hint: Interesting objects to be classified as critical, would be for example *R3TR XPRA* (XPRAs) or *R3TR TABL* (table definitions) or *R3TR TABU* (table content, for interesting tables)

Code Freezing and Testing

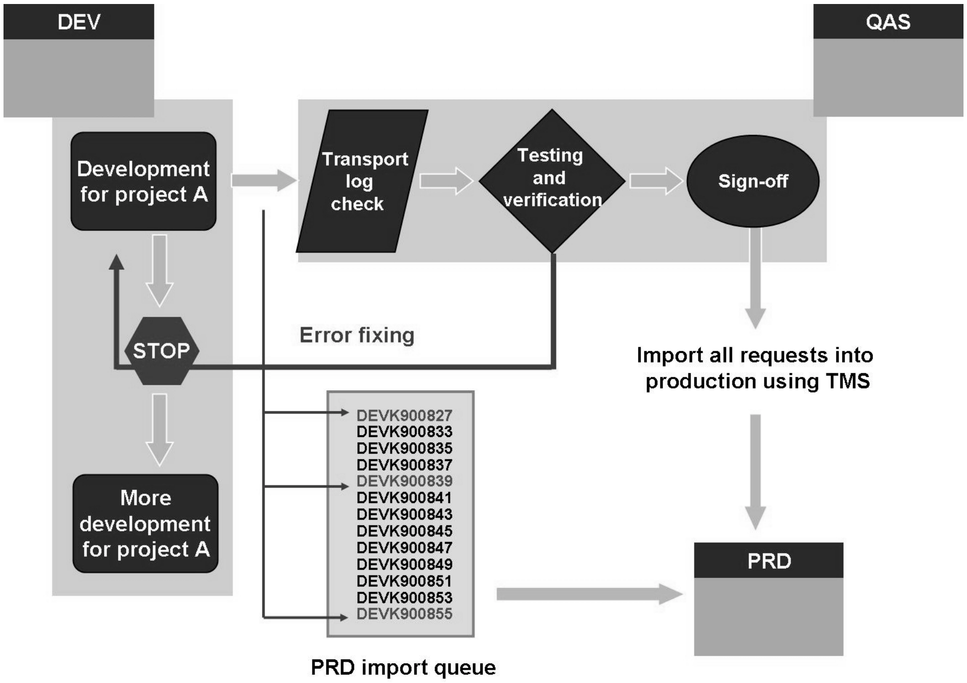


Figure 118: Code Freeze and Testing

To ensure a stable test and development environment, use a developer deadline to freeze work on objects in the development system until the completion of quality assurance verification.

Follow this procedure during object development and testing:

1. Release the transport request containing the developed objects.
2. Freeze development of the objects in the development system.
3. Import the objects and verify the changes in the quality assurance environment.
4. Sign off the changes.
5. If necessary, allow further development on the objects in the development system.

With the following tools an administrator can force code freezing for a system landscape:

- **stop exports** by creating a file in transport directory: */usr/sap/trans/bin → file: T_OFF.<SID>*

  The first line of this file will be displayed, whenever you start an export from the Transport Organizer.

- **stop imports** by creating a file in transport directory: */usr/sap/trans/tmp → file: NOIMPORT.<SID>*

  The content of this file is not evaluated.

When errors are discovered during quality assurance testing, the affected objects are sent back to the development system for correction. However, if additional development has been performed on these or related objects, the problem is more difficult to correct.

If an object has gone through the error fixing cycle, it is included in at least two different transport requests within the import queue of the production system. If you import the complete import queue, using *Import all*, into the production system after sign-off, the pre-corrected object has no impact on your production environment.

Change management includes creating, releasing, distributing, and verifying transport requests for all SAP systems in the system landscape. Ensure that your distribution procedures are supported by the TMS setup. Determine release, import, and verification requirements for all transport points in your system landscape. For each transport point, define the following:

- When transports will take place?
- Who is responsible for a transport request during its various transport phases?
- How transports will be verified before being distributed or redistributed?
- What happens if a transport is successful but testing proves that the contents are incorrect?
- Are sign-offs required for transporting?

Assign persons to be responsible for all transport steps. Use authorizations to restrict access to transport request management tasks and to the operating system level.

SAP provides techniques for the distribution and management of changes using import buffers. These buffers indicate the changes to be transported to the target system and the transport order. You must establish and control the buffers and the distribution or redistribution process. To verify all imports, review transport logs, establish testing standards and quality assurance procedures, and define sign-off procedures.

SAP recommends transporting a whole change project when all objects are in an acceptable state rather than each single object as soon it is ready.

Transport Directory Naming Conventions

Because the transport control program tp runs on many different operating systems, restrictive naming conventions are required. Change requests are always represented in the following format *<source SID>K9<5 digits or characters>*, where source SID is the system in which the transport request is created and K9 indicates a customer request. The five digits form a serial number, which can be expanded using characters.

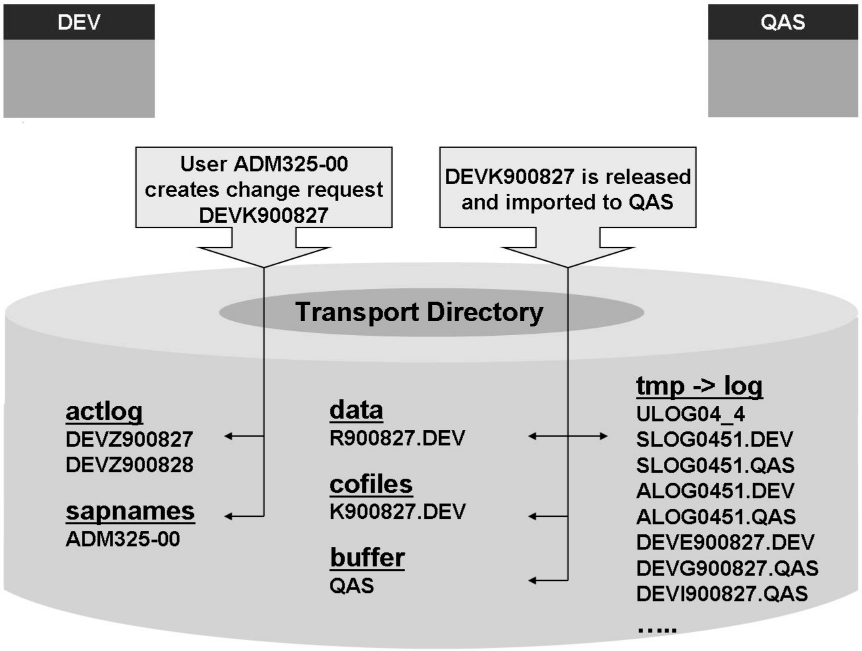


Figure 119: The Transport Directory

The transport directory contains subdirectories, such as bin, data, cofiles, buffer, log, tmp, actlog, sapnames, EPS.

bin

This subdirectory contains the configuration files for the transport domain.

data

This subdirectory contains files named *R9<5 digits >.<source SID>* containing the exported objects.

cofiles

This subdirectory contains command files named *K9<5 digits>.<source SID>*. They contain, for example, import steps to be performed.

buffer

This subdirectory contains an import buffer for each SAP System named after the SID. When a transport request is released, the import buffer of the target systems is updated.

log

This subdirectory contains all log files, such as *ULOGs*, *ALOGs*, *SLOGs*, log files named *<source SID><action>9<5 digits>.<action SID>* for each executed step, and log files named *<action><date>.<action SID>* for steps that are collectively executed, for example, step N (structure conversion) and step P (move nametabs).

tmp

This subdirectory contains log files before written to log.

actlog

This subdirectory contains files named *<source SID>Z<6 digits>* recording each action on a request or task, for example, creation, release, or change of ownership.

sapnames

This subdirectory contains files named after the user's logon name. A file is created for each SAP system user, who performs transport actions, and updated when the user releases a request.

EPS

This subdirectory (Electronic Parcel Service) contains the directory *in*, to which Support Packages are copied in order to apply them with the Support Package Manager (transaction SPAM).

Troubleshooting Steps

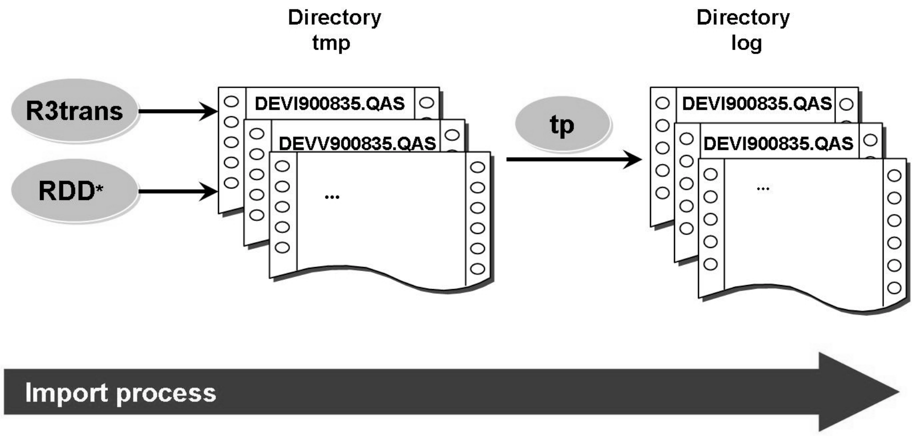


Figure 120: Monitoring and Analysis: Transport Tool Log Files

The various transport tools write a log for each transport action to the transport subdirectory *tmp*. After completion of this step, tp moves these logs from *tmp* to *log*. The log files are named *<source SID><action><6 digits>.<target SID>*, where the action is represented by a single character and the 6 digits are taken from the corresponding transport request.

Each log file can be seen in the SAP system. You can choose different detail-levels in the log file.

The level for the transport protocol display in the SAP system. The possible levels are as follows:

- Performed actions and return code
- Additional error messages
- End-user logs
- Details for developers and hotline

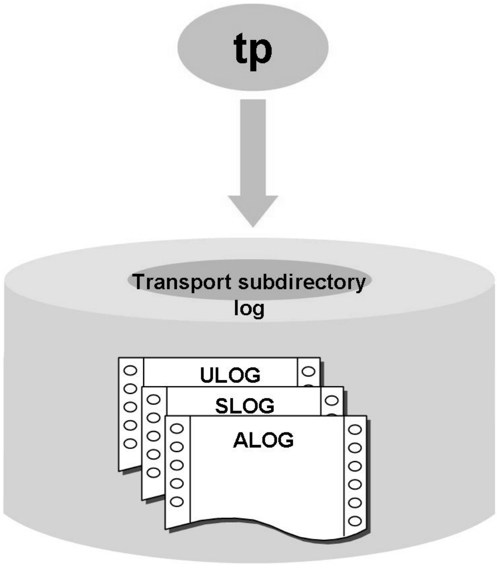


Figure 121: Monitoring and Analysis: tp Log Files

For long-running imports, it may be helpful to monitor the log files written at the operating system level. All logs in the transport environment are stored in the transport subdirectory log. These logs include logs created by tp (*ULOG*, *SLOG* and *ALOG*), and logs created by the various transport tools.

The current *ULOG* file records all tp commands that are free of syntax errors, and is named using the name convention *ULOG<YY>_<one digit>*. Each line in the ULOG file represents a tp command.

The *SLOG* file is used to monitor the transport activities of a specific SAP system. It contains a general overview of performed imports indicating the return code and thus the success of each import. The name of the SLOG file can be set as tp parameter using the global parameter *SYSLOG*. The default setting is *SLOG<YY><WW>.<SID>*.

The *ALOG* file records the return code for all transport steps handled in the common transport directory. The name of the ALOG file can be set as tp parameter using the global parameter *ALLLOG*. The default value is *ALOG<YY><WW>*.

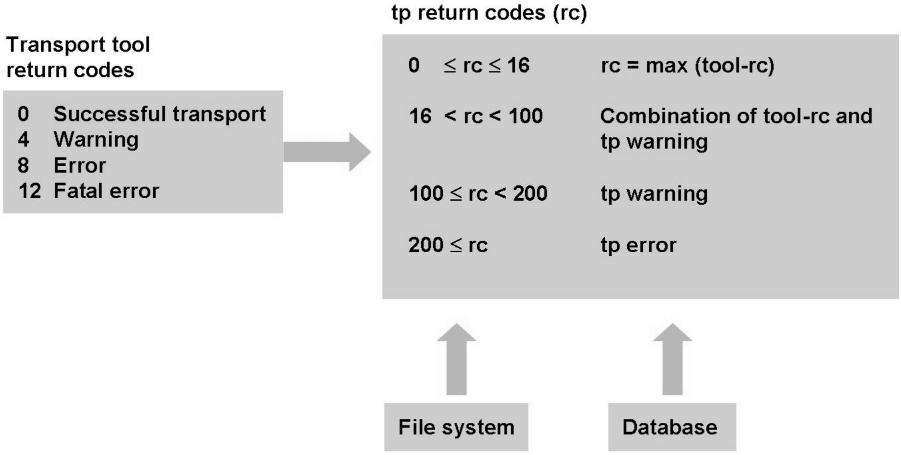
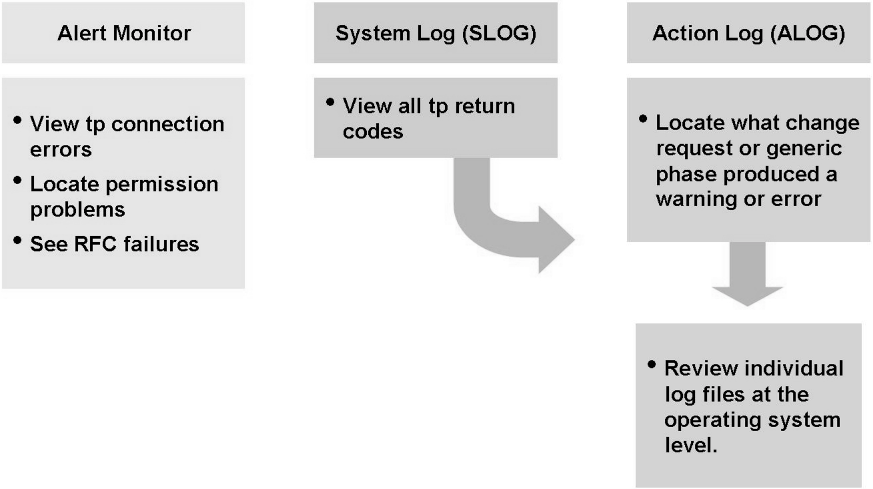

tp return codes (rc)
Transport tool return codes
0 Successful transport
4 Warning
8 Error
12 Fatal error
0 ≤ rc ≤ 16 rc = max (tool-rc)
16 < rc < 100 Combination of tool-rc and tp warning
100 ≤ rc < 200 tp warning
200 ≤ rc tp error
File system
Database

Figure 122: tp Return Codes

tp receives return codes from all the transport tools involved in an import process. tp's own return code following exit is interpreted as follows:

- 0 to 16 indicate the maximum value of all return codes from transport tools.
- 17 to 99 are values that are calculated from the return codes of the transport tools and a tp warning, for example, the buffer of the target system has no write permission.
- 100 to 199 indicates warnings. Warnings mean that something went wrong and tp could not perform all the tasks. 100 to 149 are normal tp warnings. For example, RDDIMPDP cannot be triggered by sapevt. Return codes of 150 to 199 are rare and indicate incorrect operation by a user. For example, a return code of 152 is received if tp tries to import a request that is not included in the buffer.
- 200 or more indicates tp errors. For example, if a file could not be accessed as required by the import process, the return code is 212.

The accompanying text is more significant than the value of a return code. To display the text of a specific tp return code, use the tp command *tp explain rc <value of return code>*.



Alert Monitor

- View tp connection errors
- Locate permission problems
- See RFC failures

System Log (SLOG)

- View all tp return codes

Action Log (ALOG)

- Locate what change request or generic phase produced a warning or error

- Review individual log files at the operating system level.

Figure 123: Troubleshooting (1)

The first step in troubleshooting transport errors is use the alert monitor, which records all TMS transport actions. The alert monitor is available in transaction STMS by choosing *Monitor → Alert Viewer*. The information displays the date and time, the user name, the TMS status message, and the target SAP system. To display the full text of an error message, double-click the error message.

More detailed information can be seen in the *SLOG* file, which is used to monitor the transport activities of SAP systems and determine the success of import requests.

If import failures are recorded in the *SLOG* file, drill down to the *ALOG* file and locate the import step that produced the return code listed in *SLOG*.

Use *ALOG* file to identify the detailed log file, which is written for each step of a transport request. These log files can be accessed from the SAP system using the Transport Organizer, transaction *SE09*. To locate the log file for the request that produced an error use *ALOG* file.

All log files that are not dependent on specific requests, such as the log files for structure conversion and nametab moving (mvntab), cannot be displayed in the SAP system. These files can be accessed at the operating system level in the transport directory.

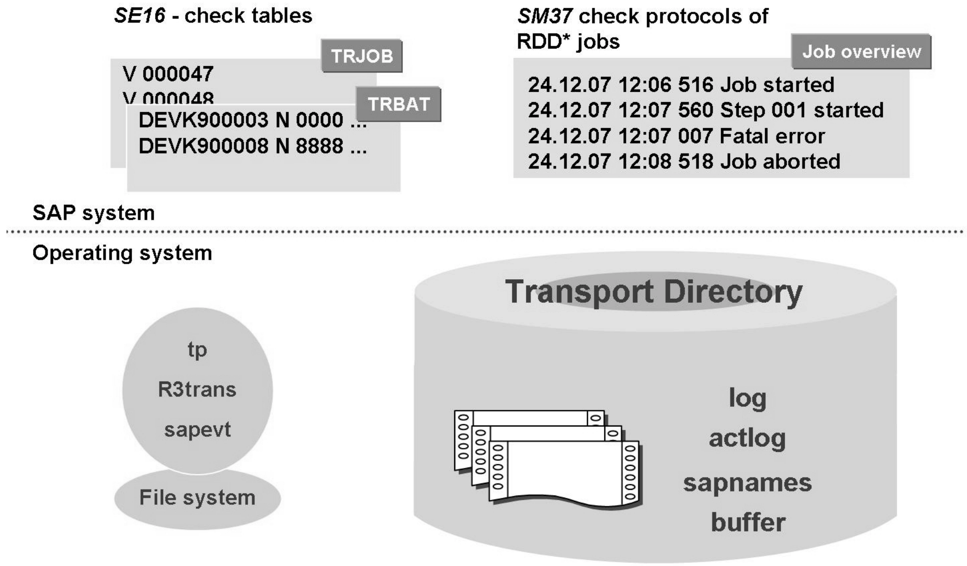


Figure 124: Troubleshooting (2)

Additionally you can check whether the import dispatcher *RDDIMPDP* is scheduled correctly and is event-triggered. Use the job overview *SM37* to monitor the related background jobs (RDD*). Enter **`RDD*`** in the *Job name* field and an asterisk **`*`** in the *or after event* field.

Problems might result from wrong versions of tp or R3trans; tp not running, as in UNIX “ps -ef | grep tp”; permission or share problems with the common transport directory; or no free disk space.

When analyzing a problem, compare the logs and buffer entries with the entries in tables *TRBAT* and *TRJOB*, that is, transaction SE16. If required, insert the transport request or header in *TRBAT* and restart *RDDIMPDP*.

If a communication problem between tp and the SAP system is indicated, try to start *sapevt* at operating system level and trigger *RDDIMPDP*.

Exercise 10: Monitoring Tools

Exercise Objectives

After completing this exercise, you will be able to:

- Analyze problems that occur while importing transport requests
- Follow the steps for using transport log files for troubleshooting

Business Example

When transports cannot be finished successfully, it is essential to analyze the transport errors and solve the problem.

Task 1:

Reviewing export and import files in the transport directory

1. You want to see transport logs generated by a transport request during export and import within your SAP system. To which system (DEV, QAS, ...) do you have to log on?
2. Have a look at the files in the transport directory related to one of your **released and exported** transport requests. Use transaction AL11 to view files at operating system level.
3. Have a look at the files in the transport directory related to one of your **imported** transport request. Use transaction AL11 to view files at operating system level.

Task 2:

Troubleshoot the import problem #1.

1. While the instructor imports the transport requests into the quality assurance system, monitor the import process.
2. The import monitor reports a return code *0008*. What import phase and/or transport request produced the import error?
3. How can you locate the log files for the transport request that produced the import error?
4. How would you correct this error?

Continued on next page

Task 3:

Troubleshoot the import process problem #2.

1. While the instructor imports the transport requests into the quality assurance system, monitor the import process.
2. The import monitor reports a return of *0008*. What import phase and/or transport request produced the import error?
3. How can you locate the log files for the transport requests that produced import errors? What are the errors that are reported?
4. What caused this error? How could it have been prevented?
5. How would you correct this error?

Solution 10: Monitoring Tools

Task 1:

Reviewing export and import files in the transport directory

1. You want to see transport logs generated by a transport request during export and import within your SAP system. To which system (DEV, QAS, ...) do you have to log on?

   a) The DEV and QAS system share the same transport directory and thus are in the same transport group. Therefore, it does not matter from which SAP system you view the transport directory.

      Note: If your transport domain contains multiple transport groups, you need to log on to a system that accesses the transport directory where the log files reside.

2. Have a look at the files in the transport directory related to one of your **released and exported** transport requests. Use transaction AL11 to view files at operating system level.

   a) Start transaction AL11 and select the directory parameter *DIR_TRANS*. The following files are created or modified in the transport directory during the **export process** for transport requests. The solutions use the transport request named *DEVK900132* as an example. Substitute your transport request number.

      - The **cofiles** subdirectory contains important control information on how to import and process transport requests. You can find the command file *K900132.DEV* in this subdirectory.
      - The **data** subdirectory contains data file *R900132.DEV*. This file stores all the data that was extracted from the development system based on the objects recorded in the transport request.

        Note: The data files are not in character format; therefore, the contents are not viewable with transaction AL11.

      - The **buffer** subdirectory contains control information on which transports are to be imported into which systems and the order in which imports should take place. Each buffer file is named with the <*SID*> of the corresponding SAP system. You can see an entry in

Continued on next page

the buffer file of QAS. If you are using transport target groups to import into multiple clients, there may be multiple entries, one for each client in the target group.

Note: Comment lines begin with a number sign (#). Comments are used for informational purposes only.

- The **log** subdirectory contains transport logs, trace files, and statistics. The fourth letter of the filename specifies the step that was performed. The <*SID*> at the end of each filename specifies the SAP system on which the actions were performed. The following example log files were written during the **export process** of request *DEVK900132*:

  DEVEK900132.QAS: export log file

  DEVPK900132.QAS: test import log file

  Note: The tp parameter *TESTIMPORT* controls whether a test import is automatically executed during the export phase. If this is deactivated, the test import log will not exist.

- The **sapnames** subdirectory contains information for each SAP user who released a request. Select the file for your user name *ADM325-##*.

Continued on next page

3. Have a look at the files in the transport directory related to one of your **imported** transport request. Use transaction AL11 to view files at operating system level.

   a) Start transaction AL11 and choose the directory parameter *DIR_TRANS*. The following files are created or modified in the transport directory during the **import process** for transport requests. The solutions use the transport request named *DEVK900132* as an example. Substitute your transport request number.

      Choose subdirectory **buffer**, which contains each buffer file named with the corresponding <*SID*>. If you choose *QAS*, you can see request *DEVK900132* still listed, if the transport request was imported using the preliminary import functionality. After import into QAS, an entry for the request appears in the buffer of PRD.

      The **log** subdirectory contains log files written during both the export and **import process**. Examples of log files that may be written during the import process of request *DEVK900132* include:

      - DEV**H**900132.QAS - import log file
      - DEV**A**900132.QAS - dictionary activation log file
      - DEV**I**900132.QAS - main import log file
      - DEV**M**900132.QAS - enqueue log file
      - DEV**V**900132.QAS - versioning log file
      - DEV**R**900132.QAS - user-defined activities
      - DEV**G**900132.QAS - generation log file

      Note: Each transport request requires different import phases and log files are written only for the phases required by that transport request. Some import phases are generic and do not create individual log files for each transport request, rather, one log file for the import phase.

Continued on next page

Task 2:

Troubleshoot the import problem #1.

1. While the instructor imports the transport requests into the quality assurance system, monitor the import process.

   a) You can monitor the import process for example in one of the following three ways:

      - Start transaction STMS, choose *STMS → Import Queue: System QAS*, choose *Show more*. Now you can directly see the return code of the transport requests in the import queue. By double-clicking the symbol for the return code, you can see the detailed log files for this transport request.
      - Start transaction STMS, choose *STMS → Import Queue: System QAS*, choose *Goto → Import Monitor*. The TMS Import Monitor shows the current import process. To follow the progress of the import, choose *Refresh*.
      - Start transaction STMS, choose *STMS → Import Queue: System QAS*, choose *Goto → tp System Log*. Go to the bottom of the SLOG listing. To follow the progress of the import, choose *Refresh*.
      - From the *tp System Log - System QAS* screen, choose *Goto → Transport steps*. Go to the bottom of the ALOG listing. To follow the progress of the import, choose *Refresh*.

2. The import monitor reports a return code *0008*. What import phase and/or transport request produced the import error?

   a) To check whether there are obvious setup errors, start transaction STMS, choose *Monitor → TMS alerts → CCMS Alert Monitor* and drill down to the QAS branch of the tree. No configuration errors are recorded.

   b) To determine the success of imports, start transaction STMS, from the *Import Queue: System QAS* screen, choose *Goto → tp system log*. You can see as the last step, an imp all (import all) stopped with a return code *0008*.

   c) To view the action log file *ALOG*, choose *Goto → Transport steps*. To locate the transport request(s) or generic phase(s) that produced error *0008*, go to the bottom of the listing and check the return codes of all individual transport steps. You can now detect the request that caused problems, displaying a return code of *0008* (error) for step *G* (generation).

Continued on next page

3. How can you locate the log files for the transport request that produced the import error?

    a) View the *ALOG* and double-click the error.

    Hint: As long as the transport request is still in the import queue, you can directly reach the detailed log files by double-clicking the symbol for the return code of the transport request. the return codes are only shown in the import queue, if you switch to the extended view (*Show more*).

4. How would you correct this error?

    a) Correct the syntax error in the development system. Corrections should always be made in the development system.

    b) Use the program check to ensure that the program is correct.

    c) Release and export the new request. Then, import the request with the corrected object into the target system.

Task 3:

Troubleshoot the import process problem #2.

1. While the instructor imports the transport requests into the quality assurance system, monitor the import process.

    a) (See exercise *Troubleshoot the import problem #1* above)

2. The import monitor reports a return of *0008*. What import phase and/or transport request produced the import error?

    a) A transport request returned *0008* during the activation phase *A*.

Continued on next page

3. How can you locate the log files for the transport requests that produced import errors? What are the errors that are reported?

   a) View the *ALOG* or the return code in the import queue and double-click on the error.

   b) Expand the log file completely.

      The red lines indicate where the problem is. During the ABAP Dictionary activation step of the import process, the system detected that either the field's data element of a table from the transport request or its underlying domain is either not active in the ABAP Dictionary in QAS, or does not exist. Therefore, the runtime object (nametab) of the table could not be generated and the table could not be activated.

4. What caused this error? How could it have been prevented?

   a) This error occurred probably because the table and the corresponding date element were developed and transported in two different transport requests, and the data element was not imported together in the same import step as the table that refers to the data element. Such problems can occur if imports of single requests are triggered and the correct sequence is not considered.

      To prevent such problems, developers should transport all related objects in the same transport requests to ensure proper activation.

5. How would you correct this error?

   a) Import the transport request that contains the missing data element. Import the transport requests, that belong together, in on step into the subsequent systems.

Lesson Summary

You should now be able to:

- Identify the tools for monitoring transport activity
- Identify the way to ensure a stable development environment

Lesson: Cleaning Up the Transport Directory

Lesson Overview

In this lesson, you will learn how to clean up the transport directory.

Lesson Objectives

After completing this lesson, you will be able to:

- Follow the steps for cleaning up a transport directory

Business Example

From time to time it makes sense to clean up the transport directory.

Cleaning Up the Transport Directory

To clean up the transport directory, use the commands *tp check all* and *tp clearold all*. In the course of time, many large files might accumulate in the transport directories. These files contain important information, but they also require disk space and become obsolete over time.

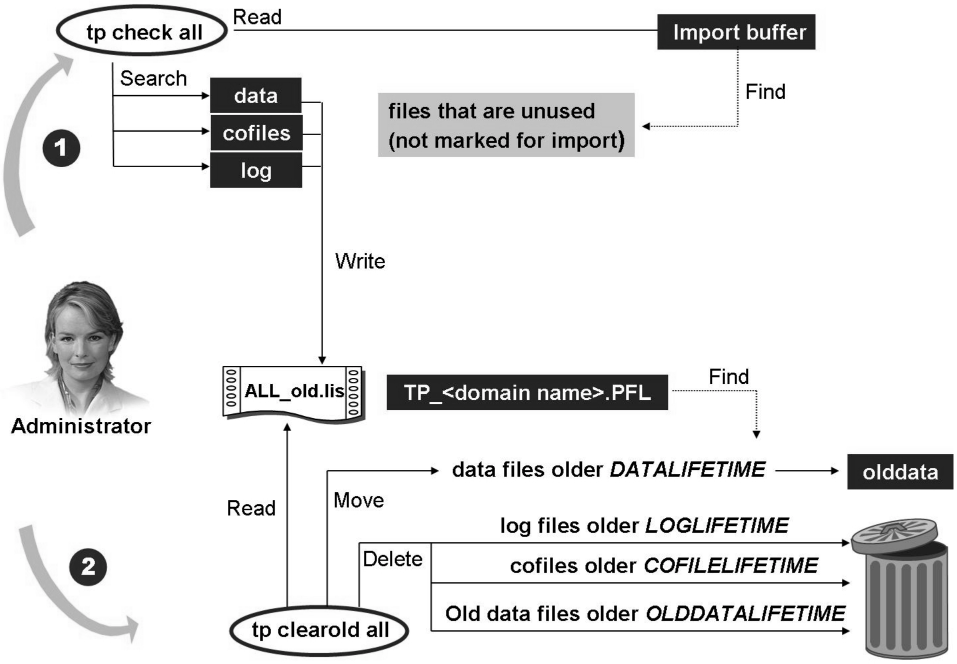

tp check all
Read
Import buffer
Search
data
cofiles
log
Find
files that are unused (not marked for import)
1
Write
Administrator
ALL_old.lis
TP_<domain name>.PFL
Find
data files older *DATALIFETIME*
olddata
Read
Move
log files older *LOGLIFETIME*
Delete
cofiles older *COFILELIFETIME*
2
Old data files older *OLDDATALIFETIME*
tp clearold all

Figure 125: Cleaning Up the Transport Directory

The command *tp check all* searches for transport requests that are not marked for import into any SAP system, as revealed by the buffer entries. The transport requests are listed in file *ALL_old.lis* which is located in transport subdirectory *tmp*.

The command *tp clearold all* uses the result list of the command *tp check all*, to find files that have exceeded a maximum age. To specify the maximum age, use the transport profile parameters *DATALIFETIME*, *OLDDATALIFETIME*, *COFILELIFETIME*, and *LOGLIFETIME*. Data files in the transport subdirectory *data* that are older than the parameter *DATALIFETIME* are moved to the transport subdirectory *olddata*. Files in the subdirectories *log* and *cofiles* are immediately deleted if they exceed the age specified by *LOGLIFETIME* or *COFILELIFETIME*. Files in *olddata* are deleted if they are older than *OLDDATALIFETIME*.

Before cleaning up the transport directory, SAP recommends making a copy of the transport directory and saving it for audit purposes.

Hint: With the command *tp testold all* it is possible to create a list of files which will be deleted by the command *tp clearold all*. The result is located in subdirectory *tmp*.

Lesson Summary

You should now be able to:

- Follow the steps for cleaning up a transport directory

Related Information

- Read SAP note 41732 *Deletion of data in transport directory* at SAP Service Marketplace, service.sap.com/notes.

Lesson: Transport of non-ABAP objects

Lesson Overview

After you have done all necessary configuration steps on both the Transport Domain Controller (TDC) System and the Systems that are included in the transport landscape (runtime systems), you are able to transport non-ABAP objects into subsequent target systems.

This lesson starts with a section that contains the “big picture”. Here the main steps for transporting non-ABAP objects are listed. In the following sections, these steps are presented in more detail with respect to the usage type of the systems in question

Lesson Objectives

After completing this lesson, you will be able to:

- Outline the process of transporting non-ABAP objects with the help of ABAP Change and Transport System

Business Example

Your company uses the Change and Transport System (CTS) to perform transports in transport landscapes based on SAP systems with AS ABAP. In addition your company is performing some portal development and some development on Exchange Infrastructure (XI) systems.

As a member of the transport administration team you want to know how to transport portal content files (EPA files) and XI transport files (TPZ files) with the help of CTS as well.

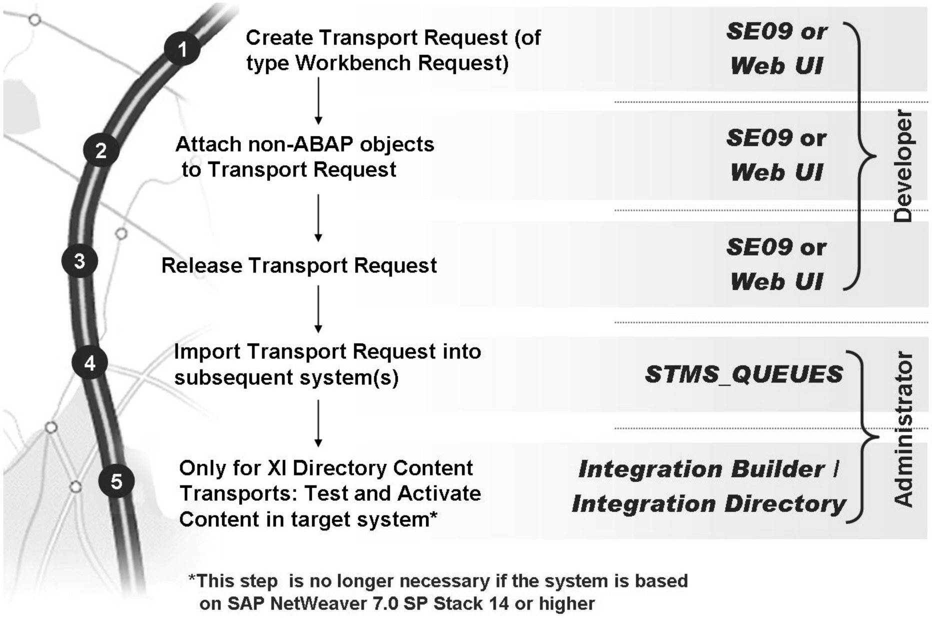
The “big picture”

After you have done all necessary configuration steps on both the Transport Domain Controller System / the systems on which the CTS Deploy Web Service is configured (abbreviated as CTS system in this lesson) and the systems that are included in the transport landscape, you are able to transport non-ABAP objects into subsequent target systems.

The following roadmap contains the main steps to transport non-ABAP content with the help of the enhanced Change and Transport System (CTS).

Note: For the sake of simplicity it is assumed that the Transport Domain Controller (TDC) system is a system with AS ABAP + Java runtime environment based on SAP NetWeaver 7.0 SP Stack 13 or higher and that the CTS Deploy Web Service is configured on this system.

As a prerequisite, all necessary configuration steps both on the TDC system and the target systems must have been performed. The TDC system should be at least on SAP NetWeaver 7.0 SP Stack 13.


1
Create Transport Request (of type Workbench Request)
SE09 or Web UI
2
Attach non-ABAP objects to Transport Request
SE09 or Web UI
3
Release Transport Request
SE09 or Web UI
Developer
4
Import Transport Request into subsequent system(s)
STMS_QUEUES
5
Only for XI Directory Content Transports: Test and Activate Content in target system*
Integration Builder / Integration Directory
Administrator

*This step is no longer necessary if the system is based on SAP NetWeaver 7.0 SP Stack 14 or higher

Figure 126: Transporting non-ABAP content – The big picture

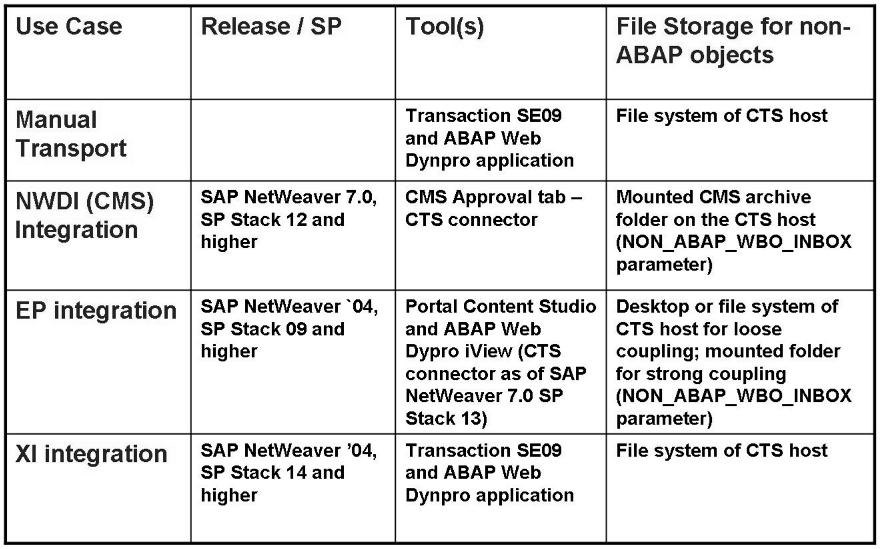
First you need to create a transport request (of type “workbench request”). This step is performed on the TDC system either with transaction SE09 (for development systems based on AS ABAP + Java) or with the help of the Transport Organizer Web UI (for development systems based on AS Java only).

Next you may attach non-ABAP object (for example SCAs, EARs, SDAs, EPAs, TPZs) to this transport request. This can also be done either directly in transaction SE09 of the development system (in case of a development system based on AS ABAP + Java 7.0 SP Stack 12 or higher) or with the help of a Web UI based on the Transport Organizer Web Service in the TDC system. Then you may release the transport request. Again this step can be done either directly in transaction SE09 of the development system or with the help of a Web UI based on the Transport Organizer Web Service in the TDC system.

After the export has finished successfully, you may want to import the transport request to a subsequent system (for example the quality assurance portal system). The import can be triggered with transaction STMS_QUEUES on either the target system (if it is based on AS ABAP) or on the TDC system.

Hint: Transaction STMS_QUEUES offers a subset of the STMS functions. In case of Integration Builder Content Transports (TPZ files) – depending on the release and SP stack level – it might be necessary to activate the changes manually in the Integration Directory of the target system.

Note that in detail the transactions or Web UIs for the steps listed on the roadmap above depend on the SP stack level of the TDC system.



Use Case	Release / SP	Tool(s)	File Storage for non-ABAP objects
Manual Transport		Transaction SE09 and ABAP Web Dynpro application	File system of CTS host
NWDI (CMS) Integration	SAP NetWeaver 7.0, SP Stack 12 and higher	CMS Approval tab – CTS connector	Mounted CMS archive folder on the CTS host (NON_ABAP_WBO_INBOX parameter)
EP integration	SAP NetWeaver `04, SP Stack 09 and higher	Portal Content Studio and ABAP Web Dypro iView (CTS connector as of SAP NetWeaver 7.0 SP Stack 13)	Desktop or file system of CTS host for loose coupling; mounted folder for strong coupling (NON_ABAP_WBO_INBOX parameter)
XI integration	SAP NetWeaver ’04, SP Stack 14 and higher	Transaction SE09 and ABAP Web Dynpro application	File system of CTS host

Figure 127: Different Use Cases for the enhanced CTS

The table above lists different use cases for the enhanced Change and Transport system (CTS system) together with the supported releases, the transport tools (export and import part) and the file storage location.

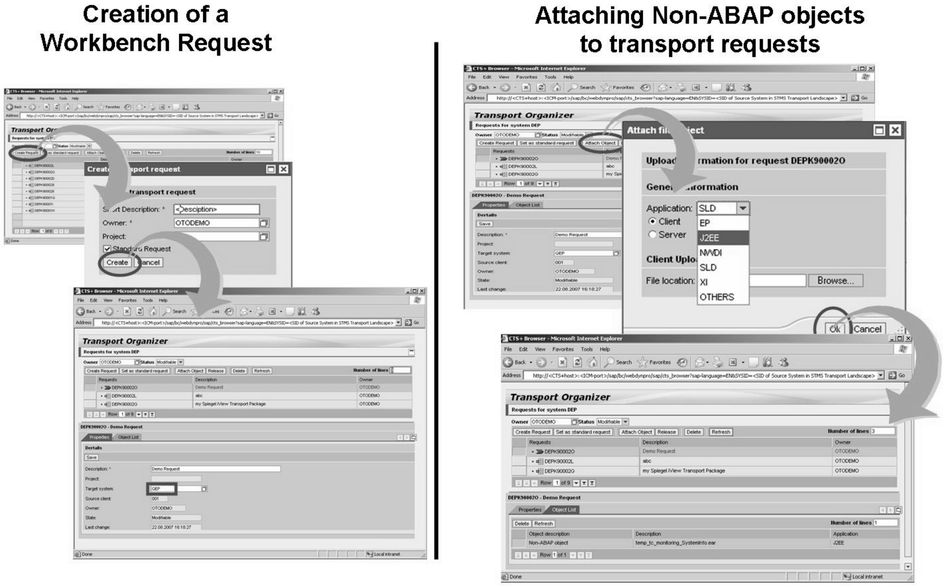
Note: Note that in any cases the Transport Domain Controller system should be at least on SAP NetWeaver 7.0 SP Stack 13.

Generic Steps for Performing Transports

Before you can perform the integrated transport scenarios, a transport request of type *workbench* is needed.

Creation of Workbench Requests and Attaching non-ABAP objects

This transport request can either be created on the development system in transaction SE09 (Transport Organizer) or it can be created in a Web UI for the *CTS_BROWSER* Web Dynpro Application that runs on the AS ABAP part of the TDC system.



CTS+ Browser - Microsoft Internet Explorer

Address: http://<CTS+host>:<ICM-port>/sap/bc/webdynpro/sap/cts_browser?sap-language=EN&SYSID=<SID of Source System in STMS Transport Landscape>

Create transport request

Short Description: * <Desciption>
Owner: * OTODEMO
Project:
☑ Standard Request

Transport Organizer

Requests for system DEP

Owner OTODEMO Status Modifiable

Requests	Description	Owner
DEPK900020	Demo Request	OTODEMO
DEPK90002L	abc	OTODEMO
DEPK90002G	my Spiegel iView Transport Package	OTODEMO

Row 1 of 9

DEPK900020 - Demo Request

Properties | Object List

Details

Description: * Demo Request
Project:
Target system: QEP
Source client: 001
Owner: OTODEMO
State: Modifiable
Last change: 22.08.2007 16:18:27

Attach file object

Upload Information for request DEPK900020

General Information

Application: SLD (EP, J2EE, NWDI, SLD, XI, OTHERS)
● Client
○ Server

Client Upload

File location:

Object List — Number of lines 1

Object description	Description	Application
Non-ABAP object	temp_tc_monitoring_SystemInfo.ear	J2EE

Row 1 of 1

Figure 128: Creation of Workbench Requests and Attaching non-ABAP objects

Note: Note that the *CTS_BROWSER* Web Dynpro is available as of SAP NetWeaver 7.0 SP Stack 13 (for the TDC system). It can be accessed directly from the entry screen of transaction STMS using the Transport Organizer Web UI pushbutton

Hint: You may delete the user in the tasks list, as a task is not needed for non-ABAP transports.

The transport request that you create with the help of this Web Dynpro on the Transport Domain Controller will be named <SID>K9<number> where number is a running number and <SID> is the name of the development system according to your transport landscape.

There are two options to attach non-ABAP transport objects (for example an EAR or an SCA file) to existing Workbench Requests:

- If you use transaction SE09 on the development system, you need to store the file in question on this host. In this case, select the new transport request and choose *Include Objects → Non-ABAP object*. Then you need to specify the directory and file name which you want to deploy via CTS to the runtime systems configured in the transport route. Furthermore the type of application and deployment needs to be specified.

  As a result, you should see the Non-ABAP object in the object list of the request. Finally you may want to double-click the GUID of the Non-ABAP-Object to check the attributes and content.

- The other option is to use the coupling to the transport organizer using the ABAP Web Dynpro application *CTS_BROWSER*.

  In this case, start a browser and enter the URL http://<TDC-Host>:<ICM-Port>/sap/bc/webdynpro/sap/cts_browser?sap-client=<client number as specified by the tp parameter NON_ABAP_WBO_INBOX> (or press the corresponding pushbutton on the start screen of transaction STMS). On the resulting screens, enter the client for which you want to perform the transport, your user ID on the TDC system and the password. Within that dialog, you can select the transport request you want to use for attaching files from your file system to it.

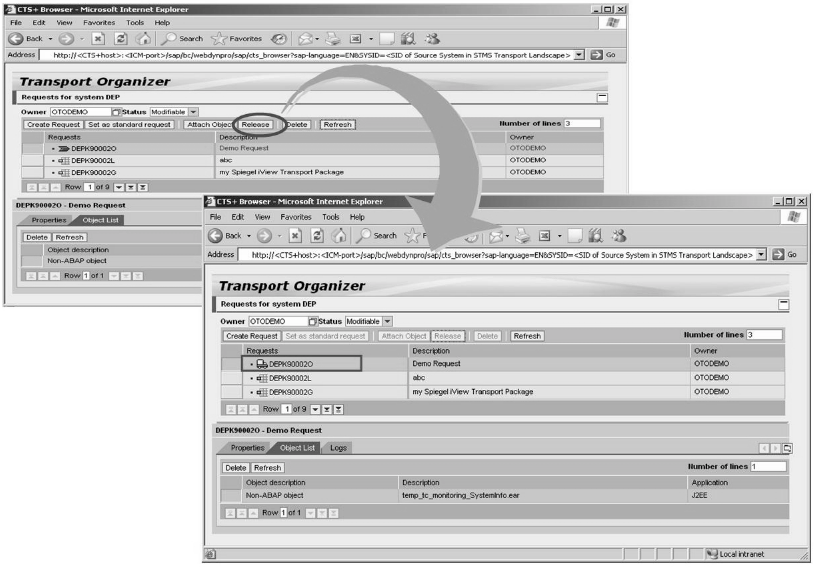
  Note: If no transport request is available that you can use, you have to create a new transport request either within the Transport Organizer – transaction SE09 (in case of development systems based on AS ABAP + Java) or with the help of the Transport Organizer Web UI (for development systems based on AS Java only).

  Several types of objects that can be attached to a change request. You can browse your file system and choose the file that you want to attach to the transport request. When you have found the file in question, click the *Attach* object button. To see the attached files, click the *Display* request button.

  Hint: Note that for portal transports, since SP stack 13 of SAP NetWeaver 7.0 it is also possible to attach portal files (for example epa files) directly to a transport request. In this case you do not need to browse the file system.

Releasing Transport Requests

After assigning Non-ABAP objects to a transport request, you can release this transport request. The release of a transport request starts the export of the files to file system level (transport directory). After releasing the transport request, this transport request is ready for import into a subsequent system (according to the definition of the transport landscape in transaction STMS on the transport domain controller).



Address: http://<CTS+host>:<ICM-port>/sap/bc/webdynpro/sap/cts_browser?sap-language=EN&SYSID=<SID of Source System in STMS Transport Landscape>

Transport Organizer

Requests for system DEP

Owner: OTODEMO Status: Modifiable

Number of lines: 3

Requests	Description	Owner
DEPK90002O	Demo Request	OTODEMO
DEPK90002L	abc	OTODEMO
DEPK90002G	my Spiegel iView Transport Package	OTODEMO

Row 1 of 9

DEPK90002O - Demo Request

Object description
Non-ABAP object

Row 1 of 1

Address: http://<CTS+host>:<ICM-port>/sap/bc/webdynpro/sap/cts_browser?sap-language=EN&SYSID=<SID of Source System in STMS Transport Landscape>

Transport Organizer

Requests for system DEP

Owner: OTODEMO Status: Modifiable

Number of lines: 3

Requests	Description	Owner
DEPK90002O	Demo Request	OTODEMO
DEPK90002L	abc	OTODEMO
DEPK90002G	my Spiegel iView Transport Package	OTODEMO

Row 1 of 9

DEPK90002O - Demo Request

Number of lines: 1

Object description	Description	Application
Non-ABAP object	temp_tc_monitoring_SystemInfo.ear	J2EE

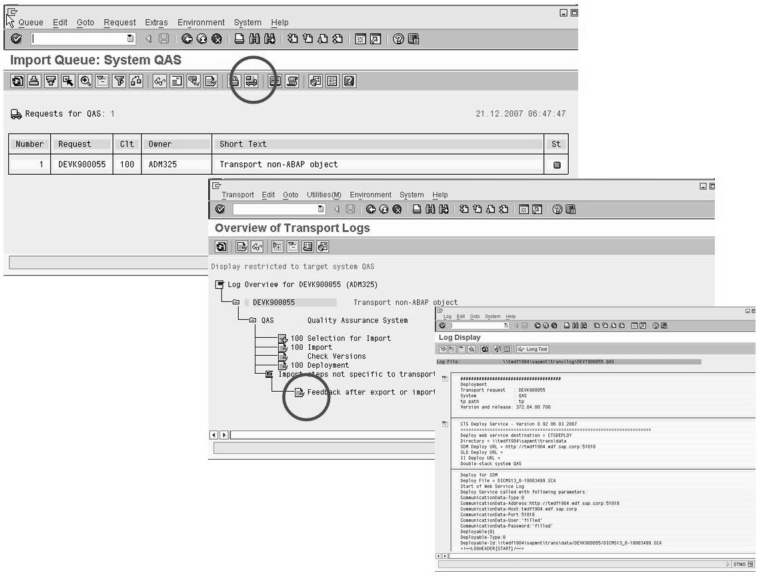
Row 1 of 1

Figure 129: Releasing Transport Requests

The release of the transport request <SID>K9<number> can either be done in transaction SE09 on the development system (in case of a development system based on AS ABAP + Java 7.0 SP Stack 12 or higher) or using the Web UI for the *CTS_BROWSER* Web Dynpro application (as shown in the figure above).

Importing Transport Requests into a Subsequent System

The import of transport requests into subsequent systems is done in transaction STMS_QUEUES (which is a subset of transaction STMS) on the transport domain controller (TDC) system.


Queue Edit Goto Request Extras Environment System Help
Transport Edit Goto Utilities(M) Environment System Help
Log Edit Goto System Help
Long Text
STMS

Import Queue: System QAS

Requests for QAS: 1 21.12.2007 06:47:47

Number	Request	Clt	Owner	Short Text	St
1	DEVK900055	100	ADM325	Transport non-ABAP object	■

Overview of Transport Logs

```
Display restricted to target system QAS
Log Overview for DEVK900055 (ADM325)
  DEVK900055            Transport non-ABAP object
    QAS        Quality Assurance System
          100 Selection for Import
          100 Import
              Check Versions
          100 Deployment
        Import steps not specific to transport
          Feedback after export or import
```

Log Display

```
Log File:           \\tedf1904\sapmnt\trans\log\DEVT900055.QAS

####################################
Deployment
Transport request   : DEVK900055
System              : QAS
tp path             : tp
Version and release: 372.04.08 700

CTS Deploy Service - Version 0.92 06.03.2007
================================================================
Deploy web service destination = CTSDEPLOY
Directory = \\tedf1904\sapmnt\trans\data
SDM Deploy URL = http://tedf1904.wdf.sap.corp:51018
SLD Deploy URL =
XI Deploy URL =
Double-stack system QAS

Deploy for SDM
Deploy File = DICMS13_0-10003499.SCA
Start of Web Service Log
Deploy Service called with following parameters:
CommunicationData-Type:0
CommunicationData-Address:http://tedf1904.wdf.sap.corp:51018
CommunicationData-Host:tedf1904.wdf.sap.corp
CommunicationData-Port:51018
CommunicationData-User:'filled'
CommunicationData-Password:'filled'
Deployable(0)
Deployable-Type:0
Deployable-Id:\\tedf1904\sapmnt\trans\data/DEVK900055/DICMS13_0-10003499.SCA
<!—LOGHEADER[START]/—>
```

During import, RDDIMDP triggers job RDDEPLOY

Figure 130: Importing Transport Requests into a Subsequent System

In transaction STMS_QUEUES, double-click the consolidation system in which you want to import the transport request (this is the system which was assigned as "target" in the workbench request before). Select the transport request in question and choose the *Import Request* button. After the import has finished, you may want to check the deployment protocol.

During the import, the transport control program *tp* triggers the import dispatcher *RDDIMPDP* that in turn triggers the job *RDDEPLOY*. This job opens a connection to the CTS Deploy Proxy. This connection needs to be kept open during the complete deployment process. The CTS Deploy Proxy finally calls the import tool on the AS Java side (for example the Software Deployment Manager, SDM).

Hint: This is why you need to maintain an ICM timeout parameter during the configuration of the enhanced capabilities of CTS.

As a result, the transport request is now listed in the import queue of the production system (according to the definition of the transport landscape in transaction *STMS → Transport Routes*).

Hint: Click the refresh button, if your import queue is not automatically refreshed.

Appendix: Transport for objects from SAP NetWeaver Development Infrastructure (NWDI)

The usage type Development Infrastructure (DI) is also referred to as SAP NetWeaver Development Infrastructure (NWDI). It provides an infrastructure for developing based applications on the SAP NetWeaver platform and is responsible for versioning, build and life cycle management of these applications.

The NWDI takes care of all parts of the development process in a project-specific way:

- Central source file management – in the Design Time Repository (DTR), a file storage in a database with export mechanisms that allow you to synchronize the instances of the DTR in a distributed development.
- Central build and archive management – in the Component Build Service (CBS), gives developers access to the latest archive versions in a central DB storage and a central build triggered by the developer.
- Central landscape and transport management – in the Change Management Service (CMS), gives administrators a central service to set up development landscapes for all development tasks and manage all transport processes for these tasks in the same UI.

All the development processes in the NWDI are based on SAP's component model, which enhances the public/private concept of Java by metadata regarding the use of objects without implying any changes to the development objects themselves: a Java interfaces stays an interface, a public class stays a public class. By clearly defining the visibility of and the dependencies between objects, the component model helps you structure applications into reusable components.

The figure below shows the integration of SAP NetWeaver Development Infrastructure with the enhanced Change and Transport System CTS.

Note: In this figure the scenario of SAP systems based on AS ABAP + Java is described, but the transport process is similar in case of SAP systems based on AS Java only.

This figure shows in the upper part the system landscape (as defined in transaction STMS). The runtime systems (based on AS ABAP + Java) are shown in the middle. In the lower part, the system landscape as defined in an NWDI track is shown.

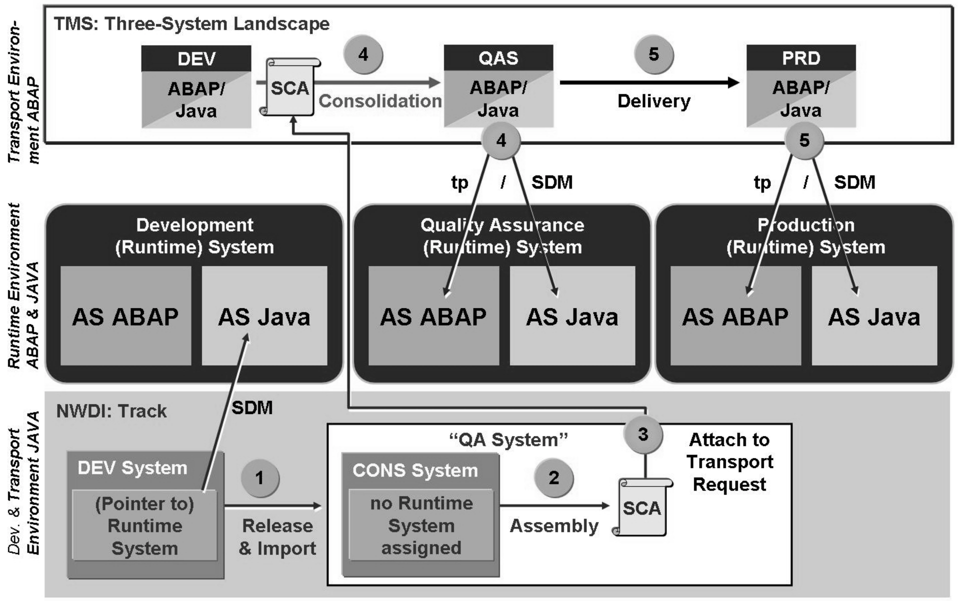

Transport Environ-ment ABAP
TMS: Three-System Landscape
DEV
ABAP/ Java
SCA
4
Consolidation
QAS
ABAP/ Java
5
Delivery
PRD
ABAP/ Java
4
tp / SDM
5
tp / SDM
Runtime Environment ABAP & JAVA
Development (Runtime) System
AS ABAP
AS Java
Quality Assurance (Runtime) System
AS ABAP
AS Java
Production (Runtime) System
AS ABAP
AS Java
Dev. & Transport Environment JAVA
NWDI: Track
SDM
DEV System
(Pointer to) Runtime System
1
Release & Import
"QA System"
CONS System
no Runtime System assigned
2
Assembly
SCA
3
Attach to Transport Request

Figure 131: NWDI-Integration – The basic idea

After the approval step (step 2) in the NWDI (which comes directly after the assembly step as there is no TEST runtime system configured), the SCA file can be handed over from the NWDI to the ABAP CTS (step 3).

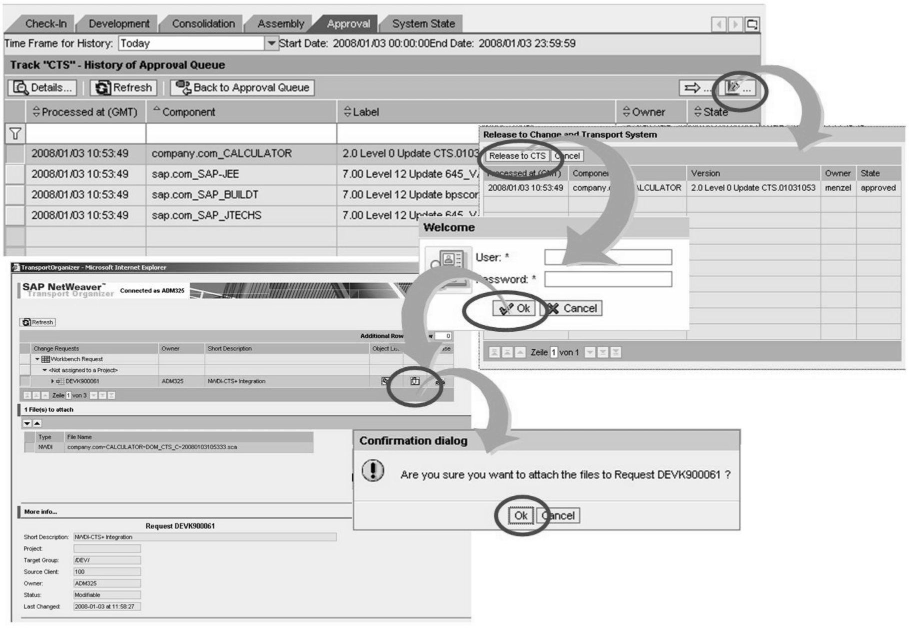
Next, the SCA file can be included into a transport request and can be imported into the QA system with transaction STMS (step 4). After the approval of these changes either using the QA approval procedure or the Change Request Management with SAP Solution Manager, the transport request can then be imported in the production system (step 5).

Before you can transport objects from NWDI, there must exist a transport request of type “workbench request”. This workbench request can be created in two different ways:

- You can create a transport request in transaction SE09 on the Transport Domain Controller (TDC) system. In this case the transport request is called <SID>K9<number> where <SID> denotes the SID of the TDC system.
- As of SAP NetWeaver 7.0 SP Stack 14 (for both the NWDI system and the TDC system) a transport request can be generated automatically when you forward an approved SCA file from NWDI to CTS. With SP Stack 14, the Configuration Adapter in the Visual Administrator will be replaced by the Destination Service that allows a more flexible configuration. As a prerequisite, the following tp parameters must be set for the source system on the transport domain controller (transaction STMS):

  – *WBO_GET_REQ_STRATEGY* = **`smart`**
  – *WBO_REL_REQ_STRATEGY* = **`auto`**

  These two parameters allow the automatic creation and automatic release of a transport request.



Time Frame for History: Today Start Date: 2008/01/03 00:00:00End Date: 2008/01/03 23:59:59

Track "CTS" - History of Approval Queue

Processed at (GMT)	Component	Label	Owner	State
2008/01/03 10:53:49	company.com_CALCULATOR	2.0 Level 0 Update CTS.0103		
2008/01/03 10:53:49	sap.com_SAP-JEE	7.00 Level 12 Update 645_V		
2008/01/03 10:53:49	sap.com_SAP_BUILDT	7.00 Level 12 Update bpscor		
2008/01/03 10:53:49	sap.com_SAP_JTECHS	7.00 Level 12 Update 645_V		

Release to Change and Transport System

Processed at (GMT)	Component	Version	Owner	State
2008/01/03 10:53:49	company.c…ALCULATOR	2.0 Level 0 Update CTS.01031053	menzel	approved

Zeile 1 von 1

User: *
Password: *

TransportOrganizer - Microsoft Internet Explorer

SAP NetWeaver™ Transport Organizer Connected as ADM325

Additional Row 0

Change Requests	Owner	Short Description
Workbench Request		
<Not assigned to a Project>		
DEVK900061	ADM325	NWDI-CTS+ Integration

Zeile 1 von 3

1 File(s) to attach

Type	File Name
NWDI	company.com~CALCULATOR~DOM_CTS_C~20080103105333.sca

More info...

Request DEVK900061

Short Description:	NWDI-CTS+ Integration
Project:	
Target Group:	/DEV/
Source Client:	100
Owner:	ADM325
Status:	Modifiable
Last Changed:	2008-01-03 at 11:58:27

Confirmation dialog

Are you sure you want to attach the files to Request DEVK900061 ?

The import into subsequent systems can be performed with transaction STMS on the Transport Domain Controller system as usual

Figure 132: Assign SCA to a transport request

In the *Approval history* of the CMS Transport Studio (on the NWDI system) you will see the icon *Release to CTS* . Select the SCA you want to release for transport using CTS and press the mentioned icon. A dialog-box comes up where you choose the *Release to CTS* button. On the next screen, you are asked to enter the TDC ABAP user and password for the target transport client and choose the *Ok* button. As a result, you get an overview of your Workbench requests and the file that you want to attach. Click the *Attach* button in the *Attach* column. When the file has been successfully attached, a message is displayed: "Attachment has been executed with success". You can also view the object list to see which files are attached to the selected transport request.

Hint: If you encounter sharing/mounting problems, you may copy the SCA file to the file system of the TDC server and to use transaction SE09 on the TDC host to include a non-ABAP object.

After assigning an SCA file to a transport request, you can release the transport request using the *Release* button. After releasing the transport request, this request appears in the import queue of the subsequent system. It can be imported to the subsequent system with transaction STMS as usual for ABAP transport requests.

Note: If you have created the transport request manually with the help of the Transport Organizer Web UI on the Transport Domain Controller system, you can also release the transport request from there.

Hint: Of course, you need to configure the NWDI system for the integration with the CTS system first. To do this, you need to maintain connection data to the TDC system and to the Java Development System. These settings can be done with the help of the Visual Administrator on the NWDI system. Start the Visual Administrator on the NWDI server and navigate to the Configuration Adapter (area *Server → Services → Configuration Adapter*). Within the tab *Runtime* choose *Configurations → apps → sap.com → tc~com.sap.tmw.clients~_ctsa_manager → appcfg → Propertysheet application.global.properties* and maintain the conifiguration data (in change mode).

- For the *target.url* property enter the value **`http://<TDC host>:<ICM-port>/sap/bc/srt/rfc/sap/SI_CORE_WBO_API?sap-client=<client>`**
- For the *tms.system* value enter the logical development system (as defined in the STMS transport routes screen).

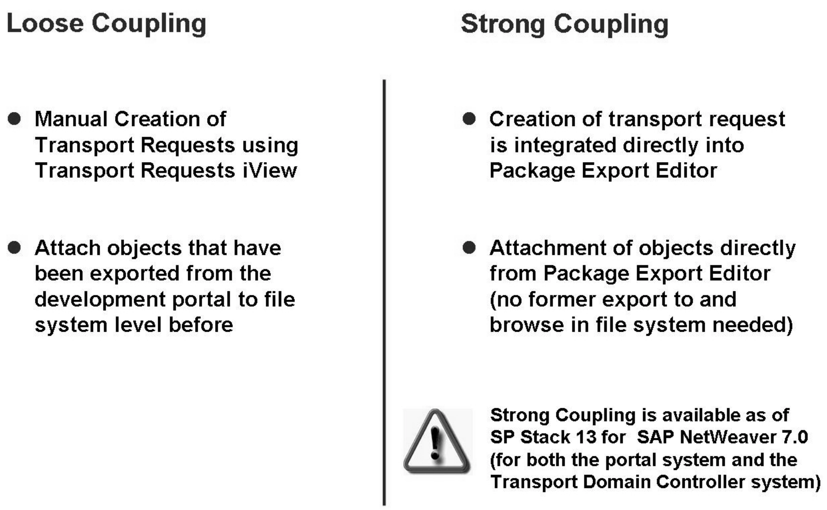
This hint also applies for the configuration of SAP NetWeaver Portal systems for the use with the enhanced capabilities of CTS.

Note that with SP Stack 14 of SAP NetWeaver 7.0, the *Configuration Adapter* will be replaced by the *Destination Service* that allows a more flexible configuration of the integration between the NWDI / SAP NetWeaver Portal systems and the enhanced CTS.

Appendix: Transport for Systems based on Usage Type EP

As of SP Stack 13 for SAP NetWeaver 7.0 (for the system based on usage type EP) a strong coupling between the Transport Domain Controller (TDC) system and the EP system is available. Strong coupling means that a transport request can be created directly from the package editor. In addition objects (for example epa files that contain iViews) can be attached directly to this transport request within the Package Explorer.

For EP systems that are based on SAP Web AS 6.40 or SAP NetWeaver 7.0 with SP Stack 12 or lower, you can only configure a loose coupling between the portal system and the TDC system. Loose coupling means that transport requests need to be created manually. In addition, non-ABAP objects need to be exported to file system level first before they can be attached to the transport request.



Loose Coupling

- Manual Creation of Transport Requests using Transport Requests iView
- Attach objects that have been exported from the development portal to file system level before

Strong Coupling

- Creation of transport request is integrated directly into Package Export Editor
- Attachment of objects directly from Package Export Editor (no former export to and browse in file system needed)

Strong Coupling is available as of SP Stack 13 for SAP NetWeaver 7.0 (for both the portal system and the Transport Domain Controller system)

Figure 133: Portal integration: Loose Coupling vs. Strong Coupling

For strong coupling, the following (optional) tp parameters can be set for the source system on the transport domain controller (transaction STMS). This two parameters allow the automatic creation and automatic release of a transport request:

- *WBO_GET_REQ_STRATEGY* = `smart`
- *WBO_REL_REQ_STRATEGY* = `auto`

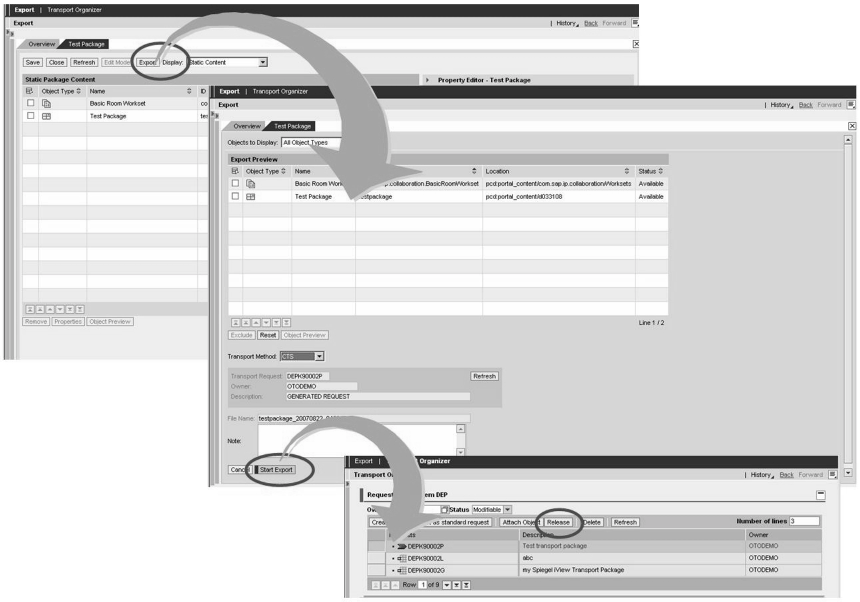
In the following let us consider the strong coupling scenario.

With strong coupling, it is possible to create and release transport requests automatically and directly from the Package Export Explorer in the development portal. In this scenario, you need to create an object that can be transported within the portal system (transport package). Then you include your changes to the portal objects in question in this transport package.

In this case you need to select the Transport Method “CTS”. If you press *Start Export* the transport request will be released and therefore will be written on file system level (transport directory) so that is ready for import into subsequent systems.

Note: *Start Export* does not start the export (copy of files into transport directory) but means “attach objects to transport request”.

Finally you need to release the transport request manually – unless the tp parameter *WBO_REL_REQ_STRATEGY* is not set to **`auto`**.



Display: Static Content

Static Package Content

Object Type	Name	ID
☐	Basic Room Workset	co
☐	Test Package	tes

Objects to Display: All Object Types

Export Preview

Object Type	Name		Location	Status
☐	Basic Room Work	.collaboration.BasicRoomWorkset	pcd:portal_content/com.sap.ip.collaboration/Worksets	Available
☐	Test Package	estpackage	pcd:portal_content/d033108	Available

Line 1 / 2

Transport Method: CTS

Transport Request: DEPK90002P
Owner: OTODEMO
Description: GENERATED REQUEST

File Name: testpackage_20070822_

Note:

Transport O

Request em DEP

Ow Status Modifiable

Number of lines 3

	sts	Description	Owner
	DEPK90002P	Test transport package	OTODEMO
	DEPK90002L	abc	OTODEMO
	DEPK90002G	my Spiegel iView Transport Package	OTODEMO

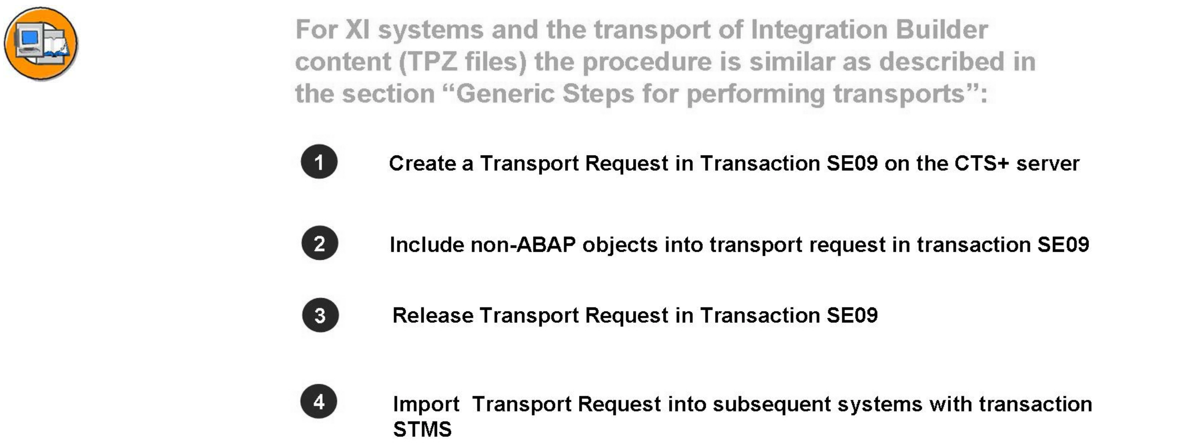
Row 1 of 9

Figure 134: Strong Coupling: Create Transport Requests and attach Transport Packages

To import the transport request into a subsequent system, you can use transaction STMS_QUEUES on the TDC system.

Appendix: Transport for Systems based on Usage Type PI

As SAP Exchange Infrastructure systems contains both an AS ABAP and an AS Java part, you can use transaction SE09 (of the AS ABAP part) to create and release transport requests and use transaction STMS to import the change requests into subsequent systems.



For XI systems and the transport of Integration Builder content (TPZ files) the procedure is similar as described in the section “Generic Steps for performing transports”:

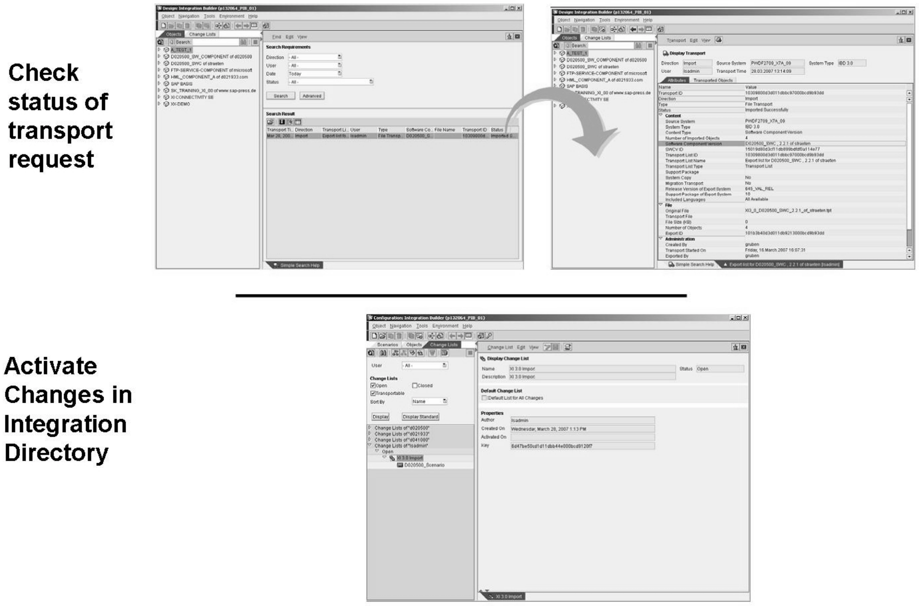
1. Create a Transport Request in Transaction SE09 on the CTS+ server
2. Include non-ABAP objects into transport request in transaction SE09
3. Release Transport Request in Transaction SE09
4. Import Transport Request into subsequent systems with transaction STMS

Figure 135: Transport for Systems based on Usage Type PI: General Procedure

Exchange Infrastructure (XI) systems can play the role of the CTS system to transport non-ABAP as well (as they are based on AS ABAP + Java). As of SP Stack 13 of SAP NetWeaver 7.0 for the CTS system it is possible to include TPZ files directly from the XI application itself. In this case you do not need to include the non-ABAP object manually in transaction SE09. As a prerequisite the following tp parameters must be set on the Transport Domain Controller (Section *System Overview*) for the XI development system (double click on the system in question and switch to the *Transport Tool* tab):

- *WBO_GET_REQ_STRATEGY* = `smart`
- *WBO_REL_REQ_STRATEGY* = `auto`

Hint: As of SP Stack 12 of SAP NetWeaver 7.0 (for the XI systems) it is no longer necessary to create separate system landscapes for the AS ABAP and the AS Java part of the XI. Therefore the workbench requests in question can contain both ABAP and non-ABAP objects.



Check status of transport request

Activate Changes in Integration Directory

Figure 136: Activate Changes in the Integration Directory

After the import into a subsequent system, you can check the status of your XI transports in the receiver system by using the Integration Builder for the Integration Repository or the Integration Directory.

In order to do this, choose *Tools* → *Find Transports*, define your search criteria and choose *Search*. To access the details of the transport (*Attributes* and *Transported Objects* tab pages), double-click the transport line.

Hint: If your system is based on SAP NetWeaver 7.0 SP Stack 13 or lower, you must activate the changes manually in the Integration Directory. If your system is based on SP Stack 14 or higher, this activation will be done automatically.

Note: You will find the objects for the service user that you have created in the XI system before (for example user *NWDI_CTSADM*). This user must be assigned to the *SAP_XI_CMS_SERV_USERS* role.

Lesson Summary

You should now be able to:

- Outline the process of transporting non-ABAP objects with the help of ABAP Change and Transport System

Related Information

Further Information on the enhancements of the ABAP Change and Transport System can be found on SDN in the area *SDN Community → SAP NetWeaver Capabilities → Lifecycle Management → Software Logistics → Change and Transport System* in the *Knowledge Center → Software Change Management* area (https://www.sdn.sap.com/irj/sdn/cts)

Unit Summary

You should now be able to:

- Describe the transport flow within a three-system landscape
- Outline the transport process
- Identify the tools used to perform imports using the Transport Management System (TMS)
- Explain the relationship between import queues and import buffers
- Follow the procedure for setting up the date and time for an import
- Decide whether to import single transport requests or to import the entire import queue
- List the different transport strategies
- Define Quality Assurance (QA) procedure
- Describe a situation in which using special transport workflow would be practical
- Explain why the transport proposal is necessary
- Identify the steps of the import process
- Identify the relationship between tp, R3trans and RDDIMPDP
- Identify the tools for monitoring transport activity
- Identify the way to ensure a stable development environment
- Follow the steps for cleaning up a transport directory
- Outline the process of transporting non-ABAP objects with the help of ABAP Change and Transport System

Test Your Knowledge

1. Using TMS, you can import all transport requests listed in the production system import queue in the given sequence.

   Determine whether this statement is true or false.

   □ True

   □ False

2. Describe the relationship between import queues and import buffers.

   ____

   ____

   ____

   ____

3. SAP recommends that you reject transport requests containing errors.

   Determine whether this statement is true or false.

   □ True

   □ False

4. State how to create a transport proposal.

   ____

   ____

   ____

   ____

5. Which additional tools are called by tp to perform all import steps.

   Choose the correct answer(s).

   □ A tp_abap.exe

   □ B RDDIMPDP

   □ C R3trans

   □ D R3import

6. To execute transport steps, tp uses the tables _____________ and _____________ to communicate with the various ABAP programs.

   Fill in the blanks to complete the sentence.

7. To display tp systemlog, you have to log on to operating system and navigate in transport directory */usr/sap/trans/log*.

   Determine whether this statement is true or false.

   □ True

   □ False

8. SAP recommends transporting each object as soon as it is syntax error free.

   Determine whether this statement is true or false.

   □ True

   □ False

9. The command ___________________ searches for transport requests that are not marked for import into any SAP system.

   Fill in the blanks to complete the sentence.

10. Which statements concerning the transport of non-ABAP obejcts with the help of the ABAP Change and Transport System (CTS) are correct?

    Choose the correct answer(s).

    □ A You must use Customizing requests to transport non-ABAP objects.

    □ B You must use Workbench requests to transport non-ABAP objects.

    □ C Transport requests for non-ABAP objects can be created using the CTS_BROWSER Web Dynpro application.

    □ D Transport Requests for non-ABAP objects can be created using transaction SE09 on the transport domain controller.

Answers

1. Using TMS, you can import all transport requests listed in the production system import queue in the given sequence.

   Answer: True

   Using TMS, you can import all transport requests listed in the production system import queue in the given sequence.

2. Describe the relationship between import queues and import buffers.

   Answer: The import queue in the SAP System represents the import buffer located on the operating system level. Import queues display the requests that are to be imported to the target system, in the order of their release.

3. SAP recommends that you reject transport requests containing errors.

   Answer: False

   SAP does not recommend the rejecting of transport requests containing errors, but instead of correcting them using subsequent transport requests and approving the affected transport requests as an entire package.

4. State how to create a transport proposal.

   Answer: To create a transport proposal, go to the Change and Transport Organizer and select released requests. Choose Display. Position the cursor on a request and choose *Utilities(M) → Create transport proposal*. Enter a short text, the target system and you can other transport requests.

5. Which additional tools are called by tp to perform all import steps.

   Answer: B, C

   RDDIMPDP is used for communication with ABAP. R3trans is used for communication with database.

6. To execute transport steps, tp uses the tables TRBAT and TRJOB to communicate with the various ABAP programs.

   Answer: TRBAT, TRJOB

7. To display tp systemlog, you have to log on to operating system and navigate in transport directory */usr/sap/trans/log*.

   Answer: False

   To display tp systemlog, choose*STMS → Overview → Imports → Goto → Import Monitor* or use transaction *AL11*

8. SAP recommends transporting each object as soon as it is syntax error free.

   Answer: False

   SAP recommends importing a whole change project when all objects are in an acceptable state rather than each object as soon as it is ready.

9. The command tp check all searches for transport requests that are not marked for import into any SAP system.

   Answer: tp check all

10. Which statements concerning the transport of non-ABAP obejcts with the help of the ABAP Change and Transport System (CTS) are correct?

    Answer: B, C, D

    You can only use Workbench requests to transport non-ABAP objects with the help of the enhancements of CTS. Transport requests can be created both with transaction SE09 and using the Web Dynpro application. If you use SE09 on the Transport Domain Controller, the transport request is named TDCK9xxxxx where TDC is the System ID of the transport domain controller. If you use the CTS_BROWSER Web Dynpro application, the transport request can also be named JADK9xxxx where JAD is the System ID of your Java Development system.

Unit 5

Client Tools

Unit Overview

In this unit, you will learn about the concepts and functions of client tools. You will also identify methods to handle errors and limitations of the tools. Finally, you will learn to compare customizing settings across SAP clients and systems and learn how to delete clients in an SAP system.

Unit Objectives

After completing this unit, you will be able to:

- Describe client copy and client transport tools
- Perform a client copy
- Troubleshoot client copy and client transport errors
- Compare and maintain clients

Unit Contents

Lesson: Client Copy and Client Transport Tools

Lesson Overview

In this lesson, you will get an overview of the client tools available in an SAP system landscape. You will learn the processes of client copy and client transport. You will explore system requirements for processing a client copy.

Lesson Objectives

After completing this lesson, you will be able to:

- Describe client copy and client transport tools
- Perform a client copy
- Troubleshoot client copy and client transport errors

Business Example

Several clients are required during the implementation of an SAP system. Your job as a system administrator is to create and set up clients and then manage all the clients across the SAP system landscape.

Using Client Copy and Transport Tools

Client copy and transport tools are available.

- **The client copy and client transport tools:**
  - **Are designed to replace the data of one client with data from another client**
  - **Are not designed to combine the data of different clients**
- **Client data is categorized as either:**
  - **Client dependent customizing data**
  - **User master data**
  - **Application data as defined by the data table's delivery class**
- **Application data can only be copied with the corresponding customizing data**

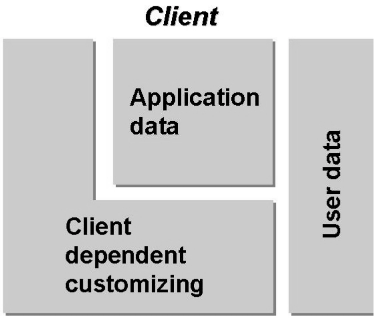


Figure 137: Client Tools: Overview

Profiles define the tables to be copied through the delivery class of the table. Special data such as user master data can be selected by specifying criteria. These table delivery classes are explained later in this lesson.

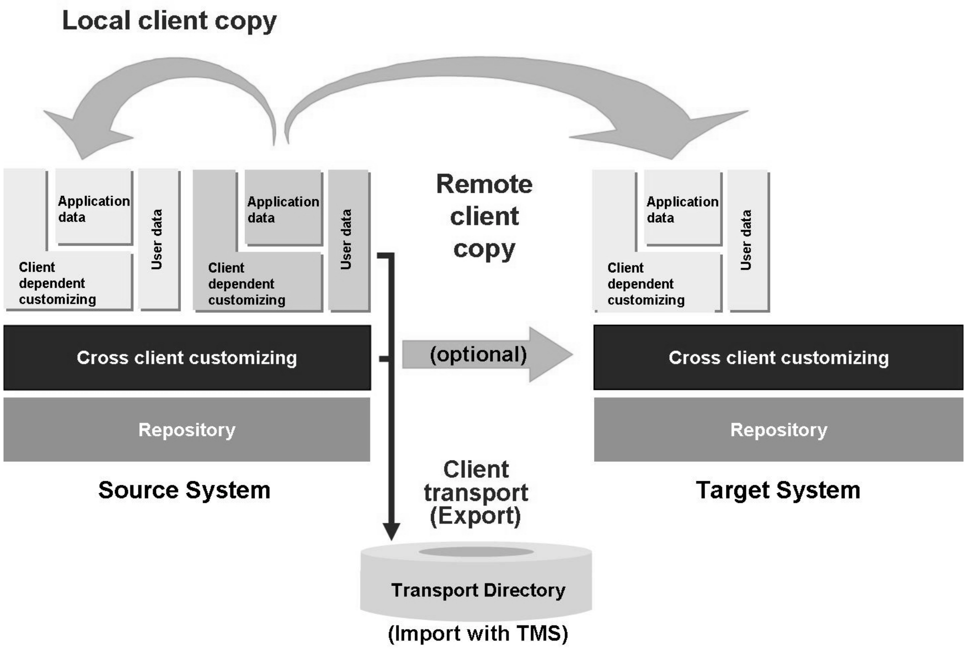


Figure 138: Client Copy and Transport Tools

The client copy tools can copy the following components of the source client into the target client:

- User master data: user master data is only deleted in the target system if a profile with user master data is copied. Authorization profiles and roles belong to Customizing and are therefore always copied with it. Copying users without authorization profiles is problematic. The copy profile *SAP_USER* therefore contains authorization profiles and roles.
- Customizing: all profiles except *SAP_USER* contain customizing. Customizing data is generally in tables with delivery classes *C*,*G*,*E* and *S*.
- Cross-client Customizing: choose a profile with this data type for example when you want to create a test system which is analogous to the production system, and no system copy is to be made.
- Master/transaction (application) data: select this option, for example, if you want to set up a test system from the production client.

Note: Application data depends on customizing data, so it can only exist consistently together with it. Application data is therefore always deleted from the target client, exception for copies with *SAP_USER*. Application data is generally in tables with delivery class *A*.

If you have maintained a logical system in each of the source and target clients and assigned it to the client, the logical system names in the target client application data are automatically converted after copying the application data (e.g. with copy profile *SAP_ALL*). See transaction BD54.

Testing Customizing Change Requests

Transport requests for customizing are usually created in the development system. They are then released and imported into the quality assurance system. It can be a good idea to additionally perform some pre-testing in a separate client of the development system. This should be done before releasing the transport request.

Before releasing a customizing transport request, perform a unit test to:

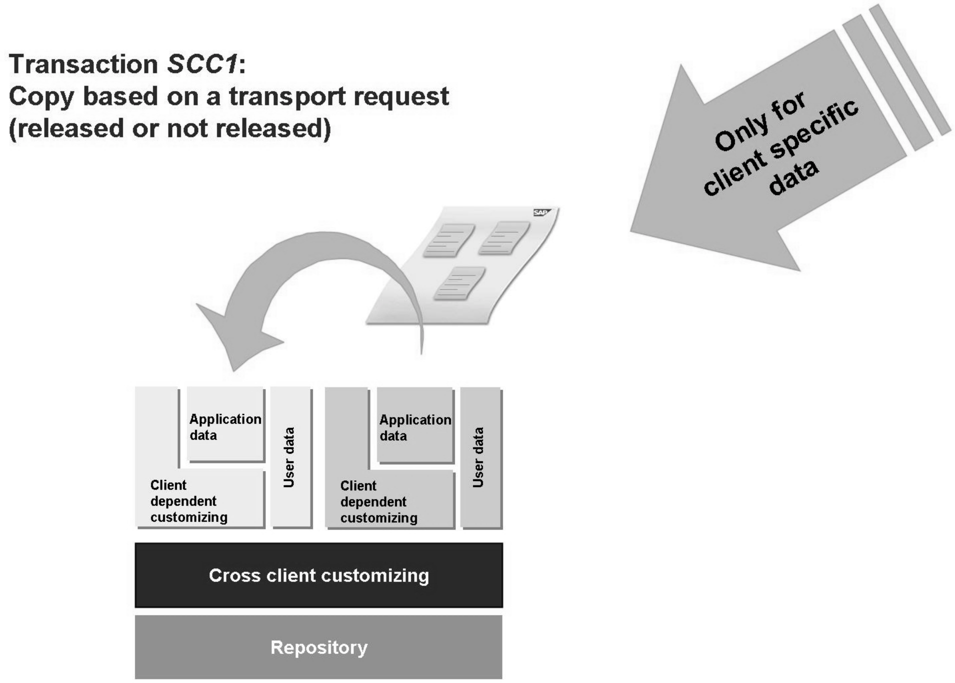
- Test the functionality of the transport request.
- Verify that the content of the transport request is complete.

Maintaining a separate client for testing allows:

- True unit testing
- Maintenance of test data without the risk of creating customizing-dependent data

Transaction SCC1 copies changes from one client to another based on:

- A task
- A transport request
- A transport request and its tasks



Transaction *SCC1*:
Copy based on a transport request
(released or not released)

Only for client specific data
Application data
User data
Client dependent customizing
Application data
User data
Client dependent customizing
Cross client customizing
Repository

Figure 139: Testing Customizing Transport Requests

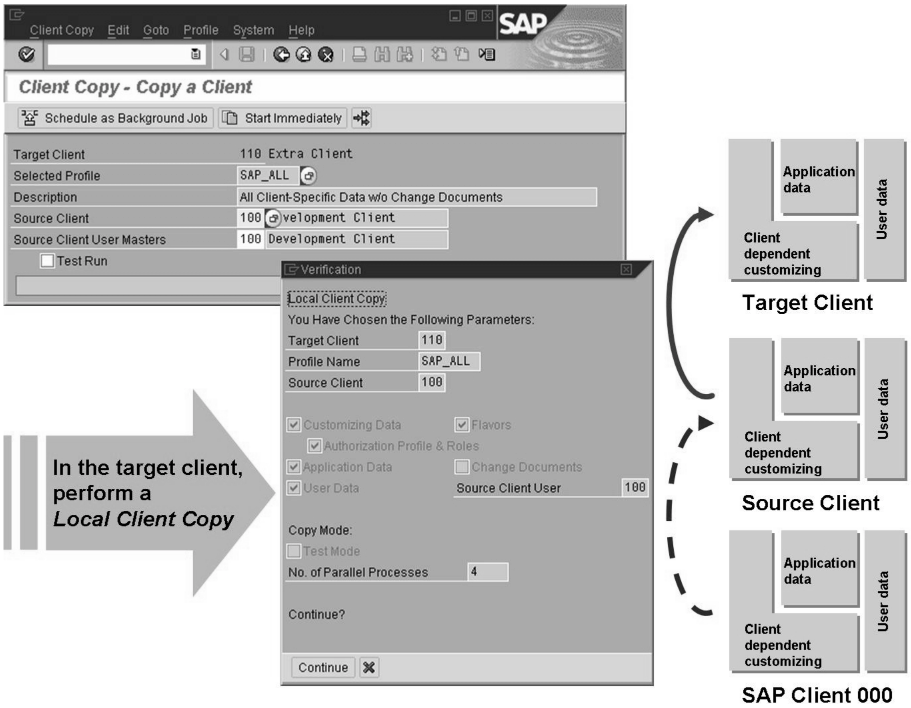
To copy the contents of a transport request from one client to another client, use transaction SCC1 from the target client. Enter the source client and the transport request to be copied.

Only one transport request may be copied at a time. As copying each transport request individually may be time consuming, use the *Include Objects* option in transaction SE09 to bundle several transport requests into one transport request. This larger transport request can then be copied to the target client, saving time.

Note: If the transport request contains cross-client objects, these objects are not copied, of course.

Local Client Copy

A local client copy copies between clients within the same SAP system.


Client Copy Edit Goto Profile System Help
SAP
In the target client, perform a *Local Client Copy*
Application data
User data
Client dependent customizing
Target Client
Application data
User data
Client dependent customizing
Source Client
Application data
User data
Client dependent customizing
SAP Client 000

Client Copy - Copy a Client

Schedule as Background Job | Start Immediately

Target Client 110 Extra Client

Selected Profile SAP_ALL

Description All Client-Specific Data w/o Change Documents

Source Client 100 velopment Client

Source Client User Masters 100 Development Client

☐ Test Run

Verification

Local Client Copy

You Have Chosen the Following Parameters:

Target Client 110

Profile Name SAP_ALL

Source Client 100

☑ Customizing Data ☑ Flavors

☑ Authorization Profile & Roles

☑ Application Data ☐ Change Documents

☑ User Data Source Client User 100

Copy Mode:

☐ Test Mode

No. of Parallel Processes 4

Continue?

Continue ✖

Figure 140: Local Client Copy

Both local and remote client copy must be initiated from the target client using the following steps:

1. To create an entry for the target client in the client maintenance table, choose transaction SCC4.
2. Log on to the target client as **`SAP*`** with the initial password **`pass`**. Perform the client copy using transaction SCCL. Select the data to be copied using a profile.

   Note: The user name *SAP** with the password *pass* is only active if the profile parameter *login/no_automatic_user_sapstar* does not have the value *1*.

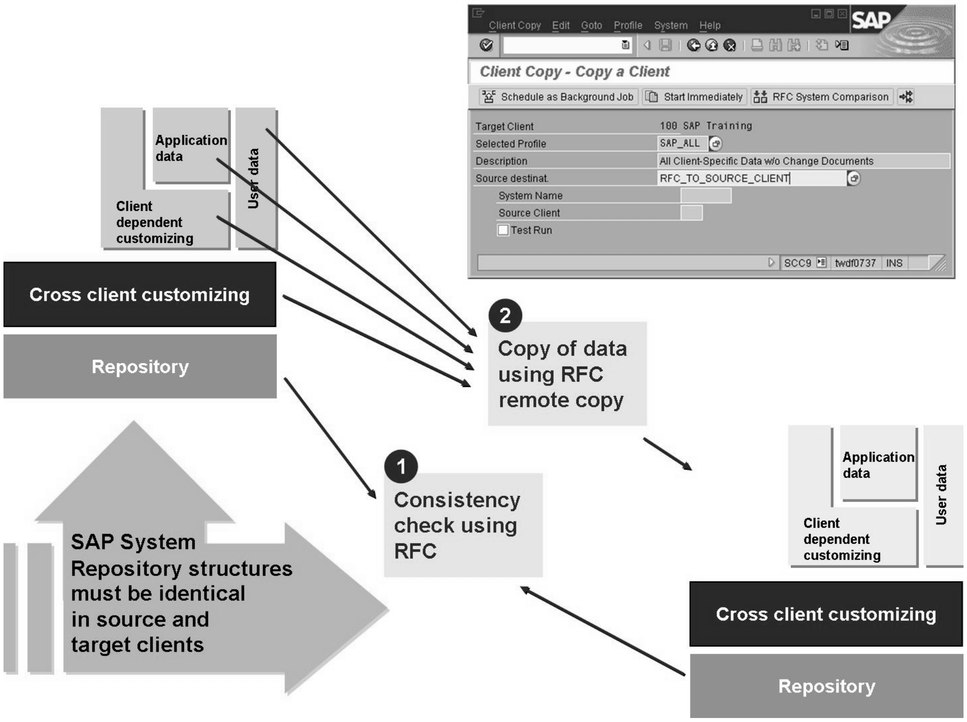
3. Assign the source client for Customizing data, application data, and user master records.
4. Begin the client copy. Because copying is a lengthy process, use background processing.

Caution: To ensure data consistency, you may not log on to the target client during a client copy. SAP also recommends that you do not work in the source client during a client copy.

Note: SAP delivers the software with standard client 000. You may not work in client 000. But you may use client 001 as your productive, if it exists. Client 001 is a copy of client 000. However, if no client 001 exists, SAP recommends that you begin SAP system implementation by creating a new client as a copy of client 000.

You can improve the performance of the client copy, for example, by excluding tables or packages, with *Edit -> Expert Settings*. You can exclude tables from the client copy, for example if they are not relevant for the target client, in the *Tables* tab. For further information about excluding tables from the client copy, choose the *Help* button in the expert settings. For further information about expert settings in the client copy, see the SAP note 446485.

Remote Client Copy


Client Copy Edit Goto Profile System Help
SAP
Schedule as Background Job
Start Immediately
RFC System Comparison
SCC9
twdf0737
INS
Application data
User data
Client dependent customizing
Cross client customizing
Repository
2
Copy of data using RFC remote copy
1
Consistency check using RFC
SAP System Repository structures must be identical in source and target clients
Application data
User data
Client dependent customizing
Cross client customizing
Repository

Client Copy - Copy a Client

Target Client 100 SAP Training
Selected Profile SAP_ALL
Description All Client-Specific Data w/o Change Documents
Source destinat. RFC_TO_SOURCE_CLIENT
System Name
Source Client
□ Test Run

Figure 141: Remote Client Copy

A remote client copy allows you to copy data between clients in different SAP systems. You can use a remote client copy to, for example, copy client-dependent and client-independent customizing data between SAP systems.

A remote client copy proceeds in the same way as a local copy, but sends the data through a remote function call (RFC) connection to the target client.

A remote client copy is easy to use and does not require file system space on operating system level. The limitations of a remote client copy are as follows:

- A remote client copy does not create a file at the operating system level, so there is no **hard copy** of the client to be copied. Therefore, the same, identical client copy cannot be duplicated at a later time.
- Starting with release 4.5B, cross-client customizing objects can be copied during a remote client copy, depending on the client copy profile.
- To be able to copy all data during the client copy, the structures of all copied tables in both systems must be identical. During remote client copy, an automatic repository consistency check is performed. If inconsistencies are detected, the client copy is terminated and an error message is displayed. You can also perform the consistency check yourself with the *RFC System Comparison* push-button.

If you experience problems copying from another system, see the SAP note 557132.

For more information about remote client copies, see also SAP Note 47502 at the SAP Service Marketplace, http://service.sap.com/notes.

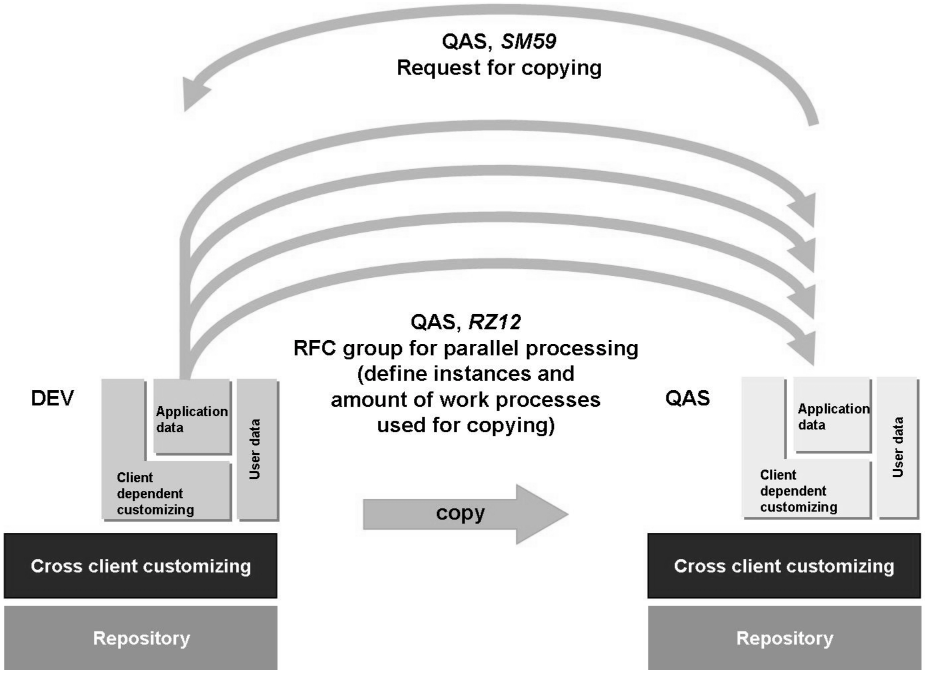


Figure 142: Client Copy Using Parallel Processing

The client copy function can be used with multiple parallel processes, which speeds up the copying process. Other factors that should be taken into consideration to optimize a client copy include the network performance and the database performance.

The client copy processes are generated at runtime. They are parallel RFC processes, managed by server groups. You must specify the maximum number of processes you want to use in the client copy program. Parallel processing can only be used when performing a local or remote client copy, it cannot be used for client transports.

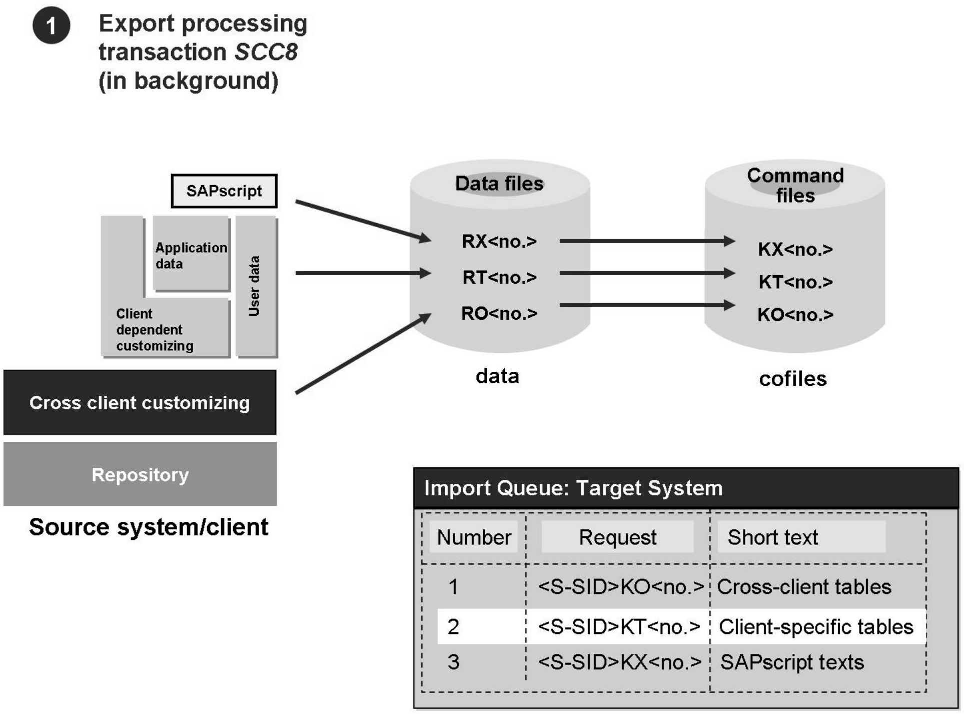
To specify the maximum number of client copy processes, use transaction SCCL for a local copy or transaction SCC9 for a remote copy and then *Goto* → *Parallel Processes*.

To define a server group, use transaction SM59, *RFC* → *RFC Groups*. For RFC groups, the limitations on the system parameters are explained in SAP Note 99284 in SAP Service Marketplace.

During the execution of a client copy using parallel processing, each parallel process checks the other to see if it is still running. If a process has failed, it will be restarted automatically.

Hint: Remember: in an existing client, application data, client-specific Customizing, client-independent Customizing and the Repository rely on each other! So even when including the client-independent Customizing: a remote client copy is only 100% consistent, if the complete Repository of the source and target system is identically. This will never be the case in “real” systems!

Client Transport



Import Queue: Target System

Number	Request	Short text
1	<S-SID>KO<no.>	Cross-client tables
2	<S-SID>KT<no.>	Client-specific tables
3	<S-SID>KX<no.>	SAPscript texts

Figure 143: Client Transport: Export Process

A **client transport** differs from a client copy in that it does not use RFC. Like a client copy, however, a client transport is used to copy data between different clients. A client transport can be used to replace a local or remote client copy.

A client transport consists of two steps: First a client export extracts client data from the source client to files at the operating system level. Then the data is imported from the operating system files into the target client.

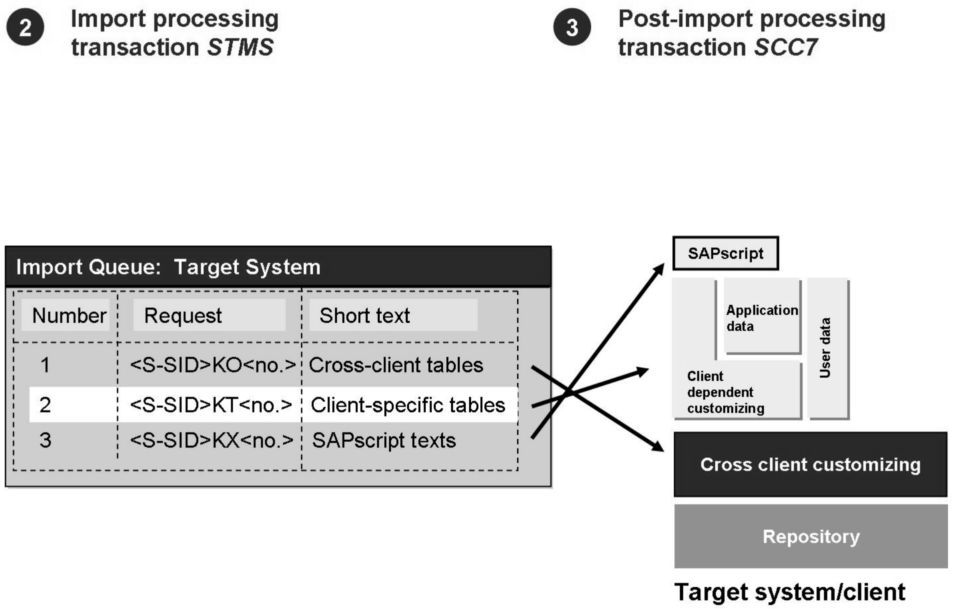
To perform a client export, follow these steps:

1. Log on to the source client. Choose transaction SCC8 (client export). Select the data to be copied using a profile.
2. Indicate the target system to which the client will be copied. The target system must be defined in TMS as part of the transport domain.
3. Begin the client export. Because copying is a lengthy process, use scheduled background processing.

The client export performed in the source system <*SID*> exports the client data asynchronously by calling the transport program *tp* at the operating system level. This export process will generate up to three data files at the operating system level:

- *RO< number >*: this file contains client-independent data
- *RT< number >*: this file contains client-specific data
- *RX< number >*: this file contains SAPscript texts

Note: The data export is asynchronous and still runs if transaction SCC8 has already finished. Do not run any other client copy tool before the data export is finished. You can check the status of the export with the transaction SE01. Display the logs for the request <*SID*>*KT*<*NO*>.



2 Import processing transaction *STMS*

3 Post-import processing transaction *SCC7*

Import Queue: Target System

Number	Request	Short text
1	<S-SID>KO<no.>	Cross-client tables
2	<S-SID>KT<no.>	Client-specific tables
3	<S-SID>KX<no.>	SAPscript texts

Figure 144: Client Transport: Import Process

Depending on the type of data selected through the client transport profile, the client copy command files added to the buffer of the target system are as follows:

- *<SID>KO<number>*: this file is for client-independent data
- *<SID>KT<number>*: this file is for client-specific data
- *<SID>KX<number>*: this file is SAPscript texts

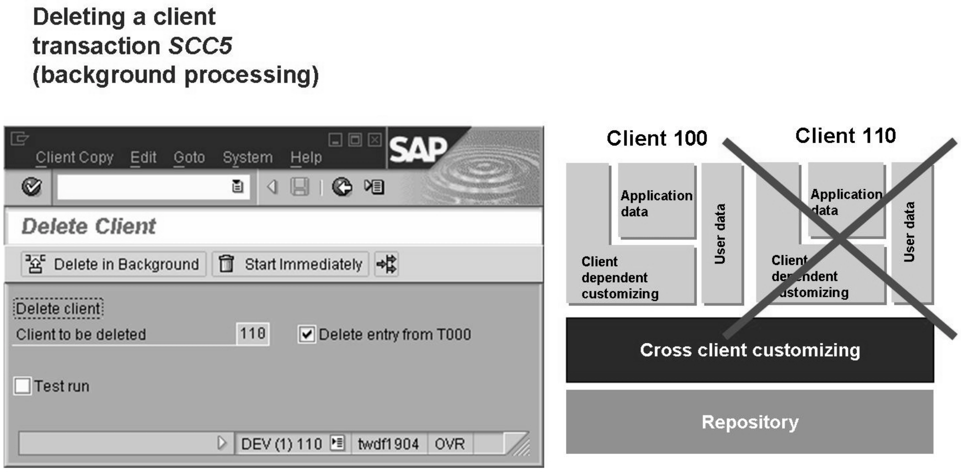
The client export transport requests are not imported when an *Import all* takes place. You can import these requests into the target client using TMS. Choose one of the transport requests of the client transport in the Transport Management System (TMS). The other transport requests belonging to this client transport are then automatically added in the correct order.

After the import process has completed, post-import activities are required for object generation steps. After completing the import, log on to the target client. Choose transaction SCC7.

To display client transport logs, use the Transport Organizer (transaction SE09).

During client transport, a repository consistency check can be performed by clicking the *RFC system check* button in transaction SCC8. If inconsistencies are detected, a list of the ABAP Dictionary table definitions missing in the target system is generated. This will help you recognize in advance formal problems that may occur during the import of the source data.

Client Delete



Deleting a client
transaction *SCC5*
(background processing)

Delete Client

Delete client
Client to be deleted 110 ☑ Delete entry from T000

□ Test run

Figure 145: Client Delete

To delete a client from within an SAP system:

- Log on to the client to be deleted.
- Use transaction SCC5.
- Start the deletion of the client, preferably using background processing.

When you delete a client entry from table *T000* with client maintenance, transaction SCC4, you can no longer log on to the client or update it using transport requests. This, however, does not eliminate the data that belongs to the client. This means the client-specific data remains in your SAP system, occupying space in the database. Therefore, to eliminate an SAP client entirely, that is, to delete both the client and the client-specific data, use the client delete function, transaction SCC5.

Deleting a client entry with client maintenance, transaction SCC4, allows you to temporarily lock the client. This deletion procedure preserves the data for the client but prevents users from logging on to the client or accessing the data belonging to the client. To restore the client and allow logon, recreate the client entry using client maintenance.

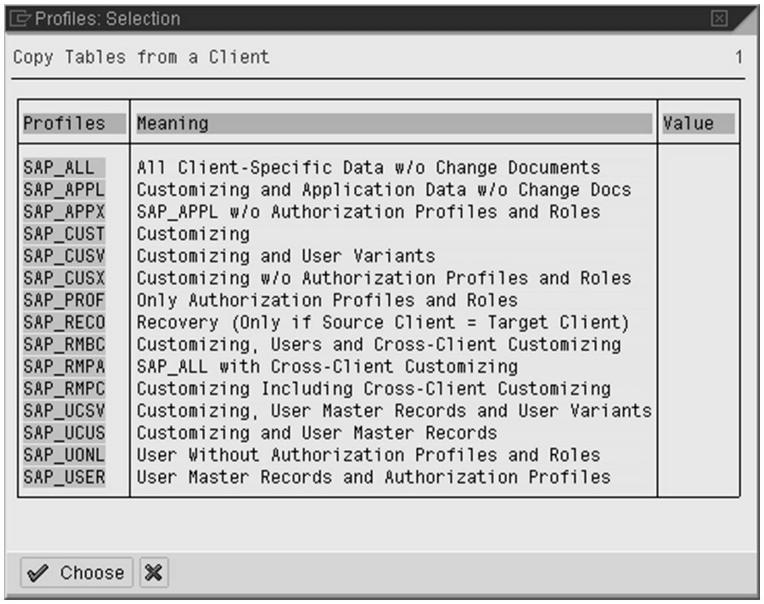
The amount of time required for the deletion of a client can be reduced by performing the deletion using parallel processes.

Note: In most databases, space which has become free is only available after a reorganization.

Client Copy Profiles

SAP delivers all necessary profiles to be used by a client copy or a client transport.

Starting with release 4.5 roles (the former *activity groups*) are also copied with the user master records using the profile *SAP_USER*.



Copy Tables from a Client 1

Profiles	Meaning	Value
SAP_ALL	All Client-Specific Data w/o Change Documents	
SAP_APPL	Customizing and Application Data w/o Change Docs	
SAP_APPX	SAP_APPL w/o Authorization Profiles and Roles	
SAP_CUST	Customizing	
SAP_CUSV	Customizing and User Variants	
SAP_CUSX	Customizing w/o Authorization Profiles and Roles	
SAP_PROF	Only Authorization Profiles and Roles	
SAP_RECO	Recovery (Only if Source Client = Target Client)	
SAP_RMBC	Customizing, Users and Cross-Client Customizing	
SAP_RMPA	SAP_ALL with Cross-Client Customizing	
SAP_RMPC	Customizing Including Cross-Client Customizing	
SAP_UCSV	Customizing, User Master Records and User Variants	
SAP_UCUS	Customizing and User Master Records	
SAP_UONL	User Without Authorization Profiles and Roles	
SAP_USER	User Master Records and Authorization Profiles	

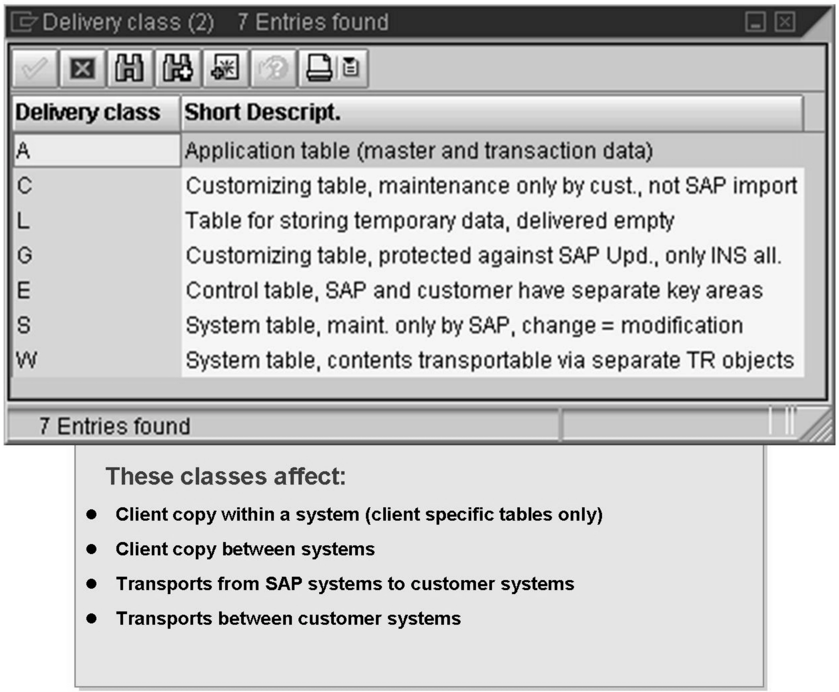
The selection depends on the type of client copy.

This example shows the selection for a remote client copy.

Figure 146: SAP Standard Client Copy Profiles

For all typical client copy scenarios, SAP provides profiles affecting various kinds of data in the SAP system and using various copy functions (as shown above).

Depending on the copy option you have selected, the client copy tool determines the purpose of the table in the system when the table is copied. For example, if you selected the *Customizing* profile and the table belongs to Customizing, then the table is copied to the target client. If the table contains application data, however, the table contents are deleted, not copied. To determine the purpose of a table, the client copy tool uses the delivery class assigned. The client copy tool assumes that all tables belong to Customizing except those that have classes **L**, **A**, and **W**.



Delivery class (2) 7 Entries found

Delivery class	Short Descript.
A	Application table (master and transaction data)
C	Customizing table, maintenance only by cust., not SAP import
L	Table for storing temporary data, delivered empty
G	Customizing table, protected against SAP Upd., only INS all.
E	Control table, SAP and customer have separate key areas
S	System table, maint. only by SAP, change = modification
W	System table, contents transportable via separate TR objects

7 Entries found

These classes affect:

- Client copy within a system (client specific tables only)
- Client copy between systems
- Transports from SAP systems to customer systems
- Transports between customer systems

Figure 147: Table Delivery Classes

To understand the effect of table delivery classes during upgrading, transport, or client copy, especially if there are development projects that create new tables, see the SAP Online Documentation and SAP Note 2857 in the SAP Service Marketplace.

Table delivery classes are assigned to tables as follows:

- In general tables containing application data should have delivery class *A*.
- Tables containing customizing data have different delivery classes, depending on how SAP changes the customizing settings in them.
- Delivery class *C* may not contain new entries from SAP.
- Delivery class *G* may be changed by SAP.
- Delivery classes *S*, *E*, and *W* are for tables containing control and SAP system data, such as tables *TADIR*, *VARI*, *TSTC*, and *CCCEXIT*.

  W tables are system tables that are filled internally. Their contents are purely technical. W tables are not taken into account by the client copy tool.
- Delivery class *L* is for tables for temporary data such as spool data and background processing data, for example, tables *TSP** and *TBT**.

  These tables should be empty in the new system and are deleted in the target client.

Use background processing for client copies because, depending on the profile used during a client copy, the client copy may take several hours.

As of release SAP_BASIS 4.6C with 4.6D kernel, the time required for a client copy is much less due to parallel processing.

To ensure data consistency in the target client, users, except for SAP* and DDIC, cannot log on to the target client during copying. Similarly, no one should work in the source client during the client copy.

When you start a client copy, take into account runtime restrictions and database space. Storage requirements can only be estimated, because space already allocated, but not yet used, is not taken into account. A client without application data needs approximately 500 MB in the database. In most databases, space made free by deleting data is only available after a reorganization.

The maximum online runtime (parameter *rdisp/max_wprun_time*) must be longer than the time needed to copy the largest table. A value of at least 2000 seconds is recommended.

To test whether there is sufficient database space, perform a test client copy, either as a simulation or as a considerably faster resource check; the resulting display shows the database memory requirements (for Oracle, Informix, Max DB and DB2/6000).

On how to copy large production clients, see SAP Note 489690 in the SAP Service Marketplace.

Hint: To estimate the size of a client you can also run program *RSTABLESIZE*. Besides other options, you can distinguish between application and customizing data.

Monitoring

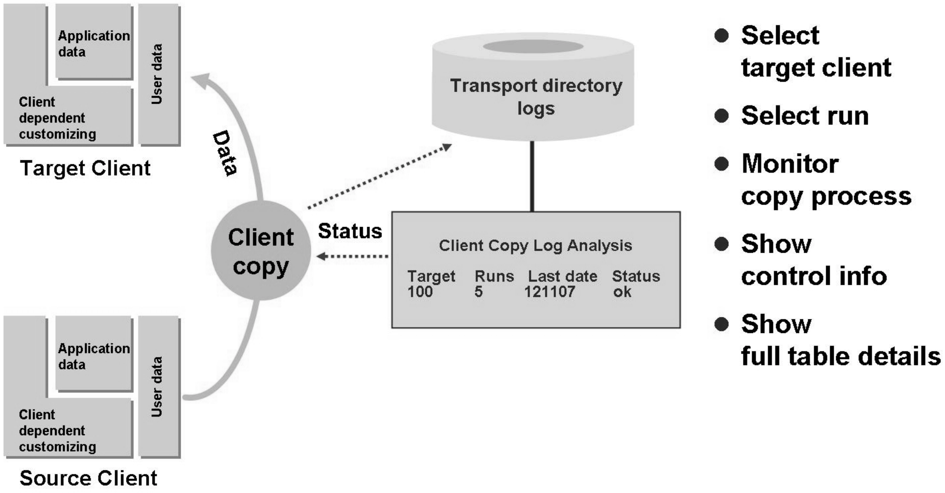


Figure 148: Copy Logs and Control Information

To display all log files for a completed local or remote client copy, choose transaction SCC3. Select the relevant target client and the relevant copying process. These copy logs provide the following information:

- Table statistics
- Control information
- Information about each table copied, including the link to the appropriate IMG component

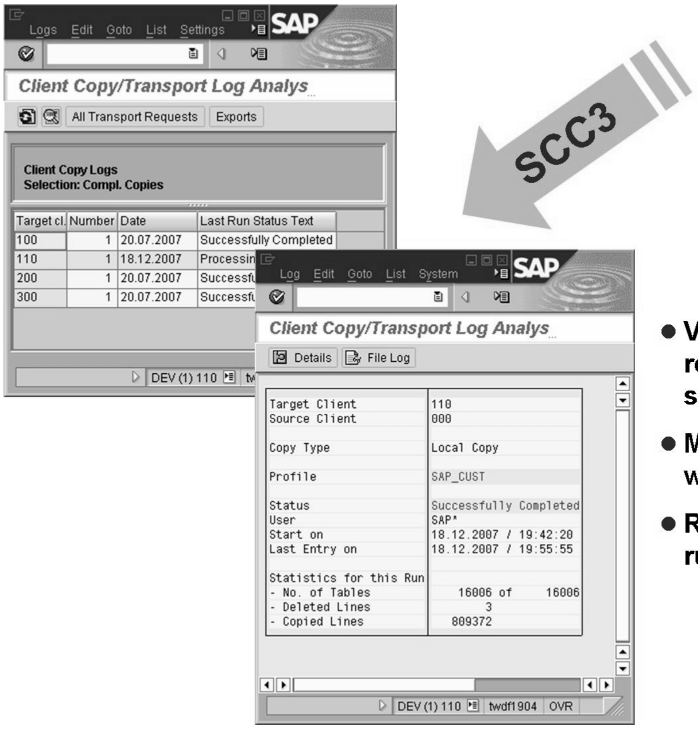
Double-click the highlighted entries for further information. You get a selection list in which you can get information about the copied tables (e.g. runtime and number of entries copied) and the exit programs from other application components used. If you had tables with copy problems in a local or remote client copy, you can recopy them with the *Recopy Errors* function. You can display the original log in the file system with the *File Log* push-button. You can also display the export and import logs of the main transport request.

To display client transport logs, use the Transport Organizer (transaction SE01).

Technically, copy log information is located in the view *V_CCCFLOW*. This view contains, for example, the runtime and processing status of a local client copy, the number of already copied tables, and the name of the table being currently copied.

All log files are physically stored in the transport log directory at operating system level. Log files are named *CC<number>.<SID>*, where *<number>* is the serial client copy number, and *<SID>* is the source system ID.

If copying is unexpectedly terminated, the system uses the name of the table being currently copied to restart the copy process.


Logs Edit Goto List Settings
SAP
Client Copy/Transport Log Analys...
All Transport Requests
Exports
SCC3
Log Edit Goto List System
SAP
Client Copy/Transport Log Analys...
Details
File Log
DEV (1) 110
twdf1904
OVR

Client Copy Logs
Selection: Compl. Copies

Target cl.	Number	Date	Last Run Status Text
100	1	20.07.2007	Successfully Completed
110	1	18.12.2007	Processin
200	1	20.07.2007	Successfu
300	1	20.07.2007	Successfu

```
Target Client             110
Source Client             000

Copy Type                 Local Copy

Profile                   SAP_CUST

Status                    Successfully Completed
User                      SAP*
Start on                  18.12.2007 / 19:42:20
Last Entry on             18.12.2007 / 19:55:55

Statistics for this Run
- No. of Tables              16006 of     16006
- Deleted Lines                  3
- Copied Lines              809372
```

- **Verify that database resources are sufficient**
- **Monitor a Client Copy with *SCC3***
- **Restart a terminated run if necessary**

Figure 149: Client Copy: Monitoring and Troubleshooting

When performing client copy error analysis, check not only the copy log, but also the system log, which tells you whether database problems are responsible for the client copy error. Correct any database problems before restarting the client copy.

When a client copy run terminates abnormally, the restart option is proposed by default, which continues the run from the point at which it was terminated. If the original run was recent, as indicated by the status line, it is advisable to choose *Restart*. Alternatively, to start the run from the beginning, choose *Restart* → *New start*.

If a copy terminates for technical reasons, for example, due to a database shutdown, you can restart the process from the point of termination. If a client copy or client import postprocessing did not finish, the system automatically proposes restart mode when you call the transaction. The same parameter settings are used automatically.

Note: The last step is restarted. You cannot continue to copy an incompletely copied table, the table is reinitialized and recopied.

Note: If the restart fails, the log displays possible reasons for the error. Before you try to restart the program again, eliminate the error.

To avoid the problem of insufficient free space in the database, perform a simulation test run or a resource check.

A simulation test run estimates the space required by reading all records to be copied without updating them in the database.

A resource check estimates the space required by counting the records to be copied.

Client copies ignore tables in the local development class *$TMP*. If you want to copy these tables, modify the development class in the object directory.

On error analysis with client copy tools, see also SAP Notes 22514 and 69444 in the SAP Service Marketplace.

If you discover a write error in the target client, usually insufficient space in the database or simultaneous processing in the target client is the reason. Check the system log to determine the name of the tablespace. Extend the tablespace and repeat the copy. You do not have to delete it first. Do not work in either the source or target client during the copy.

System log messages like *Syn. MC maintenance deactivated fully* or *Buffer TABL/TABLP reset* or *Table logging* do not indicate an error. These messages document special functions that are used to improve performance and guarantee consistency.

If a termination in an exit program (the termination is explained in an ABAP runtime error) occurs, run the log display *RSCCPROT* to determine the name of the last exit program that caused the termination. The client copy program has not crashed, this is an application error.

In any case, if you discover an unexplained cancellation or error, check

- copy log (transaction SCC3)
- system log (transaction SM21)
- dump analysis (transaction ST22)
- job overview (transaction SM37)
- and possibly spool (transaction SP01)

Exercise 11: Client Copy and Client Transport Tools

Exercise Objectives

After completing this exercise, you will be able to:

- Create a new client
- Perform a local client copy

Business Example

Several clients are required during the implementation of an SAP application. Your job as a system administrator is to manage all the clients across your landscape. You need to create and set up clients across your SAP system landscape.

Task 1:

Create a client.

1. Create a new client in the *DEV* system and name it **2##**. As always, ## signifies your group number.

Task 2:

Log on to the DEV system to your new client and start a local client copy with profile *SAP_USER* in background.

1. Log on to your new client **2##**. Start a local client copy with client *100* as the source client. Only copy users and authorizations and schedule the client copy to run in the background.

   Note: To avoid that two client copies run at the same time in the system, this exercise should be done in a distributed way. Your instructor will schedule the client copies.

Solution 11: Client Copy and Client Transport Tools

Task 1:

Create a client.

1. Create a new client in the *DEV* system and name it **`2##`**. As always, ## signifies your group number.

   a) Start transaction SCC4.

   b) Switch to change mode by choosing *Table View → Display/Change*.

   c) To create a new client, choose *New Entries*.

   d) Enter **`2##`** in the *Client* field. Enter a client description. As *Client role* choose **`Test`**.

Continued on next page

Task 2:

Log on to the DEV system to your new client and start a local client copy with profile *SAP_USER* in background.

1. Log on to your new client **2##**. Start a local client copy with client *100* as the source client. Only copy users and authorizations and schedule the client copy to run in the background.

   Note: To avoid that two client copies run at the same time in the system, this exercise should be done in a distributed way. Your instructor will schedule the client copies.

   a) Log on to your new client **2##** with user **SAP*** and password **pass**.

   b) Choose transaction SCCL.

   c) Select client *100* as the source client. For the selected profile, choose **SAP_USER**.

      Choose *Enter*.

   d) Create and schedule the copy job.

      Choose *Schedule as Background Job*.

      Choose *Schedule Job*. Ensure that the target client, profile name, and source client are correct. Copy only user data and authorizations, *SAP_USER*. The data should come from client *100*. Choose *Continue*.

   e) Once the background job is scheduled, you can monitor the copy process with transaction SCC3.

   f) After the copy run is completed, log on to the new client, using your **ADM325-##** user with your password from client *100*.

Lesson Summary

You should now be able to:

- Describe client copy and client transport tools
- Perform a client copy
- Troubleshoot client copy and client transport errors

Lesson: Client Compare and Maintenance Tools

Lesson Overview

In this lesson, you will learn how to perform a client compare and maintain clients by a client adjustment.

Lesson Objectives

After completing this lesson, you will be able to:

- Compare and maintain clients

Business Example

You have a specific problem in production that is related to a customizing table. You have multiple SAP systems in production. The customizing settings work in one system but not in another. You would like to compare a specific customizing table across multiple SAP systems to help you debug the problem in production.

Using Client Compare Tools

When several SAP systems and clients are being implemented, it may be necessary to compare and adjust customizing settings between different systems and clients. The compare function enables you to compare and adjust the contents of a table or view in two different clients, using RFC connections.

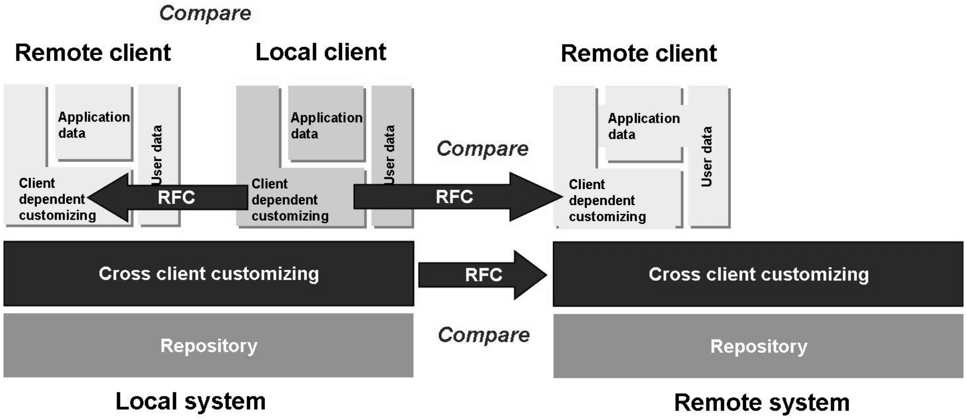
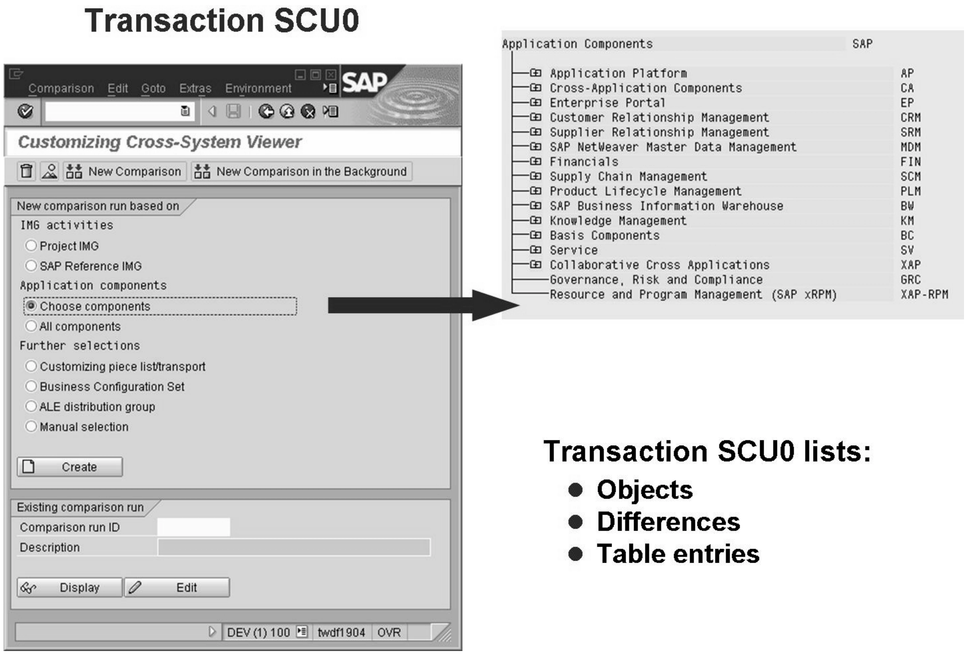


Figure 150: Client Compare

To use the compare function, from the SAP systems initial screen, choose transaction SCU0.

Use Compare to:

- Compare a client with a reference client, such as a customer-owned client or the SAP client 000. This is especially useful after an upgrade or language import, since only client 000 is provided with data from tables belonging to table delivery class *C*.
- Compare clients during a rollout scenario. For example, if subsidiaries wish to adjust their Customizing with respect to the reference Customizing of a master client.
- Compare client-independent Customizing before combining different clients into one SAP system in a roll-in scenario. For example, if subsidiaries wish to receive Customizing changes from the reference client, master client of their parent organization.


Transaction SCU0
Comparison Edit Goto Extras Environment
SAP
Customizing Cross-System Viewer
New Comparison
New Comparison in the Background
Create
Display
Edit
DEV (1) 100 twdf1904 OVR

New comparison run based on

IMG activities
- ○ Project IMG
- ○ SAP Reference IMG

Application components
- ● Choose components
- ○ All components

Further selections
- ○ Customizing piece list/transport
- ○ Business Configuration Set
- ○ ALE distribution group
- ○ Manual selection

Existing comparison run

Comparison run ID

Description

Application Components	SAP
Application Platform	AP
Cross-Application Components	CA
Enterprise Portal	EP
Customer Relationship Management	CRM
Supplier Relationship Management	SRM
SAP NetWeaver Master Data Management	MDM
Financials	FIN
Supply Chain Management	SCM
Product Lifecycle Management	PLM
SAP Business Information Warehouse	BW
Knowledge Management	KM
Basis Components	BC
Service	SV
Collaborative Cross Applications	XAP
Governance, Risk and Compliance	GRC
Resource and Program Management (SAP xRPM)	XAP-RPM

Transaction SCU0 lists:

- **Objects**
- **Differences**
- **Table entries**

Figure 151: Client Compare: Selection Options

To select the objects to be compared in a client compare, you can use a project-oriented approach utilizing the IMG to define the objects. You can also select from application components' customizing object lists and transport requests. You can also select the objects manually.

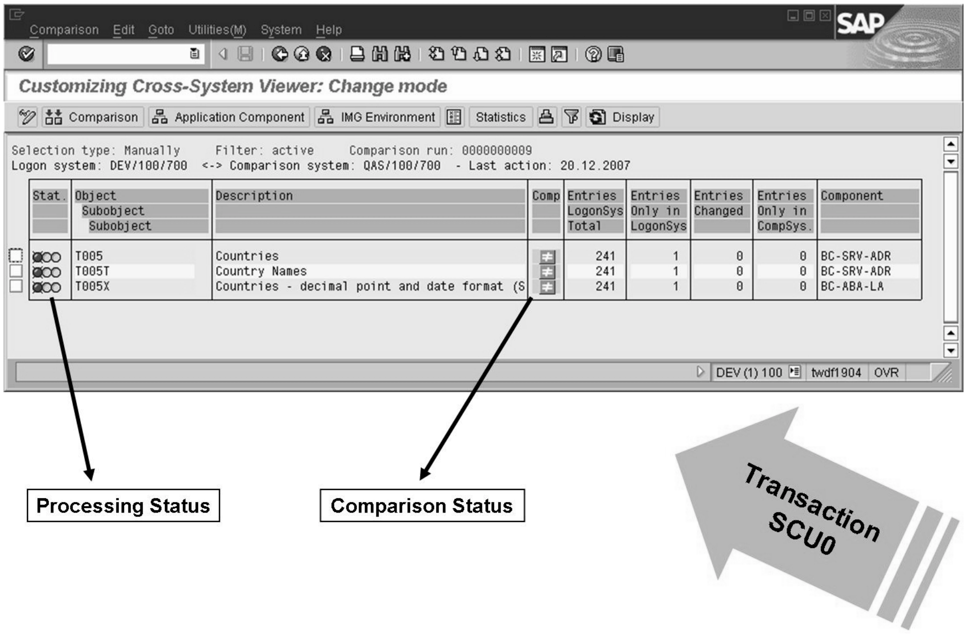
To start a client compare, use transaction SCU0 in background processing. From the initial screen, select the component and choose *Create*. In the pop-ups that appear, make the appropriate entries. Transaction SCU0 can be restricted to, for example, client-dependent data.

Transaction SCU0 first displays an overview of the tables belonging to views for the selected IMG, application component or request. Next, the comparison itself is performed. The system creates a list of differences and indicates whether these differences are encountered in the table structure or in the table content.

To display the differing table entries, select an object. This enables you to perform a detailed comparison.

The comparison client is generally an external environment, so you must access it through a remote function call (RFC). For information on how to use an RFC call, in the selection popup for the RFC destination, choose *Info*.

For information on client compares, see also SAP Note 91096 in SAP Service Marketplace.



Customizing Cross-System Viewer: Change mode

```
Selection type: Manually     Filter: active     Comparison run: 0000000009
Logon system: DEV/100/700  <-> Comparison system: QAS/100/700  - Last action: 20.12.2007
```

	Stat.	Object	Description	Comp	Entries	Entries	Entries	Entries	Component
		Subobject			LogonSys	Only in	Changed	Only in	
		Subobject			Total	LogonSys		CompSys.	
□	●○○	T005	Countries	≠	241	1	0	0	BC-SRV-ADR
□	●○○	T005T	Country Names	≠	241	1	0	0	BC-SRV-ADR
□	●○○	T005X	Countries - decimal point and date format (S	≠	241	1	0	0	BC-ABA-LA

Figure 152: Client Compare: Difference List

The results of each comparison run is an overview of existing differences between the logon client and the comparison client. This overview serves as a starting point for subsequent processing of differences. The comparison runs are stored in a worklist or difference list.

Using Client Maintenance Tools

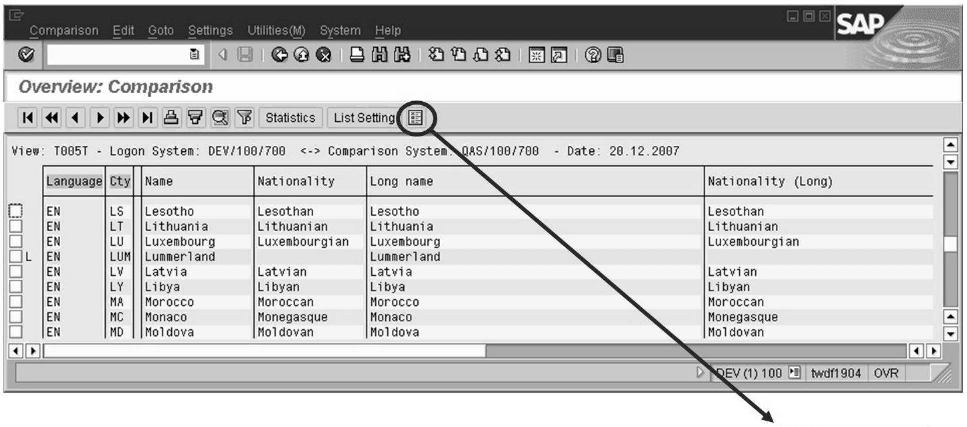
For each compared object in a comparison run, the objects compared are listed along with a description. The most important information is the status indicators. The *comparison status* indicates the existence and nature of any differences and is set automatically.

Process status allows you to distinguish between the objects that have already been processed, that is, have been made identical in both clients, and those that have not. This kind of processing is also referred to as **adjusting** the respective object. The processing status is indicated by a traffic light, where red indicates not processed, yellow indicates the objects in both clients are in the process of being made identical, and green indicates the completion of this adjustment process.

An explanation of the possible statuses under comparison status and process status can be obtained by clicking the *Legend* icon.

For a detailed analysis of a comparison run, choose *Comparison*.

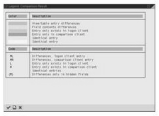
- **Adjustment of Individual objects with SCU0:**



Overview: Comparison

View: T005T - Logon System: DEV/100/700 <-> Comparison System: QAS/100/700 - Date: 20.12.2007

		Language	Cty	Name	Nationality	Long name	Nationality (Long)
□		EN	LS	Lesotho	Lesothan	Lesotho	Lesothan
□		EN	LT	Lithuania	Lithuanian	Lithuania	Lithuanian
□		EN	LU	Luxembourg	Luxembourgian	Luxembourg	Luxembourgian
□	L	EN	LUM	Lummerland		Lummerland	
□		EN	LV	Latvia	Latvian	Latvia	Latvian
□		EN	LY	Libya	Libyan	Libya	Libyan
□		EN	MA	Morocco	Moroccan	Morocco	Moroccan
□		EN	MC	Monaco	Monegasque	Monaco	Monegasque
□		EN	MD	Moldova	Moldovan	Moldova	Moldovan



- **Generates detailed information in work list**
- **Displays a comparison status for each record in work list**
- **To understand the comparison status, choose the legend icon**

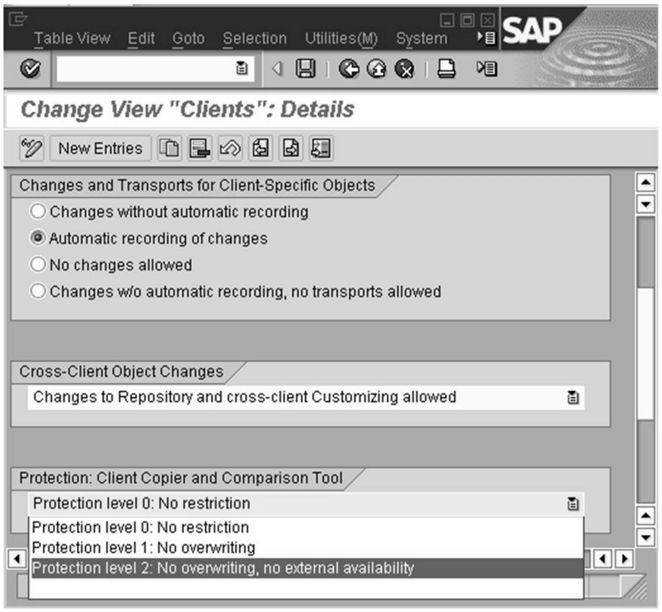
Figure 153: Adjusting Customizing Objects

You can adjust one object at a time. The objects that can be adjusted are some of the tables and views that can be maintained using transaction SM30. You can also use the IMG, of course. All other objects can only be compared.

To perform an adjustment, from the *Comparison* screen, choose *Edit → Interact Copy*. The *Overview Adjustment* screen is displayed, showing the details of the differences between the two clients, record by record. To the left of each record in this adjustment worklist is the comparison status, which indicates whether or not each respective entry in the record exists in the comparison client and the logon client.

An explanation of the possible statuses under the comparison status and process status can be obtained by clicking the *Legend* icon.

- **Transaction *SCC4* set client option:**
  Protection level 2: No overwriting, no external availability



Change View "Clients": Details

Changes and Transports for Client-Specific Objects

○ Changes without automatic recording
● Automatic recording of changes
○ No changes allowed
○ Changes w/o automatic recording, no transports allowed

Cross-Client Object Changes

Changes to Repository and cross-client Customizing allowed

Protection: Client Copier and Comparison Tool

Protection level 0: No restriction

Protection level 0: No restriction
Protection level 1: No overwriting
Protection level 2: No overwriting, no external availability

- **Default value is: *No restriction***

Figure 154: Client Maintenance Settings

The client option *Protection* regarding client copy and client comparison can be set to prevent a client from being overwritten by the client copy or client compare and client adjustment. It can also be set to ensure that sensitive data is not viewable from another client during a client compare.

To set this option from the SAP systems initial screen, choose transaction SCC4. Select a protection level:

Protection level 1: No overwriting

This protection level ensures that the client cannot be overwritten by the client copy program. This setting should be used if Customizing is being performed in this client or the client contains critical settings or data that should not be overwritten.

Protection level 2: No overwriting and no external availability

This protection level also protects the client against overwriting and against read access from another client during a client copy or customizing comparison. This setting should be used if the client contains sensitive data. All production clients should use this setting.

To set the option *Protection regarding client copy/comparison*, you need full authorization for the Workbench Organizer (authorization *S_CTS_ADMIN*).

Exercise 12: Client Compare and Maintenance Tools

Exercise Objectives

After completing this exercise, you will be able to:

- Compare Customizing across clients
- Maintain Customizing across clients

Business Example

In a multi-client system landscape it is necessary to compare and maintain customizing settings across clients.

Task 1:

As part of your customizing settings, compare country texts across clients. The table, that contains the country texts is *T005T*. The corresponding maintenance view is *V_T005_BAS*.

1. Log on to the **DEV** system to client **2##** with your user **ADM325-##**.

   Compare Customizing in client **2##** with Customizing in client **100** using transaction SCU0 in regard to view **V_T005_BAS** and the RFC connection, provided by your administrator.

Task 2:

Adjust country texts across clients.

1. The Customizing you did for an earlier lesson does not exist in the new client 2##. Replicate your country definition to client 2## in the country maintenance.

Solution 12: Client Compare and Maintenance Tools

Task 1:

As part of your customizing settings, compare country texts across clients. The table, that contains the country texts is *T005T*. The corresponding maintenance view is *V_T005_BAS*.

1. Log on to the **DEV** system to client **2##** with your user **ADM325-##**.

   Compare Customizing in client **2##** with Customizing in client **100** using transaction SCU0 in regard to view **V_T005_BAS** and the RFC connection, provided by your administrator.

   a) After entering transaction SCU0, choose *Manual Selection* and then *Create*. Enter the object name **V_T005_BAS** and choose **V** (for view) as type. Choose *Copy*.

   b) Enter a description and name the RFC connection.

   c) Choose *Total comparison*. Select the check box next to table *V_T005_BAS*. Choose *Comparison* without restricting the number of entries to compare.

   d) View the result of the following Comparison. Look at the legend of the result screen.

Continued on next page

Task 2:

Adjust country texts across clients.

1. The Customizing you did for an earlier lesson does not exist in the new client 2##. Replicate your country definition to client 2## in the country maintenance.

   a) In client 2##, open the IMG via transaction SPRO_ADMIN or SPRO. Select the country maintenance.

   > **Note:** If the view contains exactly **one** entry, choose *Back* before the next step

   b) Choose *Utilities* → *Adjustment*. Use the RFC connection, provided by your instructor.

   c) Mark **only** the entry **you created** in an earlier exercise and choose *Adjust*.

   d) Choose *All*, go back using *Back* twice until the pop up for saving your customizing appears. Confirm it. Save again. Depending on your client settings in transaction SCC4 for your client, a transport request is required. In this case, create a customizing request.

   e) Check your actions.

Lesson Summary

You should now be able to:

- Compare and maintain clients

Unit Summary

You should now be able to:

- Describe client copy and client transport tools
- Perform a client copy
- Troubleshoot client copy and client transport errors
- Compare and maintain clients

Test Your Knowledge

1. Client transport consists of two processes: export and import (with post import activities).

   Determine whether this statement is true or false.

   □ True

   □ False

2. The client copy and client transport tools are designed to replace the data of one client with data from another client.

   Determine whether this statement is true or false.

   □ True

   □ False

3. Explain the steps involved in client transport.

4. To avoid the problem of insufficient free space in the database, perform a simulation test run or a resource check.

   Determine whether this statement is true or false.

   □ True

   □ False

5. List the steps to delete a client from within an SAP system.

6. What is the difference between a client compare and a client maintenance/adjustment?

7. What should be the settings for a productive client, concerning client copy/comparison?

Answers

1. Client transport consists of two processes: export and import (with post import activities).

   Answer: True

   Client transport consists of two processes: export and import (with post import activities).

2. The client copy and client transport tools are designed to replace the data of one client with data from another client.

   Answer: True

   The client copy and client transport tools are designed to replace the data of one client with data from another client.

3. Explain the steps involved in client transport.

   Answer: A client transport consists of two steps: .

   - First, a client export extracts client data from the source client to files at the operating system level.
   - Second, the data is imported from the file system files into the target client.

4. To avoid the problem of insufficient free space in the database, perform a simulation test run or a resource check.

   Answer: True

   To avoid the problem of insufficient free space in the database, perform a *simulation test run* or a *resource check*.

5. List the steps to delete a client from within an SAP system.

   Answer: To delete a client from within an SAP system:

   - Log on to the client to be deleted.
   - Use transaction SCC5.
   - Either delete the client entry from table *T000* by using this function in transaction SCC5 or by deleting it with transaction SCC4.

6. What is the difference between a client compare and a client maintenance/adjustment?

   Answer: A client compare looks for differences in the Customizing of two clients, a client maintenance/adjustment copies these differences from one client to the other.

7. What should be the settings for a productive client, concerning client copy/comparison?

   Answer: Protection level 2: No overwriting, no external availability.

Unit 6

SAP Note Assistant, Support Packages, SAP system upgrades and Enhancement Packages

Unit Overview

In this unit, you will learn about the SAP software component model. You will learn about the concept of SAP Notes and how to work with the SAP Note Assistant. You will also learn how to apply Support Packages to your SAP system and system landscape. Finally, you will learn, as an overview, how to perform an SAP system upgrade for your system landscape. As an outlook you will learn about the concept of the Enhancement Packages.

There are four ways for an SAP System:

- by applying SAP kernel patches (not part of this unit)
- by applying SAP Notes with transaction SNOTE
- by applying Support Packages with transaction SPAM
- by performing an SAP system upgrade with the tool *SAPup*
- by applying Enhancement Packages with transaction SAINT

By implementing an SAP Note an error in a single program is removed. When applying an Support Package, all errors are corrected that were known when creating this SAP Support Package. Through an SAP system upgrade or by applying an Enhancement Package many functions are replaced by new, enhanced functions.

Also SAP Notes and Support Packages are not only corrections of errors in the SAP standard, but also can bring new functions into the system.

Unit Objectives

After completing this unit, you will be able to:

- List the advantages of automatically implementing SAP Notes
- List the goals, functions, and the features of SAP Note Assistant

- Describe Note Browser and list the preconditions for handling SAP Notes
- Describe the import of Support Packages
- Reduce the downtime when applying Support Packages
- Perform a modification adjustment during the import of Support Packages
- Have an idea about planning an SAP system upgrade
- Roughly explain how an SAP system upgrade functions

Unit Contents

Lesson: SAP Note Assistant

Lesson Overview

In this lesson, you will learn about SAP Notes and the advantages of implementing an SAP Note automatically with the SAP Note Assistant. You will also learn about the goals, functions, and the features of the SAP Note Assistant. You will also learn about the Note Browser. Additionally, you will learn about how to cancel an SAP Note implementation and about the preconditions for handling SAP Notes.

Lesson Objectives

After completing this lesson, you will be able to:

- List the advantages of automatically implementing SAP Notes
- List the goals, functions, and the features of SAP Note Assistant
- Describe Note Browser and list the preconditions for handling SAP Notes

Business Example

As the system administrator, you should be aware of the tools used for applying error fixes to the SAP software components. SAP supplies many tools the system administrator can use to keep the SAP system running the latest fixes and functions. These tools enable you to install various degrees of fixes and functions. You can use them to apply individual fixes to ABAP programs and to add transactions that are important to meet business requirements.

Maintenance Using SAP Note Assistant

There are SAP Notes for different reasons. SAP Notes can provide information of any kind for customers and partners of SAP. But often they also contain solutions for known problems with an SAP system.

When talking about SAP Notes in this lesson, we refer only to those SAP Notes that contain a solution for an error in a Repository object. Not all objects can be handled with the SAP Note Assistant, for example no Dictionary objects. These SAP Notes will then describe how to remove these errors in the SAP system by providing the correct source code. They include a description of the following:

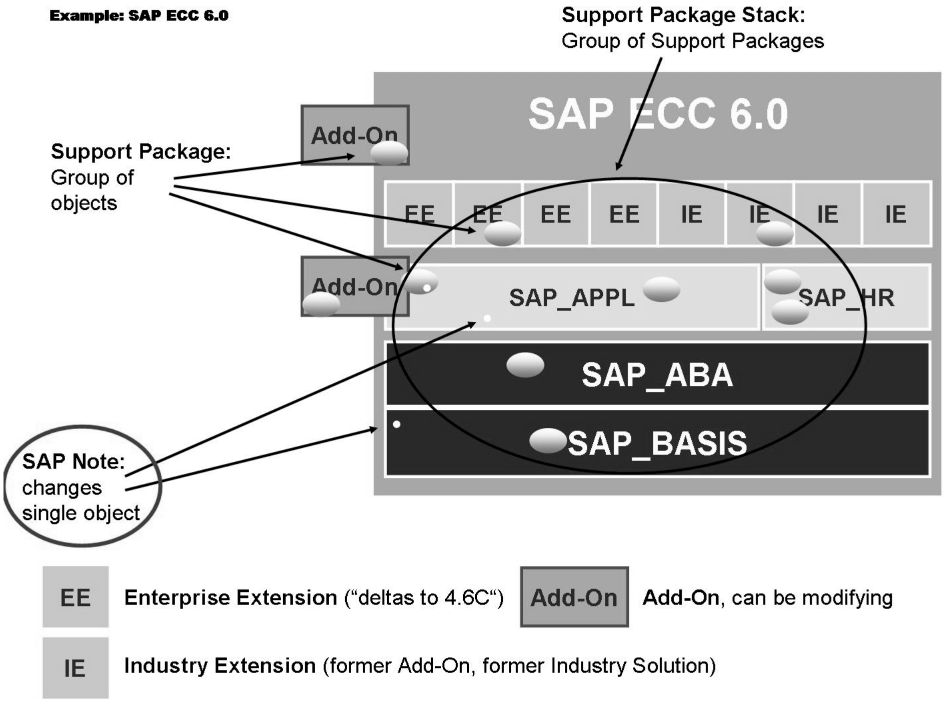


Figure 155: Concept

- The symptoms of the error
- The cause of the error
- Descriptions of how to correct the source code
- The SAP release and Support Package status for which the note is valid
- Links to Support Packages that correct the problem

Carefully read the SAP Note before you implement it. The SAP Note can contain prerequisites, interactions, and references to post processing activities, such as making changes to a table that you must consider when you implement it.

While implementing an SAP Note, you can remove the error it refers to.

- Costs
- Quality Assurance

Manually implementing an SAP Note leads to high costs. Resources, including personnel and time to implement the note are necessary.

If an SAP Note is implemented incorrectly, the quality of the system suffers.

Prior to making any modification based on SAP Notes, verify that the SAP Note is applicable to your SAP release and that the symptoms in the SAP Note are the same as the symptoms apparent in your SAP system. In the case of uncertainty, contact SAP.

All modifications of SAP objects should be performed in the development client, tested in the quality assurance client and distributed to production clients only after thorough testing and verification.

Because modifications often lead to the need for a modification adjustment during a later SAP system upgrade or when applying Support Packages, document the modification changes to help to speed up the modification adjustment process. When documenting, be sure to include the SAP Note number and SAP system release dependencies.

With SAP R/3 4.6B you can set attributes for change requests to help you identify them. For example, after setting appropriate attributes, you can identify which change request in the import queue contains a manual advance correction according to an SAP Note.

Automatic implementation of SAP Notes contributes to enhanced quality, because it reduces errors due to manual implementation.

- Automatically implement SAP Notes which handle ABAP source code corrections
- Handle dependencies with other SAP Notes, Support Packages, and modifications
- Display all SAP Notes implemented in the system
- Handle updates (Upgrades and Support Packages)
- Correct only a single bug, not a replacement for Support Packages

Note: SAP Notes cannot change ABAP Dictionary Objects.

The SAP Note Assistant includes the following functions:

- Reporting: you can display an overview of all existing SAP Notes and their processing status, and all implemented source code corrections.
- Workflow functions: You can assign SAP Notes to users who then specify a processing status for the notes. The system informs you if an SAP Note has an inconsistent status.
- Logging: the system logs all processing steps.
- Corrections: you can automatically implement the source code corrections (correction instructions) contained in SAP Notes into your system.
- Integration: when you apply Support Packages or upgrade your system, the system automatically recognizes which SAP Notes have already been completed by an imported Support Package or Upgrade, and which correction instructions still need to be implemented.

You can apply the SAP Notes in two different ways with the SAP Note Assistant:

1. If a connection to the SAP Service Marketplace is configured in the SAP System, you can directly download the SAP Notes to your system.
2. You can also separately download the SAP Notes from SAP Service Marketplace (for example, with the Download Manager) and then upload them to the SAP system.

   The upload then takes place from within the SAP Note Assistant.

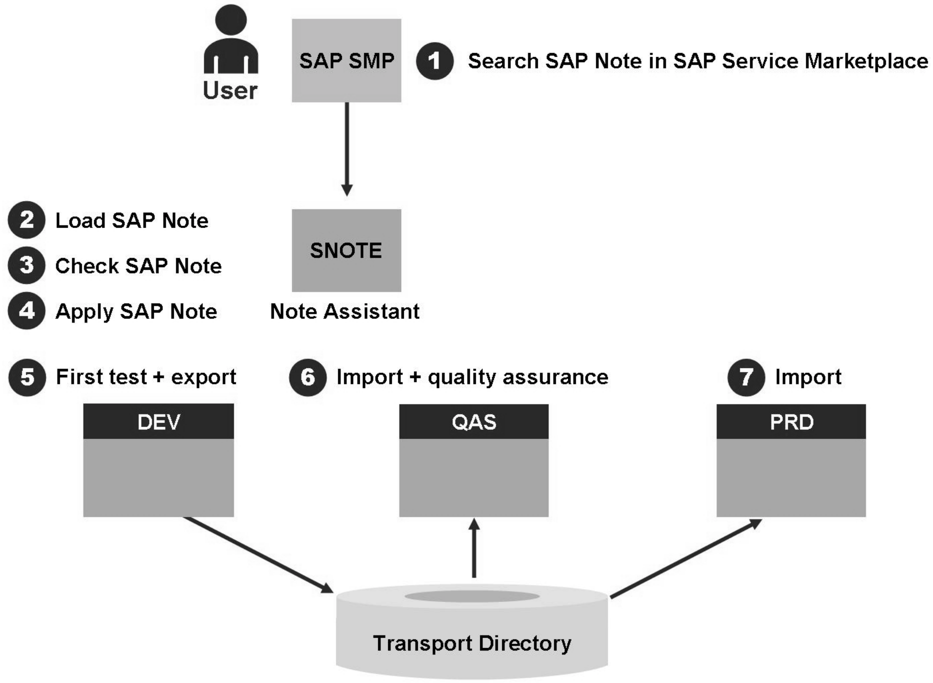


Figure 156: Distribution of Corrections

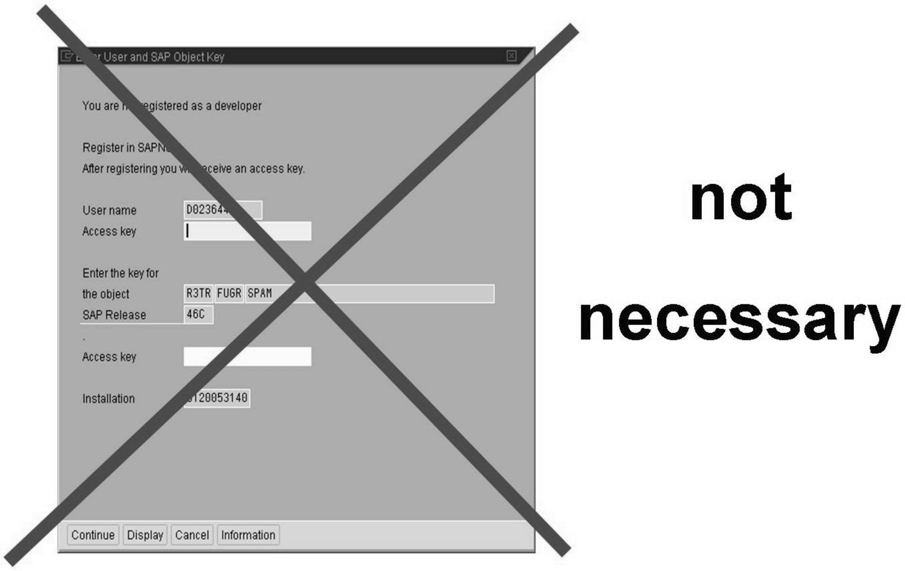
Follow these steps to apply SAP Notes in your system landscape:

1. Implement the SAP Notes with the SAP Note Assistant in the development system (DEV).
2. Transport the SAP Notes from the development system to the quality assurance system (QAS). Do this with the transport request that contains the corrected objects. Do not implement the SAP Note with the SAP Note Assistant in QAS. Now you can test these corrections.

   Caution: Do not transport the implemented SAP Notes in the quality assurance system (QAS) before applying the same SAP Support Packages in QAS, because you need to perform the same implementation sequence as in DEV. It is important that all systems in a system landscape have the same SAP Support Package level.

3. Import the SAP Notes into the productive system (PRD). Use the transport request that contains the corrected objects. Do not implement the SAP Note with the SAP Note Assistant in PRD.

The SAP Note Assistant is available from SAP R/3 4.5B. From SAP Web Application Server 6.10, it is integrated in the standard delivery. For releases prior to SAP Web AS 6.10, the SAP Note Assistant is delivered as an Add-on. You can download the Add-on SAP NOTES from the SAP Service Marketplace. Please read the related SAP Notes before you install the SAP Note Assistant.



Enter User and SAP Object Key

You are not registered as a developer

Register in SAPNet
After registering you will receive an access key.

User name: D02364
Access key:

Enter the key for the object: R3TR FUGR SPAM
SAP Release: 46C

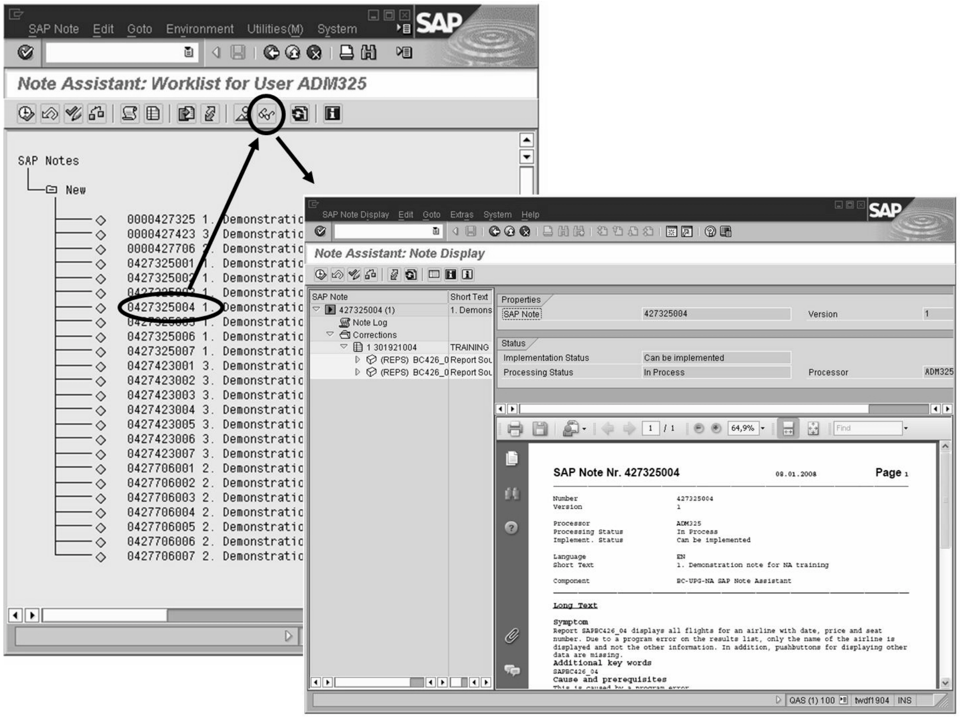
Access key:

Installation: 0120053140

Figure 157: SSCR Key

Changes to SAP objects that are introduced using the SAP Note Assistant do not require an SSCR key.

Handling the SAP Note Assistant


SAP Note Edit Goto Environment Utilities(M) System
SAP
SAP Note Display Edit Goto Extras System Help
SAP

Note Assistant: Worklist for User ADM325

```
SAP Notes
 └─ New
      ◇ 0000427325 1. Demonstratio
      ◇ 0000427423 3. Demonstratio
      ◇ 0000427706 2. Demonstratio
      ◇ 0427325001 1. Demonstratio
      ◇ 0427325002 1. Demonstratio
      ◇ 0427325003 1. Demonstratio
      ◇ 0427325004 1. Demonstratio
      ◇ 0427325005 1. Demonstratio
      ◇ 0427325006 1. Demonstratio
      ◇ 0427325007 1. Demonstratio
      ◇ 0427423001 3. Demonstratio
      ◇ 0427423002 3. Demonstratio
      ◇ 0427423003 3. Demonstratio
      ◇ 0427423004 3. Demonstratio
      ◇ 0427423005 3. Demonstratio
      ◇ 0427423006 3. Demonstratio
      ◇ 0427423007 3. Demonstratio
      ◇ 0427706001 2. Demonstratio
      ◇ 0427706002 2. Demonstratio
      ◇ 0427706003 2. Demonstratio
      ◇ 0427706004 2. Demonstratio
      ◇ 0427706005 2. Demonstratio
      ◇ 0427706006 2. Demonstratio
      ◇ 0427706007 2. Demonstratio
```

Note Assistant: Note Display

SAP Note	Short Text
427325004 (1)	1. Demons
Note Log	
Corrections	
1 301921004	TRAINING
(REPS) BC426_0	Report Sou
(REPS) BC426_0	Report Sou

Properties

SAP Note: 427325004 Version: 1

Status

Implementation Status: Can be implemented

Processing Status: In Process Processor: ADM325

1 / 1 64,9% Find

```
SAP Note Nr. 427325004                        08.01.2008          Page 1

Number                      427325004
Version                     1

Processor                   ADM325
Processing Status           In Process
Implement. Status           Can be implemented

Language                    EN
Short Text                  1. Demonstration note for NA training

Component                   BC-UPG-NA SAP Note Assistant

Long Text

Symptom
Report SAPBC426_04 displays all flights for an airline with date, price and seat
number. Due to a program error on the results list, only the name of the airline is
displayed and not the other information. In addition, pushbuttons for displaying other
data are missing.
Additional key words
SAPBC426_04
Cause and prerequisites
This is caused by a program error
```

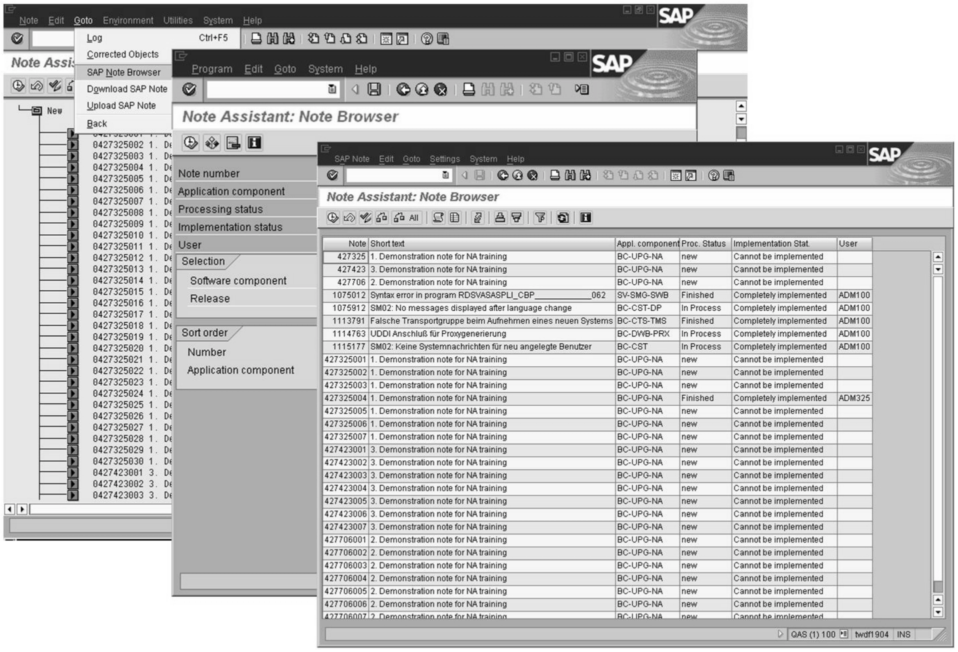
QAS (1) 100 twdf1904 INS

Figure 158: Worklist

The Note Assistant is started with transaction SNOTE. The worklist is your personal view on the SAP Notes (unlike the Note Browser that displays all SAP Notes that are available in the system).

All users see inconsistent and new SAP Notes in their worklists. Only you see the SAP Notes under the category *In Processing*. They are the ones that you process.

From your worklist you can perform all actions such as upload, implement, and display SAP Notes.



Note	Short text	Appl. component	Proc. Status	Implementation Stat.	User
427325	1. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427423	3. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427706	2. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
1075012	Syntax error in program RDSVASASPLI_CBP____________062	SV-SMG-SWB	Finished	Completely implemented	ADM100
1075912	SM02: No messages displayed after language change	BC-CST-DP	In Process	Completely implemented	ADM100
1113791	Falsche Transportgruppe beim Aufnehmen eines neuen Systems	BC-CTS-TMS	Finished	Completely implemented	ADM100
1114763	UDDI Anschluß für Proxygenerierung	BC-DWB-PRX	In Process	Completely implemented	ADM100
1115177	SM02: Keine Systemnachrichten für neu angelegte Benutzer	BC-CST	In Process	Completely implemented	ADM100
427325001	1. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427325002	1. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427325003	1. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427325004	1. Demonstration note for NA training	BC-UPG-NA	Finished	Completely implemented	ADM325
427325005	1. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427325006	1. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427325007	1. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427423001	3. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427423002	3. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427423003	3. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427423004	3. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427423005	3. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427423006	3. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427423007	3. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427706001	2. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427706002	2. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427706003	2. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427706004	2. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427706005	2. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427706006	2. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	
427706007	2. Demonstration note for NA training	BC-UPG-NA	new	Cannot be implemented	

Figure 159: Note Browser

The Note Browser gives an overview of all SAP Notes available in the system.

To reduce the number of SAP Notes that are displayed in the results list, you can specify various criteria in the selection screen (processing status, application component, implementation status).

From the Note Browser, you can also implement or display SAP Notes.

SAP Note Assistant provides additional functions.

Viewing information

From your worklist, you can view the following information:

- SAP Notes that are assigned to you as a user
- All SAP Notes that are inconsistent, even if they are assigned to another user
- New SAP Notes that have not yet been processed

Finding SAP Notes

You can use the Note Browser to find any SAP Note that is available in the system. Choose *Display Note* to read through the SAP Note and see the source code changes.

Upload SAP Notes

Download an SAP Note from the SAP Service Marketplace to your front end. Then choose *Upload* in transaction SNOTE to load the SAP Note into the SAP Note Assistant.

Checking notes

Choose *Check Note* to check the implementation status.

Specifying processing status

You can specify the processing status by choosing *Set Processing Status*.

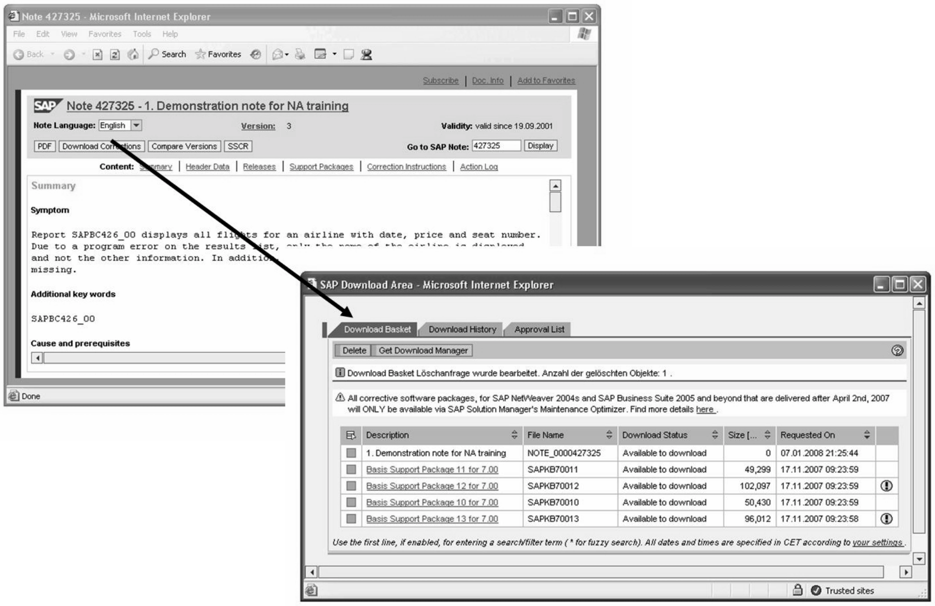
You can implement the correction instructions that are contained in the SAP Note and then remove them if necessary. In the Log, you can display the processing steps and add your own comments to the log.

You choose *Corrected Objects* to switch to the Modification Browser where you can see all the objects that an SAP Note has corrected.

After importing an Support Package, you can use the Modification Assistant to perform follow-up activities.

If you want to correct an error using an SAP Note, you must already know the number of the SAP Note that corrects the error.

You can download the SAP Note from http://service.sap.com/notes with the SAP Download Manager. The SAP Download Manager and its requirements are available in the SAP Service Marketplace.



Note 427325 - Microsoft Internet Explorer

File Edit View Favorites Tools Help
Back
Search
Favorites
Subscribe | Doc. Info | Add to Favorites

Note 427325 - 1. Demonstration note for NA training

Note Language: English **Version:** 3 **Validity:** valid since 19.09.2001

PDF
Download Corrections
Compare Versions
SSCR

Go to SAP Note: 427325

Display
Content: Summary | Header Data | Releases | Support Packages | Correction Instructions | Action Log

Summary

Symptom

```
Report SAPBC426_00 displays all flights for an airline with date, price and seat number.
Due to a program error on the results list, only the name of the airline is displayed
and not the other information. In additio
missing.
```

Additional key words

```
SAPBC426_00
```

Cause and prerequisites

SAP Download Area - Microsoft Internet Explorer

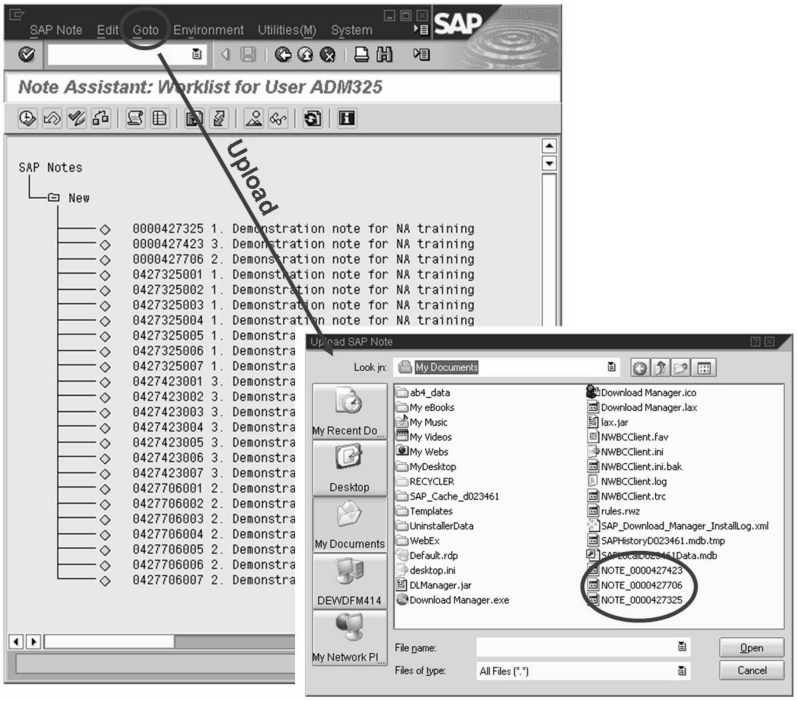
Download Basket Löschanfrage wurde bearbeitet. Anzahl der gelöschten Objekte: 1 .

All corrective software packages, for SAP NetWeaver 2004s and SAP Business Suite 2005 and beyond that are delivered after April 2nd, 2007 will ONLY be available via SAP Solution Manager's Maintenance Optimizer. Find more details here .

	Description	File Name	Download Status	Size [...	Requested On	
▨	1. Demonstration note for NA training	NOTE_0000427325	Available to download	0	07.01.2008 21:25:44	
▨	Basis Support Package 11 for 7.00	SAPKB70011	Available to download	49,299	17.11.2007 09:23:59	
▨	Basis Support Package 12 for 7.00	SAPKB70012	Available to download	102,097	17.11.2007 09:23:59	!
▨	Basis Support Package 10 for 7.00	SAPKB70010	Available to download	50,430	17.11.2007 09:23:59	
▨	Basis Support Package 13 for 7.00	SAPKB70013	Available to download	96,012	17.11.2007 09:23:58	!

*Use the first line, if enabled, for entering a search/filter term (* for fuzzy search). All dates and times are specified in CET according to your settings .*

Figure 160: Download from SAP Service Marketplace



Note Assistant: Worklist for User ADM325

```
SAP Notes
 └─ New
      ──◇  0000427325 1. Demonstration note for NA training
      ──◇  0000427423 3. Demonstration note for NA training
      ──◇  0000427706 2. Demonstration note for NA training
      ──◇  0427325001 1. Demonstration note for NA training
      ──◇  0427325002 1. Demonstration note for NA training
      ──◇  0427325003 1. Demonstration note for NA training
      ──◇  0427325004 1. Demonstration note for NA training
      ──◇  0427325005 1. Demonstra
      ──◇  0427325006 1. Demonstra
      ──◇  0427325007 1. Demonstra
      ──◇  0427423001 3. Demonstra
      ──◇  0427423002 3. Demonstra
      ──◇  0427423003 3. Demonstra
      ──◇  0427423004 3. Demonstra
      ──◇  0427423005 3. Demonstra
      ──◇  0427423006 3. Demonstra
      ──◇  0427423007 3. Demonstra
      ──◇  0427706001 2. Demonstra
      ──◇  0427706002 2. Demonstra
      ──◇  0427706003 2. Demonstra
      ──◇  0427706004 2. Demonstra
      ──◇  0427706005 2. Demonstra
      ──◇  0427706006 2. Demonstra
      ──◇  0427706007 2. Demonstra
```

Upload SAP Note

Look in: My Documents

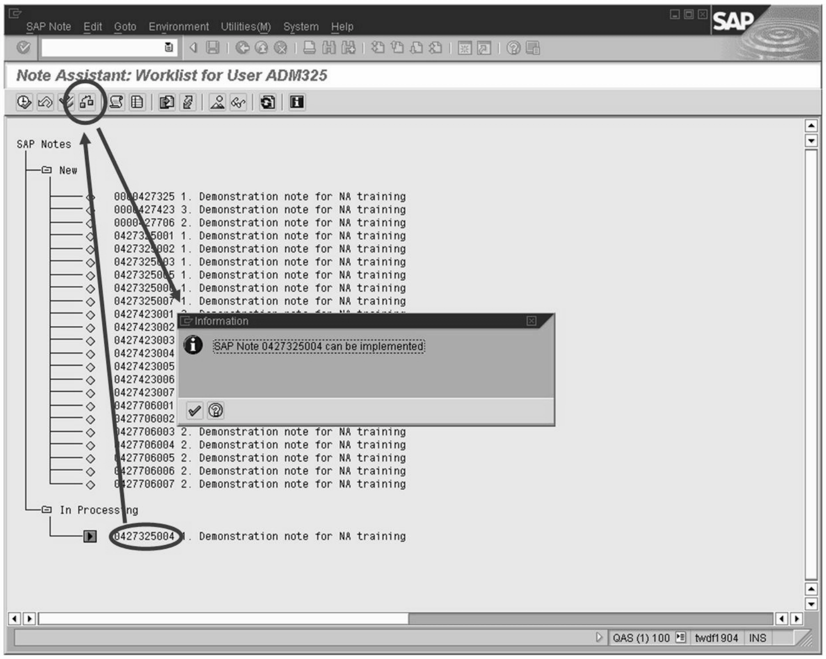
```
ab4_data                 Download Manager.ico
My eBooks                Download Manager.lax
My Music                 lax.jar
My Videos                NWBCClient.fav
My Webs                  NWBCClient.ini
MyDesktop                NWBCClient.ini.bak
RECYCLER                 NWBCClient.log
SAP_Cache_d023461        NWBCClient.trc
Templates                rules.rwz
UninstallerData          SAP_Download_Manager_InstallLog.xml
WebEx                    SAPHistoryD023461.mdb.tmp
Default.rdp              SAPLocalD023461Data.mdb
desktop.ini              NOTE_0000427423
DLManager.jar            NOTE_0000427706
Download Manager.exe     NOTE_0000427325
```

File name:

Files of type: All Files (*.*)

Figure 161: Upload Notes

To upload the SAP Notes, choose *Goto* → *Upload SAP Note*.



Note Assistant: Worklist for User ADM325

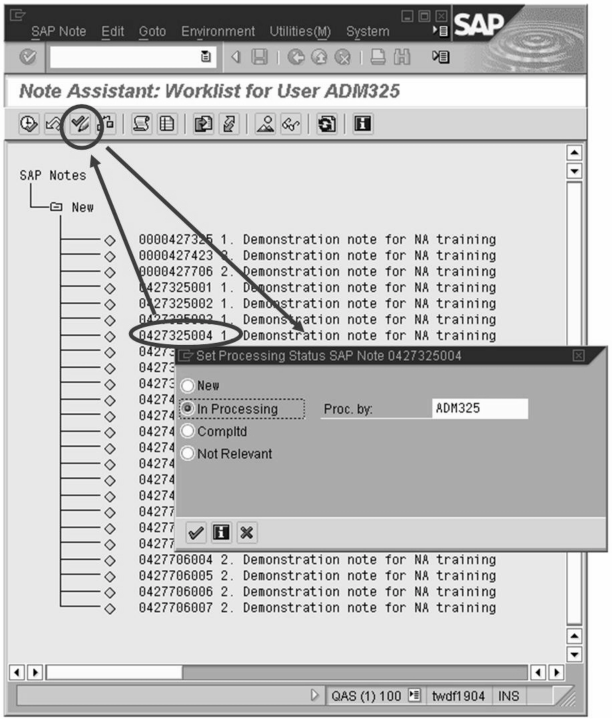
```
SAP Notes
  New
    0000427325 1. Demonstration note for NA training
    0000427423 3. Demonstration note for NA training
    0000427706 2. Demonstration note for NA training
    0427325001 1. Demonstration note for NA training
    0427325002 1. Demonstration note for NA training
    0427325003 1. Demonstration note for NA training
    0427325005 1. Demonstration note for NA training
    0427325006 1. Demonstration note for NA training
    0427325007 1. Demonstration note for NA training
    0427423001
    0427423002
    0427423003
    0427423004
    0427423005
    0427423006
    0427423007
    0427706001
    0427706002
    0427706003 2. Demonstration note for NA training
    0427706004 2. Demonstration note for NA training
    0427706005 2. Demonstration note for NA training
    0427706006 2. Demonstration note for NA training
    0427706007 2. Demonstration note for NA training
  In Processing
    0427325004 1. Demonstration note for NA training
```

SAP Note 0427325004 can be implemented

Figure 162: Check Note

You can check the implementation status of an SAP Note.

For more comprehensive information, click the yellow question mark in the information window.



Note Assistant: Worklist for User ADM325

```
SAP Notes
  New
    0000427325 1. Demonstration note for NA training
    0000427423 2. Demonstration note for NA training
    0000427706 2. Demonstration note for NA training
    0427325001 1. Demonstration note for NA training
    0427325002 1. Demonstration note for NA training
    0427325003 1. Demonstration note for NA training
    0427325004 1 Demonstration note for NA training
    0427
    04273
    04273
    04274
    04274
    04274
    04274
    04274
    04274
    04274
    04277
    04277
    04277
    0427706004 2. Demonstration note for NA training
    0427706005 2. Demonstration note for NA training
    0427706006 2. Demonstration note for NA training
    0427706007 2. Demonstration note for NA training
```

Set Processing Status SAP Note 0427325004

○ New
● In Processing   Proc. by: ADM325
○ Compltd
○ Not Relevant

QAS (1) 100 | twdf1904 | INS

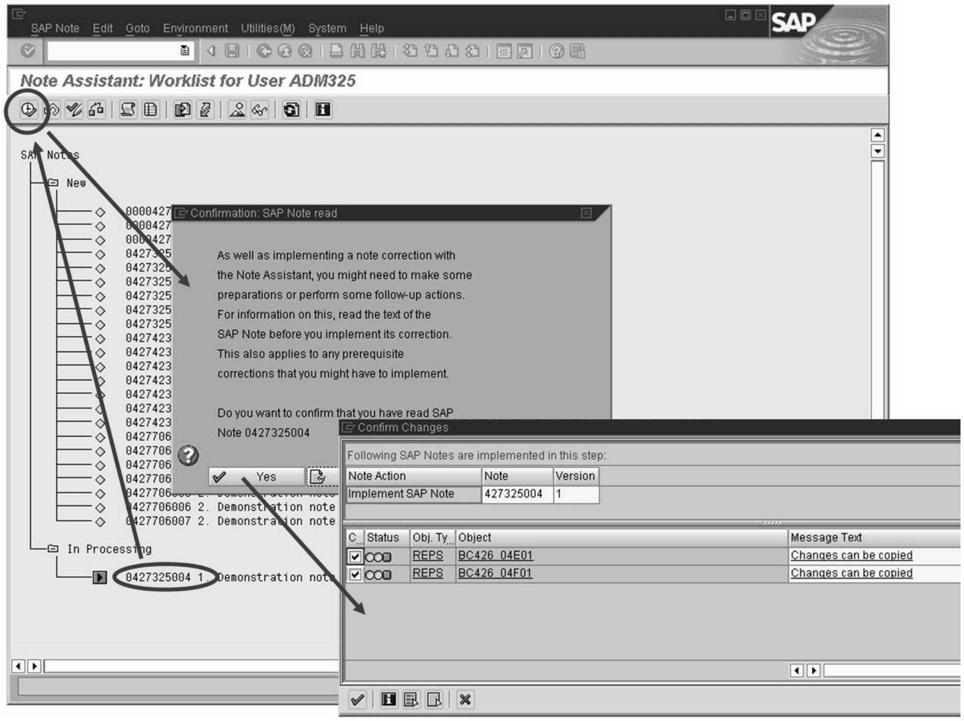
Figure 163: Set Processing Status

You can classify an SAP Note or set the processing status.

If you want your colleague to process the SAP Note, enter your colleague's user name and set the note to *In Processing*. It will appear in his or her worklist.

If the SAP Note does not correspond to your situation, you can set it to *Not Relevant*.

After implementing an SAP Note, you can set the status to *Compltd.*.



Note Assistant: Worklist for User ADM325

```
SAP Notes
  New
    0000427
    0000427
    0000427
    0427325
    0427325
    0427325
    0427325
    0427325
    0427325
    0427423
    0427423
    0427423
    0427423
    0427423
    0427423
    0427423
    0427706
    0427706
    0427706
    0427706
    0427706
    0427706006 2. Demonstration note
    0427706007 2. Demonstration note
  In Processing
    0427325004 1. Demonstration note
```

Confirmation: SAP Note read

As well as implementing a note correction with the Note Assistant, you might need to make some preparations or perform some follow-up actions. For information on this, read the text of the SAP Note before you implement its correction. This also applies to any prerequisite corrections that you might have to implement.

Do you want to confirm that you have read SAP Note 0427325004

Confirm Changes

Following SAP Notes are implemented in this step:

Note Action	Note	Version
Implement SAP Note	427325004	1

C...	Status	Obj. Ty...	Object	Message Text
☑	○○●	REPS	BC426_04E01	Changes can be copied
☑	○○●	REPS	BC426_04F01	Changes can be copied

Figure 164: Implement Note

To implement the SAP Note, select the respective note and choose *Implement SAP Note*.

You must confirm that you have carefully read the SAP Note because it may contain pre-steps or postprocessing activities that are described in the long text only.

You are prompted to include your changes in a transport request.

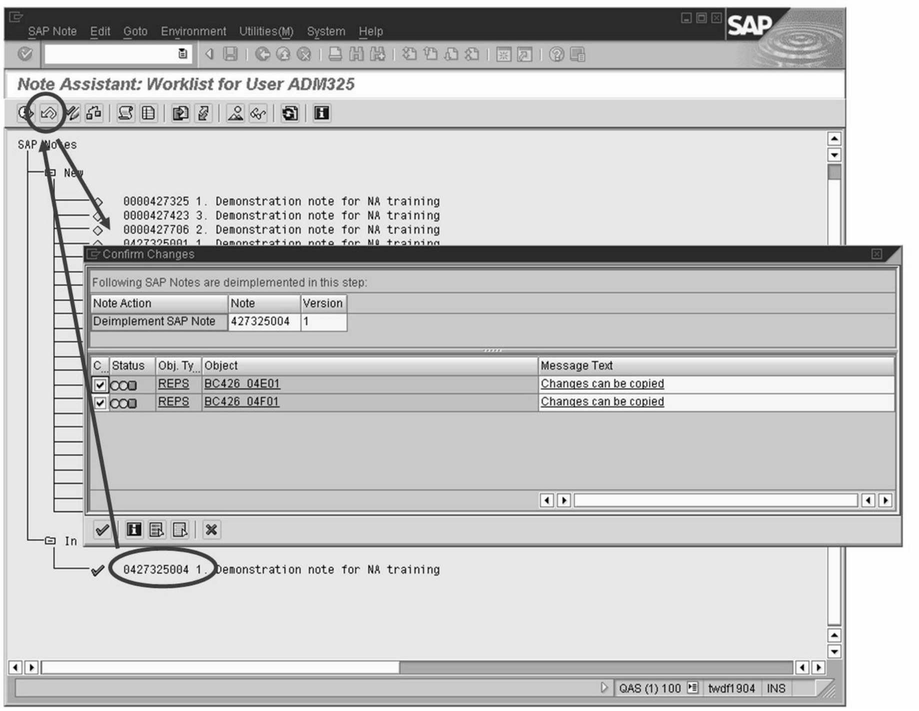
Traffic light indicates whether the changes can be copied, i.e. whether the correction instructions can be implemented. Double-clicking the message gives you more information.

Always perform a syntax check before you confirm the changes.

A log file is automatically created for each note that records all the steps that are performed in the SAP Note Assistant.

For example, you can enter a comment to leave a message for another processor.

You cannot change the information that is entered in the log file.



Note Assistant: Worklist for User ADM325

SAP Notes
New
0000427325 1. Demonstration note for NA training
0000427423 3. Demonstration note for NA training
0000427706 2. Demonstration note for NA training
0427325001 1. Demonstration note for NA training

Confirm Changes

Following SAP Notes are deimplemented in this step:

Note Action	Note	Version
Deimplement SAP Note	427325004	1

C..	Status	Obj. Ty..	Object	Message Text
☑	○○■	REPS	BC426_04E01	Changes can be copied
☑	○○■	REPS	BC426_04F01	Changes can be copied

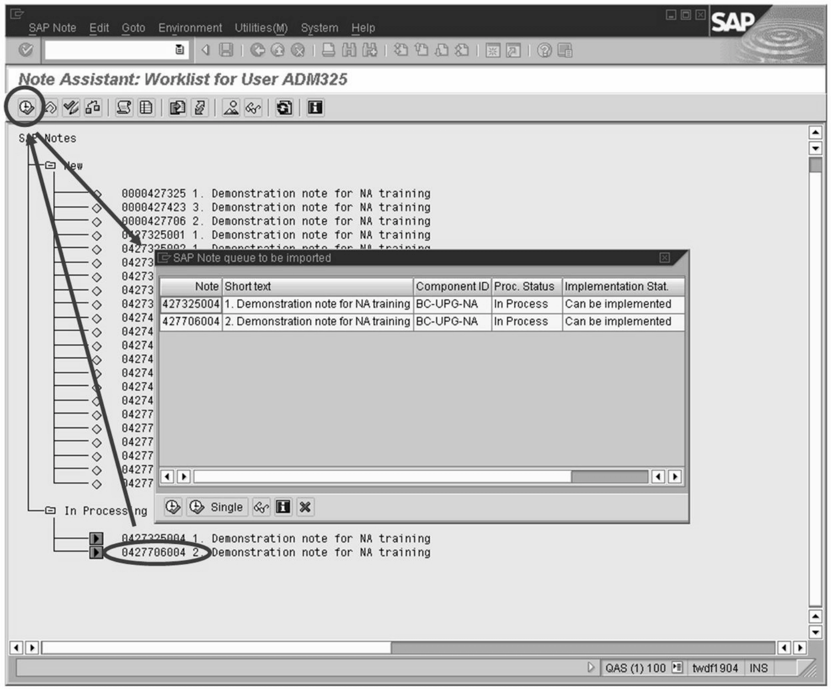
In
✓ 0427325004 1 Demonstration note for NA training

QAS (1) 100 twdf1904 INS

Figure 165: Cancel SAP Note Implementation

In exceptional cases, you may want to deimplement an SAP Note that you have implemented. You can do this by choosing *Reset SAP Note Implementation*.

If you do this, the changes that have been introduced with the implementation of the SAP Note are reset and the SAP Note is reset to the status that it had before you implemented it.



Note Assistant: Worklist for User ADM325

```
SAP Notes
  New
    0000427325 1. Demonstration note for NA training
    0000427423 3. Demonstration note for NA training
    0000427706 2. Demonstration note for NA training
    0427325001 1. Demonstration note for NA training
    0427325002 1. Demonstration note for NA training
    04273
    04273
    04273
    04273
    04274
    04274
    04274
    04274
    04274
    04274
    04274
    04277
    04277
    04277
    04277
    04277
    04277
  In Processing
    0427325004 1. Demonstration note for NA training
    0427706004 2. Demonstration note for NA training
```

SAP Note queue to be imported

Note	Short text	Component ID	Proc. Status	Implementation Stat.
427325004	1. Demonstration note for NA training	BC-UPG-NA	In Process	Can be implemented
427706004	2. Demonstration note for NA training	BC-UPG-NA	In Process	Can be implemented

Figure 166: Handling Notes with Preconditions

Sometimes correction instructions can only be implemented if you implement other correction instructions before. These preconditions are described in the SAP Note. Because the SAP Note Assistant strictly follows the information in the SAP Note, you must implement the required SAP Notes before.

You must download the required SAP Notes and upload them in the SAP Note Assistant, before you can implement them.

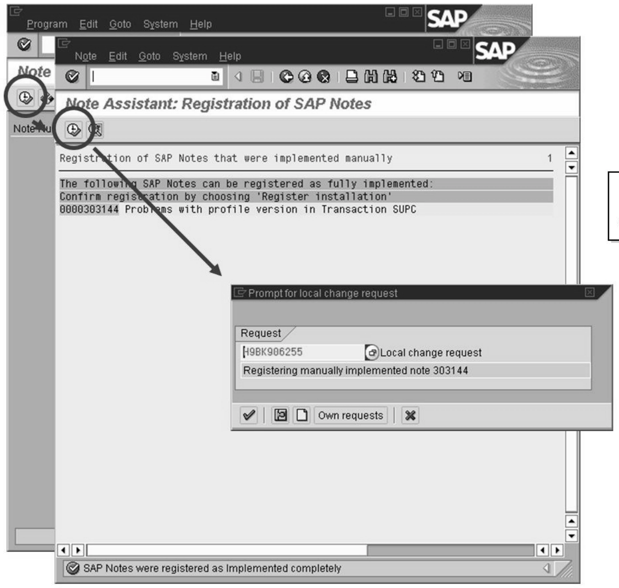
Note: Make sure that you carefully read the prerequisite SAP Notes, too.

The SAP Note Assistant then tries to implement as many SAP Notes as possible in one step.

The SAP Note Assistant is closely related to the Modification Assistant.

From within the SAP Note Assistant you can directly switch to the Modification Browser (transaction SE95).

The Modification Browser displays an overview of all objects that are changed with this SAP Note.


Program Edit Goto System Help
Note Edit Goto System Help
SAP
Note
Note Nu
Note Assistant: Registration of SAP Notes
Prompt for local change request
Request
Local change request
Own requests

```
Registration of SAP Notes that were implemented manually                1

The following SAP Notes can be registered as fully implemented:
Confirm registration by choosing 'Register installation'
0000303144 Problems with profile version in Transaction SUPC
```

Request: H9BK906255 Local change request

Registering manually implemented note 303144

SAP Notes were registered as Implemented completely

Program SCWN_REGISTER_NOTES

Figure 167: Registering manually implemented SAP Notes

If you have implemented corrections from SAP Notes manually before using the SAP Note Assistant, you must make these SAP Notes known to the SAP Note Assistant. Otherwise, the SAP Note Assistant does not know that this correction has already been implemented in your system and implements it again. This may lead to inconsistencies.

Registration offers the following benefits:

1. All the SAP Notes that have been implemented in your system are displayed in the Note Browser.
2. If a manually implemented SAP Note is a prerequisite for an SAP Note that you want to implement using the SAP Note Assistant, the SAP Note Assistant recognizes that the prerequisite SAP Note has already been implemented and does not prompt you to implement it again.
3. For future SAP Support Packages or upgrades, the manually implemented SAP Notes are displayed during the modification adjustment in the category SAP Note corrections. The system determines whether these SAP Notes have to be implemented again or reset to their original status.

To register the SAP Notes, the system loads the SAP Note and checks whether it is valid for your release and SAP Support Package level.

Problems with the SAP Note Assistant:

- If implementation fails, create a problem message in SAP Service Marketplace and assign it to the component of the SAP Note (for example, FI-AR-CR).
- If you experience any other problems with the SAP Note Assistant, create a problem message and assign it to the component BC-UPG-NA.

Exercise 13: SAP Note Assistant

Exercise Objectives

After completing this exercise, you will be able to:

- Use the SAP Note Assistant

Business Example

Single errors in an SAP system can be fixed by SAP Notes. SAP Notes can contains the corrected coding. The administrator should be able to use the SAP Note Assistant to automatically implement these SAP Notes into an SAP system. The implementation of SAP Notes with the SAP Note Assistant is also often a task of a developer. In any case the administrator has to be able to monitor the implemented SAP Notes.

Task 1:

Apply an SAP Note with the SAP Note Assistant.

1. The SAP program *SAPBC426_##* should display flights with their seats. But the program is not working properly. (## stands for your group number)

   Apply the SAP Note *4273250##* to the program *SAPBC426_##*, using transaction SNOTE. First, check the program and its result using transaction SE38. Also, read the description of the SAP Note and check if the SAP Note can be implemented. Then, implement it. You may complete the implementation of the SAP Note.

Task 2:

Use the SAP Note Assistant to deimplement or remove an SAP Note.

1. After setting the status for a SAP Note to *Completed*, it is somewhat more difficult to find the SAP Note for deimplementation. For this, you start transaction SNOTE and choose *Goto → SAP Note Browser* and select SAP Notes in processing status *Finished*.

   Note: If you had not chosen to set the processing status for your SAP Note to *Completed* in the previous step, it would have been more obvious how to deimplement the SAP Note, because it would not have vanished from the entrance screen of transaction SNOTE. You could have chosen to deimplement the SAP Note from this screen.

Continued on next page

Task 3:

Use the SAP Note Assistant to implement two dependent SAP Notes.

1. Try to implement SAP Note *4277060##*.

Solution 13: SAP Note Assistant

Task 1:

Apply an SAP Note with the SAP Note Assistant.

1. The SAP program *SAPBC426_##* should display flights with their seats. But the program is not working properly. (## stands for your group number)

Continued on next page

Apply the SAP Note *4273250##* to the program *SAPBC426_##*, using transaction SNOTE. First, check the program and its result using transaction SE38. Also, read the description of the SAP Note and check if the SAP Note can be implemented. Then, implement it. You may complete the implementation of the SAP Note.

a) In client 100 start transaction SE38, enter the program name **SAPBC426_##** and choose *Execute*.

Using *F4 help*, enter a valid airline carrier ID (such as **LH**) and start the selection. Note that the result looks very poor.

Start transaction SNOTE and display the SAP Note text of your SAP Note *4273250##* by double-clicking the corresponding line. Read the SAP Note text.

➡ **Note:** If there is some *ADS SOAP runtime exception*, you can ignore this.

Change the processing status of the SAP Note to *In Processing* by choosing the *Set Processing Status* icon. Make sure that the status of your SAP Note is *In Processing*, not the status *New*.

Check if the SAP Note can be implemented by clicking *Check SAP Note*.

Choose *Implement Note*.

- Click away the message that the connection to SAPNet failed.
- Confirm that you have read the SAP Note with *Yes*.
- Create a new transport request with the description **SAP Note ##**.
- Confirm changes by choosing *Continue*.
- Set the processing status for your SAP Note to *Completed*.

Start transaction SE38, enter the program name **SAPBC426_##** and choose *Execute*.

Using *F4 help*, enter a valid airline carrier ID (such as **LH**) and start the selection. Please see that the last column *Seats max* remains empty.

Continued on next page

Task 2:

Use the SAP Note Assistant to deimplement or remove an SAP Note.

1. After setting the status for a SAP Note to *Completed*, it is somewhat more difficult to find the SAP Note for deimplementation. For this, you start transaction SNOTE and choose *Goto → SAP Note Browser* and select SAP Notes in processing status *Finished*.

   Note: If you had not chosen to set the processing status for your SAP Note to *Completed* in the previous step, it would have been more obvious how to deimplement the SAP Note, because it would not have vanished from the entrance screen of transaction SNOTE. You could have chosen to deimplement the SAP Note from this screen.

   a) If you followed the instructions given above, you are now able to call the function *Reset SAP Note Implementation*. Ensure that only your SAP Note is selected for deimplementation. Confirm the following pop-ups.

      Check that the program runs again in its old, incorrect fashion.

Continued on next page

Task 3:

Use the SAP Note Assistant to implement two dependent SAP Notes.

1. Try to implement SAP Note *4277060##*.

   a) Start transaction SNOTE and display the SAP Note text of your SAP Note *4277060##* by double-clicking the corresponding line. Read the SAP Note text.

   Change the processing status of the SAP Note to *In Processing* by choosing the *Set Processing Status* icon. See that in the entry screen of transaction SNOTE your SAP Note now vanishes from display below the status *New* and appears below status *In Processing*.

   Click *Check SAP Note* to check whether the SAP Note can be implemented.

   Choose *Implement Note*.

   - Click away the message that the connection to SAPNet failed.
   - The *SAP Note queue to be imported* popup is displayed. Note that the dependency to SAP Note *4273250##* has been identified automatically and that the SAP Note has *In Processing* status. This works in the training system because the needed SAP Note is already available in the system (through upload).
   - Choose *Install*.
   - Confirm the following pop-ups by choosing *Continue*.
   - Set the processing status for your SAP Notes to *Completed*.

   Start transaction SE38, enter the program name **`SAPBC426_##`** and choose *Execute*.

   Using *F4 help*, enter a valid airline carrier ID (such as **`LH`**) and start the selection. Please see that the last column *Seats max.* is now showing some values. **Great! :-)**

   Now the program works as designed.

Lesson Summary

You should now be able to:

- List the advantages of automatically implementing SAP Notes
- List the goals, functions, and the features of SAP Note Assistant
- Describe Note Browser and list the preconditions for handling SAP Notes

Lesson: Support Packages

Lesson Overview

In this lesson, you will learn about modification adjustment and downtime when applying Support Packages. In addition, you will learn about the application of Support Packages in a three-system landscape.

Lesson Objectives

After completing this lesson, you will be able to:

- Describe the import of Support Packages
- Reduce the downtime when applying Support Packages
- Perform a modification adjustment during the import of Support Packages

Business Example

You want to import functional enhancements into your system and correct errors that may occur. SAP makes Support Packages available for this purpose.

Maintenance Using SAP Support Patches

An SAP system consists of different software components. All of these components are regularly updated using Support Packages; that is, errors are corrected and new functions are provided using Support Packages, for example, due to legal changes.

The SAP system should always be kept at the current correction status so that errors in the standard release are preventativly removed.

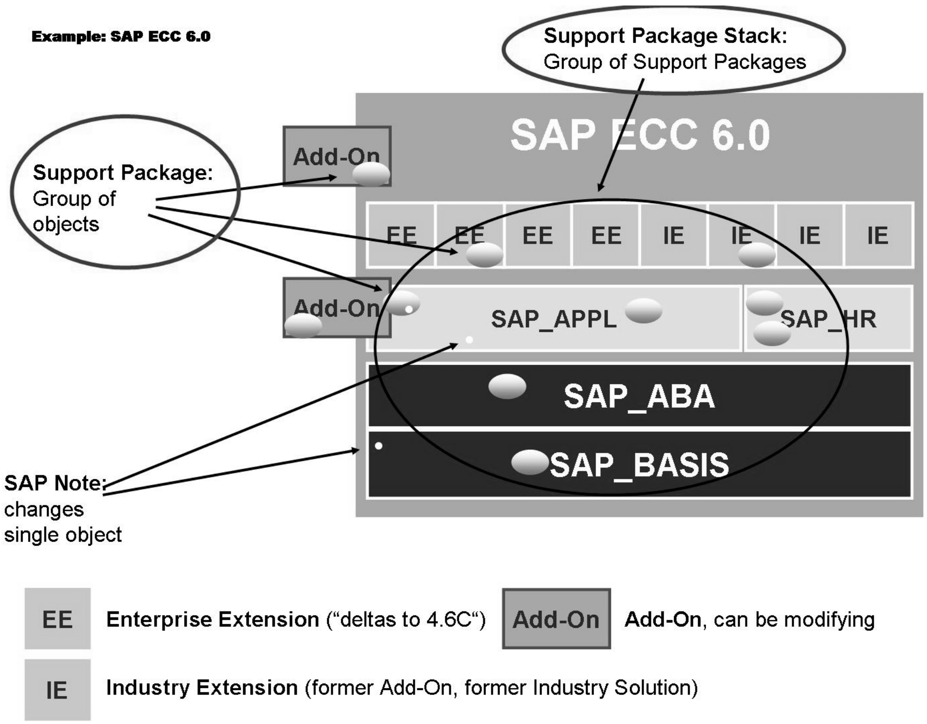

Example: SAP ECC 6.0
Support Package Stack: Group of Support Packages
SAP ECC 6.0
Add-On
Support Package: Group of objects
EE
EE
EE
EE
IE
IE
IE
IE
Add-On
SAP_APPL
SAP_HR
SAP_ABA
SAP_BASIS
SAP Note: changes single object
EE
Enterprise Extension ("deltas to 4.6C")
Add-On
Add-On, can be modifying
IE
Industry Extension (former Add-On, former Industry Solution)

Figure 168: What is an Support Packages, what is a Support Package Stack?

What Is an SAP Support Package?

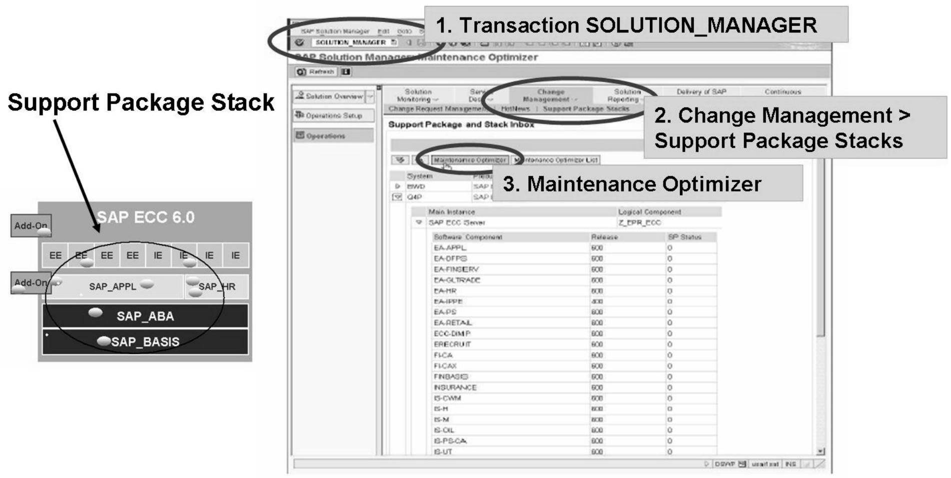
An SAP Support Package is a quantity of corrected or new SAP objects. Support Packages are required to correct errors in various components. This is done by replacing erroneous objects with corrected versions of these objects.

A Support Package Stack is a combination of Support Packages, recommended by SAP.

Each software component has a separate sequence of Support Packages.

Requesting and Applying Support Packages

To request Support Packages and Support Package Stacks for SAP NetWeaver 7.0 (and above) based systems requires the Maintenance Optimizer!



1. Transaction SOLUTION_MANAGER

2. Change Management > Support Package Stacks

3. Maintenance Optimizer

System	Product
BWD	SAP
Q4P	SAP

Main Instance	Logical Component
SAP ECC Server	Z_EPR_ECC

Software Component	Release	SP Status
EA-APPL	600	0
EA-DFPS	600	0
EA-FINSERV	600	0
EA-GLTRADE	600	0
EA-HR	600	0
EA-IPPE	400	0
EA-PS	600	0
EA-RETAIL	600	0
ECC-DIMP	600	0
ERECRUIT	600	0
FI-CA	600	0
FI-CAX	600	0
FINBASIS	600	0
INSURANCE	600	0
IS-CWM	600	0
IS-H	600	0
IS-M	600	0
IS-OIL	600	0
IS-PS-CA	600	0
IS-UT	600	0

Maintenance Optimizer in Solution Manager

Figure 169: Requesting Support Packages

Before you can import Support Packages, you must first download them from the SAP Service Marketplace. You should apply whole Support Package stacks rather then other combinations of Support Packages.

For requesting Support Packages for newer releases you need the Maintenance Optimizer for approving the download. The Maintenance Optimizer helps you to calculate the correct queue of Support Packages for you individual SAP system.

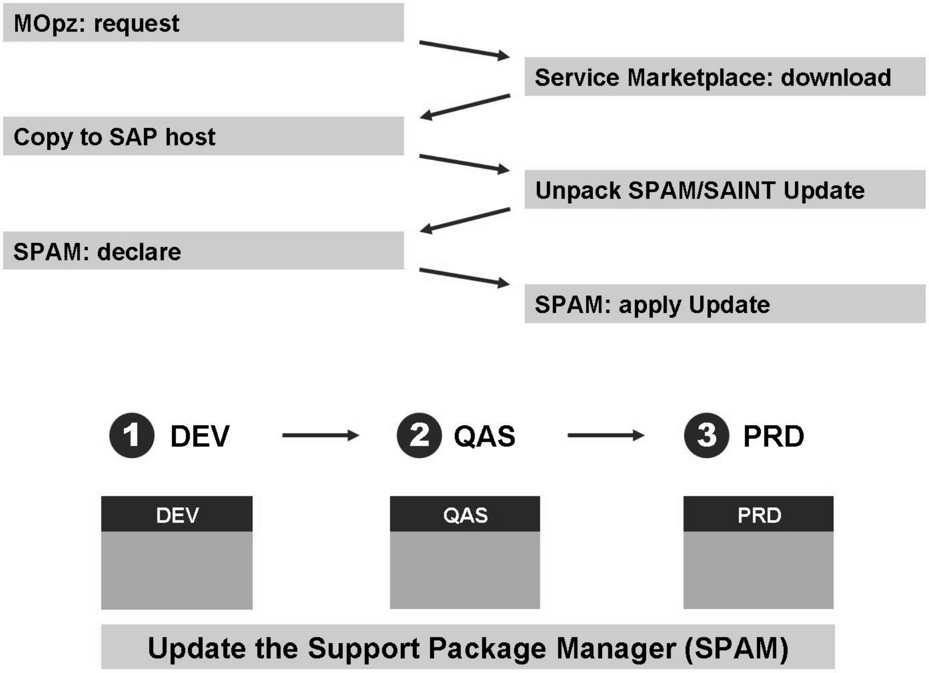


Figure 170: Support Package Manager: SPAM/SAINT Update

The Support Package Manager (transaction SPAM) is the tool for applying Support Packages.

Before using the Support Package Manager, you should patch the tool itself with a *SPAM/SAINT update*. This is a patch to enhance the functions of SPAM and SAINT, for example when dealing with new attributes from new Support Packages.

Transaction SAINT allows you to install and upgrade an Add-Ons.

Note: Please note that *SPAM/SAINT updates* come in German and English only. If you work in another language, new or updated user interface elements and texts might not be properly displayed. This is why SAP recommend to log on in German or English if you want to use the Support Package Manager.

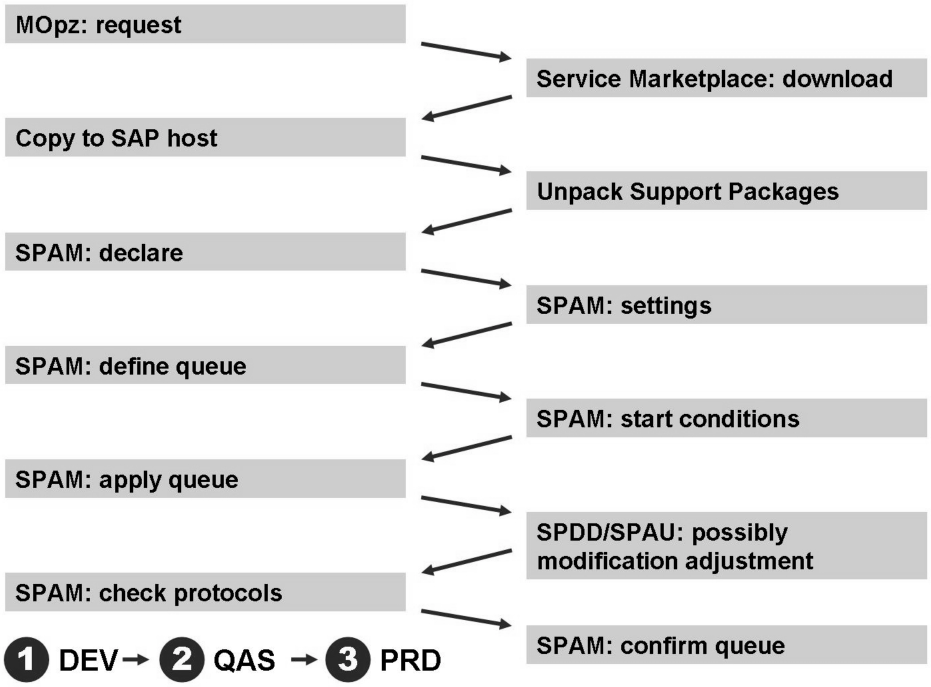


Figure 171: Applying Support Packages

The SAP Support Package Manager ensures that only Support Packages that match your system are displayed in the queue. Support Packages that are intended for a different release or an Add-on that is not installed do not appear in the queue, even if you have loaded them into your SAP system.

Alternatively, you can compile a shared queue for all software components in the system, by choosing *All Components*.

The queue contains the Support Packages available for the selected component(s) in your system and any required Conflict Resolution Transports (CRTs) and associated Add-on Support Packages.

If Support Packages in the queue have connections to Support Packages for another software component (other predecessor relationship, required CRT), the queue is extended with other Support Packages until all predecessor relationships are fulfilled.

The following rules apply when you create a queue:

- Support Packages for a selected component are placed in the queue according to their sequence.
- If Support Packages in the queue are related to Support Packages for another software component (predecessor relationship, CRT required), the extra Support Packages are added to the queue until all predecessor relationships are included.

If the system has calculated a consistent queue, you can display it and check it by choosing the *Calculated Queue* tab.

Test scenario or standard scenario: To set the desired scenario, choose *Extras → Settings*.

- Use the test scenario to check whether a modification adjustment is necessary, or whether there are any conflicts or problems (such as unreleased repairs), before you actually import the Support Package. Use this scenario to estimate and possibly minimize the time and effort needed to import the Support Packages. This scenario imports no data or objects into your SAP system. If errors occur, you can continue the import without needing to remove the errors. You need to select the test scenario explicitly. Note that the queue is empty after you run the test scenario and you need to define it again. You then have to select the standard scenario explicitly.
- The standard scenario imports all Support Packages in the queue completely. If errors occur, you can only continue and complete the import after you have removed the errors. If you selected the standard scenario, you can choose between the standard import mode and the "Downtime-minimized" import mode to reduce downtime.

Import Process

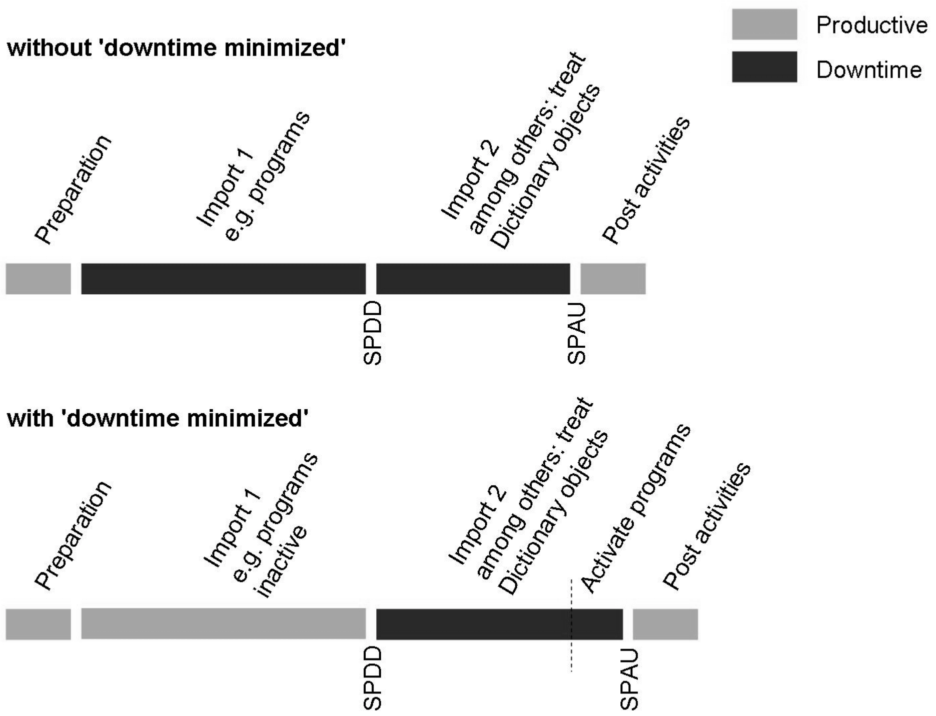


Figure 172: Inactive Import

Note: Note that during the time between the modules *Preparation* and *Import 2* (or rather, until the modification adjustment), no transports other than the modification adjustment transport can take place, and no manual changes can be made to Repository objects (ABAP programs, dictionary objects). The time period between *Import 1* and *Import 2* must be kept as short as possible.

Due to the size and scope of current Support Packages, importing Support Packages requires longer system downtime. The system is not stopped and restarted during the import process, but it should also not be used productively during this time. This restriction is a disadvantage for many production systems. To reduce the downtime during the import of Support Packages, an import procedure has been developed that allows you to import a large proportion of the objects to be imported during production operation. These objects are report sources and report texts. The downtime can be significantly reduced if a SAP Support Package contains a high proportion of report sources and report texts (the proportion is up to 80%.). With the inactive import, the objects are imported into the database in an inactive state and are therefore largely 'invisible' to the system. You can therefore continue to use the system productively.

Modification Adjustment

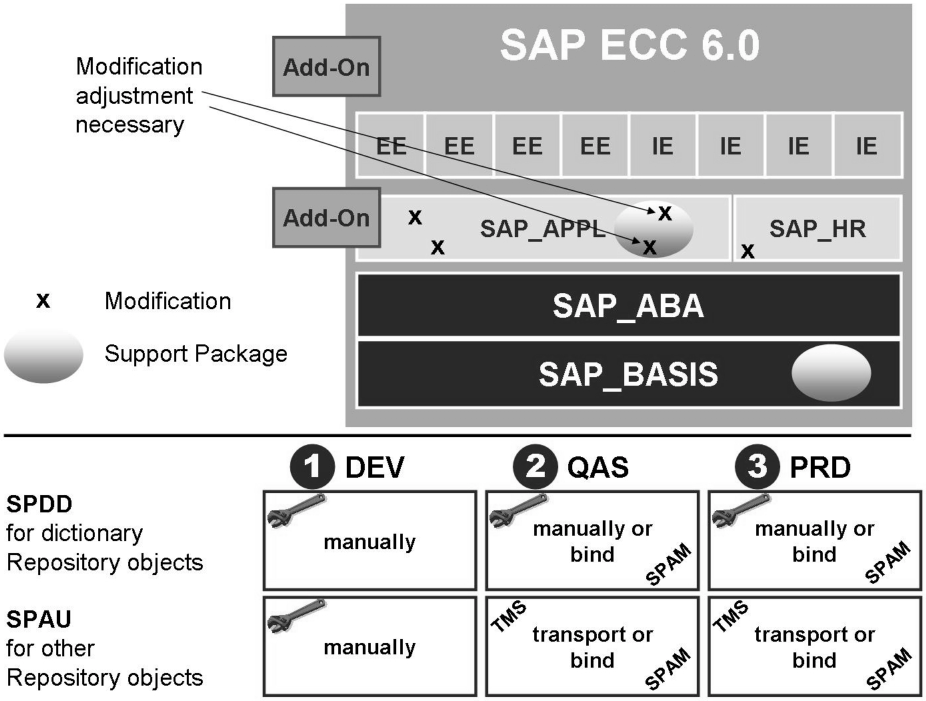


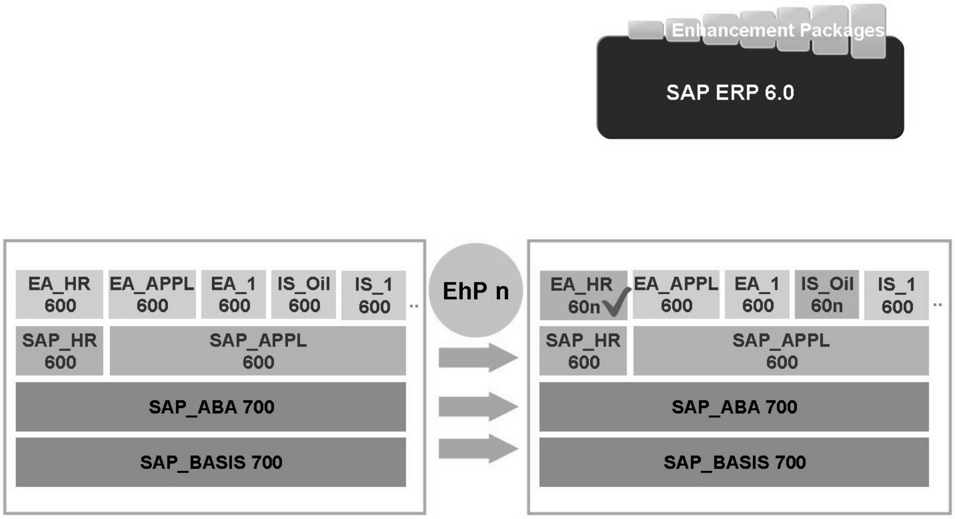
Figure 173: SPAM: Performing the Modification Adjustment

1. The Support Package Manager stops to allow the modification adjustment.
2. So that your developers can perform the modification adjustment, create a transport request in the Transport Organizer and tasks inside the request for the developers.
3. Ask the developers to perform the modification adjustment for their objects. After the adjustment is complete, the developers must release the tasks and inform you. **The adjustment can be performed in any client.**
4. If you left the transaction, call the SAP Support Package Manager (transaction SPAM), and choose *Support Package → Import Queue*. The system prompts you again to perform the modification adjustment. As it has already been completed, ignore the message and choose *Continue*. The Support Package Manager completes the processing and displays the status.

The *Import logs of the queue* function displays logs for steps of the Support Package Manager that use the transport control program tp. After the queue has been successfully imported, you should always check these logs. To jump to the log display, choose *Goto → Log → Queue* in the initial screen of the Support Package Manager.

Confirm the successful import of the queue into your system. This ensures that you can import other Support Packages in future. Without this confirmation, it is not possible to import additional Support Packages. If you have not yet confirmed successfully imported Support Packages, you are prompted to confirm these packages during the upgrade of the system.

Enhancement Packages



Example: Enhancement Package n delivers new version of EA_HR and IS_OIL, but only new functionality in EA_HR is being activated

Figure 174: Enhancement Packages

Hint: For detailed information look at http://service.sap.com/erp-ehp

Exercise 14: Support Packages

Exercise Objectives

After completing this exercise, you will be able to:

- Perform the modification adjustment when implementing Support Packages

Business Example

In order to correct errors in the SAP standard and also to apply the newest functions, the system administrator has to implement Support Packages on a regular basis.

Task 1:

Adjust Dictionary Repository objects using transaction SPDD.

1. When applying an Support Package, the system checks if Dictionary Repository object modification adjustment is necessary. If this is the case, the system requires you to use transaction SPDD to adjust the modification or to return to the SAP standard. With the Support Package for this exercise this is the case for table *ADM325X##*. The instructor will call transaction SPDD to display the objects to be adjusted and ask you to adjust your own object.
2. After all participants have completed the adjustment, the instructor returns to the Support Package Manager (transaction SPAM) and continues the application of the Support Package.

Task 2:

Adjust the non-Dictionary Repository objects using transaction SPAU.

1. When applying a Support Package, the system also checks if for other Repository objects, as those from SPDD, modification adjustment is necessary. If this is the case, the system requires you to use transaction SPAU to adjust the modification or to return to the SAP standard. With the Support Package for this exercise this is the case for program *ADM325X##*. This is because this program was repaired in a previous exercise. The instructor will call transaction SPAU to display the objects to be adjusted and ask the participants to adjust their own objects.
2. After all participants have completed the adjustment, the instructor returns to the Support Package Manager (transaction SPAM) and continues the application of the Support Package.

Continued on next page

Result

After adjusting all modifications, that are necessary, and after successfully applying the Support Package queue, the instructor will check the logs in transaction SPAM. Afterwards, the instructor will confirm the Support Packages using transaction SPAM.

Solution 14: Support Packages

Task 1:

Adjust Dictionary Repository objects using transaction SPDD.

1. When applying an Support Package, the system checks if Dictionary Repository object modification adjustment is necessary. If this is the case, the system requires you to use transaction SPDD to adjust the modification or to return to the SAP standard. With the Support Package for this exercise this is the case for table *ADM325X##*. The instructor will call transaction SPDD to display the objects to be adjusted and ask you to adjust your own object.

   a) Log on to the development system with your user in client **`100`** and start transaction SPDD. The *Modification Adjustment Dictionary: Object Selection* screen is displayed.

   In the *Selections* area, enter your user ID in the *Last changed by* field and choose *Execute*. The *Modification Adjustment Dictionary* screen appears.

   Expand the folder in front of *Modifications and enhancements Without Modification Assistant* until you see the dictionary object that you modified.

   Choose the legend icon to get an explanation of the status on the object.

   b) Select the yellow traffic light of the object *ADM325X##*. An adjustment proposal is displayed. You can either edit or accept the proposal. Please accept the proposal in this case. To accept the SAP proposal, choose *Accept Proposal*. Accept the name of the append with *Continue* and choose the package *ZPACKAGE_##*, you created, for the append structure. Assign the retained modification to the transport request, the instructor created for SPDD adjustment. SPDD suggests to keep all SAP standard fields and to move your customer field into an append structure. Use the *back arrow* to return to the previous screen. SPDD now asks, if you want to return to reset your modification. This is ok, because your customer field is now in the append structure. An append structure is not a modification, but an enhancement. So the table itself can be reset.

Continued on next page

Once the object has been adjusted, the system automatically assigns the status *Object has been adjusted*, if you confirm the corresponding popup.

Note: The object *ADM325X##* now contains your old modification (the field *ZZBIRTHPLACE*) in an append structure and the new field *TELNO* that came with the Support Package from SAP.
If you do not adjust your modification, you will lose the field *ZZBIRTHPLACE* and its data from your table.

So it is very important to perform the modification adjustment **now**. Later you would be able to add the field *ZZBIRTHPLACE* again, of course. But you would have **lost** the **data**.

Assign your object to the transport request for SPDD adjustment, which is provided by the instructor.

Release your task using transaction SE09.

Do not release the transport request itself.

2. After all participants have completed the adjustment, the instructor returns to the Support Package Manager (transaction SPAM) and continues the application of the Support Package.

   a) The instructor continues the application of the Support Package.

Task 2:

Adjust the non-Dictionary Repository objects using transaction SPAU.

1. When applying a Support Package, the system also checks if for other Repository objects, as those from SPDD, modification adjustment is necessary. If this is the case, the system requires you to use transaction SPAU to adjust the modification or to return to the SAP standard. With the Support Package for this exercise this

Continued on next page

is the case for program *ADM325X##*. This is because this program was repaired in a previous exercise. The instructor will call transaction SPAU to display the objects to be adjusted and ask the participants to adjust their own objects.

a) Log on to the development system in client `100` and start transaction SPAU. The *Modification Adjustment Dictionary: Object Selection* screen is displayed.

   In the *Selections* area, enter your user ID in the *Last changed by* field and choose *Execute*. The *Modification Adjustment Dictionary* screen appears.

   Expand the folder in front of *Modifications and enhancements With Modification Assistant* until you see the program you modified.

   Choose the legend icon to get an explanation of the status on the object.

b) Discard the changes and thus return to the SAP standard. Position your cursor on the object and choose *Modification → Reset to original*.

   The *Reset* object popup appears. Choose *Yes* to confirm the reset of the object.

   Assign the retained modification to the transport request for SPAU adjustment, which is provided by the instructor.

   Note: If you want to retain the modification, select the traffic light beside the program *ADM325-##*. Depending on your modification, you now enter the adjustment process using the Modification Assistant. Then, the split screen editor appears displaying the original object on the left and the modification on the right. Look over the code to see if you want to save your modifications or return to the original. Return to the main screen of SPAU, when you are finished. The reports traffic light is replaced by a green check mark to indicate that the adjustment process is complete.

   Release your task using transaction SE09.

   Do not release the transport request itself.

Continued on next page

2. After all participants have completed the adjustment, the instructor returns to the Support Package Manager (transaction SPAM) and continues the application of the Support Package.

   a) The instructor continues the application of the Support Package.

Result

After adjusting all modifications, that are necessary, and after successfully applying the Support Package queue, the instructor will check the logs in transaction SPAM. Afterwards, the instructor will confirm the Support Packages using transaction SPAM.

Lesson Summary

You should now be able to:

- Describe the import of Support Packages
- Reduce the downtime when applying Support Packages
- Perform a modification adjustment during the import of Support Packages

Lesson: SAP System Upgrade

Lesson Overview

In this lesson, you will get a brief overview about planning and performing an SAP system upgrade.

Lesson Objectives

After completing this lesson, you will be able to:

- Have an idea about planning an SAP system upgrade
- Roughly explain how an SAP system upgrade functions

Business Example

The SAP administrator of your company should at least roughly understand how an SAP system upgrade works.

Planning and performing an SAP system upgrade

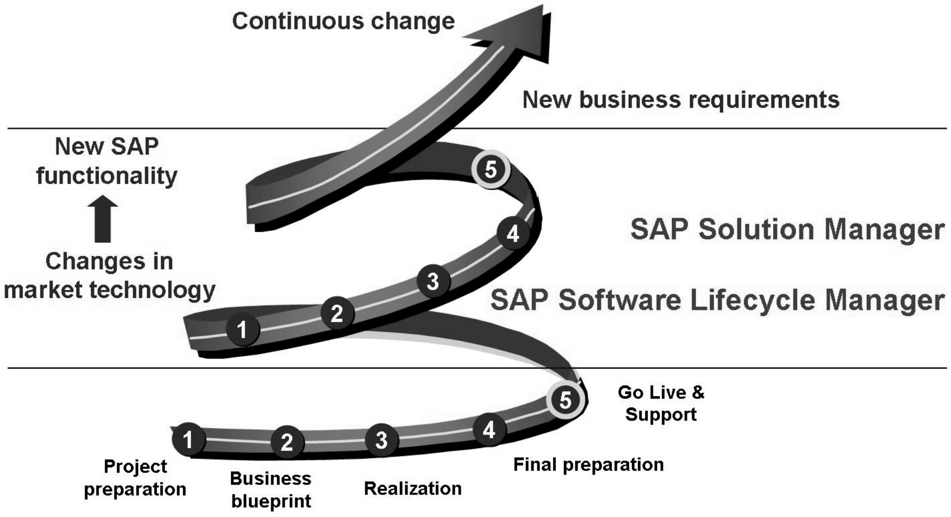


Figure 175: Upgrade Project Overview

The first thing that must happen is to recognize the need for an upgrade. This recognition of the need to upgrade could be at a number of different levels of upgrade recognition triggers; some are very operational in nature, while others are much more strategic. On the operational side, you may realize that you have a serious technical system limitation within your current environment.

There is a deadline for every release, at which time maintenance is stopped. This means that SAP delivers no more SAP Support Packages that adapt your processes for legal requirements and correct errors. See alias *pam* (product availability matrix) in the SAP Service Marketplace.

The new release can contain many new functions that your company requires. These new features can be on the basis side (new administration tools) or on the application side (new and enhanced business functionality).

The following factors are also important when deciding whether to upgrade:

- Costs: what will the upgrade cost me in total?
- Payback/ROI: will the upgrade have financial advantages for me?
- Benefit: what advantages will the upgrade have?
- Risks: are there risks involved in the upgrade?

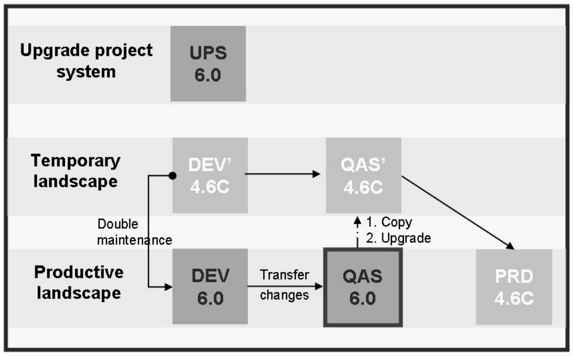
Change and improvements are integral parts of today’s business environment. SAP supports and embraces the process of continuous change as:

- New SAP functions becomes available
- Market technology changes
- New business requirements are defined

Focus on the typical problem areas such as solution gaps: assuming you documented your initial implementation, one of the easiest things to evaluate during upgrade planning is if any of the original gaps can be replaced by standard functions in the target release.

Review each component of new release functions. Determine how these enhancements can add value to your company’s operations.

System landscape



Schedule:

- Plan system downtime

PREPARE:

- OS upgrade?
- Hardware check
- DB upgrade?
- Modifications?
- Inconsistencies
- Checks
- ...

Perform Upgrade including follow-up activities:

- Technical Upgrade
- Adjust modifications
- Backup?

- Delta customizing
- New authorizations...
- Delta development
- Quality assurance
- User acceptance tests

Figure 176: Upgrading a Three System Landscape

When planning an SAP system upgrade, consider the following topics:

- Release Customizing
- Modification adjustment and migration of modifications into SAP standard
- Adaptation and testing of customer enhancements
- Adaptation and testing of interfaces
- End-user training
- Validation and testing of new releases

The biggest part of the upgrade project represents the adaptation of the current business processes to the new features of the new SAP system. There are not only changes to the Customizing and the repository. There are also changes for the end users, how to work with the new applications. When the development system has been upgraded to the new release it can still be necessary to develop bug fixes for the productive system which is still working with the start release.

Some additional technical upgrade process related issues are:

- Hardware requirement on new release: server, front-end, network
- Sizing forecast and system configuration for new release
- Planning of OS, DB, and SAP system upgrades
- Testing and validating of backup strategy for upgrade and on the new release
- Performing technical upgrade on the whole system landscape
- Post-upgrade activities including performance monitoring

Since the requirements of the new release are changing we need a sizing forecast which in turn requires changes to the configuration. Finally, the whole environment is concerned, for example, the server hardware, the client hardware, the network, the operating system, and the database. Especially when upgrading the operating system, other software on this computer has to be tested. For example, the backup software has to be tested.

Because of the new requirements, performance monitoring gets more important after the upgrade.

When deciding to which SAP release you want to upgrade, consider

- SAP's release strategy
- IT costs for upgrade and maintenance
- Cost for adapting business processes
- Business requirements versus SAP functions
- Costs versus Payback/ROI

An SAP system Upgrade is more than just a technical upgrade! The upgrade of a system landscape should be executed as a project. A significant amount of planning and lots of steps are required in order to perform a release change for the system landscape. On an average, customer upgrade projects last 6 months. But there are also upgrade projects with a duration of more than a year! So do not forget to think about

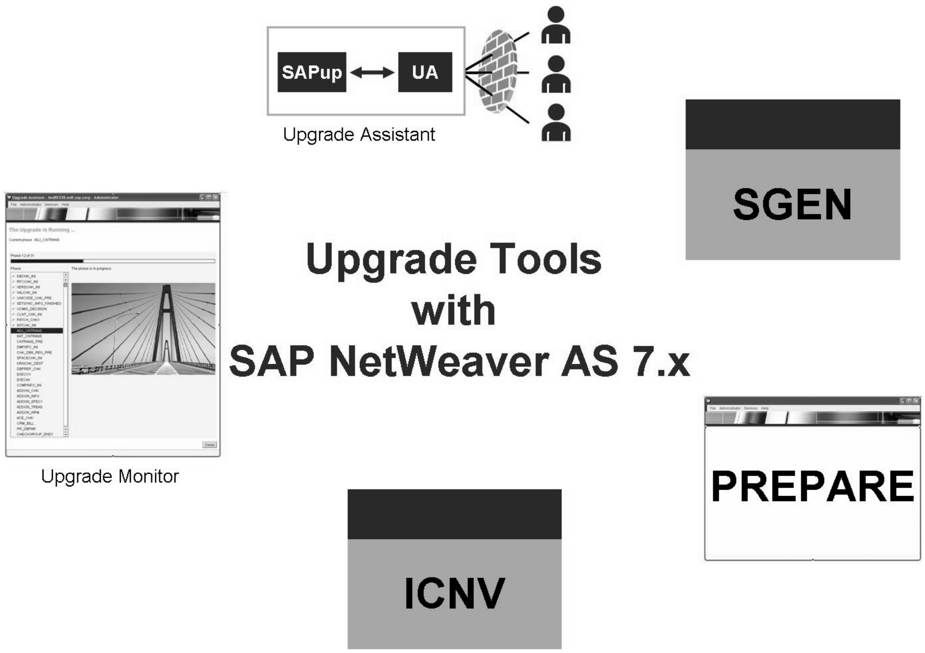
- Budgeting and resource planning
- Upgrade project in relation to other implementation/roll out activities
- Availability of resources
- End of maintenance of current release

A technical upgrade of one SAP system involves the following steps:

- When planning the upgrade, you should take decisions concerning the upgrade strategy. An exact upgrade schedule should be drawn up.

Hint: It is vital to read the SAP Notes concerning the upgrade! If you don't, your SAP system upgrade likely will fail!

- Certain requirements (software, hardware, operating system, database) must be met before you can upgrade the SAP system. Careful preparation of the upgrade is the best guarantee that it will run without errors.
- After you have performed the preparations for PREPARE, you can start it.
- PREPARE automatically checks if your system is properly configured for performing the upgrade. However, you must perform many tests and actions manually before starting the upgrade.
- Perform the upgrade.
- Follow-up activities include all actions necessary for completing the upgrade. SAP recommends completing the actions in the order given in the upgrade guidelines.



SAPup ↔ UA

Upgrade Assistant

SGEN

Upgrade Tools
with
SAP NetWeaver AS 7.x

Upgrade Monitor

PREPARE

ICNV

Figure 177: Tools for the Upgrade

Several tools help perform the upgrade. They are part of the technical upgrade process.

- PREPARE: PREPARE has to run prior to the SAP system upgrade. You have to repeat PREPARE so often, until it is error-free. The error messages are written to the log file *CHECKS.LOG*.

  PREPARE performs checks on the source release. For example, if the source release of the SAP system, the database, and the operating system is sufficient for this upgrade, if there is enough space in the database available, if no modified SAP objects are still in unreleased transport requests.

  PREPARE collects SAP Support Packages and Add-Ons for binding them to the SAP system upgrade. This is very important. If you do not bind enough SAP Support Packages to the upgrade, this will result in a loss of data during the SAP system upgrade. If you do not maintain your Add-Ons, the whole SAP system can become unstable and inconsistent.

  Furthermore, PREPARE imports tools in the source SAP system that are needed for the SAP system upgrade.

- ICNV (incremental table conversion): this transaction is one of the tools imported by PREPARE. With each new release, the structure of the tables in the database changes. Depending of the changes on a table structure, a table conversion may be necessary. These table conversions occur during the upgrade downtime (e.g. phase PARCONV).

  By using ICNV, you can convert many of these data records before downtime, during productive usage of the SAP system.

- Upgrade Assistant: with the Upgrade Assistant the SAP system upgrade process runs independently from a certain front-end. So you can control and monitor the progress of the upgrade from a number of different places. Front-end and upgrade processes are divided into separate components so that the entire upgrade does not terminate if a connection fails.

  This provides optimal support for a remote upgrade. The Upgrade Assistant provides an alert mechanism that lets you start an external program (for example: sending an SMS to your mobile).

  The Upgrade Assistant (server and GUI) is implemented in Java.

- Upgrade Monitor: the Upgrade Monitor lets you monitor the upgrade and helps you recognize any processes that have stopped.

  The monitor shows, which phase in currently running, which phases are succeeded and which ones will follow. So you get an overview of the whole Upgrade process.

- SGEN: with the new release 100% of all ABAP programs are exchanged. They are delivered in their source code. But not with their load. This is not possible, because the load depends on the local environment (operating system and hardware). When you call a program, its load is automatically generated if it does not already exist. This may, however, reduce production system performance for some time after the upgrade. To avoid this, you can use transaction SGEN to generate the loads.

Hint: If there is time left in your downtime window, generate the loads directly after the upgrade.

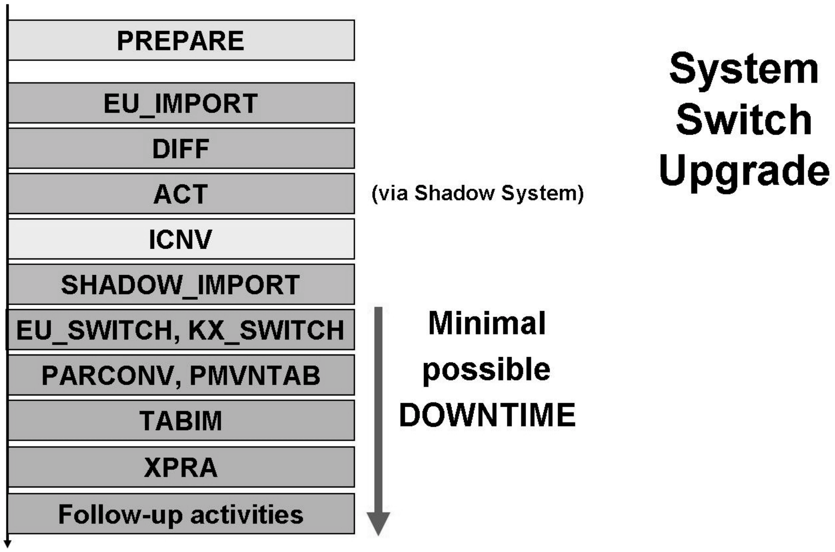


Figure 178: Phases of an SAP system upgrade

- PREPARE: preparations before the actual upgrade
- EU_IMPORT: import of the new Repository (as a shadow Repository)
- DIFF: coping of all modifications, own development, bounded Support Packages and Add-Ons to the shadow Repository
- ACT: SPDD modification adjustment and activation of table definitions in the shadow Repository
- ICNV: converting of large table structures during productive operation
- SHADOW_IMPORT: import of new tables, that do not harm the productive running system
- EU_SWITCH, KX_SWITCH: switching to the new Repository (the shadow Repository) and switching to the new Kernel
- PARCONV, PMVNTAB: conversion of table structures and move nametab of table structures
- TABIM: import of new data (standard Customizing, language dependent text elements, ...)
- XPRAS: restructuring of business data
- Follow-up activities: manual follow-up activities

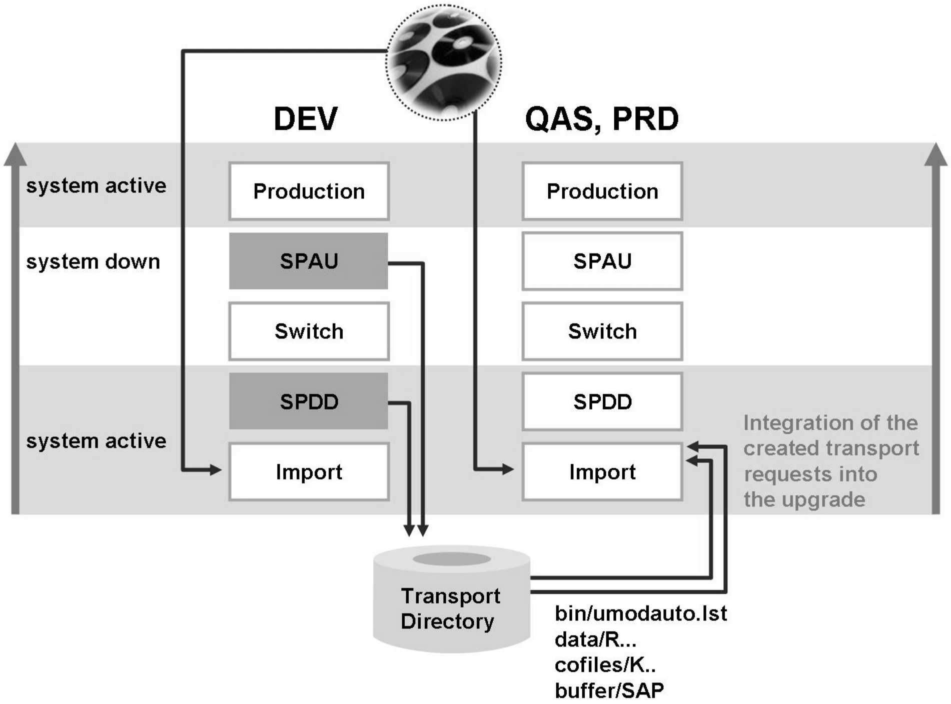


Figure 179: Modification Adjustment

The start of the upgrade process involves the transfer of new data to the system from the shipped DVDs. SAP Repository objects are imported into the system, and the customer objects are compared. All Repository objects that have been modified by customers must be compared to the new SAP standard during this upgrade process.

In order to avoid loss of data and table fields that customers may have created, conflicting structures must be merged before the mass activation of Dictionary objects in the upgrade process.

If objects need to be adjusted, use the transactions SPDD and SPAU. All modifications made by customers are then merged with the new SAP object versions to retain data; otherwise, the new SAP version will be activated and data may be lost.

The system is available during the activation phase. The activation takes place on the shadow instance.

When the upgrade is completed, the SAP system is successfully running at the new release level. Customer-developed objects and modifications have been preserved.

Hint: The two transport requests from SPDD and SPAU (one from SPDD and one from SPAU) can be included (**not imported**) into the upgrade of the subsequent SAP system in the same SAP system landscape. By doing this, the modification adjustment is performed automatically in this second SAP system. Here you only have to adjust those objects manually that were modified in the second SAP system, but not in the first SAP system.

The ABAP Dictionary objects (tables, data elements, domains, and so on) are adjusted during downtime before the activation of the ABAP Dictionary. The adjusted objects are collected in a transport request. You cannot release this transport request; instead it must be flagged for export in transaction SPDD. Towards the end of the upgrade, SAPup exports this transport request into the transport directory, and then registers it for transport in the file <transport directory>\bin\umodauto.lst.

Non-Dictionary Repository objects (reports, screens, and so on) are adjusted towards the end of the upgrade. At this stage, the import of SAP objects has already been completed. However, the old, modified version is still available in the versions database. As with ABAP Dictionary objects, all adjustments are released to a transport request that is noted and then exported and registered by SAPup.

Hint: Do not attempt to import adjustment transport requests into the system manually during SPDD. This can lead to a loss of data!

Do not activate any objects during SPDD. Activation is carried out automatically after the adjustment.

After you have completed the upgrade phases, you have a maximum of 14 days to execute transaction SPAU without a key check (SAP Software Change Registration, SSCR) for the objects that you changed.

For further information, see the upgrade manual and online application help for SPDD/SPAU.

Lesson Summary

You should now be able to:

- Have an idea about planning an SAP system upgrade
- Roughly explain how an SAP system upgrade functions

Unit Summary

You should now be able to:

- List the advantages of automatically implementing SAP Notes
- List the goals, functions, and the features of SAP Note Assistant
- Describe Note Browser and list the preconditions for handling SAP Notes
- Describe the import of Support Packages
- Reduce the downtime when applying Support Packages
- Perform a modification adjustment during the import of Support Packages
- Have an idea about planning an SAP system upgrade
- Roughly explain how an SAP system upgrade functions

Test Your Knowledge

1. The SAP Note Assistant can be used for changing for example programs and DDIC objects.

   Determine whether this statement is true or false.

   □ True

   □ False

2. List the functions of the SAP Note Assistant.

   ____

   ____

   ____

   ____

3. Identify the functions of the Note Browser.

   ____

   ____

   ____

   ____

4. The ____ lets you apply the Support Packages in the correct order by creating a patch queue.

   Fill in the blanks to complete the sentence.

5. To apply Support Packages, use the Support Package Manager (transaction SPAM).

   Determine whether this statement is true or false.

   □ True

   □ False

6. Support Packages of one component can be imported in any order.

   Determine whether this statement is true or false.

   □ True

   □ False

7. An important aspect of an SAP system upgrade is preserving customer ______________________ and ______________________.

   Fill in the blanks to complete the sentence.

8. Modification adjustments should be performed by SAP system administrators.

   Determine whether this statement is true or false.

   □ True

   □ False

9. List the conditions for which modification adjustments are necessary.

   ____

   ____

   ____

   ____

Answers

1. The SAP Note Assistant can be used for changing for example programs and DDIC objects.

   Answer: False

   The SAP Note Assistant cannot be used for changing DDIC objects.

2. List the functions of the SAP Note Assistant.

   Answer: The functions of an SAP Note Assistant include implementing SAP Notes, checking dependencies, reporting, workflow functions and logging.

3. Identify the functions of the Note Browser.

   Answer: The Note Browser gives an overview of all SAP Notes available in the system. You can also implement or display SAP Notes from the Note Browser.

4. The Support Package Manager lets you apply the Support Packages in the correct order by creating a patch queue.

   Answer: Support Package Manager

5. To apply Support Packages, use the Support Package Manager (transaction SPAM).

   Answer: True

   To apply Support Packages, use the Support Package Manager (transaction SPAM) in client 000 of the SAP system.

6. Support Packages of one component can be imported in any order.

   Answer: False

   Support Packages of one component must be imported in ascending order, without any gaps.

7. An important aspect of an SAP system upgrade is preserving customer developments and modifications.

   Answer: developments, modifications

8. Modification adjustments should be performed by SAP system administrators.

   Answer: False

   Modification adjustments should be performed by developers and not by SAP system administrators.

9. List the conditions for which modification adjustments are necessary.

   Answer: The conditions for which modification adjustments are necessary include:

   1. an SAP object is modified by the customer
   2. and the same SAP object has been changed by SAP in the new SAP release

Unit 7

Advanced User Administration Topics

Unit Overview

This unit deals with two advanced user administration topics.

In a landscape with multiple SAP systems, Central User Administration (CUA) allows you to maintain user master records centrally in one location and to distribute them to any number of child systems. After an explanation of the concept, you will set up a CUA yourself and get to know various settings.

The CUA is restricted to ABAP-based SAP systems. If user master records are to be synchronously maintained in a heterogeneous landscape, directory services can be useful. They act as a central repository for information (such as user data) that can be accessed from different applications using a standardized protocol. In this unit, you will connect an SAP NetWeaver AS to a directory service.

You can implement each of these concept independently of the other; however, they can also be used together.

Unit Objectives

After completing this unit, you will be able to:

- Outline the concept of Central User Administration
- Evaluate the usefulness of Central User Administration for your company
- Define logical systems
- Correctly set up the required technical users for RFC connections
- Create the RFC connections required for CUA
- Configured the ALE communication required for CUA
- Set up Central User Administration
- Maintain and distribute user data using the CUA
- Perform simple troubleshooting
- Remove a CUA

- Explain the concept of user administration as applied to the integration of directory services
- Identify prerequisites for connecting an AS ABAP to a directory server
- Configure the AS ABAP for connection to a directory server
- Exchange data between the AS ABAP and a directory server using the LDAP protocol

Unit Contents

Lesson: Introduction to CUA

Lesson Overview

This lesson introduces the principles of Central User Administration (CUA), and provides decision-making aids.

Lesson Objectives

After completing this lesson, you will be able to:

- Outline the concept of Central User Administration
- Evaluate the usefulness of Central User Administration for your company

Business Example

In your system group of SAP systems, the same users exist in a number of clients. You want to learn how you can reduce the costs for maintaining users in the different SAP systems. You can reduce the costs for user administration by centrally administering the users in distributed systems.

Fundamentals

User information is stored in a client-specific manner in SAP systems. That is, every client in a system has its own, independent group of permitted users. If you have a complex system landscape in your company, you might have more than one of the three standard system types (DEV, QAS, and PRD).

Hint: For example, you have three SAP production systems: an SAP SCM system for production planning, an SAP ECC system as an ERP production system, and a CRM system. These systems would all be regarded as “PRD” systems.

Classically, users are maintained in the respective client. If, for example, a developer retires from your company, you must ensure that the relevant system user is removed from all affected clients; ideally, you must ensure that the relevant user is locked. If you do not know exactly which areas your colleague worked in as a developer, you need to log onto every single client that he may have used, and manually search for and remove the user. In complex landscapes, this task can easily become confusing. There may, therefore, be potential security gaps in your user administration. This can result in large costs for consistent user maintenance.

Note: Users that are no longer to be used for logging on to the SAP system should be locked and not deleted.

The following figure should clarify the structure of user administration in a “classic” scenario with multiple SAP systems.

The users are administered individually in each client

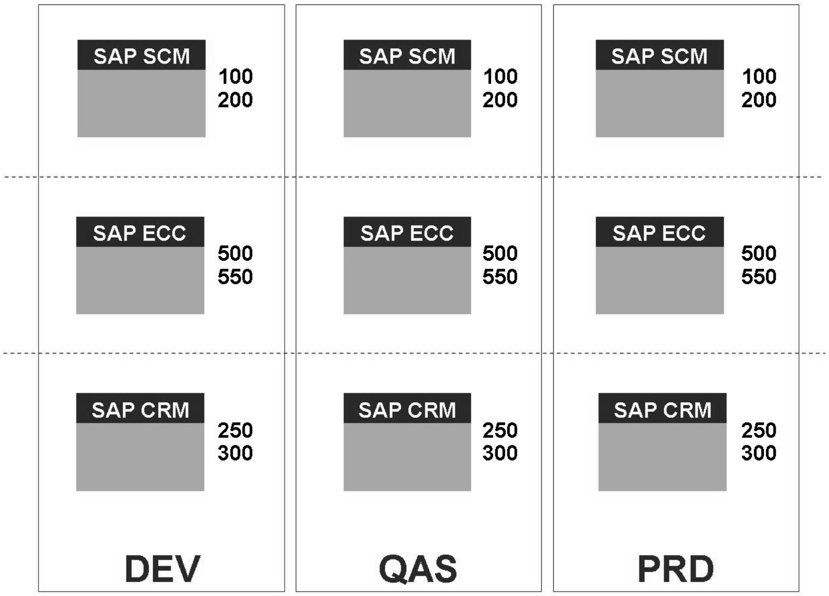


Figure 180: Classic User Administration

As you can see from the figure, the number of "interesting" clients (that is, clients other than 000, 001, and 066) to be administrated is 17, even in this simple example. The clients shown are used for different purposes, for example, client 200 is not required in the production system in the SAP SCM system (in the figure above) (because this is a test client, for example).

Concept of Central User Administration

The aim of Central User Administration is to reduce the cost for user administration and to make user administration more secure by centralizing the work. The concept of Central User Administration (CUA) envisages only a single client for maintaining user data; that is, you determine the following list of points at a central location in your system landscape.

What You Define with Central User Administration

- Which user master records exist in which clients?
- Which roles and profiles are assigned to these master records?
- What are the initial passwords?
- Is the user locked or not?

The above user data is managed in only one place. This means that you can have an overview of the situation for managing your user data. Before you decide to implement CUA, the following prerequisites should exist in your system landscape; otherwise the work reduction due to the implementation of CUA will not be optimal.

Prerequisites for Efficient CUA Implementation

- There are a large number of identical users in multiple clients
- There is a complex system landscape with large number of clients to be managed
- You want to reduce the costs of complex, distributed user administration by implementing Central User Administration

The prerequisites above are not entirely self-explanatory. If, for example, you have a “standard three-system landscape” in which the users each primarily work in one client, and the user groups in the different clients are almost disjunctive, the cost saving due to the implementation of CUA is not particularly large. The situation is similar if you use identical users in multiple clients, but your system landscape only contains a few clients, or the number of administered users is not especially high in total.

Hint: CUA can also simplify user administration in small system landscapes, as the user master records are then only administered in one client of your system landscape, which is **very clear**. This increases the security of the systems that you administer.

As the following figure illustrates, you can perform all work relating to user master records from one central client when you implement CUA.

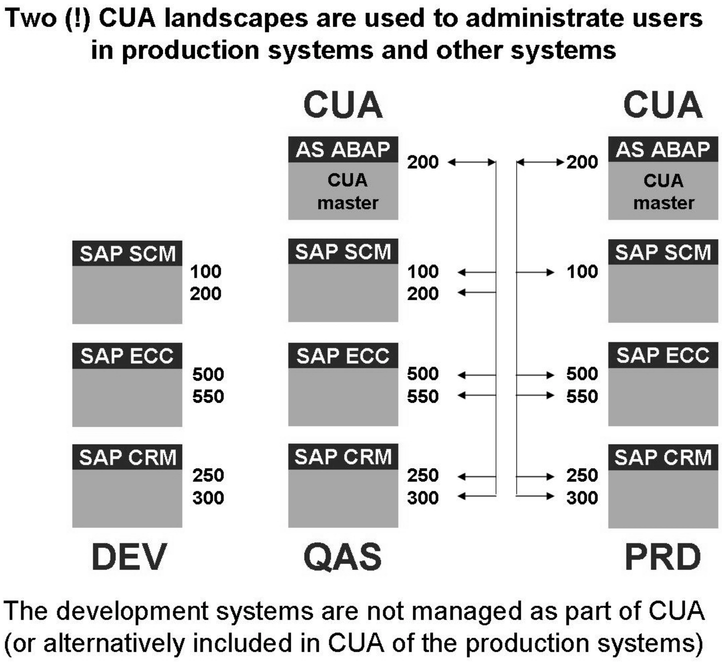

Two (!) CUA landscapes are used to administrate users in production systems and other systems
CUA
CUA
AS ABAP
CUA master
200
200
AS ABAP
CUA master
SAP SCM
100
200
SAP SCM
100
200
100
SAP SCM
SAP ECC
500
550
SAP ECC
500
550
500
550
SAP ECC
SAP CRM
250
300
SAP CRM
250
300
250
300
SAP CRM
DEV
QAS
PRD
The development systems are not managed as part of CUA (or alternatively included in CUA of the production systems)

Figure 181: User Administration Using CUA (Example)

Using CUA, which resides in a dedicated system, you can now distribute the following information from the central system to the child systems by Application Link Enabling (ALE):

Which users exist in which system, in which client, and with which roles and profile assignments. You can also distribute the initial password.

Hint: ALE is a powerful tool, and only a small portion of its many possibilities can be explained in the context of CUA.

The following list contains important notes for implementing CUA.

Notes on Implementing CUA

1. Use a system with a release status of at least SAP Basis 4.6C. Your system should be as up to date as possible, with the latest Support Packages and kernel patches.
2. CUA can be used cross-release, although a linked system must have at least SAP Basis 4.5B (and should also have as current a Support Package status as possible).
3. CUA can also be used for user administration in BW, CRM, and SRM systems, for example. However, you should search for SAP Notes about this topic in the SAP Notes database before implementing CUA, and take these into account.

In an ALE integrated system union, SAP and non-SAP systems may be in contact. The systems linked to each other are called **logical systems**. The logical systems in the context of CUA are, in this case, the clients involved. That is, a single SAP system can house multiple logical systems in terms of ALE (and CUA). Data exchange in an ALE integrated system is performed by exchanging **intermediate documents** (IDocs) by **Remote Function Call** (RFC).

Caution: Ensure also that if CUA is in use and role definitions are transported (by means of transport requests from one system to another system), the user assignments to the roles are **not** transported.

If CUA is in use and role definitions, including user assignments, are transported at the same time, discrepancies might arise in the user assignment to roles.

You can avoid these problems as described in SAP Note 571276: *PFCG: Transporting roles* using transaction SM30 by setting the **`USER_REL_IMPORT`** entry in table PRGN_CUST to **`NO`** in all the child systems in CUA.

Lesson Summary

You should now be able to:

- Outline the concept of Central User Administration
- Evaluate the usefulness of Central User Administration for your company

Related Information

- SAP Note 159885: *Collective SAP Note for the Central User Administration (CUA)*
- Quick Link */security* on the SAP Service Marketplace
- Online documentation for SAP NetWeaver 7.0: *SAP Library → SAP NetWeaver Library → SAP NetWeaver by Key Capability → Security → Identity Management → Identity Management of the Application Server ABAP → Configuration of Identity Management → Central User Administration*

Lesson: Setting Up a CUA

Lesson Overview

In this lesson, you will learn about setting up Central User Administration (CUA). All of the required steps are presented in chronological order.

Lesson Objectives

After completing this lesson, you will be able to:

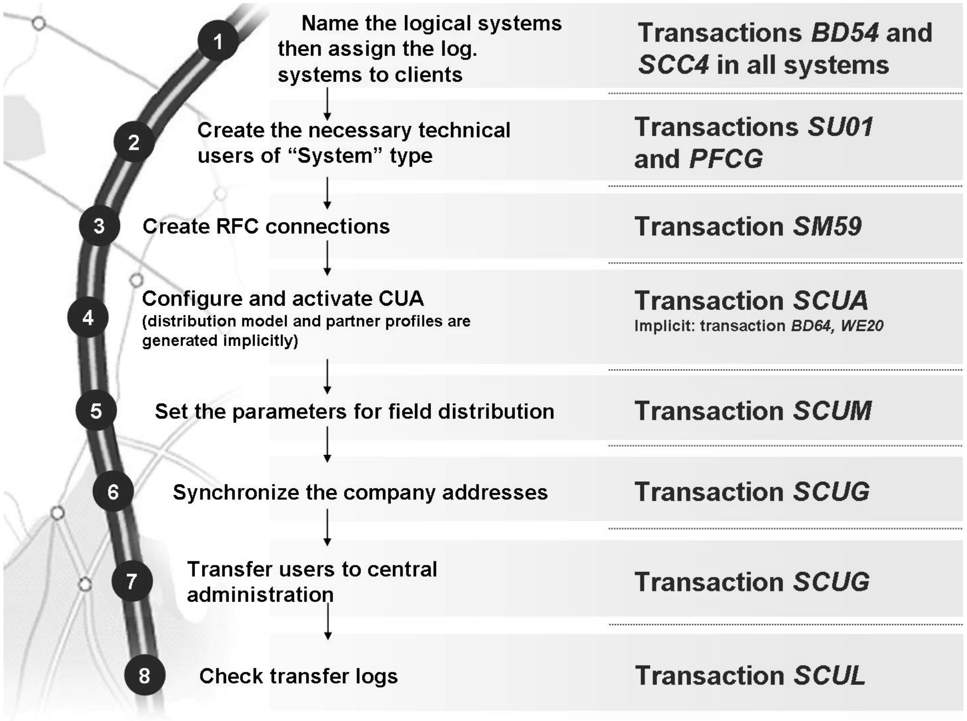
- Define logical systems
- Correctly set up the required technical users for RFC connections
- Create the RFC connections required for CUA
- Configured the ALE communication required for CUA
- Set up Central User Administration

Business Example

The user administration of the SAP systems in your company is to be organized more efficiently and more clearly. CUA is to be implemented to do this.

Central User Administration: Step by Step

The steps listed in the following “roadmap” are required to set up Central User Administration.



1	Name the logical systems then assign the log. systems to clients	Transactions *BD54* and *SCC4* in all systems
2	Create the necessary technical users of “System” type	Transactions *SU01* and *PFCG*
3	Create RFC connections	Transaction *SM59*
4	Configure and activate CUA (distribution model and partner profiles are generated implicitly)	Transaction *SCUA* Implicit: transaction *BD64, WE20*
5	Set the parameters for field distribution	Transaction *SCUM*
6	Synchronize the company addresses	Transaction *SCUG*
7	Transfer users to central administration	Transaction *SCUG*
8	Check transfer logs	Transaction *SCUL*

Figure 182: Roadmap for Setting Up CUA

Note: Where the term “**technical users**” is used here, it refers to technical users in terms of user classification for system measurement. In this case, the class of “technical users” represents users for whom **no** license fees are charged.

The technical users that are used for CUA are of the type **“System”** (created as such in transaction SU01), although the documentation and the note database sometimes refer to users of the type “Communication”. However, this recommendation is out of date, since users of the type “Communication”, unlike users of the type “System”, are subject to the password change rules.

Hint: The following **training setup** is assumed in this lesson.

Setup for the DEV system

DEV: Your central system is client 100 in the DEV system, identified as DEVCLNT100.

QAS: The linked child system is client 200 in the QAS system, identified as QASCLNT200.

Setup for the QAS system

QAS: Your central system is client 100 in the QAS system, identified as QASCLNT100.

DEV: The linked child system is client 200 in the DEV system, identified as DEVCLNT200.

The following figure should also clarify this setup for this course.

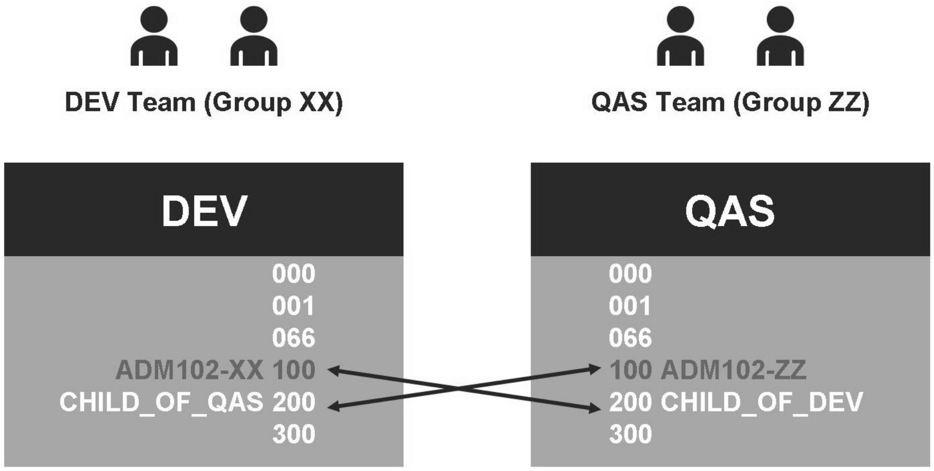


Figure 183: CUA Infrastructure in ADM102

Note: In a real system landscape, you will only set up one central system.

Caution: In the following, only the terms **central system** and **child system** are used. This refers specifically to clients (logical systems). Therefore, if your company runs the central CUA and child CUA on the same SAP system (in different clients), the terms 'central system' and 'child system' refer to different clients in the same SAP system.

Specifying and Assigning the Logical Systems

You can specify logical systems as follows:

1. Log onto your central system with an administration user.
2. In the Implementation Guide (transaction SALE), choose *Basic Settings → Logical Systems → Define Logical System*, or start transaction BD54. Alternatively, you can maintain the table view *V_TBDLS* using transaction SM30.
3. Choose *Edit → New Entries*.
4. In the *Log.System* column, create a new logical name in **uppercase letters**. So that you can later identify systems by name more easily, use the following naming convention for the logical system names:

   <SID>CLNT###, where you replace **<SID>** with the relevant system ID and ### with the relevant client number, for example, `DEVCLNT100`. Enter a name of the logical system, such as `Central System (CUA)` or `Child System (CUA)`.
5. Save your entries, which will then be included in a transport request.
6. Create the logical system name of the central system and of the relevant child system in all child systems.

You have the following options for step 6:

- Send the transport request generated in step 5 (which contains the names of all logical systems) to all other systems in the CUA.
- Perform steps 1 to 4 in all child systems of the CUA. Create the logical system name of the central system and of the relevant child system in each case.

Hint: The data for the logical systems is created cross-client. If there are multiple logical systems of the CUA in one SAP system, you only need to perform steps 1 to 5 (or to import the transport request) once.

You also use transaction SALE to assign the logical systems.

Choose *Basic Settings → Logical Systems → Assign Logical System to Client*, or start transaction SCC4 directly. Alternatively, edit table *T000* with transaction SM30.

After you have switched to change mode, you can assign an appropriate entry to the *Logical system* field in the detail view of the client attributes. When doing so, use the input help to avoid input errors. Select the appropriate entry, such as `DEVCLNT100`, if you want to assign this logical system to client 100 in the DEV system. Make this assignment of logical systems for all clients involved in the CUA (in all of the involved systems).

Creating the Technical Users and the RFC Connections

RFC connections are used to exchange data by ALE in the context of CUA. These RFC connections are used to distribute user data from the central system to the child systems, to send changes to this data in the child systems, and to send status reports back to the central system. In the context of CUA, the central system is treated like another child system in some respects. That is, in the simplest case of a CUA (two logical systems linked to each other – one central system, one child system – within the same SAP system), two RFC connections are required. If the central system and the child system are in different SAP systems, three RFC connections are required:

RFC Connection from the Master System to Itself (Loopback)

A separate, additional RFC connection is required to connect the central system "to itself". Special users are used for these RFC connections. These technical users are of the type *System* and their authorizations are clearly delimited using predefined roles.

Caution: For more information, see SAP Note 492589:*Minimum authorizations for communication users*.

RFC Connections from the Central System to the Child Systems

RFC connections are required from the central system to all child systems and to the central system itself (loopback).

These RFC connections use users in the child systems for which this naming convention should be sufficient: **`CUA_<SID>_###`**, where <SID> is replaced with the system ID of the child system and ### with the number of the client being addressed in the child system.

This user also requires authorizations that are contained in the following roles: **`SAP_BC_USR_CUA_SETUP_CLIENT`** and **`SAP_BC_USR_CUA_CLIENT`**.

However, you should not assign the delivered SAP roles to the technical users, but rather copies of these roles for which you have created authorization profiles with role maintenance (transaction PFCG). For simplicity, name the copies of the SAP roles by adding "Z_" to the start of the name, that is, **Z_SAP_BC_USR_CUA_SETUP_CLIENT** and **Z_SAP_BC_USR_CUA_CLIENT**.

Hint: Note that the technical user in the child system requires the role **Z_SAP_BC_USR_CUA_SETUP_CLIENT** only while you are setting up the CUA. You can then remove this role from the technical user again.

Hint: To increase the security and performance in your system, you can use the role Z_SAP_BC_USR_CUA_CLIENT in a "shared" way. This role contains the authorizations required for receiving the IDocs of the CUA and to update them. However, the subrole **Z_SAP_BC_USR_CUA_CLIENT_RFC** (create this in the same way as the roles described above) contains only authorization to receive the IDocs, and the role **Z_SAP_BC_USR_CUA_CLIENT_BATCH** has the update authorization for the inbound IDocs. Assign the first role to the technical user, as described above. Assign role **Z_SAP_BC_USR_CUA_CLIENT_BATCH**, however, which only allows the changes to the user master records, to a user (of the type System) that you use for background processing. This user is used (in the child systems) to schedule a periodic background job that implements the CUA changes in the child systems (periodically).

RFC Connections from the Child Systems to the Central System

RFC connections are also required from the child systems to the central system. Every child system must be able to create an RFC connection to the central system.

Hint: Since RFC connections can be used cross-client (all clients of a system can use a shared RFC connection), you only need to create one RFC connection to the central system if there are multiple child systems in one SAP system.

The connection from the child system to the central system uses a technical user of the type System that must be known in the central system. This user should have the ID **CUA_<SID>**. In this case, <SID> stands for the system ID of the child system that communicates with the central system using this user.

This user requires the authorizations for the roles **Z_SAP_BC_USR_CUA_SETUP_CENTRAL**, **Z_SAP_BC_USR_CUA_CENTRAL**, and **Z_SAP_BC_USR_CUA_CENTRAL_BDIST**.

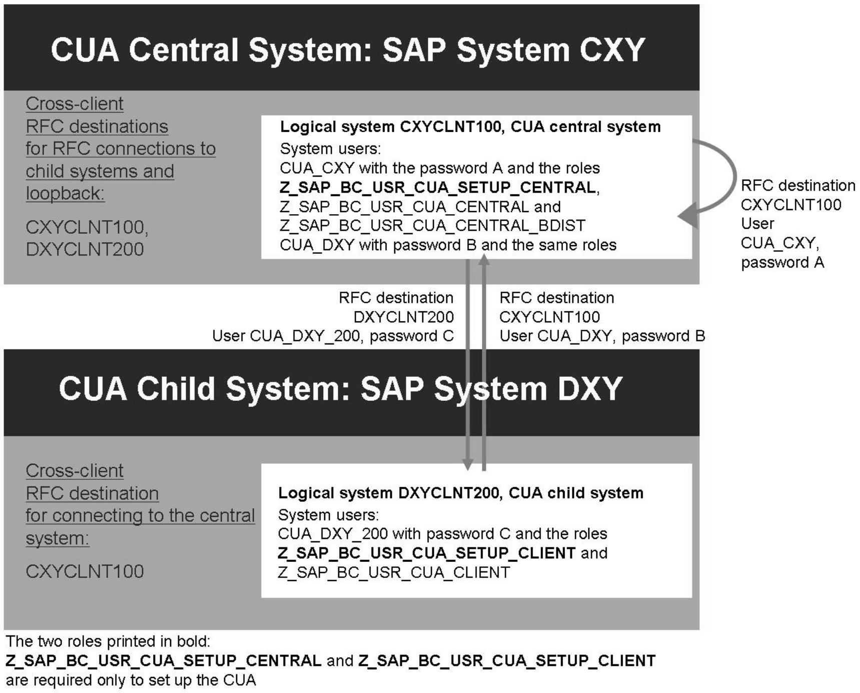
Create the roles by copying the template roles (the template names have the same names as the specified roles, but without **Z_** at the start of the name) and generating the profiles for them using transaction PFCG.

The first role listed, **Z_SAP_BC_USR_CUA_SETUP_CENTRAL**, is only required while you are setting up CUA.

Caution: The role **Z_SAP_BC_USR_CUA_CENTRAL_BDIST** is only required by the user in the RFC connection if the attributes in transaction SCUM are set to *Redistribution*. See the section "Setting the Parameters for Field Distribution".

Caution: In addition to the RFC connections from the child systems to the central system, you also require an RFC connection from the central system to the central system. This connection also uses the user with the ID **CUA_<SID>**, where <SID> is the ID of the central system. The required authorizations are contained in the roles **Z_SAP_BC_USR_CUA_SETUP_CENTRAL**, **Z_SAP_BC_USR_CUA_CENTRAL**, and **Z_SAP_BC_USR_CUA_CENTRAL_BDIST**, mentioned above.

The following figure illustrates the required RFC connections and technical users as well as the roles they require in a very simple CUA scenario.



CUA Central System: SAP System CXY

Cross-client RFC destinations for RFC connections to child systems and loopback:

CXYCLNT100, DXYCLNT200

Logical system CXYCLNT100, CUA central system

System users:
CUA_CXY with the password A and the roles **Z_SAP_BC_USR_CUA_SETUP_CENTRAL**, Z_SAP_BC_USR_CUA_CENTRAL and Z_SAP_BC_USR_CUA_CENTRAL_BDIST
CUA_DXY with password B and the same roles

RFC destination CXYCLNT100 User CUA_CXY, password A

RFC destination DXYCLNT200 User CUA_DXY_200, password C

RFC destination CXYCLNT100 User CUA_DXY, password B

CUA Child System: SAP System DXY

Cross-client RFC destination for connecting to the central system:

CXYCLNT100

Logical system DXYCLNT200, CUA child system

System users:
CUA_DXY_200 with password C and the roles **Z_SAP_BC_USR_CUA_SETUP_CLIENT** and Z_SAP_BC_USR_CUA_CLIENT

The two roles printed in bold: **Z_SAP_BC_USR_CUA_SETUP_CENTRAL** and **Z_SAP_BC_USR_CUA_SETUP_CLIENT** are required only to set up the CUA

Figure 184: Required RFC Connections and Technical Users for CUA

Hint: SAP Note 492589 – *CUA: Minimum authorizations for communication users* provides background information about the required authorizations and the roles delivered by SAP in the CUA environment.

After you have created the users listed above with the appropriate role assignments in all of the clients involved , you only need to define the RFC connections in transaction SM59. Follow the instructions below to do this:

1. In transaction SALE in the central system, choose *Communication → Create RFC Connections* (transaction SM59).
2. Choose *Create*.
3. Enter the following data:
   - In the *RFC Destination* field, enter the name of the connection, such as **`QASCLNT200`** or **`DEVCLNT200`**.

     This entry must be in uppercase letters.
   - In the *Connection Type* field, enter **`3`**
   - In the *Description* group box, enter a short text, such as **`Connection to child system.`**

   Caution: The RFC destination name must be identical to the logical system that you want to address with the RFC connection. The RFC connection must be specified in uppercase letters.
4. After choosing *Return*, make the following settings on the *Technical Settings* tab page:
   - If you want to restrict the function of the CUA to a particular server group, you can specify the logon group after selecting the *Load Balancing: Yes* radio button and choosing **`Return`**. Only the delivery logon group SPACE exists in the training systems.
   - If you do **not** want to use load distribution, specify the name of the host on which the central instance of your target system is running for *Target Host*, and specify the instance number of the central instance.
5. On the *Logon&Security* tab page, specify the client of the required logical system and the technical user

   `CUA_<SID>_###`, for example, **`CUA_QAS_200`** or **`CUA_DEV_200`**, and its password

   (in ADM102, this password was assigned by the instructor before the start of the course).
6. Save this connection.
7. Test this connection using the functions: *Utilities → Test → Connection Test (Ctrl + F3)* and *Utilities → Test → Authorization Test (Ctrl + F4)*.

To setup the required RFC connection from the child system to the central system, log onto the child system and perform steps 1 to 7 again. Use the user **CUA_<SID>** here, where <SID> is the SID of your child system. You should also set up the RFC connection from the central system to itself, described above, in the same way.

Note: You can also use trusted RFC connections for the CUA communication. This increases the security of your CUA communication further. For information about configuring the CUA using trusted RFC connections, see the system documentation.

Configuring and Activating the CUA

Once you have successfully activated the CUA, you will no longer be able to create users in the linked child systems using transaction SU01. Before you activate CUA, call transaction SU01 in a child system:

It is still possible to create new users.

The activation of CUA has been significantly simplified, and a number of necessary configuration steps are performed automatically by the system. Follow the procedure below.

1. Log onto the central system.
2. In the Implementation Guide (transaction SALE), choose *Modelling and Implementing Business Processes → Configure Predefined ALE Business Processes → Cross-Application Business Processes → Central User Administration → Select Model View for Central Administration* (transaction SCUA).
3. Enter the name of your distribution model, such as **CUA**.
4. Choose *Create*.
5. Enter the names of all child systems to be connected (such as **QASCLNT200**).
6. Save your entries.
7. The result screen *Display logs* appears. If you expand the nodes for the individual systems, you normally see the following messages for each system:

   ALE distribution model was saved,

   Central User Administration activated, and

   Text comparison was started.

   If problem messages are displayed here, follow the procedure in SAP Note 333441: *CUA: Tips for problem analysis*.

When the CUA is activated, the system carries out the following configuration steps automatically:

- The corresponding ALE model is created or adjusted to match the new CUA model if changes have been made
- Partner profiles are created
- Text comparison with the child systems is carried out for roles (and profiles)

The ALE distribution model defines which applications communicate with each other in the distributed systems and which data types are distributed. You require a separate ALE distribution model for a Central User Administration. In the central system, you define the structure of your Central User Administration in the model view, which you then distribute to the child systems. In the ALE distribution model to be defined for the CUA, two types of data are distributed: user master data (including assigned roles and profiles) and company addresses. In the distribution model, you require two methods to distribute user data and company addresses. To implement these methods, you use BAPIs of the USER and UserCompany business objects with the Clone method. You can view the partner model in transaction BD64.

Partner profiles define the conditions for electronic data exchange via the IDoc interface. If a partner profile does not exist, you cannot communicate with a partner via the IDoc interface. You can display these partner profiles in transaction WE20.

The check tables and the texts for roles, profiles, and license data in each child system are saved temporarily in the central system so they can be displayed more quickly. If they have been changed, you have to run the text comparison. If you run the text comparison in the central system, you can select the child systems from which the data is to be read. If you run the text comparison in a child system, the current data is sent to the CUA central system.

Hint: The functions of transaction SCUA have been improved through Support Packages. For more details, see SAP Note 952349.

Caution: Even if users can no longer be created in the child systems, the CUA is only “fully operational” once the following additional steps have been carried out.

Setting the Parameters for Field Distribution

For each field of transaction SU01, you can use transaction SCUM to determine the system in which the administration of the field content can be performed.

Setting the Field Distribution Parameters

Global

You can only maintain data in the central system. The data is then automatically distributed to the child systems. The corresponding fields cannot be changed there; they can only be displayed.

Local

You can only maintain data in the child system. Changes are not distributed to other systems.

Proposal

You have maintained a default value in the central system that is automatically distributed to the child systems when you create a user. After distribution, the data is only maintained locally, and is no longer distributed, if you change it in the central or child system.

Redistribution (Ret. Val.)

You can maintain the data both centrally and locally. Every time that the data is changed, the change is distributed back to the central system, and is then forwarded from there to the other child systems.

Everywhere

This option is available only on the *Lock* tab page and for initial passwords (*Logon Data* tab page). You can maintain initial passwords and lock data both centrally and locally. However, only the changes made to the data in the central system are distributed to the other systems. Local changes in child systems are not distributed.

Caution: If you subsequently change the distribution from “Local” or “Proposal” to “Global” or “Redistribution”, inconsistent data can be created. The only exception is that you can reset the indicators on the *Lock* tab page at any time without danger.

Ensure that you take into account SAP Note 611972: *SCUM: Change to field distribution parameters*.

Hint: The settings of the distribution parameters are automatically forwarded to the child systems.

Hint: Recommendations for how you should configure the parameter distribution for a number of fields are presented in the following table. The recommendation for a “Global” setting applies to all fields that are not listed (with the exception of locks).

Some Recommendations for Transaction SCUM

Field	Setting
Print parameters (under *Defaults*)	Proposal
Parameters	Proposal
User group (under *Logon Data*)	Proposal
Fields for data that the users maintain themselves	Redistribution or local

Caution: For information about lock management for your users (such as due to too many failed logon attempts), see the documentation.

Synchronizing the Company Addresses

As company address data has already been maintained in all systems of the future CUA, you must first ensure that at least the central system contains all valid company addresses. Then distribute this complete company address set to all child systems, so that a consistent status exists for the company addresses in the entire CUA.

You can use transaction SUCOMP to administrate company address data. You can use transaction SCUG in the central system to perform the synchronization activities between the central system and the child systems by selecting your child system on the initial screen of transaction SCUG and then choosing *Synchronize Company Addresses in the Central System*. For more information about this, see the documentation.

Note: Alternatively, you can compare company addresses using transaction SCUC. You cannot transfer user data in this transaction.

Synchronizing User Groups

To be able to transfer users from a child system to the central system, or to distribute them from the central system to a child system, the user group to which the user is assigned must exist in all systems in which the user exists.

Caution: For information about structuring this work as simply as possible, see SAP Note 395841: *CUA: Assign target system-specific parameters and user groups*.

After importing this SAP Note, the missing user groups from the CUA are created where required.

Hint: As of SAP Web AS 6.20, the SAP Note is included in the system; that is, if you use exclusively SAP Web AS 6.20 (or above) based systems, you do not need to take any action.

Transferring Users to Central Administration

After you have synchronized the company addresses, you can transfer the users from the newly connected child systems to central administration. You can do so using transaction SCUG in the central system. To do this, on the initial screen of transaction SCUG, select your child system and choose *Copy Users to the Central System*.

Different cases are handled differently for the user transfer.

New Users

These users are not yet contained in Central User Administration. By choosing *Transfer users*, you can transfer the selected users to the central system. All user parameters (address, logon data, and so on) and profiles and roles are transferred. The user is maintained centrally in the future.

Identical Users

In this case, there are users with identical user IDs (the user ID or ID is the string that you enter in the *User* field on the SAP Logon screen) and matching first and last names. You can transfer the roles and profile data of this user to the central system. The user is then distributed as it exists in the central system. Local data is overwritten. In this case, it is assumed that, for example, the ID “MooreJ” that exists in multiple logical systems, with the last name “Moore” and the first name “Jane”, always belongs to one and the same person. In large companies, this assumption is not always correct. For this reason, we recommend that you use unique character strings, such as the personnel number as the user ID.

Different Users

These user IDs exist both in the central and in the child system, but the user has a different first and/or last name. If, in an individual case, these IDs actually refer to the same user, you can transfer the roles and profile data for the user to the central system. The user is then distributed as it exists in the central system. If the IDs do not refer to the same user, use the CUA to create a new user ID in the child system for one of the users and delete the old ID in the child system.

Alternatively, you can assign a new ID to the user in the child system (if different employees have an identical ID in different clients). For this purpose, you can call transaction SU01 in the child system and choose *Users → Rename*. If the user transfer is restarted, the user is listed as a *New User* rather than as a *Different User*.

Already Central User

These users already exist in Central User Administration and are only administered centrally.

Hint: A function is available for copying users, which you can access using three new pushbuttons in transaction SCUG if the *Already central user* tab page is selected. With these pushbuttons, you can transfer role assignments, profile assignments, and the license classification of users from child systems, particularly in cases in which (despite CUA being in use) the corresponding administration was carried out until now in the child systems (in line with the settings in transaction SCUM).

Central User Administration is now completely set up for use.

See SAP Note 704412: *CUA support for license data maintenance*.

Hint: Until the user transfer has been completed, the child systems affected still contain (“unexpected”) processing options for users that have not (yet) been transferred, such as the delete function for users in transaction SU01.

Checking the Distribution Status

You use the log display (transaction SCUL) primarily to check the status of IDoc distribution when changing company addresses or users. If you change a company address in the central system of a Central User Administration (CUA), this is replicated and a CCLONE IDoc is sent to each child system of the CUA. If you change a user in the CUA central system, the user data is also distributed to the child systems assigned to this user. Up to three USERCLONExx IDocs are sent for each user: user attributes (USER), profile assignment (PROFILE), and role assignment (ACTGRP). You can then see in the results list of the log display whether the user or company address was replicated successfully to the child systems.

Hint: If it was not possible to completely process IDocs with user changes in the child system (*Unconfirmed* status in transaction SCUL), this can be because there are not enough dialog work processes available (in the child system or in the contacted instance). In this case, you can start postprocessing of the IDocs in the child system using transaction BD87, or opt for inbound processing of the IDocs in background processing. The latter is described in more detail in SAP Note 399271: *CUA: Tips for optimizing ALE distribribution performance*.

Exercise 15: Setting Up Central User Administration

Exercise Objectives

After completing this exercise, you will be able to:

- Set up a simple Central User Administration

Business Example

You want to familiarize yourself with the functions for setting up Central User Administration (CUA) before planning a productive implementation.

Task 1: Create Logical Systems and Assign to Clients

Implement Central User Administration between your central system and the child system that you are using.

If you are working in the DEV system (DEV group), your central system is client 100 in the DEV system, and the child system is client 200 in the QAS system.

If you are working in the QAS system (QAS group), your central system is client 100 in the QAS system, and the child system is client 200 in the DEV system.

The user that you are to use is *ADM102-##* (## corresponds to your group number) with the initial password allocated by the instructor for client 100. The user *CHILD_OF_DEV* also exists for the DEV groups in client 200 of the QAS system. At the same time, the user *CHILD_OF_QAS* exists for the QAS groups in client 200 of the DEV system.

1. Specify the required logical systems. Use the naming convention presented in the lesson when doing so. Then assign the IDs of the logical systems to the corresponding clients in the two training systems DEV and QAS.

Continued on next page

Task 2: Create Roles, Users, and RFC Destinations

Copy the roles required for Central User Administration to the customer namespace, create technical users to which you assign the required roles, and create the RFC destinations required for the CUA. To do so, use an eCATT test configuration *ZT_PREPCUA_<SID>* that exists in your CUA master system (<SID> is the system ID of your CUA master system).

1. In the system assigned to you for this course (CUA master system), go to transaction SECATT. Run the test configuration ZT_PREPCUA_<SID>, whereby <SID> is the system ID of your CUA master system. On the *Variants* tab page, select the variant *PREPCUA* only (if this has not already been selected). Then run the test configuration.

   Caution: Do not carry out any other actions on your SAP GUI while the eCATT is running. If your eCATT is terminated with an error message (red light in the log display), you must carry out all remaining steps manually.

2. Check that the eCATT test configuration was run successfully.

Task 3: Activate Central User Administration

Activate the Central User Administration and check whether a distribution model has been created.

1. Activate the CUA in the central system using transaction SCUA.

2. Optional: In the central system, check whether a distribution model has been created for the Central User Administration that can be used to exchange the BAPIs USER.Clone and UserCompany.Clone.

   Hint: In the same way, this distribution model is also created in the child system. Partner profiles are also generated automatically, but do not have to be checked as part of this exercise.

   Hint: After the CUA has been activated successfully it can take a while before the configuration settings made are processed from the child system.

Continued on next page

Task 4: Maintain Field Distribution Parameters

Use transaction SCUM in the central system to maintain the field distribution parameters.

1. Call transaction SCUM in the central system. Maintain the field distribution parameters. Do not change the default value *Global* in the parameters aside from those in the following fields. Save your entries.

Field	Setting
Print parameters (*Defaults* tab page)	Proposal
Parameters (*Parameters* tab page)	Proposal
User group (*Logon Data* tab page)	Proposal
Telephone (*Address* tab page)	Redistribution

Task 5: Compare Company Addresses

Perform a comparison of the company addresses between the central system and the child system.

1. Perform a comparison of the company addresses between the central system and the child system. Use transaction SCUG to do this.

Task 6: Transfer Child System Users to the CUA and Check Transfer

Transfer the users from the child system to the central administration. Then check the distribution log. If some user data has not been fully distributed, start postprocessing in the child system manually.

1. Use transaction SCUG (in the master system) to copy the users in the child system to the central administration. If necessary, decide how individual user master records are to be handled.
2. Use transaction SCUL in the master system to check the distribution log for central user maintenance. Have all user master records been fully distributed?
3. If your CUA group still contains unconfirmed user master records, start postprocessing in the child system manually using transaction *BD87*.

Result

Congratulations: You have now set up a simple CUA.

Solution 15: Setting Up Central User Administration

Task 1: Create Logical Systems and Assign to Clients

Implement Central User Administration between your central system and the child system that you are using.

If you are working in the DEV system (DEV group), your central system is client 100 in the DEV system, and the child system is client 200 in the QAS system.

If you are working in the QAS system (QAS group), your central system is client 100 in the QAS system, and the child system is client 200 in the DEV system.

The user that you are to use is *ADM102-##* (## corresponds to your group number) with the initial password allocated by the instructor for client 100. The user *CHILD_OF_DEV* also exists for the DEV groups in client 200 of the QAS system. At the same time, the user *CHILD_OF_QAS* exists for the QAS groups in client 200 of the DEV system.

1. Specify the required logical systems. Use the naming convention presented in the lesson when doing so. Then assign the IDs of the logical systems to the corresponding clients in the two training systems DEV and QAS.

   a) Log onto your central system with an administration user.

   In the Implementation Guide (transaction SALE), choose *Basic Settings → Logical Systems → Define Logical System*, or start transaction BD54. On the initial screen for this transaction, choose *Edit → New Entries*.

   b) In the *Log.System* column, create a new logical name in **uppercase letters**.

   Caution: You must use the naming convention **<SID>CLNT###** for the logical system names, whereby you replace **<SID>** with the relevant system identifier and ### with the number of the client involved, such as **`DEVCLNT100`** or **`QASCLNT100`** for the master system and **`QASCLNT200`** or **`DEVCLNT200`** for the child system. The following tasks require these logical system names.

   Enter a name for the logical system, for example, **`Central System (CUA)`** or **`Child System (CUA)`**. Save your entries, which will then be included in a transport request.

   c) In the child system, create the logical system name for the central system and the relevant child system as described in a) and b).

Continued on next page

d) You also use transaction SALE to assign the logical systems to clients. Here, choose *Basic Settings → Logical Systems → Assign Logical System to Client*, or start transaction SCC4 directly.

   Once you have switched to change mode (if required), you can make a suitable entry in the *Logical System* field in the detail view of the client attributes (accessed by double-clicking the relevant row in the table). You should use the input help to do so. Select the appropriate entry, for example **DEVCLNT100**, if you want to assign this logical system to client 100 in the DEV system. Then save your changes and confirm by choosing *Continue* in the dialog box that appears.

e) Make this assignment of logical systems for all clients involved in the CUA (that is, also in the child system).

Task 2: Create Roles, Users, and RFC Destinations

Copy the roles required for Central User Administration to the customer namespace, create technical users to which you assign the required roles, and create the RFC destinations required for the CUA. To do so, use an eCATT test configuration *ZT_PREPCUA_<SID>* that exists in your CUA master system (<SID> is the system ID of your CUA master system).

1. In the system assigned to you for this course (CUA master system), go to transaction SECATT. Run the test configuration ZT_PREPCUA_<SID>, whereby <SID> is the system ID of your CUA master system. On the *Variants* tab page, select the variant *PREPCUA* only (if this has not already been selected). Then run the test configuration.

Continued on next page

Caution: Do not carry out any other actions on your SAP GUI while the eCATT is running. If your eCATT is terminated with an error message (red light in the log display), you must carry out all remaining steps manually.

a) **DEV group**: Call transaction SECATT in the DEV system. On the initial screen for this transaction, select the test configuration *ZT_PREPCUA_DEV* and choose *Execute*.

   QAS group: Call transaction SECATT in the QAS system. On the initial screen for this transaction, select the test configuration *ZT_PREPCUA_QAS* and choose *Execute*.

b) Go to the *Variants* tab page. Deselect all checkboxes and then select the checkbox for the *PREPCUA* variant if this has not been carried out automatically in the SAP system. Do not change the default values for fields on all other tab pages. Finally, choose *Execute*.

2. Check that the eCATT test configuration was run successfully.

a) After the eCATT test configuration has started, the log for the eCATT run is displayed. If a green “light” is displayed here in the eCATT log, the eCATT test configuration was completed successfully. If a red “light” is displayed, at least one step has not been completed successfully. In this case, you have to carry out the steps that failed manually (expand the log display to identify the steps). The following steps are performed by the test configuration variant executed:

   1. The role *SAP_BC_USR_CUA_SETUP_CENTRAL* is copied to the role *Z_SAP_BC_USR_CUA_SETUP_CENTRAL* (in transaction PFCG in the master client) and a corresponding authorization profile is generated.
   2. The role *SAP_BC_USR CUA_CENTRAL* is copied to the role *Z_SAP_BC_USR CUA_CENTRAL* (in transaction PFCG in the master client) and a corresponding authorization profile is generated.
   3. The role *SAP_BC_USR CUA_CENTRAL_BDIST* is copied to the role *Z_SAP_BC_USR CUA_CENTRAL_BDIST* (in transaction PFCG in the master client) and a corresponding authorization profile is generated.
   4. The (system) user *CUA_<SID>* is created in transaction SU01 in the master client and the three roles created are assigned to this user (whereby <SID> is the system ID of your CUA master system).

Continued on next page

5. The (system) user *CUA_<Partner-SID>* is created in transaction SU01 in the master client and the three roles created are assigned to this user (<Partner-SID> refers to the system ID of the CUA child system).
6. The role *SAP_BC_USR_CUA_SETUP_CLIENT* is copied to the role *Z_SAP_BC_USR_CUA_SETUP_CLIENT* (in transaction PFCG in the child client) and a corresponding authorization profile is generated.
7. The role *SAP_BC_USR_CUA_CLIENT* is copied to the role *Z_SAP_BC_USR_CUA_CLIENT* (in transaction PFCG in the child client) and a corresponding authorization profile is generated.
8. The (system) user *CUA_<Partner-SID>_200* is created in transaction SU01 in the child system and the two roles created in the child system are assigned to this user.
9. The RFC destination *<Partner-SID>CLNT200* that points to the client of the CUA child and in which the user *CUA_<Partner-SID>_200* is defined, is created in transaction SM59 in the master system.
10. The RFC destination *<SID>CLNT100* that points to the client of the CUA master and in which the user *CUA_<SID>* is defined, is created as a loopback connection in transaction SM59 in the master system.
11. The RFC destination *<SID>CLNT100* that points to the client of the CUA master and in which the user *CUA_<Partner-SID>* is defined, is created in transaction SM59 in the child system (whereby <SID> is the system ID of your CUA master system).

Continued on next page

Task 3: Activate Central User Administration

Activate the Central User Administration and check whether a distribution model has been created.

1. Activate the CUA in the central system using transaction SCUA.

   a) In the Implementation Guide (transaction SALE), choose *Modelling and Implementing Business Processes → Configure Predefined ALE Business Processes → Cross-Application Business Processes → Central User Administration → Select Model View for Central Administration* (transaction SCUA).

   b) Enter the name of your distribution model such as **CUA** and choose *Create*.

   c) Use the input help to select the name of your child system to be connected (for example, **QASCLNT200**) and save your entries.

   d) The result screen *Display Logs* appears. If you expand the nodes for each system, you see the following messages for processes completed successfully:

   ALE distribution model was saved,

   Central User Administration activated, and

   Text comparison was started.

2. Optional: In the central system, check whether a distribution model has been created for the Central User Administration that can be used to exchange the BAPIs USER.Clone and UserCompany.Clone.

   Hint: In the same way, this distribution model is also created in the child system. Partner profiles are also generated automatically, but do not have to be checked as part of this exercise.

   Hint: After the CUA has been activated successfully it can take a while before the configuration settings made are processed from the child system.

   a) Call transaction BD64 (distribution model) in the central system. Check whether the path *Model Views → Central User Administration* exists in the tree structure with your logical systems for the CUA central system and CUA child system as additional subpoints and with the two BAPIs specified in the task description.

Continued on next page

Task 4: Maintain Field Distribution Parameters

Use transaction SCUM in the central system to maintain the field distribution parameters.

1. Call transaction SCUM in the central system. Maintain the field distribution parameters. Do not change the default value *Global* in the parameters aside from those in the following fields. Save your entries.

Field	**Setting**
Print parameters (*Defaults* tab page)	Proposal
Parameters (*Parameters* tab page)	Proposal
User group (*Logon Data* tab page)	Proposal
Telephone (*Address* tab page)	Redistribution

a) For a solution, see the task.

Continued on next page

Task 5: Compare Company Addresses

Perform a comparison of the company addresses between the central system and the child system.

1. Perform a comparison of the company addresses between the central system and the child system. Use transaction SCUG to do this.

   a) Call the comparison in the CUA master system by selecting your child system on the initial screen of transaction SCUG and choosing *Synchronize Company Addresses in the Central System*. By expanding the resulting tree structure, you determine that the logical systems involved contain different company addresses.

   b) Replicate the company address that exists only in the central system to the child system by selecting the corresponding entry and choosing *Distribute to Child System*. Then choose *Back*.

   c) Replicate the company address that exists only in the child system to the central system by selecting the corresponding entry and choosing *Copy from Child System*. Then choose *Back*.

   d) Overwrite the old address in the child system with the current address in the central system by selecting the corresponding entry in the *Different Company Addresses* area and choosing *Distribute to Child System*. Then choose *Back*.

   e) Use transaction SUCOMP in the child system to check that your actions have been successful. You should be able to find all companies transferred using the input help. The current version of the company address originally maintained differently should have been copied to the CUA child from the central system (this can be seen in the change mode for this company).

Continued on next page

Task 6: Transfer Child System Users to the CUA and Check Transfer

Transfer the users from the child system to the central administration. Then check the distribution log. If some user data has not been fully distributed, start postprocessing in the child system manually.

1. Use transaction SCUG (in the master system) to copy the users in the child system to the central administration. If necessary, decide how individual user master records are to be handled.

   a) In the CUA master system, call up the function for copying users by selecting your child system on the initial screen of transaction SCUG and choosing *Copy Users to the Central System*.

   b) On the *New Users* tab page, you can select all users and copy them to Central User Administration by choosing *Transfer Users*.

   c) You follow the same procedure on the *Identical Users* tab page.

   d) On the *Different Users* tab page, you can see that the master and child clients contain users with the same ID but different first/last name combinations. Select all users here and choose *Transfer Users*. This transfers the user master data for these users from the central system to the child system according to the field distribution parameters.

2. Use transaction SCUL in the master system to check the distribution log for central user maintenance. Have all user master records been fully distributed?

   a) Call transaction SCUL in your CUA central system. On the initial screen, choose the *Users* tab page. In the *Selection by Distribution Status* area, select all fields (that is, you also want to display user master records that have been distributed successfully). Choose *Execute*.

   b) The next screen displays the status of the individual user master records. Green entries indicate successfully distributed master records. If white entries are displayed, this indicates that the distribution of the corresponding user master record has not yet been confirmed.

Continued on next page

3. If your CUA group still contains unconfirmed user master records, start postprocessing in the child system manually using transaction *BD87*.

   a) If your CUA group still contains unconfirmed user master records, call transaction BD87 in your CUA child system. On the initial screen for this transaction, choose *Execute*.

   b) On the next screen, expand the tree structure of the child system. You find that there are still IDocs of type USERCLONE with the status *IDoc ready to be transferred to application*. Select the entry USERCLONE in this area of the tree structure and choose *Process selected node*. After choosing *Back*, all application documents should be posted.

   c) Use transaction SCUM in the master system to refresh the user transfer log from the previous subtask. All user master records should now be confirmed.

Result

Congratulations: You have now set up a simple CUA.

Lesson Summary

You should now be able to:

- Define logical systems
- Correctly set up the required technical users for RFC connections
- Create the RFC connections required for CUA
- Configured the ALE communication required for CUA
- Set up Central User Administration

Related Information

- Online documentation for SAP NetWeaver 7.0: *SAP Library → SAP NetWeaver Library → SAP NetWeaver by Key Capability → Security → Identity Management → Identity Management of the Application Server ABAP → Configuration of Identity Management → Central User Administration*
- SAP Note 159885: *Collective SAP Note for the Central User Administration (CUA)*
- SAP Note 952349: *SCUA: Functional improvements*
- SAP Tutorial on SAP Service Marketplace at *service.sap.com/security* . Choose *Security in Detail → Secure User Access → Identity Management → SAP Tutor: Central User Administration*

Lesson: User Administration with the CUA

Lesson Overview

In this lesson, you learn about using Central User Administration (CUA) based on simple examples. You will also learn tips and tricks to ensure reliable and secure operation, as well as the steps required to remove Central User Administration.

Lesson Objectives

After completing this lesson, you will be able to:

- Maintain and distribute user data using the CUA
- Perform simple troubleshooting
- Remove a CUA

Business Example

You have set up Central User Administration (CUA) in your company and now want to administer the user data for the connected SAP systems. You also want to know about the options for troubleshooting, and to learn the steps required to remove a CUA.

Distributing User Data Using the CUA

With an activated CUA, you continue to use transaction SU01 for user administration. However, there are a number of differences in its operation. The following list should provide you with an overview of the differences in user maintenance when CUA is implemented.

New *Systems* tab page:

This tab page specifies the logical systems in which it is possible for a particular user to log on. However, you only rarely enter values directly on this tab page. Note the explanations regarding the *Roles* and *Profiles* tab pages described below. If a system assignment is removed from a user on the *Systems* tab page, the user is deleted from the relevant child system.

***Roles* tab page:**

On this tab page, you can assign roles to a user in the systems of the CUA. The central system should know all role names of all connected systems. It is always possible to manually assign a role that exists only in a child system, but not in the central system. However, you can only use the F4 help with current data if you have first performed the *Text Comparison from Child Systems* . When setting up the CUA, an initial text comparison is automatically performed. Therefore, F4 help is available immediately. If you want to use F4 help with current data, ensure that you perform a new text comparison between the central system and the child system after creating new roles in the child system.

During the text comparison, specify as the *Target System* the system from which the central information is to query the new information.

Note: If roles were created or changed in a child system, call transaction PFCG in the child system and choose *Environment* → *Text Comparison for CUA Central System*.

***Profiles* tab page:**

The same statements apply to this tab page as to the *Roles* tab page except for the *Role Assignment* dialog box. This dialog box is not displayed when assigning profiles.

Change to the *New Password/Change Password* and *Lock User* functions

The “New Password/Change Password” function now contains an option for specifying the child system for which the user is to receive a new password. When locking a user, you can decide – depending on the field distribution parameters in transaction SCUM – whether this user is to be locked locally or globally on all CUA systems.

The following changes occur for transaction SU01 in the child system.

- All tab pages: restricted changeability
- It is no longer possible to create new users
- It is no longer possible to delete existing users (after complete user transfer)
- User in the *Last Changed on* field

All tab pages:

On all tab pages of transaction SU01 in the child system, you are now able to administer only the fields that you released for local administration during the configuration of the parameters. Local administration is allowed if you chose an administration setting other than *Global*.

It is no longer possible to create new users or delete existing users (as soon as the users have been completely transferred to CUA).

***Last Changed on* field:**

The name of the user that was stored with the RFC connection from the central system to the child system, such as CUA_QAS_200, is specified in this field in the child system for all users that have already been centrally changed.

Hint: There are alternative options for the text comparison described above. You can also start the comparison from the child system by choosing, in transaction PFCG, *Environment → Text Comparison for CUA Central System*, or by calling report *SUSR_ZBV_GET_RECEIVER_PROFILES* in transaction SA38. You do not need to make any entries for this report; the data from the child system is sent to the central system anyway.

To force the text comparison, you should also run report **SUSR_ZBV_GET_RECEIVER_PROFILES** in transaction SA38. In this case, however, you should explicitly specify the child systems with which you want to perform a text comparison.

You can automate the comparison by scheduling the report to be regularly executed in the background on the central system. In this case, select a variant that contains all of your child systems.

Fields whose field distribution parameters are set to *Redistribution* send their content back to the central system, from which this field value is then forwarded to all child systems. Existing entries are updated in this way.

Special Authorization Objects for CUA

For Central User Administration, the following authorization objects exist that allow you to define user administrator authorizations for specific systems. This means that all user administrators of a CUA system group can exist in the central CUA, but each user can assign authorizations only for one part of the connected systems.

Special Authorization Objects for CUA

Authorization Object	Meaning
S_USER_SYS	System assignment in CUA
S_USER_SAS	Role and profile assignment to systems in CUA

S_USER_SYS is the authorization object for system assignment in Central User Administration (CUA). You can distribute users from a central system to different child systems of a system group. The object *S_USER_SYS* checks which child systems the user administrator can assign to the users.

The system checks the authorization object *S_USER_SAS* in transactions SU01, SU10, PFCG, and PFUD when roles, profiles, and systems are assigned to users. It represents the further development of the authorization objects *S_USER_GRP*, *S_USER_AGR*, *S_USER_PRO*, and *S_USER_SYS*, which were previously checked during assignment.

Hint: You have to activate the authorization object S_USER_SAS using the Customizing switch, otherwise the previous authorization objects will be checked.

Troubleshooting for the CUA

If the distribution of user data is not functioning, follow the checklist below:

- Transaction SCUL in the central system
- Transaction SM59 in the central system
- Transaction BD64 in the central system
- Transaction SCUA in the central system
- Transaction SUIM in the central system
- Transaction BD87 in the child system
- SAP Note 333441: *CUA: Tips for problem analysis*

1. View the distribution log in transaction SU01: *Environment → Distribution log*. Alternatively, call transaction SCUL . If *Errors* are displayed here, you can choose *Resend User* as a first step. Trace whether the problem persists or whether the data is sent successfully this time.
2. If the display of the distribution log informs you of *tRFC with Errors*, you should check the RFC connections used in transaction SM59.
3. In transaction SCUM, choose *Environment → ALE Distribution Model* (transaction BD64) to check the consistency of the ALE model you are using. To do this, select the model for your CUA and check whether the required BAPIs *USER.Clone* and *UserCompany.Clone* are assigned to your model.
4. Check whether your CUA distribution model is displayed in transaction SCUA.
5. You can use transaction SUIM to determine cross-system information (area *User Information System → User → Cross-System Information (Central User Administration))*. Here, for example, you can display the list of all users in

child systems (together with the IDoc number that was used to update users in the child system). You can also display the roles assigned to users or the users assigned to certain roles for specific systems.

6. If it was not possible to completely process IDocs with user data changes in the child system (*Unconfirmed* status in transaction SCUL), this can be because there are not enough dialog work processes available (in the child system or in the contacted instance). In this case, you can start postprocessing of the IDocs in the child system using transaction BD87, or opt for inbound processing of the IDocs in a background job. The second variant is described in more detail in SAP Note 399271: *CUA: Tips for optimizing ALE distribution performance*.

   You should also take a look at SAP Note 555229: *IDocs hang in status 64 for tRFC with immediate processing*.

   Hint: Every sent IDocs requires at least one work process in the receiving system. If a large number of IDocs is sent at the same time, you should set inbound processing to “Processing in the background” in the partner profiles with transaction WE20 and process these IDocs in the background with report RBDAPP01. (Alternatively, IDocs can also be processed with RBDSER04.)

7. For more advanced analysis, use the tips in SAP Note 333441: *CUA: Tips for problem analysis*.

Caution: Ensure that you read the important SAP Notes, which are listed in the composite SAP Note 159885: *Collective SAP Note for the Central User Administration (CUA)*.

Performance Considerations

For more information about optimizing the performance of the data distribution of your CUA landscape, see the documentation in the area *BC-SEC-USR* → *Central User Administration* → *Activated Background Processing*. To do this, you should have at least basic experience using ALE.

Hint: You can use transaction SCUL to distribute individual user master records to the child systems again.

If you want to redistribute a large number of users simultaneously, transaction SCUL is a good option, as described in SAP Note 632486: *Improvements to transaction SCUL*.

Alternatively, you can use report *RSCCUSND* to do this, as described in SAP Note 503247.

Removing the CUA

It is important to differentiate between two (three) possible scenarios for removing the CUA.

1. A child system is to be removed from the CUA (permanently or temporarily).
2. The CUA is to be completely removed.

These scenarios are presented here.

Steps to permanently remove a child system from the CUA:

1. Choose *Delete* in transaction SCUA or call report **RSDELCUA** directly in transaction SA38.
2. In the *Delete* area, choose the option *Child System*, and then select the child system to be removed using the F4 help.
3. Choose *Test* and *Execute*.
4. On the following screen, the system lists the data to be deleted and the tables affected. By double-clicking a table name, you jump to the display of the table content (as it is displayed using transaction code SE16).
5. If you are sure that you want to delete the displayed data, choose *Back* (F3) and deselect the *Test* option. Choose *Execute*.
6. In transaction WE20 , under *Partner Profiles* for *Partner Type LS*, delete the message types CCLONE and USERCLONE for the child system.
7. Start transaction BD64, and switch to change mode.
8. Delete the methods of the child system and save your entries.
9. To distribute the distribution model of your CUA, choose *Edit → Model View → Distribute* and select the child system that you have just deleted in the dialog window.
10. Use transaction SCUA to check whether the child system was deleted from the CUA.
11. If there are no additional child systems of the CUA in the SAP system that contains the child system to be deleted from the CUA, change the communication user (such as CUA_QAS) in transaction SU01 by removing the role **`Z_SAP_BC_USR_CUA_CLIENT`** on the *Roles* tab page. If there are no longer any roles or profiles assigned to the user, you can delete it.

Hint: As the RFC destinations that you created may also be used for ALE connections other than the CUA, you should not delete the RFC destinations, but rather remove the roles with CUA authorizations from the communication user used.

Steps to temporarily remove a child system from the CUA:

If you want to remove the child system from the CUA only temporarily, see SAP Note 801877: *RSDELCUA: Adjustment to child systems*. You can delete the child system from the CUA group in the child system using report RSDELCUA. If you want to include the system in the CUA again at a later point, use transaction SCUA in the central system to add it to the system group and run transaction SCUG again in the central system for this child system. This transfers all local changes to the central CUA (in accordance with the field distribution parameters in transaction SCUM).

Steps to delete the entire CUA:

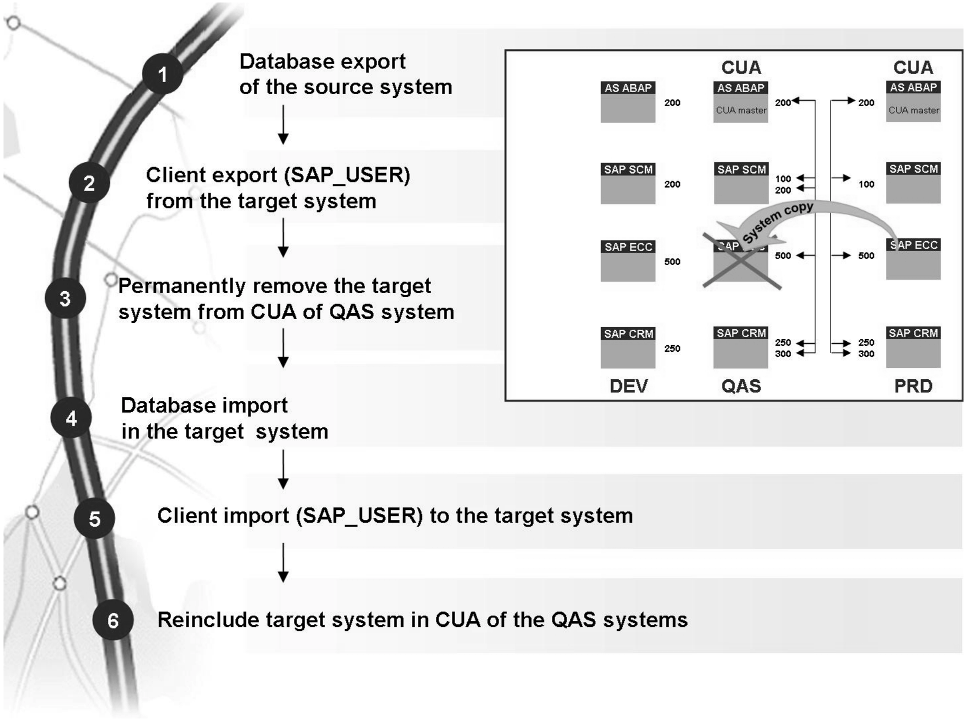
1. Start report **RSDELCUA** on the central system with the options *Complete CUA* and *Test*.
2. If you are satisfied with the list of data to be deleted, run the report again, but this time without the *Test* option.
3. In the central system, delete the data about the central system and the child system(s) in transaction WE20; that is, delete the message types CCLONE and USERCLONE under *Partner Profile for the Partner Type LS*. If only the standard message type SYNCH then remains, you can delete the partner profiles completely.
4. In transaction BD64, delete the distribution model of the CUA.
5. Repeat steps 3 and 4 in the child systems.
6. Remove the CUA roles (such as **`Z_SAP_BC_USR_CUA_CENTRAL`** and **`Z_SAP_BC_USR_CUA_CLIENT`**) from all communication users (such as CUA_QAS_200 or CUA_DEV).

Caution: Delete the RFC connections of the CUA only if you are certain that these connections are not used in other ALE scenarios.

Hint: The users that you have copied from the child systems to central administration are *locked* in the former central system after you have removed the CUA. These users have no authorizations or roles (unless they were assigned roles and profiles in the central system) and can be deleted manually.

Appendix: Central User Administration and System Copies

The figure below illustrates the procedure for copying a production system to a quality assurance system if both systems are included in a Central User Administration.



1 Database export of the source system

2 Client export (SAP_USER) from the target system

3 Permanently remove the target system from CUA of QAS system

4 Database import in the target system

5 Client import (SAP_USER) to the target system

6 Reinclude target system in CUA of the QAS systems

Figure 185: CUA and System Copy

First, you export the database of the source system (in this case, the production system). However, before you can import it to the target system (in this case, the quality assurance (QA) system), you have to carry out the following steps in the QA system: Since the user data in the production and QA systems usually differs, you should first export the client user data from the target system. The copy profile *SAP_USER* already exists for this purpose, which contains user master records, including assignments to roles/profiles and authorizations and profiles. You also have to permanently remove the CUA system from the CUA of the target systems (in this figure, the CUA of the QA systems). Once the database has been imported, you can reimport the previously saved user data to the system via client import and reinclude the target system in Central User Administration as a child system.

Exercise 16: Distributing User Data with the CUA

Exercise Objectives

After completing this exercise, you will be able to:

- Outline the use of the CUA with simple examples

Business Example

You want to use a CUA.

Task 1: Creating a User for the Child System

Create a new user for the child system using the CUA.

1. Create the user ADM102-<Partner-SID> for the child system, whereby <Partner-SID> refers to the system ID of your CUA child system.
   Assign the role SAP_BC_ENDUSER to this user. Assign the value **`ADM102-<Partner-SID>`** to the *XUS* parameter ID for this user. Save your entries.
2. In the child system, display user ADM102-<Partner-SID> that you have just created. Which fields are ready for input? For this user, change the *XUS* parameter to the value **`CHILD_OF_<SID>`** (<SID> is the system ID (central CUA) of the SAP system assigned to you). Assign a telephone number to this user.
3. In the central system, use transaction SU01 to check the *Telephone* and *Parameters* fields for user ADM102-<Partner-SID>.
4. In the central system, use transaction SUIM to check which profiles are assigned to the user you have just created in all systems connected to the CUA.

Task 2: Optional: Removing Central User Administration

Remove the CUA that you set up.

1. Follow the instructions in the training material to remove the CUA.
2. Check that transaction SU01 is fully functional in the child system.

Solution 16: Distributing User Data with the CUA

Task 1: Creating a User for the Child System

Create a new user for the child system using the CUA.

1. Create the user ADM102-<Partner-SID> for the child system, whereby <Partner-SID> refers to the system ID of your CUA child system. Assign the role SAP_BC_ENDUSER to this user. Assign the value **`ADM102-<Partner-SID>`** to the *XUS* parameter ID for this user. Save your entries.

   a) Use transaction SU01 in the central system. Create the user ADM102-<Partner-SID> with address data and an initial password (freely definable). Switch to the *Roles* tab page. Assign the role *SAP_BC_ENDUSER* of the child system to the user. To do so, use the input help to select the child system in the *System* column and then select the role on the *Roles* tab page using the input help. Before saving, enter the value **`ADM102-<Partner-SID>`** for the *XUS* parameter ID on the *Parameters* tab page (whereby <Partner-SID> refers to the system ID of your CUA child system).

2. In the child system, display user ADM102-<Partner-SID> that you have just created. Which fields are ready for input? For this user, change the *XUS* parameter to the value **`CHILD_OF_<SID>`** (<SID> is the system ID (central CUA) of the SAP system assigned to you). Assign a telephone number to this user.

   a) Use transaction SU01 in the child system. In change mode, view all tab pages and note the fields that are ready for input (these should be the fields that you did not set to *Global* in transaction SCUM for maintaining field distribution parameters). On the *Address* tab page, assign a telephone number to the user (including extension) and revaluate parameter *XUS* as described in the task. Save your changes.

3. In the central system, use transaction SU01 to check the *Telephone* and *Parameters* fields for user ADM102-<Partner-SID>.

   a) Follow the exercise text. If you have carried out transaction SCUM as specified in the exercise for the “Setting Up CUA” lesson, the telephone number should have been copied to the master system (field is set to *Redistribution* in transaction SCUM), whereas the change to the parameter is not transferred (field is set to *Proposal* in transaction SCUM).

Continued on next page

4. In the central system, use transaction SUIM to check which profiles are assigned to the user you have just created in all systems connected to the CUA.

   a) Call transaction SUIM (User Information System) in the central system and choose *User Information System → User → Cross-System Information (Central User Administration)*.

   b) Start report *Users by Roles*. In the User name field, enter the name of the user you created (**`ADM102-<Partner-SID>`**) and choose *Execute*. This provides you with a list of all receiver systems and roles that your new user created has in the systems connected to the CUA.

Task 2: Optional: Removing Central User Administration

Remove the CUA that you set up.

1. Follow the instructions in the training material to remove the CUA.

   a) Start report **RSDELCUA** on the central system with the options *Complete CUA* and *Test*. Alternatively, choose *Delete* on the initial screen of transaction SCUA and select the *Complete CUA* and *Test* options. Choose *Execute*.

   b) If you are satisfied with the list of data to be deleted, run the report again, but this time without the *Test* option.

   c) In the central system, delete the data about the central system and the child system in transaction WE20; that is, delete the message types CCLONE and USERCLONE under *Partner Profile* for the *Partner Type LS*. If only the standard message type SYNCH then remains, you can delete the partner profiles completely.

   d) In transaction BD64, delete the distribution model of the CUA.

   e) Repeat steps c) and d) in the child system.

   f) If you want to, you can easily reactivate the CUA (with just a few clicks of your mouse). Note that to delete the CUA completely, you might also have to delete the RFC connections used, the technical users, the roles used, and the logical systems. This would mean that the process of reactivating the CUA would involve a disproportionate amount of time and effort.

2. Check that transaction SU01 is fully functional in the child system.

   a) Follow the instructions in the exercise description. You should now be able to use the pushbuttons for deleting and creating users again on the initial screen of transaction SU01 in the child system.

Lesson Summary

You should now be able to:

- Maintain and distribute user data using the CUA
- Perform simple troubleshooting
- Remove a CUA

Related Information

- Online documentation for SAP NetWeaver 7.0: *SAP Library → SAP NetWeaver Library → SAP NetWeaver by Key Capability → Security → Identity Management → Identity Management of the Application Server ABAP → Configuration of Identity Management → Central User Administration*
- SAP Note 333441: *CUA: Tips for problem analysis*

Lesson: Introduction to Directory Services

Lesson Overview

This lesson defines what is meant by a directory service. A scenario is discussed as an example.

Lesson Objectives

After completing this lesson, you will be able to:

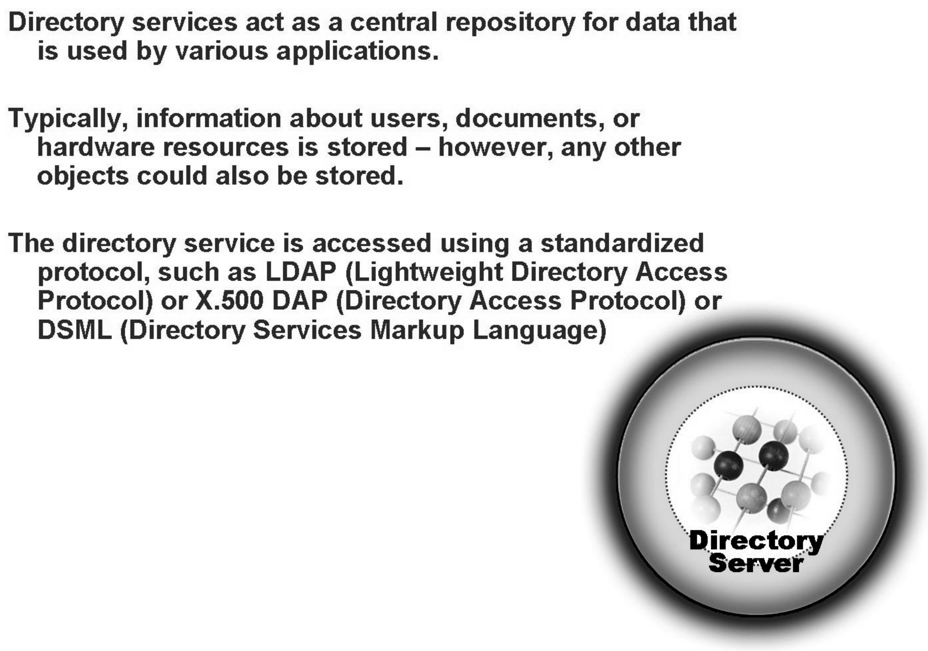
- Explain the concept of user administration as applied to the integration of directory services

Business Example

Your company uses a directory service to store personnel data centrally. As a member of the user administration team for SAP systems, you want to obtain an overview of directory services and find out how you can connect your SAP system to this directory service.

What Are Directory Services?

Not only in the Internet, but also in company networks, there is a danger that information will be lost because it is made available in an unstructured way, becomes obsolete, or no one knows that it exists. A modern information system is needed.



Directory services act as a central repository for data that is used by various applications.

Typically, information about users, documents, or hardware resources is stored – however, any other objects could also be stored.

The directory service is accessed using a standardized protocol, such as LDAP (Lightweight Directory Access Protocol) or X.500 DAP (Directory Access Protocol) or DSML (Directory Services Markup Language)

Figure 186: Properties of Directory Services

A directory server is an information system for particular information. Directory services can help us to answer questions such as “What is the project leader's e-mail address?” or “Where is the nearest color printer?” in the same way that the yellow pages help us to find a lawyer or the nearest florist.

Unlike a relational database management system (RDBMS), where information is stored in two-dimensional tables, directory services use a hierarchical (tree) structure to organize the data. Another difference is the relationship of write to read accesses. Directory services are designed for a high read load, but unlike relational databases, they do not support transaction or lock concepts, or costly queries such as joins.

Directory services are typically used to store information about users, documents, or hardware resources; however, any other objects could also be stored.

An Example Scenario

Directory services can fully demonstrate their strengths in heterogeneous system landscapes (where products from different vendors are used). The example below will illustrate this concept.

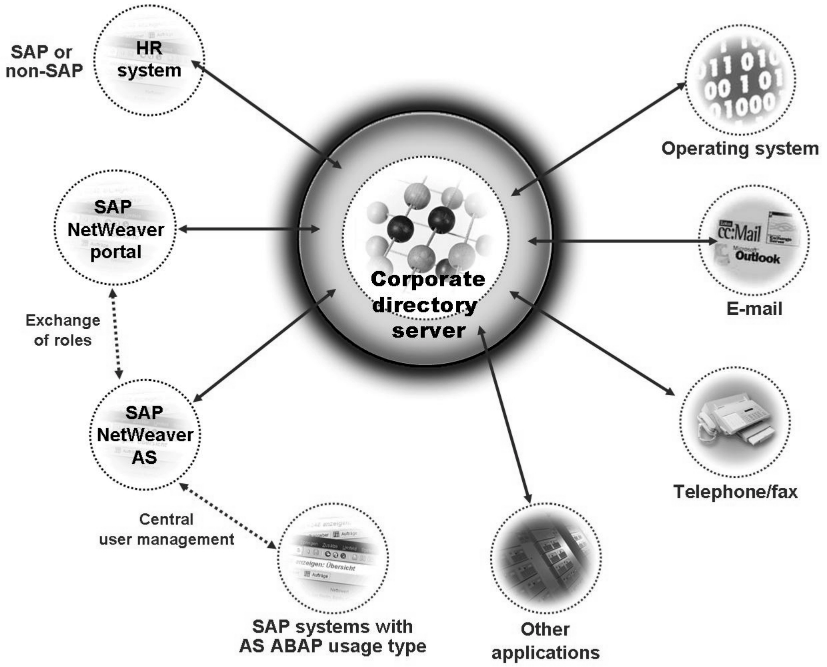


Figure 187: Example Scenario with a Directory Service

A company hires a new employee. This employee is first created in the HR system (which can be an SAP system or come from a different vendor). Some properties (such as name, personnel number, telephone number, e-mail address, department, and group assignment) are transferred to the corporate directory server, while others (such as date of employment, manager, and salary) are deliberately held only in the HR system. If different applications are connected to the directory server:

- The new employee can immediately log onto the corporate domain at operating-system level
- The employee has a functioning e-mail account immediately
- Telephone and fax numbers are assigned to the user centrally
- The employee can log onto the enterprise portal
- The employee has access to certain SAP systems (such as SAP EBP, to place orders).
- Other external systems can be easily integrated

If an employee leaves the company, he or she is deleted from the HR system and therefore also from the directory. Consequently, all of his or her accounts are deleted or locked – not an easy task if this has to be performed manually. This closes gaps in security that would result from accounts that have not been deleted.

What Is LDAP?

LDAP stands for **Lightweight Directory Access Protocol**. It describes how operations are to be formulated for a directory service. The most common operation is querying entries; however, it is also possible to create, change, or delete entries.

From a technical point of view, LDAP follows the client/server model: one or more servers hold the information that the client(s) access. Originally, LDAP implemented only communication with an X.500 server, as described by the International Organization for Standardization (ISO).

LDAPv2 was implemented in 1995 by Yeong, Howes, and Kille at the University of Michigan, to "minimize the complexity of clients so as to facilitate widespread deployment of applications capable of utilizing the Directory." LDAP uses a TCP/IP stack and simplifies access to X.500 servers. The current version of LDAP is LDAPv3. For more information about the technical details, see RFC 2251 (RFC means "Request for Comments" here; for more information about this, see www.ietf.org).

An X.500 server is no longer required; various vendors offer "standalone" servers that can be addressed using LDAP. In addition to commercial products (such as eDirectory from Novell, Sun Java System Directory Server from Sun, DirX from Siemens, and Active Directory from Microsoft), there is also an Open Source development. For more information, see www.openldap.org.

The Data Model

There are a number of models for LDAP. In addition to the access protocol, these define how information is stored and identified, and how access is protected.

A directory contains **entries** that consist of one or more **attributes**. In turn, an attribute consists of a **type** (frequently abbreviated strings such as *cn* for "common name", *c* for "country", or *mail* for e-mail address) and one or more **values**.

Certain attributes are required, while others are optional; this is controlled with LDAP using a special attribute, **object class**. Object classes have an inheritance hierarchy. The information about object classes and attribute types is stored in **schemas**, which are provided by Internet organizations and vendors of directory servers (and which can be customized by the user).

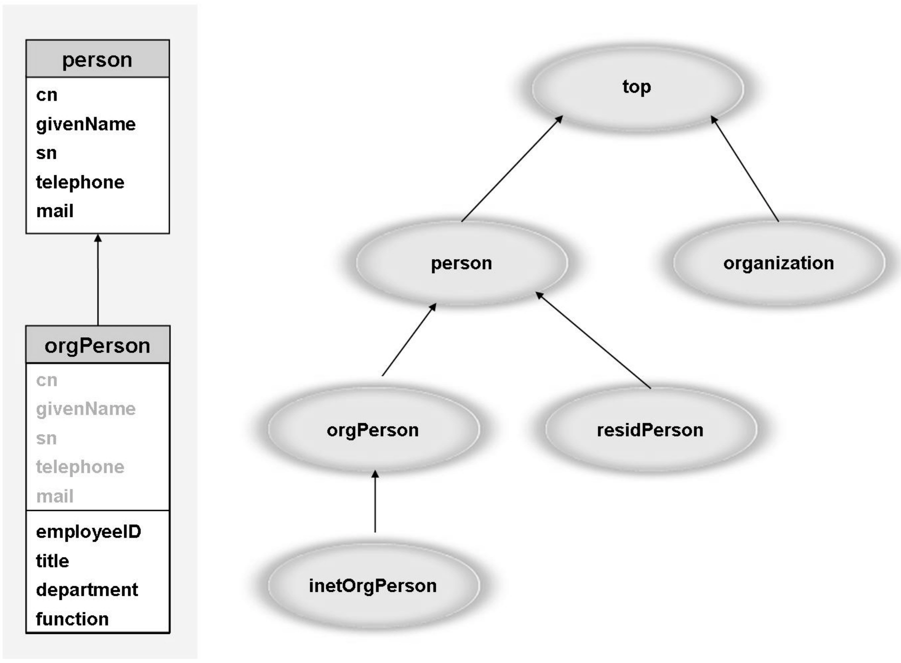
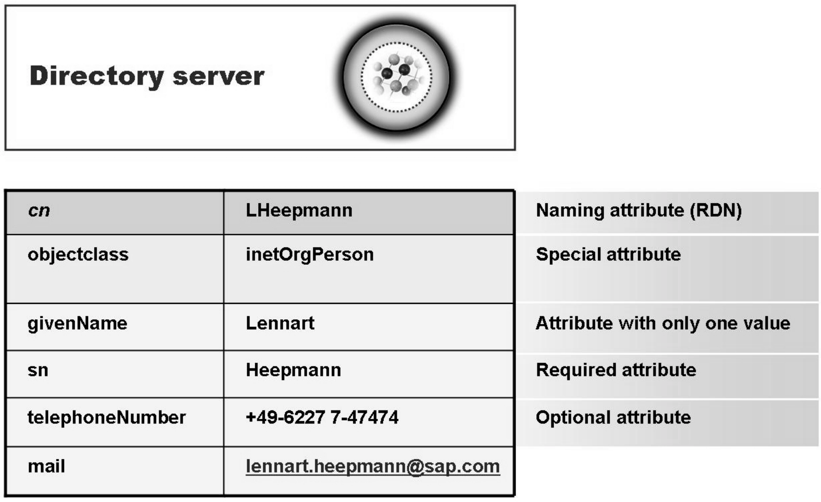


Figure 188: Hierarchy of Object Classes


Directory server

cn	LHeepmann	Naming attribute (RDN)
objectclass	inetOrgPerson	Special attribute
givenName	Lennart	Attribute with only one value
sn	Heepmann	Required attribute
telephoneNumber	+49-6227 7-47474	Optional attribute
mail	lennart.heepmann@sap.com	

Figure 189: Entry in a Directory

The entries are also organized in a hierarchical tree structure. Each entry contains a name that is unique directory-wide, the **Distinguished Name (DN)**. The DN is created by describing the path from the entry to the root of the tree (in the form of a comma-separated list) (for example: *cn=LHeepmann, ou=instructors, o=sap, c=de*).

An individual component of the DN is called the **Relative Distinguished Name (RDN)**, which always identifies the node uniquely relative to the superordinate node in the tree. Two nodes under the same superordinate node could not, therefore, have the same RDN. The entire tree of entries (which can be stored physically across multiple hosts) is called the **Directory Information Tree (DIT)**.

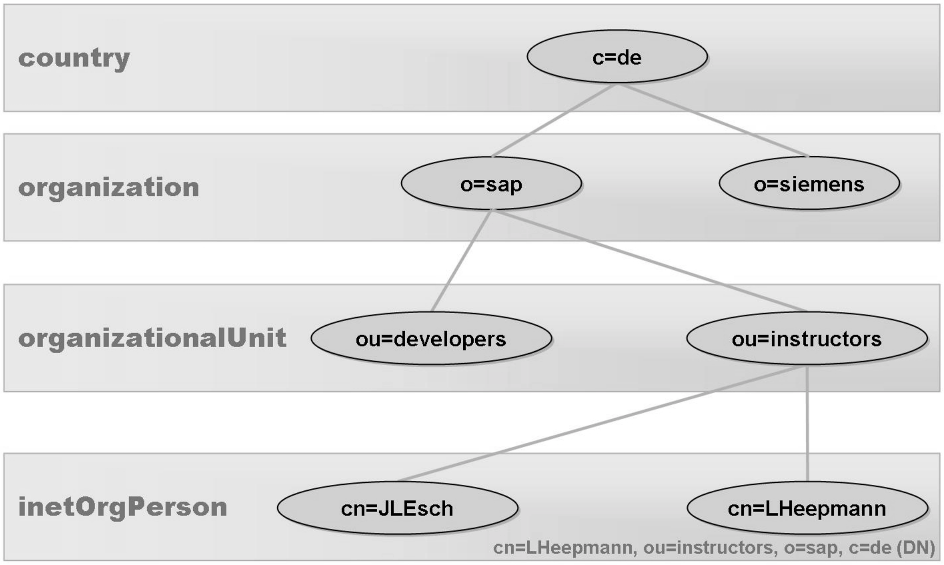


Figure 190: Directory Information Tree (DIT)

SAP and Directory Services

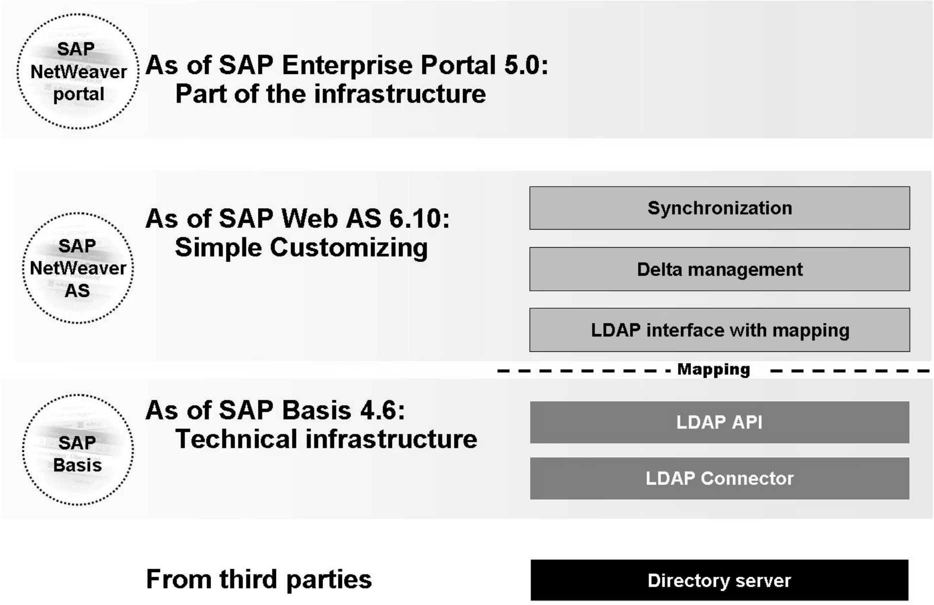


Figure 191: SAP and Directory Services

SAP has provided ways to connect SAP systems to directory services for a number of years. In 1997, the LDAP Gateway, a program that runs independently of the application server, was delivered in Java with SAP Basis 4.0A (supported up to SAP Basis 4.6B). Integration into the application server came with the LDAP Connector, which has been available since SAP Basis 4.6A. With SAP Web AS 6.10, user-friendly functions for user master synchronization (mapping, delta management, and synchronization) were added.

The SAP partner directory (http://www.sap.com/partners) contains a list of partners certified by SAP for the directory server interface (technical name: BC-LDAP-USR).

SAP's enterprise portals also require a store for user data such as master data and group assignment (user persistence store). SAP Enterprise Portal 5.0 (first delivered at the end of 2001) uses one or more directory servers for this purpose, which users are checked against when they log onto the portal. With SAP Enterprise Portal 6.0 (first shipped at the end of 2002), additional storage options were added to the directory server (SAP system or a portal database), thanks to the UME from SAP NetWeaver AS Java. Directory servers are used for most portal implementations.

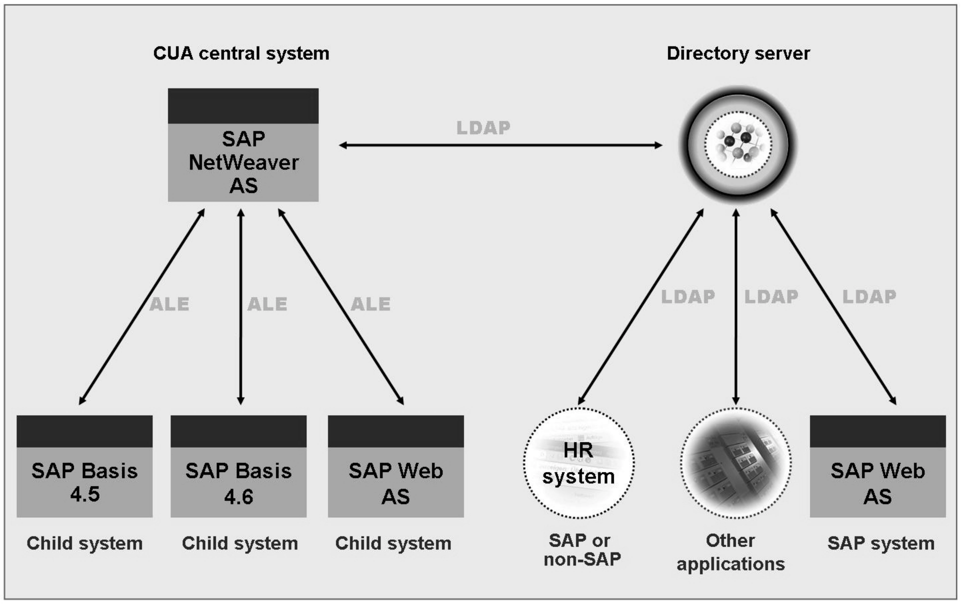


Figure 192: CUA and Directory Services

Both Central User Administration (CUA) and directory services allow you to maintain user data at one location, and to synchronize it in multiple systems. Both concepts can be implemented independently. However, you can also connect them with each other.

In this way, you could use CUA to connect SAP systems with SAP Basis 4.5 or 4.6, while other SAP systems with SAP NetWeaver AS could also be connected directly to the directory service.

In a scenario of this type, LDAP synchronization functions as a “remote control” for the CUA, meaning that all prerequisites and restrictions of the CUA continue to apply.

Exercise 17: Technical Prerequisites for LDAP Clients

Exercise Objectives

After completing this exercise, you will be able to:

- State which program libraries are required at operating-system level to be able to use LDAP clients in general

Business Example

As an administrator, it is your task to connect an SAP system to a directory server. You require information about whether there are particular prerequisites for the operating-system of the SAP system.

Task 1: Program Libraries for LDAP Clients

Depending on your operating system, program libraries for LDAP clients may already be part of the delivered system, or you may have to purchase them separately. A library of this type is also required if you are using an SAP system as an LDAP client.

1. On SAP Service Marketplace at *service.sap.com/security*, determine which program libraries are required for SAP NetWeaver AS ABAP as an LDAP client.

Task 2: Optional: LDAP Connector Version

The downward compatible LDAP Connector is made available through the SAP Service Marketplace.

1. What is the current version of the LDAP Connector for AS ABAP 7.00?

Solution 17: Technical Prerequisites for LDAP Clients

Task 1: Program Libraries for LDAP Clients

Depending on your operating system, program libraries for LDAP clients may already be part of the delivered system, or you may have to purchase them separately. A library of this type is also required if you are using an SAP system as an LDAP client.

1. On SAP Service Marketplace at *service.sap.com/security*, determine which program libraries are required for SAP NetWeaver AS ABAP as an LDAP client.

   a) Log onto the SAP Service Marketplace, http://service.sap.com/security with your user.

      You can find the document *LDAP: Client Libraries* under *Security in Detail → Secure User Access → Identity Management*. This document specifies which program libraries are required for which operating systems.

Task 2: Optional: LDAP Connector Version

The downward compatible LDAP Connector is made available through the SAP Service Marketplace.

1. What is the current version of the LDAP Connector for AS ABAP 7.00?

   a) You can determine the current version of the LDAP Connector on the basis of the last patch level, for example. To do so, go to SAP Service Marketplace at *service.sap.com/patches* and choose *Entry by Application Group → SAP NetWeaver → SAP NETWEAVER → SAP NetWeaver 7.0 (2004S)→ Entry by Component → Application Server ABAP → SAP KERNEL <Version> → <Operating system> → #Database independent*. Click the *Info* link in the row for object *ldap_rfc*.

      Alternatively, you can search for the latest changes to notes (Quick Link */notes*) that contain the search term **`ldap_rfc`**.

      The following SAP Note applied when this document was created: SAP Note 1000644 for LDAP Connector 2.8.6.

Lesson Summary

You should now be able to:

- Explain the concept of user administration as applied to the integration of directory services

Related Information

- Quick link */security → Security in Detail → Secure User Access → Identity Management* on SAP Service Marketplace
- Partner directory under http://www.sap.com/partners (path: *Search for Solutions → Certification Category → LDAP*)
- SAP Note 983808: Certified LDAP directory servers
- Request for Comments (RFCs); see also www.ietf.org and www.rfc-editor.org
  - RFC 1274: The COSINE and Internet X.500 Schema (www.ietf.org/rfc/rfc1274.txt)
  - RFC 2251: Lightweight Directory Access Protocol (v3) (www.ietf.org/rfc/rfc2251.txt)
  - RFC 2256: A Summary of the X.500 (96) User Schema for Use with LDAPv3 (www.ietf.org/rfc/rfc2256.txt)
  - RFC 2798: Definition of the inetOrgPerson LDAP Object Class (www.ietf.org/rfc/rfc2798.txt)
- www.openldap.org (Open Source LDAP implementation)

Lesson: Technical Connection of Directory Services

Lesson Overview

SAP NetWeaver AS ABAP allows you to easily connect an SAP system to a directory service using the LDAP Connector. This lesson explains how this connection is technically implemented.

Lesson Objectives

After completing this lesson, you will be able to:

- Identify prerequisites for connecting an AS ABAP to a directory server
- Configure the AS ABAP for connection to a directory server

Business Example

Your company has decided to connect SAP systems to a directory service. As a member of the user administration team for SAP systems, it is your task to configure the ABAP-based SAP systems so that this connection can be implemented.

Roadmap

The starting point for the following steps is that an SAP system based on AS ABAP and a directory server are to be connected to each other. The focus here is on the LDAP Connector, an operating system program shipped by SAP, and the associated Customizing settings.

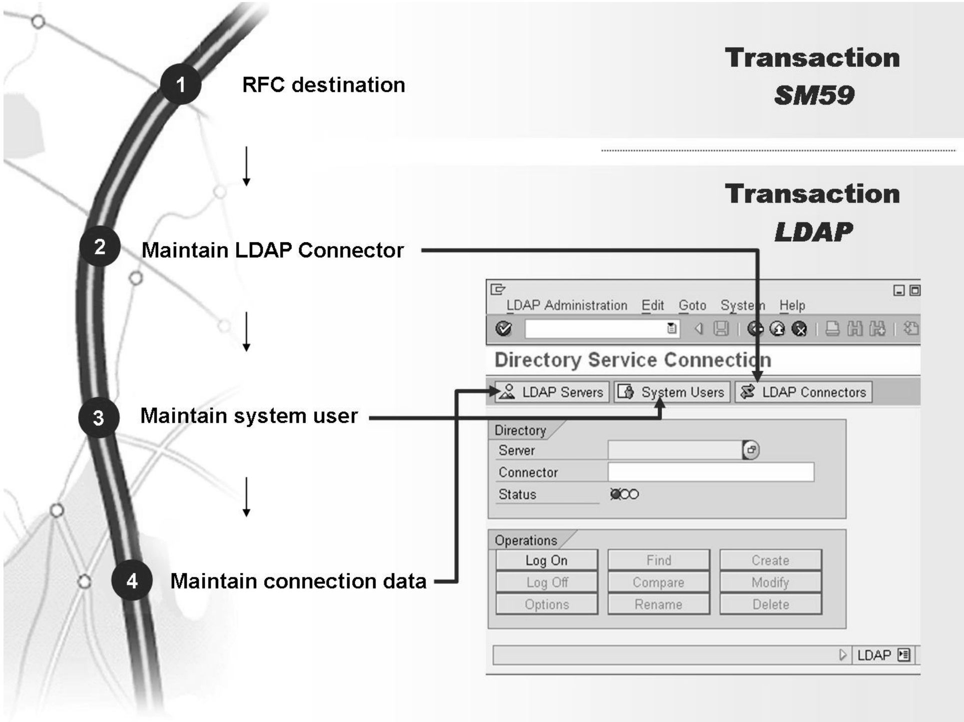


Figure 193: Roadmap for the Technical Connection of Directory Services

All of the steps for connecting the SAP system and the directory server are explained in more detail in the following:

RFC Destination and LDAP Connector

Since SAP Basis 4.6, the LDAP Connector has been delivered with every SAP system. If you start the associated program at operating-system level (the *ldap_rfc* program in the kernel directory), the program outputs a version number and additional technical details. If the LDAP program library is missing for your platform, the system displays an error message.

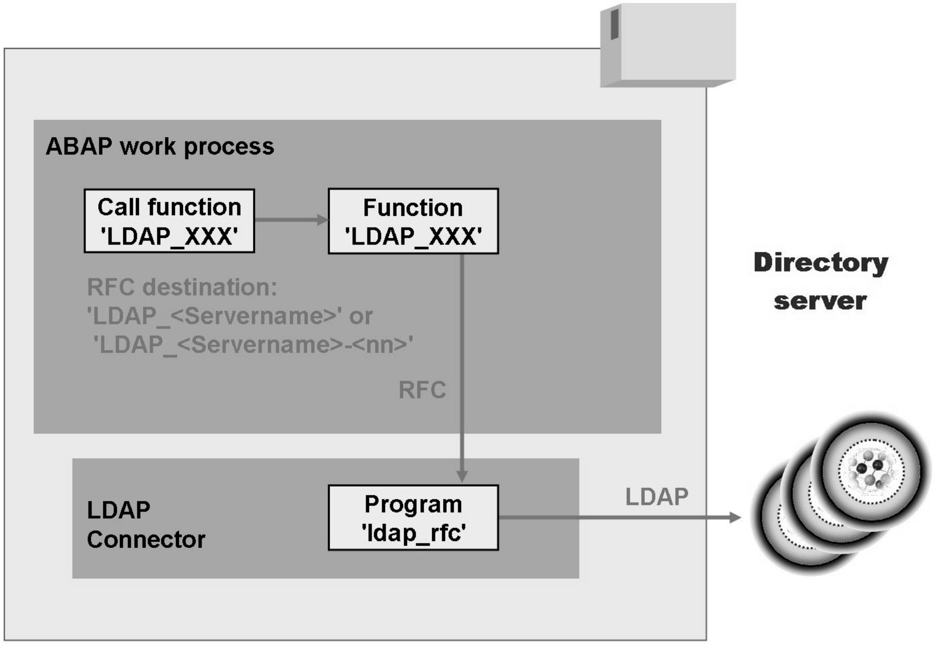


Figure 194: Connecting to Directory Services Using the LDAP Connector

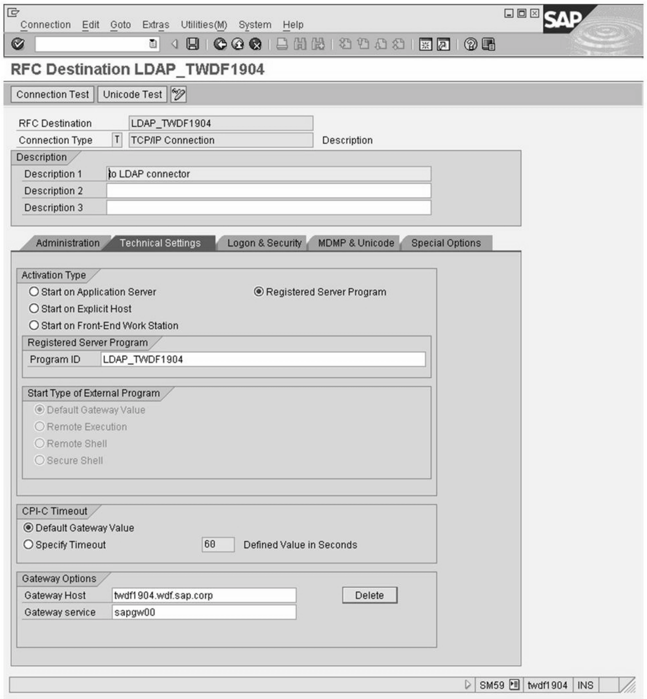
As the program name suggests, the LDAP Connector is addressed using RFC communication. To do this, an RFC destination of the type **`T`** (TCP/IP) that activates the LDAP Connector as a registered server program is required.

Caution: Note that the name of the RFC destination must be written **in uppercase letters and without spaces**.

If only one LDAP Connector is running in an SAP system for each physical host, call the RFC destination **`LDAP_<ApplServername>`** (for example: **`LDAP_TWDF1904`**). If more than one is running, add a sequence number with a hyphen, that is, **`LDAP_<ApplServername>-<Number>`** (for example: **`LDAP_TWDF1904-01`**).

The LDAP Connector should be registered at the local gateway, that is, you must maintain the gateway options so that the *Gateway Host* points to the local server and the *Gateway Service* points to **`sapgw<instance number>`**.

For the *Program ID* field, enter the name of the RFC destination.



RFC Destination LDAP_TWDF1904

RFC Destination	LDAP_TWDF1904
Connection Type	T TCP/IP Connection

Description

Description 1	to LDAP connector
Description 2	
Description 3	

Activation Type

○ Start on Application Server
○ Start on Explicit Host
○ Start on Front-End Work Station
● Registered Server Program

Registered Server Program

Program ID: LDAP_TWDF1904

Start Type of External Program

● Default Gateway Value
○ Remote Execution
○ Remote Shell
○ Secure Shell

CPI-C Timeout

● Default Gateway Value
○ Specify Timeout 60 Defined Value in Seconds

Gateway Options

Gateway Host	twdf1904.wdf.sap.corp
Gateway service	sapgw00

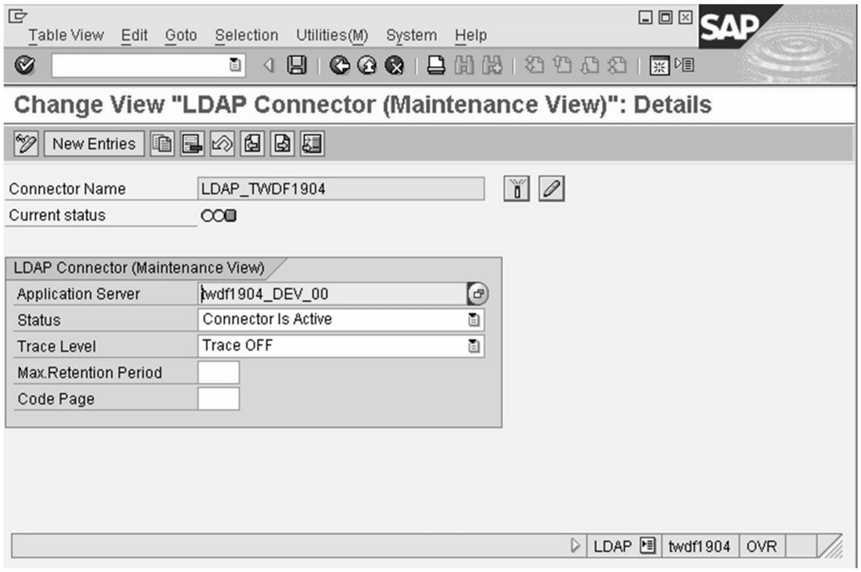
SM59 | twdf1904 | INS

Figure 195: Creating the RFC Destination

You can define all the Customizing settings for connecting directory services in the SAP Reference IMG (Implementation Guide, transaction SPRO) by choosing *SAP Customizing Implementation Guide → SAP NetWeaver → Application Server → System Administration → Directory Integration*. Alternatively, you can also use central transaction LDAP to connect to directory services. There, you can perform Customizing for the LDAP Connector in the *Connector* area. Use the field help (F1 and F4) and the documentation when creating the connection. Use the input help (F4) where possible; for example, the *Connector Name* must be identical to the RFC destination.

For the (logical) definition of the LDAP Connector in the SAP system, you must specify the application server on which the (physical) LDAP Connector is running. You can also create connections to multiple directory services using one LDAP Connector. You usually set the *Status* to *Connector Is Active* – the Computing Center Management System (CCMS) then monitors the status of the LDAP Connector periodically and attempts (for example, after a restart of the SAP system) to make or keep it available using an auto-reaction method. You can set a trace level for the error log (file *dev_<Name of the RFC destination>.trc*. You can evaluate this, for example, with transaction ST11).

After storing the Connector data, you can start the LDAP Connector manually from transaction LDAP or (if the status is set to *Connector Is Active*) wait a few minutes until the auto-reaction method has been executed.



Change View "LDAP Connector (Maintenance View)": Details

Connector Name	LDAP_TWDF1904
Current status	○○■

LDAP Connector (Maintenance View)

Application Server	twdf1904_DEV_00
Status	Connector Is Active
Trace Level	Trace OFF
Max.Retention Period	
Code Page	

Figure 196: Defining the LDAP Connector

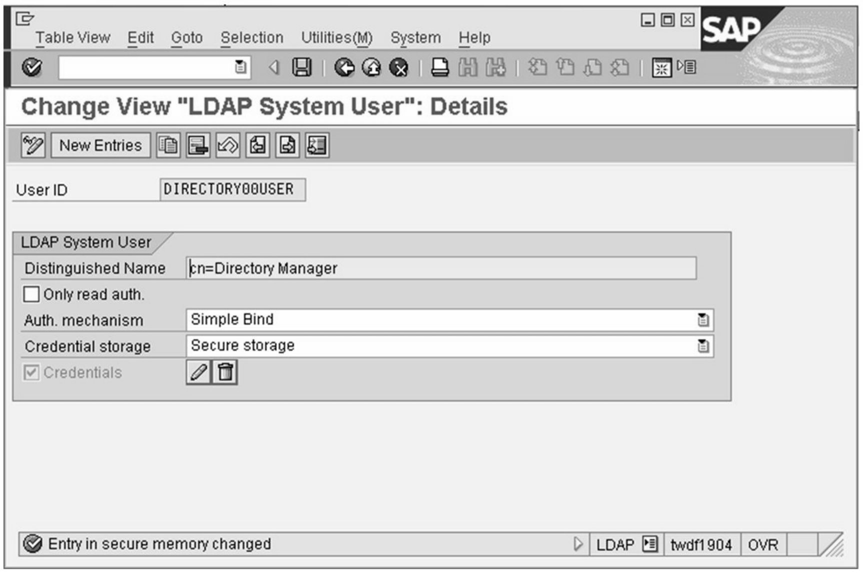
System User

The SAP system must be able to log onto the directory service for the various LDAP operations. The directory services themselves use various authorization concepts (from anyone being able to retrieve information, to extensive identity checks using a directory server).

To do this, enter the logon data for an LDAP system user in the *System Users* area in transaction LDAP using a specific *user ID*. In addition to the name and credentials (password entry), this also includes information about whether a user may access the directory service to read only, and where the credentials are stored.

Note: The name of the created *user ID* corresponds neither to a user in the SAP system, nor to a user in the directory server. Rather, it is a key that is required for the connection data.

This means that you can store the same *user ID* with multiple server definitions in the SAP system. This simplifies administration if the same physical directory service is behind the various logical servers in the SAP system (perhaps to implement a different mapping for employees and vendors).



Change View "LDAP System User": Details

User ID	DIRECTORY00USER

LDAP System User

Distinguished Name	cn=Directory Manager
□ Only read auth.	
Auth. mechanism	Simple Bind
Credential storage	Secure storage
☑ Credentials	

Entry in secure memory changed

LDAP twdf1904 OVR

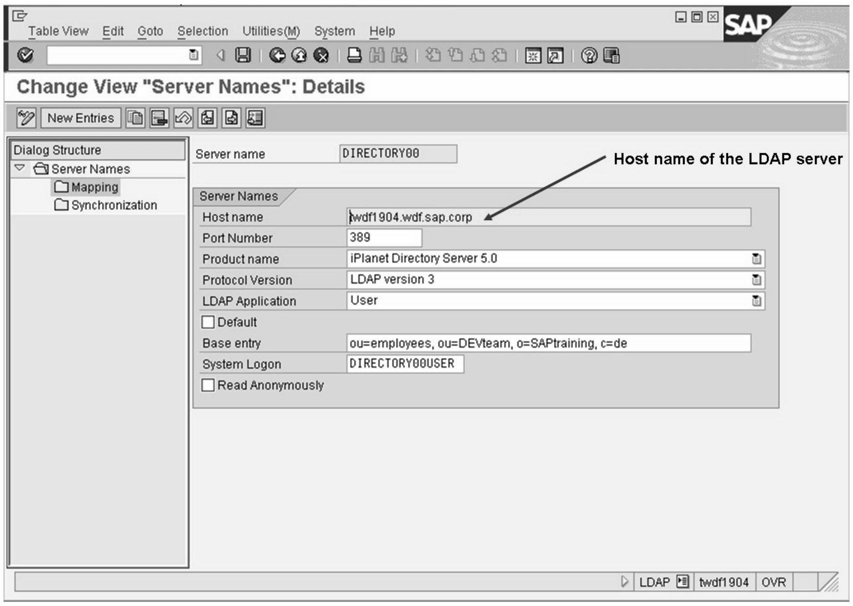
Figure 197: Defining the LDAP System User

Connection Data

After these preparations, define the actual connection data for the directory server to be connected in the *Server* area in transaction LDAP.

This includes host name, port number, and product name of the directory server in use. You also categorize the LDAP application (for example, as a directory for *Users*). The previously defined system user is also included in the connection data.

Note: In the *Server name* field in the SAP system, you can enter a name of your choice. However, you should avoid names that begin with **SAP_**, since these are reserved for SAP applications.



Change View "Server Names": Details

Server name: DIRECTORY00

Server Names

Host name	twdf1904.wdf.sap.corp
Port Number	389
Product name	iPlanet Directory Server 5.0
Protocol Version	LDAP version 3
LDAP Application	User
□ Default	
Base entry	ou=employees, ou=DEVteam, o=SAPtraining, c=de
System Logon	DIRECTORY00USER
□ Read Anonymously	

Host name of the LDAP server

Figure 198: Defining the Connection Data

Exercise 18: Connecting an AS ABAP to a Directory Service

Exercise Objectives

After completing this exercise, you will be able to:

- Make the technical connection of an AS ABAP to a directory server

Business Example

A central directory service is already operating in your company for employee master data. It is your task as an administrator to connect a newly installed SAP system to this directory service.

Task 1: Preparing and Setting Up a Directory Server

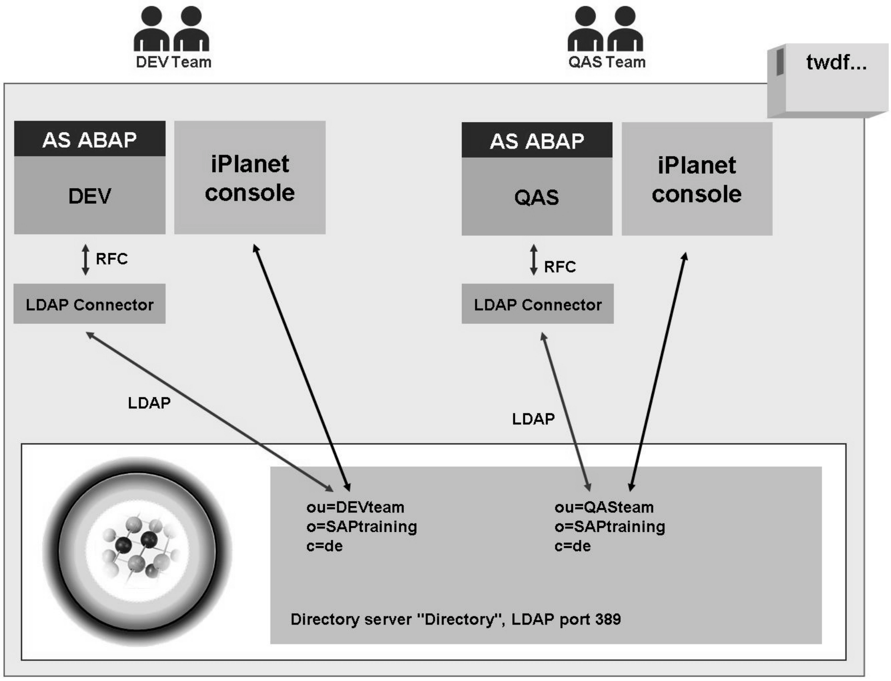


Figure 199: Complete Scenario of the Training Landscape

Start the directory server and create new organizational units.

1. Start the Windows services *iPlanet Administration Server 5.0* and *iPlanet Directory Server 5 (Directory)* on your server. Both services should run automatically if the server is restarted.

   Caution: This part of the task can be performed only **once per server**. Decide with your neighboring group which group is going to perform this part.

   Note: The iPlanet product is used as an example here. The user interfaces and functions of other directory services are different; however, the basic concepts and required entries are identical.

Continued on next page

2. On your server, use the iPlanet Console to create a connection to the directory server shared by the DEV and QAS groups. Your instructor will give you the logon data.

   Note: From this point, the tasks must be carried out by both groups (DEV and QAS).

3. In your directory, create the organizational units **`employees`** and **`partners`** (*organizationalUnit* attribute) below the entry for **your** group (*DEVteam* or *QASteam*, referred to as *<SID>team* in the following text).

   Hint: The reason for creating these two organizational units is that the organizational unit **`employees`** is to be synchronized with the SAP system later, while **`partners`** are administrated in a different way (for example, with a third-party product).

Result

You are operating a directory server (with the logical name *Directory*). On this directory server, an area in the DIT has been set up for your company (with the base entry *ou=<SID>team, o=SAPtraining, c=de*). There are two sections in your company (the organizational units *employees* and *partners*).

Task 2: RFC Destination and LDAP Connector

Technical configuration of the LDAP Connector.

1. Create an RFC destination, **`LDAP_<ApplServername>`** (ensure that you use the correct notation) of type **`T`** for "your" LDAP Connector. **`<ApplServername>`** stands for the host name on which your SAP system is running.

   Caution: Ensure that you enter the name of the RFC destination in uppercase letters and without spaces.

   Use the following settings:

Continued on next page

Activation Type	*Registered server program*
Program ID	Identical to the name of the RFC destination, that is **`LDAP_<ApplServername>`**
CPIC Timeout	*Default Gateway Value*
Gateway Host	Host on which **your** SAP system is running, such as **`twdf1904`** (lowercase)
Gateway Service	**`sapgw<instance number>`**, such as **`sapgw00`** for the DEV group and **`sapgw10`** for the QAS group

2. Define and start your LDAP Connector using transaction LDAP and the following settings:

Connector Name	**`LDAP_<ApplServername>`** (input help)
Application Server	The instance whose gateway you specified in the RFC destination (input help)
Status	*Connector Is Active*
Trace Level	*Trace OFF*
Maximum Hold Time	No entry
Code Page	No entry

Hint: Where possible, use the input help (F4 help).

Result

The LDAP Connector that implements the connection of your SAP system to your directory server is now running at operating-system level.

Task 3: System User

Maintain the logon data.

1. Store the logon data for the system user on your directory server. To do so, use the following settings:

Continued on next page

User ID	`DIRECTORY##USER`
Distinguished Name	`cn=Directory Manager`
Auth. Mechanism	*Simple Bind*
Credentials Storage	*Secure Storage*
Credentials	Provided by instructor

Result

The system user (the user *cn=Directory Manager*) is known in the SAP system.

Task 4: Connection Data and Test

Maintain the connection data and perform an initial test logon to the directory server.

1. Maintain the connection data for your directory server:

Server Name	`DIRECTORY##`, the logical name of the directory service server
Host Name	Host name of your directory service server (for example, `twdf1904.wdf.sap.corp`)
Port Number	`389`
Product Name	*iPlanet Directory Server 5.0*
Log Version	*LDAP v3*
LDAP Application	*User*
Base Entry	`ou=employees, ou=<SID>team, o=SAPtraining, c=de` (in this order)
System Logon	`DIRECTORY##USER` (using F4 help; corresponds to the previously created user ID)

2. Test the configuration by logging on to, searching in, and logging off from the directory server.

Result

The connection data (such as the product used, host name, port, and base entry) are known in the SAP system.

Solution 18: Connecting an AS ABAP to a Directory Service

Task 1: Preparing and Setting Up a Directory Server

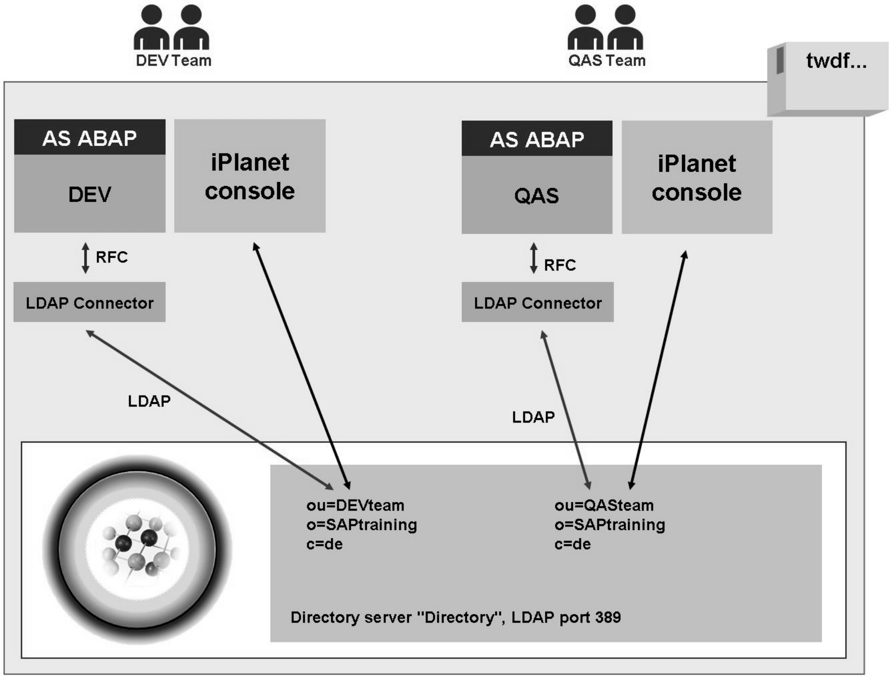


Figure 200: Complete Scenario of the Training Landscape

Start the directory server and create new organizational units.

1. Start the Windows services *iPlanet Administration Server 5.0* and *iPlanet Directory Server 5 (Directory)* on your server. Both services should run automatically if the server is restarted.

Continued on next page

Caution: This part of the task can be performed only **once per server**. Decide with your neighboring group which group is going to perform this part.

Note: The iPlanet product is used as an example here. The user interfaces and functions of other directory services are different; however, the basic concepts and required entries are identical.

a) If you have not already done so, log on to the Terminal Services Client (also known as the RDP Client) at operating system level of your SAP system using the user **`<sid>adm`**.

b) Open the Windows Service Manager (using the desktop shortcut or by choosing *Start* → *Settings* → *Control Panel* → *Administrative Tools* → *Services*).

c) Double-click the *iPlanet Administration Server 5.0* service. On the *General* tab page under *Startup Type*, choose *Automatic* and then *Start*. Close the window by choosing *OK*.

d) Repeat the last steps for the *iPlanet Directory Server 5 (Directory)* service.

Note that the following steps must be carried out by both groups (DEV and QAS).

2. On your server, use the iPlanet Console to create a connection to the directory server shared by the DEV and QAS groups. Your instructor will give you the logon data.

Note: From this point, the tasks must be carried out by both groups (DEV and QAS).

a) Start the iPlanet Console at operating system level on your server by choosing *Start* → *Programs* → *iPlanet Server Products* → *iPlanet Console 5.0*.

b) Go to the *Servers and Applications* tab page for the directory server *wdf.sap.corp* → *localhost.wdf.sap.corp* → *Server Group* → *"Directory Server" (Directory)*.

c) Open this server by double-clicking (or choosing *Open*).

Note that this directory server is used by both groups (DEV and QAS).

Continued on next page

3. In your directory, create the organizational units **`employees`** and **`partners`** (*organizationalUnit* attribute) below the entry for **your** group (*DEVteam* or *QASteam*, referred to as *<SID>team* in the following text).

   Hint: The reason for creating these two organizational units is that the organizational unit **`employees`** is to be synchronized with the SAP system later, while **`partners`** are administrated in a different way (for example, with a third-party product).

   a) Go to the *Directory* tab page in the directory server window.

   b) Select **your** organizational unit (*SAPtraining* → *<SID>team*) using the primary mouse button.

   c) Then click with the secondary mouse button and choose *New* → *Organizational Unit...*. Under *Name*, enter the value **`employees`**. All other fields are optional and can be left blank. Save your entries by choosing *OK*.

   Repeat the last step for the **`partners`** organizational unit.

Result

You are operating a directory server (with the logical name *Directory*). On this directory server, an area in the DIT has been set up for your company (with the base entry *ou=<SID>team, o=SAPtraining, c=de*). There are two sections in your company (the organizational units *employees* and *partners*).

Task 2: RFC Destination and LDAP Connector

Technical configuration of the LDAP Connector.

1. Create an RFC destination, **`LDAP_<ApplServername>`** (ensure that you use the correct notation) of type **`T`** for “your” LDAP Connector. **`<ApplServername>`** stands for the host name on which your SAP system is running.

   Caution: Ensure that you enter the name of the RFC destination in uppercase letters and without spaces.

   Use the following settings:

Continued on next page

Activation Type	*Registered server program*
Program ID	Identical to the name of the RFC destination, that is `LDAP_<ApplServername>`
CPIC Timeout	*Default Gateway Value*
Gateway Host	Host on which **your** SAP system is running, such as `twdf1904` (lowercase)
Gateway Service	`sapgw<instance number>`, such as `sapgw00` for the DEV group and `sapgw10` for the QAS group

a) In transaction SM59, choose *Create*. Enter the RFC destination `LDAP_<ApplServername>` in uppercase letters (for example, a group with the SAP system on the host twdf1904 would enter: `LDAP_TWDF1904`). Enter `T` as the connection type (start an external program using TCP/IP) and any description text.

b) Maintain the fields on the *Technical Settings* tab page, as specified in the table, and save your connection.

> **Note:** Do not be surprised if the connection test fails at this point. It will not work until the LDAP Connector has started.

2. Define and start your LDAP Connector using transaction LDAP and the following settings:

Connector Name	`LDAP_<ApplServername>` (input help)
Application Server	The instance whose gateway you specified in the RFC destination (input help)
Status	*Connector Is Active*

Continued on next page

Trace Level	*Trace OFF*
Maximum Hold Time	No entry
Code Page	No entry

Hint: Where possible, use the input help (F4 help).

a) In transaction LDAP, navigate to the *LDAP Connectors* area. Switch to change mode and choose *New Entries*.

b) Fill the fields as specified in the task. Don't forget to use the F4 help. F1 Help provides you with explanations about all input fields.

c) Leave change mode (for example, by calling transaction LDAP again). Navigate to the *LDAP Connector* view again and start your LDAP Connector by choosing *Start Connector*.

A green traffic light should now indicate that the Connector was started successfully. If a green light does not appear after you choose *Enter*, use transaction ST11 to analyze the trace file **`dev_<name of the RFC destination>.trc`**.

Result

The LDAP Connector that implements the connection of your SAP system to your directory server is now running at operating-system level.

Task 3: System User

Maintain the logon data.

1. Store the logon data for the system user on your directory server. To do so, use the following settings:

Continued on next page

User ID	**`DIRECTORY##USER`**
Distinguished Name	**`cn=Directory Manager`**
Auth. Mechanism	*Simple Bind*
Credentials Storage	*Secure Storage*
Credentials	Provided by instructor

a) On the initial screen of transaction LDAP, choose *System Users*. Switch to change mode and choose *New Entries*.

b) Fill the fields as specified in the task. Under *Credentials*, enter the password provided by your instructor twice. When saving, the system prompts you for a Customizing request. If there is not yet a Customizing request, create one.

Result

The system user (the user *cn=Directory Manager*) is known in the SAP system.

Task 4: Connection Data and Test

Maintain the connection data and perform an initial test logon to the directory server.

1. Maintain the connection data for your directory server:

Server Name	**`DIRECTORY##`**, the logical name of the directory service server
Host Name	Host name of your directory service server (for example, **`twdf1904.wdf.sap.corp`**)
Port Number	**`389`**
Product Name	*iPlanet Directory Server 5.0*
Log Version	*LDAP v3*

Continued on next page

LDAP Application	*User*
Base Entry	**`ou=employees, ou=<SID>team, o=SAPtraining, c=de`** (in this order)
System Logon	**`DIRECTORY##USER`** (using F4 help; corresponds to the previously created user ID)

a) On the initial screen in transaction LDAP, choose *LDAP Servers*. Switch to change mode and choose *New Entries*.

b) Fill the fields as specified in the task and save your entries. Use the Customizing request from the previous section of the task.

2. Test the configuration by logging on to, searching in, and logging off from the directory server.

a) On the initial screen of transaction LDAP, select your server and your Connector (using F4 help).

b) Perform the *Log On* operation. Select *Use System User* and choose *Execute*. On the initial screen of transaction LDAP, you should now see the status *Logged On*.

c) After you have logged on successfully, perform a search with the default values. To do so, choose *Find* and then *Execute* – exactly one entry should be found.

d) You can also perform another search by removing **`ou=employees`** and **`ou=<SID>team`** from the default *Base entry* – the search should now return all separate entries (including the organizational units *employees* and *partners*) and those belonging to the neighboring group.

e) Log off from the directory server again (*Log Off* pushbutton).

Result

The connection data (such as the product used, host name, port, and base entry) are known in the SAP system.

Lesson Summary

You should now be able to:

- Identify prerequisites for connecting an AS ABAP to a directory server
- Configure the AS ABAP for connection to a directory server

Related Information

- Online documentation for SAP NetWeaver 7.0: *SAP Library → SAP NetWeaver Library → SAP NetWeaver by Key Capability → Security → Identity Management → Identity Management of the Application Server ABAP → Configuration of Identity Management → Directory Services*
- Quick link */security → Security in Detail → Secure User Access → Identity Management* on SAP Service Marketplace

Lesson: Data Exchange with Directory Services

Lesson Overview

The actual synchronization process is presented in this lesson. To do this, you must define which fields in the SAP system are associated with which attributes of the directory service (possibly to be newly created). This assignment is called **mapping**.

Lesson Objectives

After completing this lesson, you will be able to:

- Exchange data between the AS ABAP and a directory server using the LDAP protocol

Business Example

As a member of the user administration team for SAP systems, you have configured a connection in an SAP NetWeaver AS ABAP to a directory service. You now want to synchronize the data between the SAP system and the directory service using the LDAP protocol.

Roadmap

After the technical connection of a directory server to an SAP system based on AS ABAP, the actual data exchange has to be prepared and performed. To do this, information is required about how SAP data fields are mapped to directory attributes, which fields are used where (mapping indicators), and the direction in which data is synchronized (synchronization indicators).

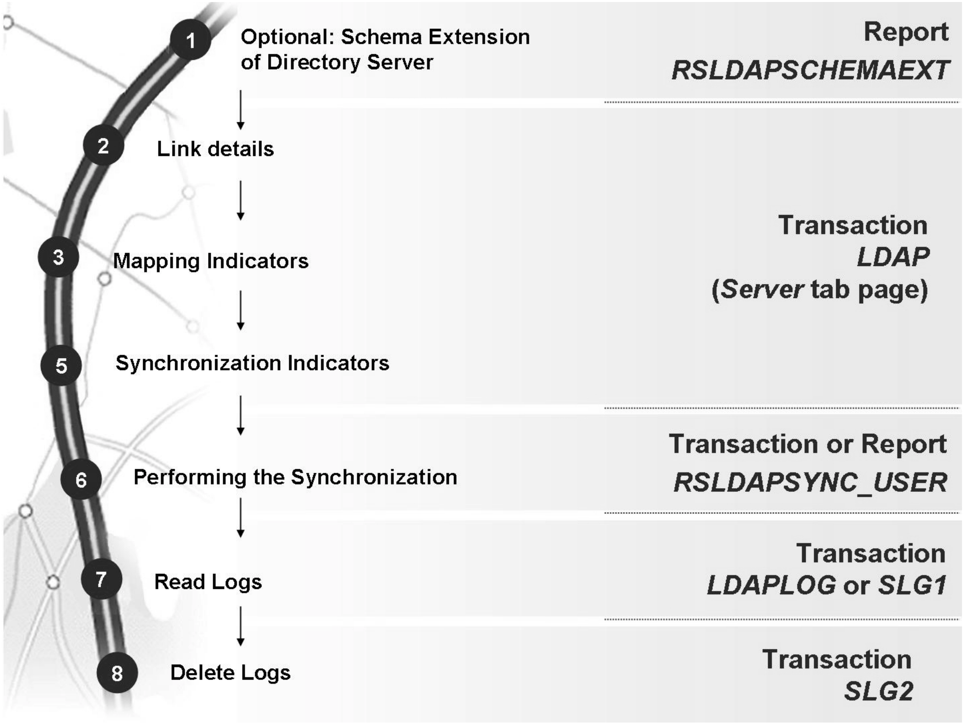

1 Optional: Schema Extension of Directory Server
2 Link details
3 Mapping Indicators
5 Synchronization Indicators
6 Performing the Synchronization
7 Read Logs
8 Delete Logs
Report *RSLDAPSCHEMAEXT*
Transaction *LDAP* (*Server* tab page)
Transaction or Report *RSLDAPSYNC_USER*
Transaction *LDAPLOG* or *SLG1*
Transaction *SLG2*

Figure 201: Roadmap for Data Exchange with Directory Services

All of the steps are described in more detail in the following:

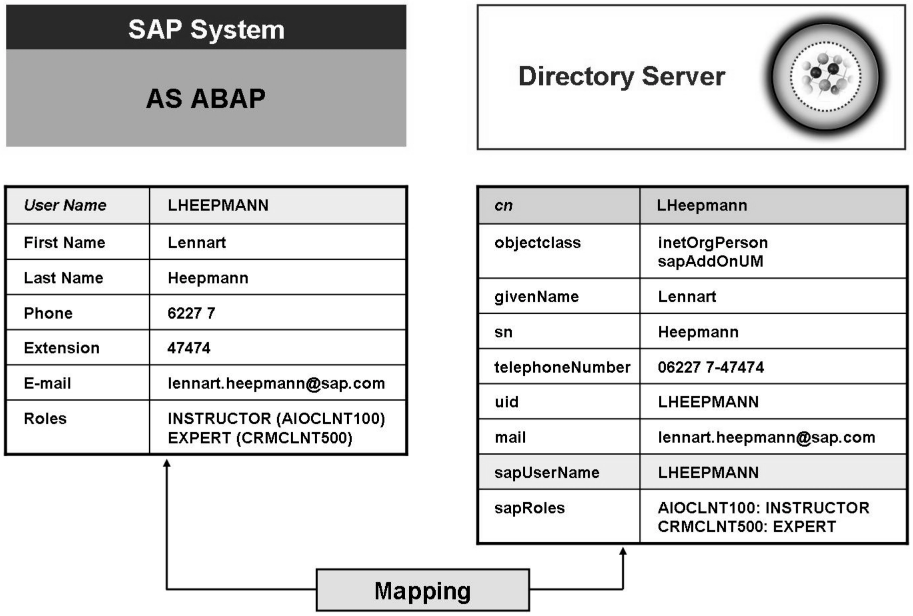
Mapping

Mapping describes the assignment of SAP fields to directory attributes. In the simplest case, this is a 1:1 assignment. However, more complex link rules may be required.

Schema Extension

The directory servers from various vendors have much in common in their data models (schemata). On the other hand, there are differences in the details. Understandably, there are no attributes for SAP-specific information in the vendor standards, such as

- SAP user name
- Alias
- Validity period
- Settings for logon language, printer, number formats, time zone
- Assigned roles
- Assigned authorization profiles



User Name	LHEEPMANN
First Name	Lennart
Last Name	Heepmann
Phone	6227 7
Extension	47474
E-mail	lennart.heepmann@sap.com
Roles	INSTRUCTOR (AIOCLNT100) EXPERT (CRMCLNT500)

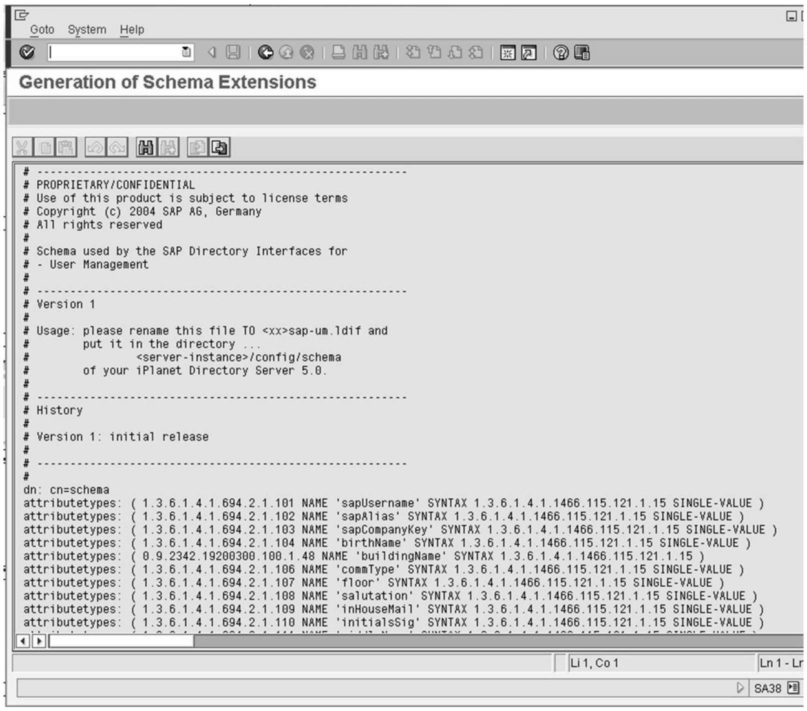
cn	LHeepmann
objectclass	inetOrgPerson sapAddOnUM
givenName	Lennart
sn	Heepmann
telephoneNumber	06227 7-47474
uid	LHEEPMANN
mail	lennart.heepmann@sap.com
sapUserName	LHEEPMANN
sapRoles	AIOCLNT100: INSTRUCTOR CRMCLNT500: EXPERT

Figure 202: Mapping SAP Data Fields to Directory Attributes

SAP has decided not to develop its own schema, but rather to support the schemata available on the market. This is important for heterogeneous system landscapes, if you want to connect non-SAP applications to your directory service in addition to SAP systems.

As the delivered schemata for the various products do not know many SAP data fields, you must use schema extensions to make these fields known. An extension of this type is provided and imported into the directory server in a special format (LDIF for “LDAP Data Interchange Format”).

You can easily create files of this type in the SAP system for all certified and some additional products using report *RSLDAPSCHEMAEXT*. After you have chosen a product, this report outputs a list that you can save as a file on your front end computer and then import into your directory server. For more information, see the comment lines in the file and SAP Note 888848.



Generation of Schema Extensions

```
# -----------------------------------------------------
# PROPRIETARY/CONFIDENTIAL
# Use of this product is subject to license terms
# Copyright (c) 2004 SAP AG, Germany
# All rights reserved
#
# Schema used by the SAP Directory Interfaces for
# - User Management
#
# -----------------------------------------------------
# Version 1
#
# Usage: please rename this file TO <xx>sap-um.ldif and
#        put it in the directory ...
#               <server-instance>/config/schema
#        of your iPlanet Directory Server 5.0.
#
# -----------------------------------------------------
# History
#
# Version 1: initial release
#
# -----------------------------------------------------
#
dn: cn=schema
attributetypes: ( 1.3.6.1.4.1.694.2.1.101 NAME 'sapUsername' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.694.2.1.102 NAME 'sapAlias' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.694.2.1.103 NAME 'sapCompanyKey' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.694.2.1.104 NAME 'birthName' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
attributetypes: ( 0.9.2342.19200300.100.1.48 NAME 'buildingName' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
attributetypes: ( 1.3.6.1.4.1.694.2.1.106 NAME 'commType' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.694.2.1.107 NAME 'floor' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.694.2.1.108 NAME 'salutation' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.694.2.1.109 NAME 'inHouseMail' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.694.2.1.110 NAME 'initialsSig' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
```

Li 1, Co 1

Ln 1 - Ln

Figure 203: Generating a Schema Extension

Linking Details

Essentially, mapping is about linking an SAP data field with a directory attribute. As stated previously, this can be a simple 1:1 assignment or a more complex rule, if, for example, multiple SAP fields are to be mapped to one attribute (or multiple attributes to one SAP field).

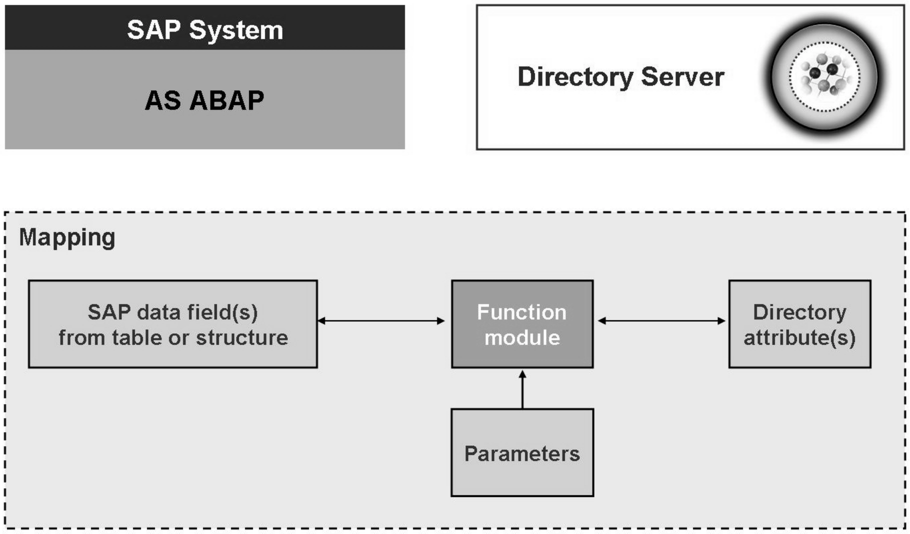


Figure 204: Structure of a Mapping

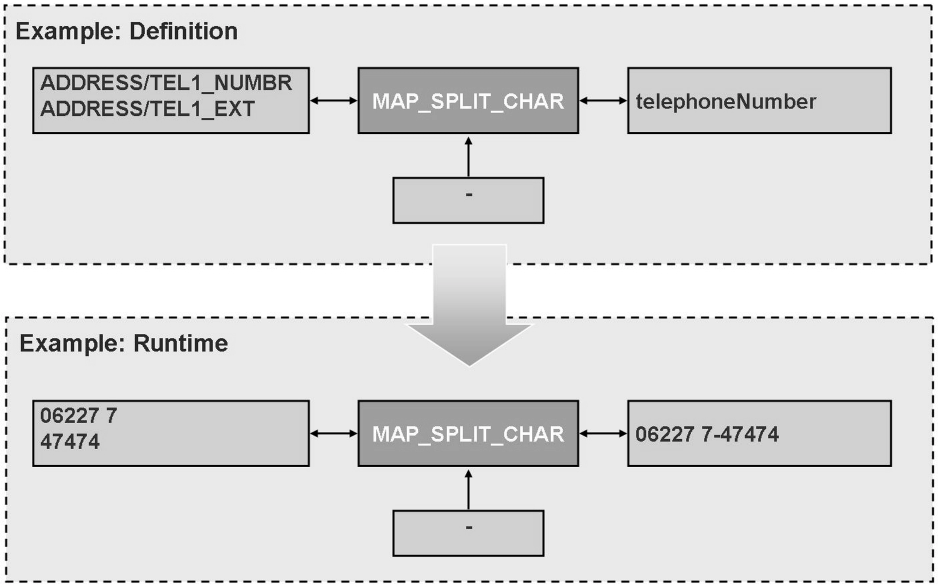


Figure 205: Example of a Mapping

SAP delivers a number of function modules that implement certain types of link. Additional parameters are required to some extent, for example, to specify a particular character at which a string is to be divided. Customers can, of course, also develop their own modules for particular purposes.

Note: The names of the delivered function modules all start with *MAP_** and are in the function group *FSLDAP_MAP*. For information about their functions, see the BC-SEC-DIR area of the SAP Library.

If no function modules are specified during mapping, function module *MAP_DEFAULT* is used implicitly; this function module is a simple 1:1 assignment.

Mapping Indicator

In addition to the link (one or more SAP data fields with one or more directory attributes), a mapping also includes its purpose. Is the mapping to be used only for export, only for import, or for both synchronization directions? Are there mandatory fields, without which it is not possible to create an entry in the directory server?

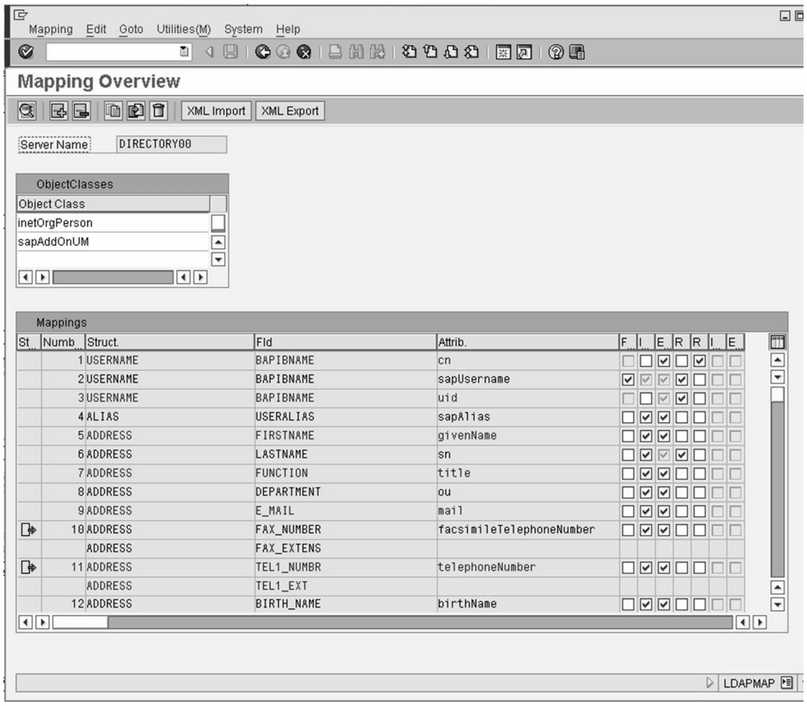
These questions and others are answered by the indicators to be maintained for each mapping. You can perform the associated Customizing in transaction LDAP on the *LDAP Servers* tab page, in the *Mapping* area.

Meaning of the Mapping Indicators

Mapping Indicator	Meaning
Filter	Mapping used for the search filter of the directory server; corresponds to the key in the SAP system (exists only once; must match the LDAP application; usually for the same mapping as the RDN flag)
Import	Mapping that can be used for reading directory entries
Export	Mapping that can be used for writing directory entries
Required	Mapping without which it is not possible to write an entry in the directory (can only be chosen for mappings with the *Export* indicator)
RDN	Mapping used to create the RDN (must exist for exactly one mapping)

SAP provides a mapping proposal for a large number of products, which you can transfer in the mapping maintenance at the push of a button.

Hint: This function is only possible if you have chosen the LDAP application *User* when maintaining the connection data and selected a supported product.



Mapping Overview

Server Name: DIRECTORY00

ObjectClasses

Object Class
inetOrgPerson
sapAddOnUM

Mappings

St	Numb	Struct.	Fld	Attrib.	F	I	E	R	R	I	E
	1	USERNAME	BAPIBNAME	cn	□	□	☑	□	☑	□	□
	2	USERNAME	BAPIBNAME	sapUsername	☑	☑	☑	☑	□	□	□
	3	USERNAME	BAPIBNAME	uid	□	□	☑	☑	□	□	□
	4	ALIAS	USERALIAS	sapAlias	□	☑	☑	□	□	□	□
	5	ADDRESS	FIRSTNAME	givenName	□	☑	☑	□	□	□	□
	6	ADDRESS	LASTNAME	sn	□	☑	☑	☑	□	□	□
	7	ADDRESS	FUNCTION	title	□	☑	☑	□	□	□	□
	8	ADDRESS	DEPARTMENT	ou	□	☑	☑	□	□	□	□
	9	ADDRESS	E_MAIL	mail	□	☑	☑	□	□	□	□
⇥	10	ADDRESS	FAX_NUMBER	facsimileTelephoneNumber	□	☑	☑	□	□	□	□
		ADDRESS	FAX_EXTENS								
⇥	11	ADDRESS	TEL1_NUMBR	telephoneNumber	□	☑	☑	□	□	□	□
		ADDRESS	TEL1_EXT								
	12	ADDRESS	BIRTH_NAME	birthName	□	☑	☑	□	□	□	□

Figure 206: Mapping (in Accordance with SAP Proposal, Extended to Include an Additional Link)

Synchronization Indicators

As the customer, you must finally decide the direction in which a synchronization works. There is no SAP proposal for this, as it depends on the scenario which data is maintained where. It is certainly conceivable that, during a synchronization for a user, some information is transferred from the directory server to the SAP system (import) and other data is transferred in the opposite direction, that is, is written to the directory (export).

You can set the synchronization indicators for each individual mapping to *Import*, *Export*, or not at all in transaction LDAP on the *LDAP Servers* tab page, in the *Synchronization* area.

Meaning of the Synchronization Indicators

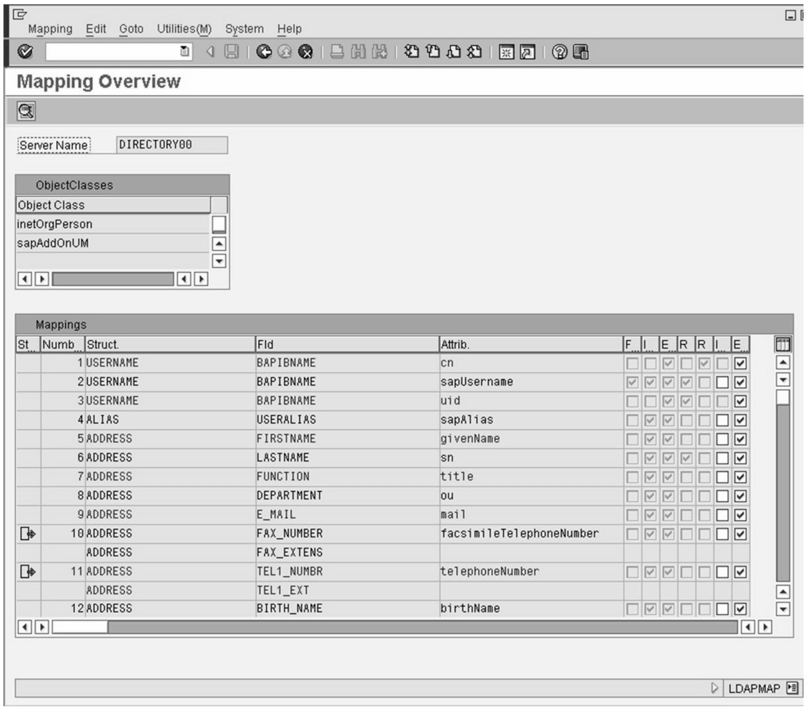
Synchronization Indicators	**Meaning**
Import	Mapping is imported during the synchronization (this can only be selected for mappings with the *Import* mapping indicator)
Export	Mapping is exported during the synchronization (this can only be selected for mappings with the *Export* mapping indicator)
None	Mapping is not transferred at all

Regardless of the indicators that you have set, the following rules apply:

- The fields that are mandatory in the directory (*Required* mapping indicator) and the key fields in the SAP system (*Filter* mapping indicator) are automatically exported when creating (but not when updating) an entry in the directory.
- The fields that are mandatory in the SAP system (this cannot be identified in the mapping maintenance) and the key fields in the SAP system (*Filter* mapping indicator) are automatically imported when creating (but not when updating) an entry in the SAP system.

For existing objects (that is, when updating objects), only the synchronization indicator settings apply.

Caution: In addition to field mapping, the synchronization indicators are vital for a correct data exchange. Incorrect Customizing can lead to data loss or inconsistencies.



Mapping Overview

Server Name DIRECTORY00

ObjectClasses
Object Class
inetOrgPerson
sapAddOnUM

Mappings

St	Numb	Struct.	Fld	Attrib.	F	I	E	R	R	I	E
	1	USERNAME	BAPIBNAME	cn	□	□	☑	□	☑	□	☑
	2	USERNAME	BAPIBNAME	sapUsername	☑	☑	☑	☑	□	□	☑
	3	USERNAME	BAPIBNAME	uid	□	□	☑	☑	□	□	☑
	4	ALIAS	USERALIAS	sapAlias	□	☑	☑	□	□	□	☑
	5	ADDRESS	FIRSTNAME	givenName	□	☑	☑	□	□	□	☑
	6	ADDRESS	LASTNAME	sn	□	☑	☑	☑	□	□	☑
	7	ADDRESS	FUNCTION	title	□	☑	☑	□	□	□	☑
	8	ADDRESS	DEPARTMENT	ou	□	☑	☑	□	□	□	☑
	9	ADDRESS	E_MAIL	mail	□	☑	☑	□	□	□	☑
⇥	10	ADDRESS	FAX_NUMBER	facsimileTelephoneNumber	□	☑	☑	□	□	□	☑
		ADDRESS	FAX_EXTENS								
⇥	11	ADDRESS	TEL1_NUMBR	telephoneNumber	□	☑	☑	□	□	□	☑
		ADDRESS	TEL1_EXT								
	12	ADDRESS	BIRTH_NAME	birthName	□	☑	☑	□	□	□	☑

Figure 207: Maintaining the Synchronization Indicators

Executing and Monitoring the Synchronization

After you have made all settings, you can perform the synchronization, perhaps first manually as a test and then later, in production, as a periodically scheduled background job.

Preparing and Performing the Synchronization

It is important that you are clear about what data is maintained where. Not all user attributes must necessarily be administered in one place – in this way, users could be created, assigned a telephone number, and assigned an SAP role in different places. Users should, of course, be created at exactly one place.

Usually, either the directory or the SAP system defines the existence of users. Users newly created there can then be automatically generated in the system and users deleted there can be automatically deleted or locked. However, this “one way street” is not mandatory. A synchronization indicator must be maintained for each mapping where attributes can be synchronized in different directions (exactly one direction for each attribute; either import or export).

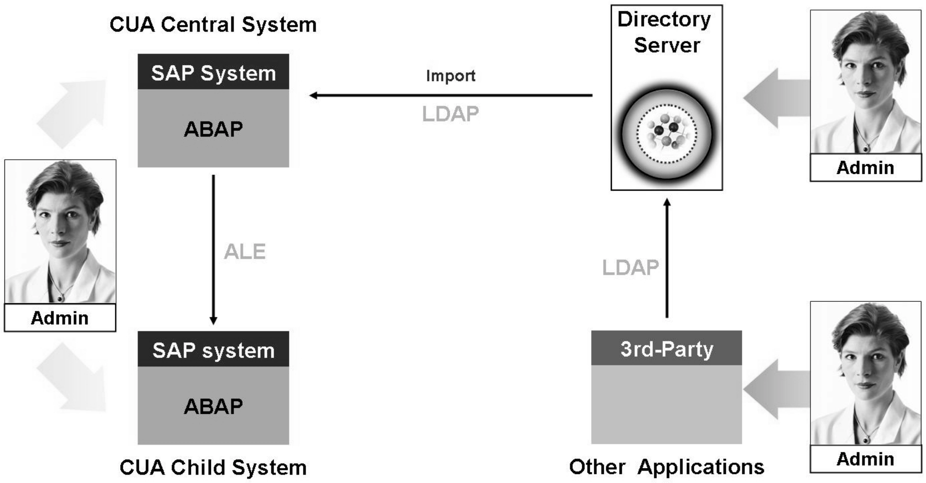


Figure 208: Data Maintenance in the Directory Server (or in an External Application)

In the case illustrated, (at least specific) users are either maintained manually in the directory server or generated by an external application such as in an HR system. Newly created users would then be created in the SAP system during the next synchronization. If this is the central system of Central User Administration (CUA), the data would also be created in the specified child systems.

In this scenario, it is still technically possible for an SAP administrator to administrate the user master record in the central system and (depending on the configuration) also in the child systems. However, these changes may be overwritten during the next synchronization process.

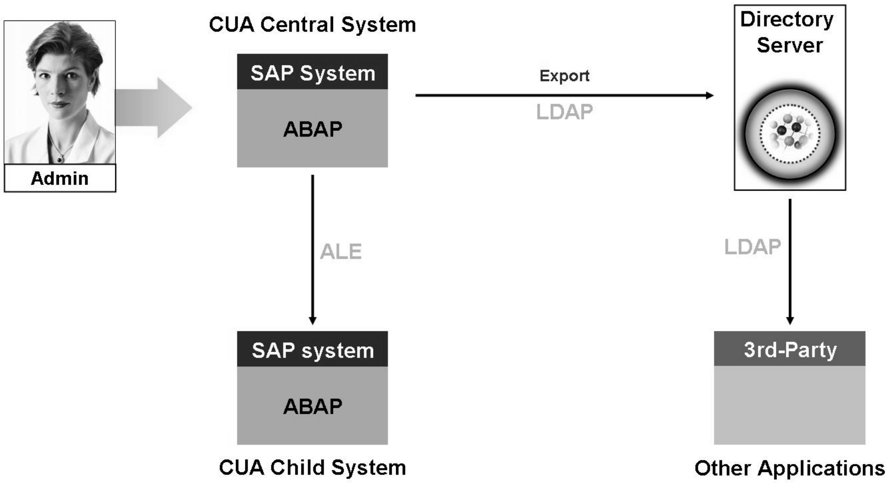


Figure 209: Data Maintenance in the SAP system

Another scenario could have the connected SAP system as the leading system. Users created or attributes maintained in this system are immediately distributed to the affected child systems. After the next synchronization, this data also exists in the directory server and can be accessed by other applications.

You always control the actual synchronization with the directory server from the SAP system, using transaction RSLDAPSYNC_USER or the report with the same name: **RSLDAPSYNC_USER**. You can start the report in dialog (for example, with transaction SA38), or schedule it as a background job with a suitable variant. You specify the desired LDAP server when you call the report (the system can determine the LDAP Connector automatically). You can also restrict the synchronization to certain user names, groups, or types.

During the synchronization process, three groups of users are identified, which are processed sequentially in accordance with the settings of report **RSLDAPSYNC_USER** and the synchronization indicators.

1. Handling of the users that exist only in the AS ABAP database (DB)
2. Handling of the users that exist only in the directory server (DS)
3. Handling of the users that exist in both repositories (DB/DS)

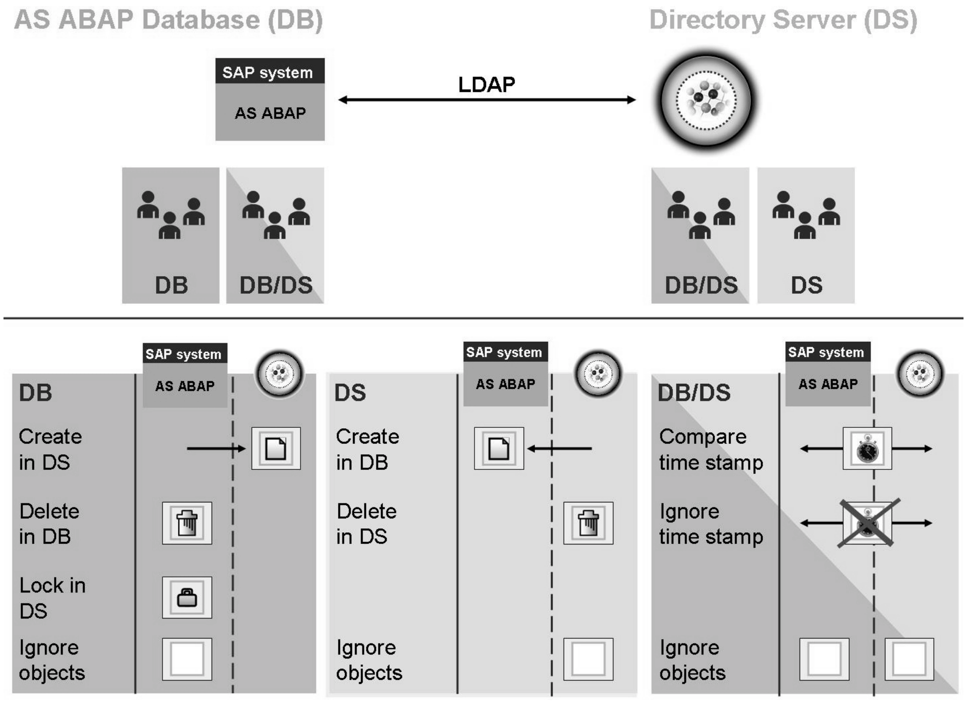
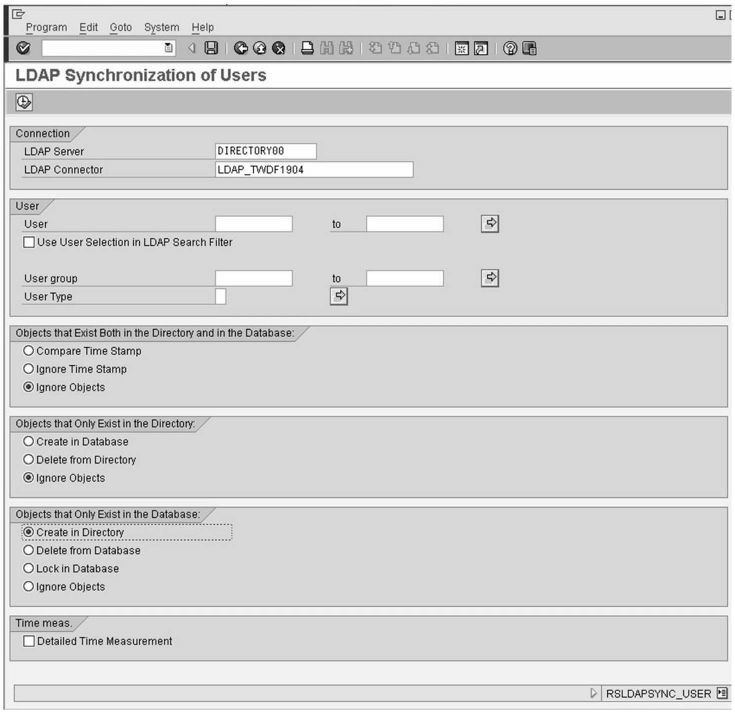


Figure 210: Sequence and Options for Synchronization

RSLDAPSYNC_USER offers different options for each of the identified user groups. The *Ignore Objects* default settings for each group are non-critical. With a run of this type, no objects are changed; however, you can find out from the log how many objects would be synchronized.

- The users SAP*, DDIC, SAPCPIC, and EARLYWATCH are always excluded from every synchronization.
- Instead of deleting users (that do not exist in the directory), we recommend that you simply lock them (for audit purposes).
- Before you delete users (that do not exist in the SAP system) from the directory, you must be sure that the entries are not required for other applications.
- *Compare Time Stamp* for the users that exist in both repositories performs a delta synchronization (that is, only users changed since the last run are synchronized).
- *Ignore Time Stamp* for the users in both repositories performs a complete synchronization (that is, all users are synchronized in accordance with the synchronization indicators).



LDAP Synchronization of Users

Connection

LDAP Server: DIRECTORY00

LDAP Connector: LDAP_TWDF1904

User

User ___ to ___

□ Use User Selection in LDAP Search Filter

User group ___ to ___

User Type ___

Objects that Exist Both in the Directory and in the Database:

○ Compare Time Stamp

○ Ignore Time Stamp

● Ignore Objects

Objects that Only Exist in the Directory:

○ Create in Database

○ Delete from Directory

● Ignore Objects

Objects that Only Exist in the Database:

● Create in Directory

○ Delete from Database

○ Lock in Database

○ Ignore Objects

Time meas.

□ Detailed Time Measurement

RSLDAPSYNC_USER

Figure 211: Performing the Synchronization with Report RSLDAPSYNC_USER

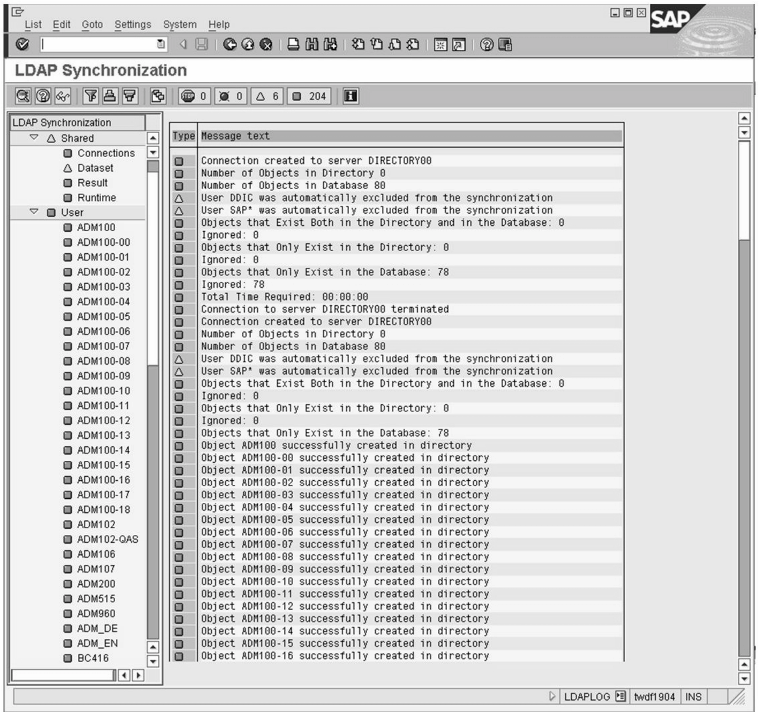
Note: If an attribute in the directory has a value that is longer than the ABAP target field, the field contents are truncated without the administrator being informed of this change. A solution is available as of NetWeaver 7.0 SP-Stack 14 (for details, see SAP Note 1081379 – *Truncated field contents with the report RSLDAPSYNC_USER*).

Logs

If you run *RSLDAPSYNC_USER* online, the system displays the associated log immediately after the program run. Small traffic light icons show you whether there were problems. The runtime required is also displayed.

If you ran the synchronization in the background (or want to access older logs), transaction LDAPLOG allows you to evaluate historical logs. The selection screen allows you to use convenient search options.

Note: As an alternative to transaction LDAPLOG, you can also use the (general) transaction SLG1 (use it with object *LDAPSYNC* and subobject *USER*).

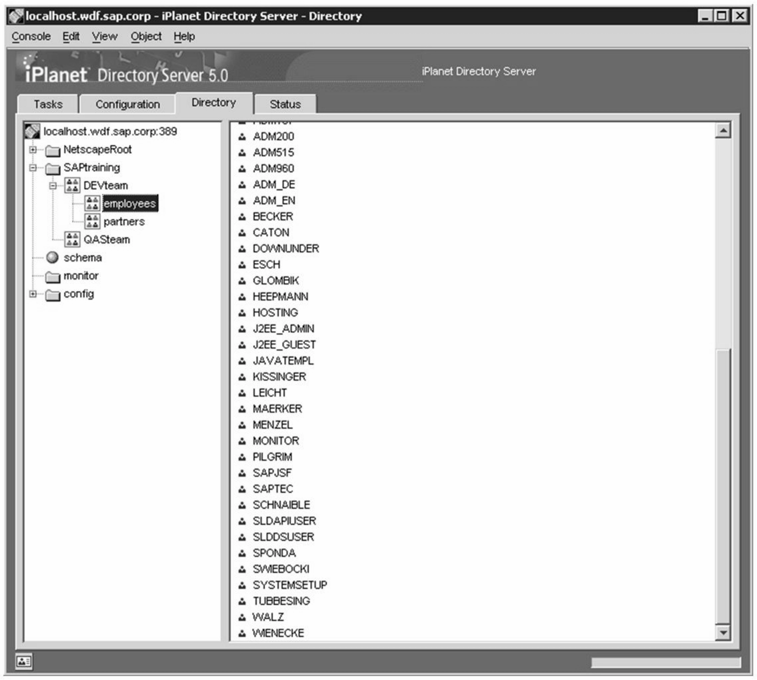


```
Type Message text

■    Connection created to server DIRECTORY00
■    Number of Objects in Directory 0
■    Number of Objects in Database 80
△    User DDIC was automatically excluded from the synchronization
△    User SAP* was automatically excluded from the synchronization
■    Objects that Exist Both in the Directory and in the Database: 0
■    Ignored: 0
■    Objects that Only Exist in the Directory: 0
■    Ignored: 0
■    Objects that Only Exist in the Database: 78
■    Ignored: 78
■    Total Time Required: 00:00:00
■    Connection to server DIRECTORY00 terminated
■    Connection created to server DIRECTORY00
■    Number of Objects in Directory 0
■    Number of Objects in Database 80
△    User DDIC was automatically excluded from the synchronization
△    User SAP* was automatically excluded from the synchronization
■    Objects that Exist Both in the Directory and in the Database: 0
■    Ignored: 0
■    Objects that Only Exist in the Directory: 0
■    Ignored: 0
■    Objects that Only Exist in the Database: 78
■    Object ADM100 successfully created in directory
■    Object ADM100-00 successfully created in directory
■    Object ADM100-01 successfully created in directory
■    Object ADM100-02 successfully created in directory
■    Object ADM100-03 successfully created in directory
■    Object ADM100-04 successfully created in directory
■    Object ADM100-05 successfully created in directory
■    Object ADM100-06 successfully created in directory
■    Object ADM100-07 successfully created in directory
■    Object ADM100-08 successfully created in directory
■    Object ADM100-09 successfully created in directory
■    Object ADM100-10 successfully created in directory
■    Object ADM100-11 successfully created in directory
■    Object ADM100-12 successfully created in directory
■    Object ADM100-13 successfully created in directory
■    Object ADM100-14 successfully created in directory
■    Object ADM100-15 successfully created in directory
■    Object ADM100-16 successfully created in directory
```

Figure 212: Result Log in the SAP System

The logs use the **Application Log**, a tool of AS ABAP, to collect messages, exceptions, and errors (component *BC-SRV-BAL*). This tool provides a range of user-friendly functions for logs. You can archive old logs (with transaction SARA, archiving object BC_SBAL) or, if they are no longer required, delete them manually or automatically (with transaction SLG2).



ADM200
ADM515
ADM960
ADM_DE
ADM_EN
BECKER
CATON
DOWNUNDER
ESCH
GLOMBIK
HEEPMANN
HOSTING
J2EE_ADMIN
J2EE_GUEST
JAVATEMPL
KISSINGER
LEICHT
MAERKER
MENZEL
MONITOR
PILGRIM
SAPJSF
SAPTEC
SCHNAIBLE
SLDAPIUSER
SLDDSUSER
SPONDA
SWIEBOCKI
SYSTEMSETUP
TUBBESING
WALZ
WIENECKE

Figure 213: Users Created in the Directory Server

Which options exist for accessing entries in the directory server directly? One approach is the *Search* operation in transaction LDAP. In addition to the directory services vendors, there are also other LDAP browsers available from third-party vendors.

Outlook: Transferring HR Data to a Directory Service

In the scenario presented so far, we have assumed that the synchronized users have a user master record in the SAP system. SAP also supports the requirement that employee data that is stored in Human Resources (HR) in an SAP system is to be transferred to an LDAP-compatible directory server. Report *RPLDAP_EXTRACT* is available for this purpose. You can use this report to retrieve and prepare the HR data in the SAP system, and to transfer this data to a directory server. For more information, see SAP Note 784697.

Exercise 19: Data Exchange Between the SAP System and the Directory Service

Exercise Objectives

After completing this exercise, you will be able to:

- Generate and import the schema extensions proposed by SAP
- Configure the mapping and synchronization of SAP data fields to directory attributes
- Perform a synchronization between AS ABAP and the directory service
- Monitor the synchronization using logs

Business Example

You have made a connection between your ABAP-based SAP system and a directory service. Employee master data is maintained in the SAP system and is to be transferred to the directory service.

Your company is also intending to implement external software for fax connections. Every employee is to receive his or her own fax number using the external software. These numbers are to be transferred to the SAP system using the directory service.

Task 1: Schema Extension

Extension of the existing directory server schema with SAP-specific fields using an LDIF file.

⚠ **Caution:** This part of the task can be performed only **once per server**. Decide with your neighboring group which group is going to perform this part.

1. Log on to your SAP system using the SAP GUI **on your server**.
2. Use report *RSLDAPSCHEMAEXT* to generate a schema extension for the iPlanet Directory Server 5.0 and save it in a suitable location.
3. Restart your directory server and check that SAP-specific attributes are known on this server.

Result

Your directory server now knows SAP-specific data fields such as SAP user name, role, and authorization profile.

Continued on next page

Task 2: Configuring the Mapping and Synchronization

Mapping (of SAP data fields to directory attributes), purpose (what the mapping is used for), and synchronization direction (import or export)

Note: From this point, the tasks must be carried out by both groups (DEV and QAS).

1. Import the SAP proposal for mapping.
2. For the iPlanet Directory Server, *User ID* (abbreviation *uid*) is a required field. Map the SAP field *USERNAME/BAPIBNAME* to the “uid” attribute of the directory server. This field is to be used for export and is required for the directory server.
3. Define that all SAP data fields are to be exported.

Result

The assignment of SAP data fields to directory attributes is performed using default values. During a synchronization, all available user data is transferred from the AS ABAP to the directory server.

Task 3: Performing the Synchronization

Export data from the SAP system to the directory server; optional: also import data from the directory server to the SAP system. Evaluate all logs that have been created for LDAP operations.

Hint: A prerequisite for this exercise is that you are working in a client in the SAP system that is assigned to a logical system. This is relevant because when the assigned SAP roles are transferred, the logical system is stored together with this information in the directory server. If you have set up CUA beforehand, this prerequisite is fulfilled.

If this prerequisite is not fulfilled, use transaction SCC4 to assign the client in which you are working a logical system **`<SID>CLNT<client>`** (such as **`DEVCLNT100`**). If no logical systems are defined, you can define one in transaction BD54.

1. **Export.** First determine how many objects (in this case, *Dialog* users) would be affected by a synchronization.

Continued on next page

Then create these dialog users in the directory server. Check the entries with the iPlanet Console.

2. **Optional: Import.** Maintain the fax number using an (accepted) external tool.

   Change the synchronization rules so that in the future, the fax number is imported from the directory server to the SAP system. Then enter a new fax number in your directory server for user *ADM102-##* and synchronize it with the SAP system.

   > **Hint:** Enter the fax number in exactly the format that is expected from the mapping (in this case, exactly one hyphen separating the main number from the extension number).

   Note whether the user master record has changed in the SAP system. If you have Central User Administration (CUA) in operation, check the user master record in the child system.

3. Display the logs for all LDAP operations for today in the SAP system.

Solution 19: Data Exchange Between the SAP System and the Directory Service

Task 1: Schema Extension

Extension of the existing directory server schema with SAP-specific fields using an LDIF file.

Caution: This part of the task can be performed only **once per server**. Decide with your neighboring group which group is going to perform this part.

1. Log on to your SAP system using the SAP GUI **on your server**.

   a) If you have not already done so, log on to the Terminal Services Client (also known as the RDP Client) at operating system level of your SAP system using the user **`<sid>adm`**.

   b) Start SAP Logon on your server and log on to your SAP system with the user **`ADM102-##`**.

2. Use report *RSLDAPSCHEMAEXT* to generate a schema extension for the iPlanet Directory Server 5.0 and save it in a suitable location.

   a) Use transaction SA38 to execute report *RSLDAPSCHEMAEXT*. As the product, enter *iPlanet Directory Server 5.0* and choose *Execute*.

   b) Save the output as **`60sap-um.ldif`** in the directory *C:\iPlanet\Servers\slapd-Directory\config\schema* (as recommended) by choosing *Save As Local File*.

   c) Log off the SAP GUI. Use the SAP GUI in the training room for the other exercises.

Continued on next page

3. Restart your directory server and check that SAP-specific attributes are known on this server.

   a) In the iPlanet Console, choose *Restart Directory Server* for your directory server on the *Tasks* tab page. Confirm the confirmation prompt and wait for the success message (this will take about 5 seconds).

   b) In the iPlanet Console, switch to the *Configuration* tab page and choose *View* → *Refresh All*.

   c) Navigate to the point *Configuration* → *Schema* → *Attributes*. The attributes described by the schema extension are listed under *User Defined Attributes*.

Result

Your directory server now knows SAP-specific data fields such as SAP user name, role, and authorization profile.

Task 2: Configuring the Mapping and Synchronization

Mapping (of SAP data fields to directory attributes), purpose (what the mapping is used for), and synchronization direction (import or export)

Note: From this point, the tasks must be carried out by both groups (DEV and QAS).

1. Import the SAP proposal for mapping.

   a) In transaction LDAP, choose *LDAP Servers*.

   b) Switch to change mode and then double-click *Mapping*.

   c) Choose *Import Proposal* (or *Utilities* → *Import Proposal*). Leave all fields unchanged and return to the previous screen, where you can save the data (Customizing request).

      Hint: A proposal of this type is possible because you are using the generated schema extension and have selected a product supported in the SAP system.

Continued on next page

2. For the iPlanet Directory Server, *User ID* (abbreviation *uid*) is a required field. Map the SAP field *USERNAME/BAPIBNAME* to the “uid” attribute of the directory server. This field is to be used for export and is required for the directory server.

   a) In server maintenance in transaction LDAP, double-click *Mapping* (in change mode).

   b) Place the cursor in the third row (the new mapping procedure should be inserted here for the sake of clarity) and choose *Add New Mapping* (or *Edit → Add New Mapping* from the menu). On the mapping details screen, select the SAP field *USERNAME/BAPIBNAME* using F4 help and enter the directory attribute **`uid`**. Leave all other fields unchanged and return to the previous screen.

   c) For this new mapping procedure, set the *Export* indicator and then the *Required* indicator. Return to the previous screen and save your work.

3. Define that all SAP data fields are to be exported.

   a) In server maintenance in transaction LDAP, double-click *Synchronization* (in change mode). You can define for each field whether it is to be imported or exported.

   b) Select the *Export* field in all rows.

      Note: Within the table control, you can change the checkboxes with the space bar and move the cursor with the arrow keys.

   c) Return to the previous screen and save your work.

Result

The assignment of SAP data fields to directory attributes is performed using default values. During a synchronization, all available user data is transferred from the AS ABAP to the directory server.

Continued on next page

Task 3: Performing the Synchronization

Export data from the SAP system to the directory server; optional: also import data from the directory server to the SAP system. Evaluate all logs that have been created for LDAP operations.

Hint: A prerequisite for this exercise is that you are working in a client in the SAP system that is assigned to a logical system. This is relevant because when the assigned SAP roles are transferred, the logical system is stored together with this information in the directory server. If you have set up CUA beforehand, this prerequisite is fulfilled.

If this prerequisite is not fulfilled, use transaction SCC4 to assign the client in which you are working a logical system **`<SID>CLNT<client>`** (such as **`DEVCLNT100`**). If no logical systems are defined, you can define one in transaction BD54.

1. **Export.** First determine how many objects (in this case, *Dialog* users) would be affected by a synchronization.

Continued on next page

Then create these dialog users in the directory server. Check the entries with the iPlanet Console.

a) In transaction SA38, run report *RSLDAPSYNC_USER* with the following entries. Use the input help for the fields.

LDAP Server	*DIRECTORY##*
LDAP Connector	*LDAP_<Hostname>*
User Type	*A* (for dialog users)

Leave the remaining fields unchanged and choose *Execute*.

As your directory server does not yet contain any entries, only users that exist only in the database (in the SAP system) are displayed.

b) Now start report *RSLDAPSYNC_USER* with the entries you made above, but **create** *objects that exist only in the database* **in the Directory**. You can see in the log which users were created in the directory server.

Hint: Special users without a last name are not included during the synchronization (with the default values, the "Last Name" SAP field is mandatory).

c) In the iPlanet Console for your directory server, navigate to the *Directory* tab page in the tree *SAPtraining* → *<SID>team* → *employees*. The users that have just been created are displayed there.

2. **Optional: Import.** Maintain the fax number using an (accepted) external tool.

Change the synchronization rules so that in the future, the fax number is imported from the directory server to the SAP system. Then enter a new fax number in your directory server for user *ADM102-##* and synchronize it with the SAP system.

Hint: Enter the fax number in exactly the format that is expected from the mapping (in this case, exactly one hyphen separating the main number from the extension number).

Continued on next page

Note whether the user master record has changed in the SAP system. If you have Central User Administration (CUA) in operation, check the user master record in the child system.

a) In change mode on the *LDAP Servers* tab page in transaction LDAP, choose *Synchronization* and set the synchronization indicator for the attribute *facsimileTelephoneNumber* to *Import*. Return to the previous screen and save this change.

b) In the iPlanet Console for your directory server, navigate to the *Directory* tab page in the tree *SAPtraining* → *<SID>team* → *employees*. Double-click the entry *ADM102-##* and change the fax number. Confirm this with *OK*.

c) Start report **RSLDAPSYNC_USER** with the setting that the time stamp is to be compared for *Objects that exist both in the directory and in the database*.

   You should be able to see in the log that a synchronization was required for your *ADM102-##* user.

d) In transaction SU01, the new fax number for the *ADM102-##* user is displayed on the *Address* tab page.

   If you have CUA set up, you can also see the changed fax number in the child system (as long as the distribution parameter for the fax number is not set to *Local*).

3. Display the logs for all LDAP operations for today in the SAP system.

   a) Run transaction LDAPLOG with the default values.

Result

Congratulations on successfully connecting an AS ABAP to a directory server using LDAP.

Lesson Summary

You should now be able to:

- Exchange data between the AS ABAP and a directory server using the LDAP protocol

Related Information

- Online documentation for SAP NetWeaver 7.0: *SAP Library → SAP NetWeaver Library → SAP NetWeaver by Key Capability → Security → Identity Management → Identity Management of the Application Server ABAP → Configuration of Identity Management → Directory Services*
- Quick link */security → Security in Detail → Secure User Access → Identity Management* on SAP Service Marketplace
- SAP Note 793191: *FAQ: User master record synchronization with LDAP directories*
- SAP Note 618095: *Structure of LDAP search filter drg. user master synchroniz*
- SAP Note 888848: *Notes on schema enhancement with RSLDAPSCHEMAEXT*
- SAP Note 748490: *Using the user selection for directory searches*

Unit Summary

You should now be able to:

- Outline the concept of Central User Administration
- Evaluate the usefulness of Central User Administration for your company
- Define logical systems
- Correctly set up the required technical users for RFC connections
- Create the RFC connections required for CUA
- Configured the ALE communication required for CUA
- Set up Central User Administration
- Maintain and distribute user data using the CUA
- Perform simple troubleshooting
- Remove a CUA
- Explain the concept of user administration as applied to the integration of directory services
- Identify prerequisites for connecting an AS ABAP to a directory server
- Configure the AS ABAP for connection to a directory server
- Exchange data between the AS ABAP and a directory server using the LDAP protocol

Test Your Knowledge

1. Which of the following statements with regard to Central User Administration are correct?

   Choose the correct answer(s).

   □ A CUA can be implemented as of SAP R/3 4.0B.

   □ B You can connect only SAP R/3 or SAP ECC systems to CUA. SAP CRM systems cannot be connected.

   □ C CUA uses an ALE setup.

   □ D The clients linked by CUA are called logical systems.

   □ E The clients linked with the CUA are known as the “central system” and “child systems.”

2. Which of the following statements about logical systems are correct?

   Choose the correct answer(s).

   □ A Logical systems are defined cross-client.

   □ B The IDs of logical systems can be assigned to clients.

   □ C The naming convention for logical systems is LOGSYS###, where ### corresponds to a client ID.

   □ D Multiple logical system IDs can be assigned to a client.

   □ E In the context of a CUA, the clients involved are addressed as logical systems.

3. Which of the following statements about the required RFC connections in CUA are correct?

   Choose the correct answer(s).

   □ A The RFC connections must be specified entirely in uppercase letters and must be identical to the name of the addressed logical system.

   □ B The user in the RFC connection must be a dialog user.

   □ C An RFC connection from the child system to itself is required.

   □ D An RFC connection from the central system “to itself” is required.

   □ E Certain roles for the communication users are required only during CUA setup, and can then be removed from the relevant user master records.

4. With which methods is it possible to create users in child systems in the context of active CUA?

   Choose the correct answer(s).

   □ A In the child system with transaction SU01, *Systems* tab page

   □ B In the central system with transaction SU01, *Systems* tab page.

   □ C In the central system with transaction PFCG

   □ D In the child system with transaction PFCG

   □ E In the central system with transaction SU01, *Roles* tab page

5. Directory services can only be used as a repository for information about users.

   Determine whether this statement is true or false.

   □ True

   □ False

6. Which prerequisites must be fulfilled on the SAP side to connect an AS ABAP with a directory server?

   Choose the correct answer(s).

   □ A Host Name and Port Number of the directory server are maintained under Connection Data.

   □ B In user maintenance (transaction SU01), there is a user of type *System* that has read and write authorization for the directory server.

   □ C The operating system program *ldap_rfc* runs on the same host as the directory server.

   □ D The names of the RFC destination and the associated Connector definition are identical.

7. During a synchronization process, data can be simultaneously imported and exported for a user.

   Determine whether this statement is true or false.

   □ True

   □ False

Answers

1. Which of the following statements with regard to Central User Administration are correct?

   Answer: C, D, E

   The CUA clients linked by ALE are known as the “central system” and “child systems,” and as “logical systems.” You can connect SAP systems to CUA that have software component SAP_BASIS (version 45B or higher).

2. Which of the following statements about logical systems are correct?

   Answer: A, B, E

   The table that contains the IDs of the logical systems is a cross-client table. Each ID is assigned to exactly one client, and the recommendation for the naming convention is <SID>CLNT###. The clients involved in CUA require a unique assignment to a logical system.

3. Which of the following statements about the required RFC connections in CUA are correct?

   Answer: A, D, E

   The names assigned to the RFC connections used must be identical to the names of the logical systems, that is, also in uppercase letters. The user behind the RFC connection should be of the type *System*. The central system requires an RFC connection to itself. The communication users require a number of authorizations only to create the CUA; the corresponding roles can be removed from the user master records after the setting up of the CUA is completed.

4. With which methods is it possible to create users in child systems in the context of active CUA?

   Answer: B, E

   If you are using CUA, you can only create users in child systems using transaction SU01 in the central system. The creation of users is possible using the *Systems*, *Roles*, and *Profiles* tab pages.

5. Directory services can only be used as a repository for information about users.

   Answer: False

   Thanks to a flexible data model, data about any objects (such as hardware resources or documents) can be stored in directory services – although their use for user information is particularly helpful.

6. Which prerequisites must be fulfilled on the SAP side to connect an AS ABAP with a directory server?

   Answer: A, D

   In the SAP system, the RFC destination used and the LDAP Connector must be maintained with the same name (in uppercase letters). The operating system program *ldap_rfc* can then be started on the selected application server of the SAP system – while the directory server(s) addressed can run on remote servers (without *ldap_rfc*). Additional Customizing settings include the definition of a system user and the connection data, as well as the host name and port number of the directory server.

   The LDAP system user is stored in Customizing for the directory service connection and has nothing to do with the SAP users maintained in transaction SU01.

7. During a synchronization process, data can be simultaneously imported and exported for a user.

   Answer: True

   The synchronization indicators do not need to be identical for all mappings. It is therefore entirely possible that, for example, SAP-specific data (such as role assignments) is exported during a synchronization process, while other data (such as a fax number) is imported.

Course Summary

You should now be able to:

- perform basic tasks within the technology environment of SAP systems

Glossary

ALE

ALE stands for Application Link Enabling and describes a link between different software systems for exchanging data.

ALE integrated system union

A group of systems linked using ALE is described as an ALE integrated system.

Backup Domain Controller

An SAP System that can assume the functions of the transport domain controller if the transport domain controller fails.

central system

In the case of systems linked with ALE, you differentiate between the system in which the distribution model of the data is managed, and the linked systems. The managing system is called the **central system**, and the linked systems are called **child systems**.

Client

A client is a self-contained unit in an SAP system in commercial, organizational, and technical terms, with its own user master data and set of table key ranges.

client change options

The client change options define, which parts of the SAP system are modifiable when working in this client.

client compare

compare customizing settings of two clients via RFC

client copy

copying the content of a hole client or parts of it (according to the client copy profile used) from one client to another; the two clients can be in the same SAP system (local client copy), in different SAP systems (remote client copy) or the copy process can be performed by creating a transport request (client transport)

Client Copy Profiles

definition of which part of the client should be copied in a client copy or client transport process

client maintenance

maintaining customizing settings of one client according to the customizing settings of another client, based on a client compare via RFC

client transport

creating a transport request out of the content of a hole client or parts of it (according to the client copy profile used) and then importing this transport request into a client either of the same SAP system or of a different SAP system

client-specific data

Client-specific data, such as user master, application data and customizing settings, which affects only one client.

Consolidation Route

An SAP System that can assume the functions of the transport domain controller if the transport domain controller fails.

correction

change to the original of an object

cross-client data

Cross-client data, such as cross-client customizing data and all Repository objects, which affects the whole system environment.

CTO

Change and Transport Organizer. Part of the Change and Transport System (CTS). Provides functions for organizing software development projects. It is designed to support projects of all sizes, whether they are carried out centrally or in a distributed environment.

CTS

Change and Transport System. These are tools for creating and organizing transport requests, exporting, distributing and importing them. The CTS consists of the Change and Transport Organizer (CTO) and the Transport Management System (TMS)

current settings

customizing settings which can be performed directly in the productive system

Customizing

Customizing adapts the SAP software to meet a company's individual business requirements by setting up the business transactions the company requires in the system.

customizing transport request

transport requests that contains client specific data, such as client-specific customizing

Development System

The SAP system of an SAP system landscape in which development and customizing take place.

Directory Information Tree (DIT)

A global tree of entries in a directory that can be physically distributed across multiple hosts.

directory server

A directory server is an information system for particular information that is stored in a hierarchical structure and that can be accessed using a standardized protocol (such as LDAP).

Distinguished Name (DN)

Distinguished Name: Name of an entry in a directory server that is unique directory-wide (for example: *cn=LHeepmann, ou=instructors, o=sap, c=de*).

Domain link

A link between different transport domains. A domain link enables transparent access to all linked systems, display of import queues in all systems and configuration of transport routes between systems in different domains.

ECC

Enterprise Central Component. Formerly known as R/3. With release 5.0 R/3 was renamed to ECC. ECC is the main SAP component and, for example, the main component in SAP ERP.

Extended Transport Control

Extended Transport Control enables you to define client-specific transport routes (consolidation, delivery), transport target groups and the assignment of clients to transport layers.

External system

External systems are like virtual systems. This means that they are accessed using a communications system. However, for this type of system, a separate transport directory is also defined.

IDoc

An IDoc (intermediate document) is a data container with a particular format used for exchanging data between SAP systems, and also for exchanging data container data with non-SAP systems. These data packages are sent by RFC. IDocs are primarily used for exchanging data in an ALE integrated system.

Implementation Guide (IMG)

The IMG provides a hierarchical list with all possible customizing activities in an SAP system. This Customizing is grouped by modules and ordered in the sequence in which the individual activity has to be performed.

import monitor

monitor used for tracking the import of transport requests

import queue

The import queue displays the transport requests that are to be imported in a specific SAP system, in the order of their release.

LDAP

Lightweight Directory Access Protocol: The protocol for accessing directory servers (directory services).

LDAP Connector

An operating system program shipped by SAP, which implements the technical connection between an AS ABAP and a directory server.

local client copy

copying the content of a hole client or parts of it (according to the client copy profile used) from one client to another client in the same SAP system

mapping

Mapping describes how SAP fields are assigned to the attributes of a directory server.

modification

repair of an SAP object

Modification Assistant

The Modification Assistant helps to modify SAP standard delivered objects and simplify the modification adjustment process.

Note Browser

the Note Browser gives an overview of all SAP Notes available in an SAP system

package

Formerly known as development class. A package provides a logical grouping of Repository objects for coordination of development efforts, defines a Repository object's transport layer and can control naming objects

Productive System

The SAP system of an SAP system landscape in which the productive work of the end users takes place.

project, IMG project

Customizing can be organized in projects. The project then contains all the customizing activities, modifications and development. A project can be used for grouping transport requests.

QA Approval Procedure

The TMS Quality Assurance (QA) approval procedure increases the quality and the availability of the production systems by letting you check transport requests in the quality assurance system before they are delivered to subsequent systems.

Quality Assurance (QA) approval procedure

The TMS Quality Assurance (QA) approval procedure increases the quality and the availability of the production systems by letting check transport requests in the quality assurance system before delivering them to subsequent systems. Transport requests are only forwarded to the delivery systems if all the QA approval steps are processed for the request in the QA system and the request has been approved.

Quality Assurance System

The SAP system of an SAP system landscape in which integrated testing and quality assurance take place.

R3trans

program at operating system level to establish a connection to the database and to perform the necessary steps in the database for exports and imports

RDDIMPDP (import dispatcher)

job and program in the SAP system, that controls post import activities, such as table structure activation and conversion, generation and versioning

Relative Distinguished Name (RDN)

Relative Distinguished Name: Components of the Distinguished Name in a directory server that are unique compared to the superordinate node (for example: *cn=LHeepmann under ou=instructors, o=sap, c=de*).

remote client copy

copying the content of a hole client or parts of it (according to the client copy profile used) from one client to another client in a different SAP system via RFC

repair

change to a copy of an object

Repository

The collectivity of all programs, function modules, table structures, data elements, domains and so on that belong to an SAP system.

SAP ERP

SAP ERP is an SAP application. It contains several SAP components. The main component is the ECC (formerly known as R/3).

SAP Net Weaver

SAP NetWeaver is a set of different components like SAP NetWeaver AS, SAP BI, SAP NetWeaver Portal, SAP PI, SAP MDM and others. These technical components of SAP NetWeaver can be installed as needed to serve different requirements.

SAP NetWeaver AS

Technical platform and environment of most SAP components, for example R/3/ECC, CRM, BI, NetWeaver Portal, PI. Can be installed as SAP NetWeaver AS ABAP+Java, SAP NetWeaver AS ABAP or SAP NetWeaver AS Java, as needed.

SAP NetWeaver Development Infrastructure

SAP NetWeaver Development Infrastructureprovides an infrastructure for developing based applications on the SAP NetWeaver platform and is responsible for versioning, build and life cycle management of these applications.

SAP Note Assistant

tool for automatically applying SAP Notes concerning program changes in an SAP system

SAP Software Change Registration (SSCR)

process for tracking, which users are allowed to develop and which SAP objects are changed in an SAP system

SAP Solution Manger

The SAP Solution Manger is the portal for implementation, operation and continuous business improvement for the SAP applications.

SAP Support Package

kind of transport request, containing bug fixes and new/changed functionality for a software component of an SAP system

SAP system upgrade

procedure for exchanging the older Repository of an SAP system with newer Repository; this includes changing table structures at SAP system and database level, converting application data, exchanging the kernel executables

schema

A set of rules for a directory server, which defines the way in which entries are organized and which information is required or optional.

special transport workflow

The special transport workflow can be used, if transport corrections are needed urgently, or if transports are required that do not follow the defined transport routes. This process works with a workflow and each step is documented.

standard system types

The three standard system types are: systems in which development and Customizing activities are performed, systems that are used for quality assurance, and systems used in production operation. These system types are often known as “DEV,” “QAS,” or “PRD” systems in the SAP environment

Support Package Manager (SPAM)

tool in the SAP system for applying Support Packages

Synchronization

During a synchronization (using the LDAP protocol), entries in a directory are compared with those in the connected system and synchronized in accordance with previously defined rules.

system change options

The system change option defines whether or not Repository objects and client-independent customizing objects are globally modifiable.

System Landscape

Any required systems and clients, their meaning and the transport routes for implementation and maintenance processes. The system landscape may, for example, comprise a development system, a test system, and a production system. Those SAP systems that work together for developing/customizing, testing and productive work form a system landscape. These could be a development system, a quality assurance system and a productive system.

Task

Organizational part of a transport request.

TMS

Transport Management System. Part of the Change and Transport System (CTS). Transport Management System (TMS) organizes, monitors, and performs transports for all SAP systems within a system landscape. In addition, TMS is used to configure and manage the setting up of SAP systems and transport routes within a system landscape.

transport control program (tp)

tp is a program at operating system level that performs exports and imports for transporting objects between SAP systems. The program also ensures that the steps in exporting and importing objects are performed in the correct order.

Transport Directory

A directory that manages all data to be transported between SAP systems.

Transport Domain

All SAP Systems that are managed jointly by the TMS.

Transport Domain Controller

SAP System in which the transport route configuration is maintained centrally for all systems in the same transport domain.

Transport Group

All SAP Systems that access a common transport directory.

Transport Management System

Set of all tools in the SAP system for organizing, performing and monitoring transports between SAP systems. The TMS is part of the Change and Transport System.

Transport Organizer

see: Change and Transport Organizer (CTO)

Transport Organizer Tools

collection of tools that support the work with the change and transport system

transport request

A transport request is used for transferring different changes together into another SAP system.

Transport target group

Transport target groups can serve as transport targets from consolidations and deliveries for simultaneously servicing different SAP systems and/or clients.

workbench transport requests

transport request that contains client-independent data, such as repository objects or client-independent customizing

Index

Feedback

SAP AG has made every effort in the preparation of this course to ensure the accuracy and completeness of the materials. If you have any corrections or suggestions for improvement, please record them in the appropriate place in the course evaluation.