Administration Guide | PUBLIC
Document Version: 1.5 – 2023-12-01

Database Administration Using the DBA Cockpit: IBM Db2 for Linux, UNIX, and Windows

For SAP Systems Based on SAP NetWeaver as of 7.02 SP26, 7.03 SP33, 7.31 SP33, 7.4 SP30, 7.5 SP27, 7.51 SP16, and 7.52 SP12

THE BEST RUN SAP

Content

1 Introduction

Learn how you administer your Db2 database using the DBA Cockpit for SAP systems based on SAP NetWeaver as of:

- 7.02 SP26
- 7.03 SP33
- 7.31 SP33
- 7.4 SP30
- 7.5 SP27
- 7.51 SP16
- 7.52 SP12

For lower SAP NetWeaver releases than 7.0 Enhancement Package 2, refer to the following documentation:

7.0 SP12 and 7.1 or higher (also available in German here)

7.0 Enhancement Package 1 (also available in German here)

With the DBA Cockpit, you can monitor, control, and configure your database. The DBA Cockpit provides you with access to all the functions and indicators for monitoring and administration such as the following:

- Checking the system status and operation modes
- Locating potential problems as quickly as possible
- Early diagnosis of potential problems, for example, resource issues in the host or database system, which could affect the SAP system
- Analyzing and tuning the SAP system and the environment (host and database systems) to optimize the throughput of the SAP system
- Configuring the database

More Information

For more information about running an SAP system on IBM Db2 for Linux, UNIX, and Windows, see References [page 345] and, for Db2-specific information, the IBM Db2 documentation for your database version.

2 Naming Conventions

Throughout this document, the following naming conventions apply:

- IBM Db2 Version 11.5 for Linux, UNIX, and Windows is referred to as Db2 11.5.
- IBM Db2 Version 11.1 for Linux, UNIX, and Windows is referred to as Db2 11.1.
- IBM Db2 Version 10.5 for Linux, UNIX, and Windows is referred to as Db2 10.5.
- IBM Db2 Version 10.1 for Linux, UNIX, and Windows is referred to as Db2 10.1. (out of mainstream maintenance)
- IBM Db2 Version 9.7 for Linux, UNIX, and Windows is referred to as Db2 9.7. (out of mainstream maintenance)

SAP Business Warehouse was formerly known as SAP NetWeaver Business Warehouse. Throughout this document, the term “SAP Business Warehouse” and the abbreviation “SAP BW” are used. If you consult additional documentation for SAP Business Warehouse on SAP Help Portal, for example, you might also find the former term “SAP NetWeaver Business Warehouse”.

IBM changed its database name from IBM DB2 for Linux, UNIX, and Windows to simply IBM Db2 (with a lowercase 'b' now in Db2). In older SAP publications, you will still find the old product name, but in future documentation, we will use the new term, sometimes extended by 'for Linux, UNIX, and Windows' to avoid confusion with other products of the Db2 family, such as Db2 for z/OS or Db2 for i.

3 Document History

> ⚠ Caution
>
> Before you start the implementation, make sure you have the latest version of this document that you can find on SAP Help Portal as follows:
>
> https://help.sap.com/viewer/db6_dbacockpit (English)
>
> https://help.sap.com/viewer/db6_dbacockpit_de (German)

The following table provides an overview of the most important document changes:

Version	Date	Description
1.5	2023-12-01	• SAP systems running on Db2 versions 9.7 and 10.1 are out of mainstream maintenance. • The screen Diagnostics: Lock-Wait Events [page 306] is now also available in SAP GUI. • New attribute "HADR" in System Attributes in the Check Environment [page 217] (on the *Configuration: Parameter Check* screen) Link added in Setting Up the DBA Cockpit for Remote Monitoring [page 34] • Information including SAP Note 3331129 added in Diagnostics: Displaying the Audit Log [page 295]
1.4	2022-05-06	Enhancements such as the following: • Support of EXPLAIN-from-Cache function on the screens Performance: SQL Cache [page 103], Performance: Top SQL Statements [page 99] and Performance: Applications [page 93] • EXPLAIN-from-Activity function for Workload Management: Threshold Violations [page 337] • EXPLAIN function: Extension for collecting additional Db2 monitoring and statistical information from `db2caem` (SAP GUI only) • EXPLAIN function: You can now download the access plan output of `db2exfmt` (SAP GUI only) • Support of Db2 Advanced Log Space Management (ALSM) on the Performance: Transaction Log [page 105] screen.
1.31	2020-11-27	Minor corrections New section Configuration: Db2 Fix Pack Level Check [page 219]

Version	Date	Description
1.3	2020-01-14	Minor updates, for example, BW Object Checks [page 325] was added under BW Administration [page 324]. References [page 345] were updated.
1.2	2018-11-28	Minor updates of, for example, Introduction [page 9], BW Administration [page 324], and References [page 345]
1.1	2017-11-23	This document has moved from SAP Service Marketplace to SAP Help Portal where it is now available in HTML and PDF format. Old SAP Service Marketplace links were replaced by links to SAP Help Portal.
1.0	2016-07-18	Initial version

4 Getting to Know the DBA Cockpit

The DBA Cockpit is a platform-independent tool that you can use to monitor and administer your database. The advantage of using the DBA Cockpit is that it has been specifically designed for the administration and monitoring of databases in an SAP system landscape. Therefore, the functions of the DBA Cockpit especially support database administrators in adapting their databases for the workload of SAP systems. The DBA Cockpit eases the work of database administrators because all important performance tuning, monitoring, and administration tasks are available in a single transaction.

Central Monitoring of the Databases in an SAP System Landscape

The DBA Cockpit is part of SAP NetWeaver ABAP systems and integrated into SAP Solution Manager. You can run the DBA Cockpit as part of your system administration activities in SAP Solution Manager. The DBA Cockpit is optimized for handling administration and monitoring the databases of your entire system landscape from a central system. In particular, you can use the DBA Cockpit to handle configuration of databases centrally. You can administer and monitor remote databases from the DBA Cockpit using remote database connections. For more information, see The DBA Cockpit on a Local System and on SAP Solution Manager [page 14].

Administration and Monitoring Functions for IBM DB2 for Linux, UNIX, and Windows

- **Performance monitoring**
  You can, for example, display performance and workload statistics, analyze top SQL statements, perform time spent analyses, and view performance information about database objects.
- **Space monitoring and administration**
  Watch the space consumption of your database including database objects such as tables, indexes, tablespaces, or containers. You can analyze space allocation and perform administration activities to change the storage layout of your database.
- **Backup and recovery overview**
  Get an overview of all performed database backups. In addition, you can display information about archived log files and about logging parameters.
- **Database configuration**
  Get an overview of your database configuration. You can also change the database configuration, including configuration settings for buffer pools, database partitions, the database optimizer, or the data collection framework (see also Data Collection Framework: Configuration [page 243]).
- **Job scheduling**
  Direct access to the DBA Planning Calendar, the DBA log, scheduled DB2 tasks, and other tools allows you to plan and keep track of all important jobs for the databases.
- **Alert monitoring**

You can view alerts in the alert monitor and enable notifications for violations of database alert thresholds.

- **Diagnostics**
  Various diagnostic functions allow you to identify critical situations in your database, such as lock-wait events or missing tables and indexes. The diagnostics tools also include, for example, the audit log, the dump directory, and the trace status.
- **Business Warehouse (BW) administration**
  If you have an SAP BW system in your system landscape, you can use the DBA Cockpit to call up the MDC advisor, a DB2 tool that helps to improve the settings for multi-dimensional clustering in partitioned databases. In addition, you can use the DBA Cockpit to perform BW health checks, for example, or perform administration activities relating to near-line storage.
- **Workload management**
  Workload management lets you distinguish and prioritize different types of work on the database. You use service classes to monitor and control resource consumption of the different workloads. In addition, the information provided on the screens in the *Workload Management* task area allows you to define thresholds on resources to capture information or even cancel database activities that violate a threshold.

Availability of Functions in the DBA Cockpit

The availability of functions in the DBA Cockpit depends on various conditions, such as the following:

- Database release level of the monitored system
- Release level of the monitored SAP system
- Application type to which the database is assigned, for example, ABAP OLTP, BW, or Java, and so on.

  > **Example**
  >
  > All BW-specific functions of the DBA Cockpit require additional BW functionality in the back end.

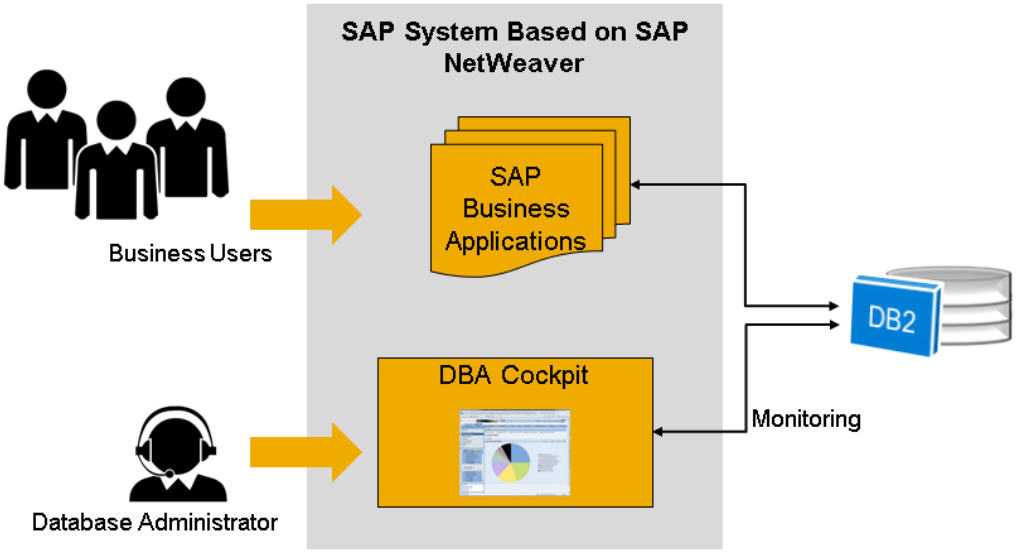
- Connections that are assigned to the system, such as the availability of RFC connections
- User permissions, which are SAP-specific and also database-specific
  For more information, see Authorizations for the DBA Cockpit and Remote Monitoring [page 30].
- Database configuration, such as monitoring settings or automatic maintenance settings
- Back-end configuration
  For more information, see Enablement of Databases for the Data Collection Framework [page 48].

Depending on the conditions that the currently selected system fulfills, functions might either be disabled or not visible in the DBA Cockpit. In addition, some screens might show a reduced amount of information and specific functions might be disabled (for example, administrative functions are unavailable if you do not have the required permissions).

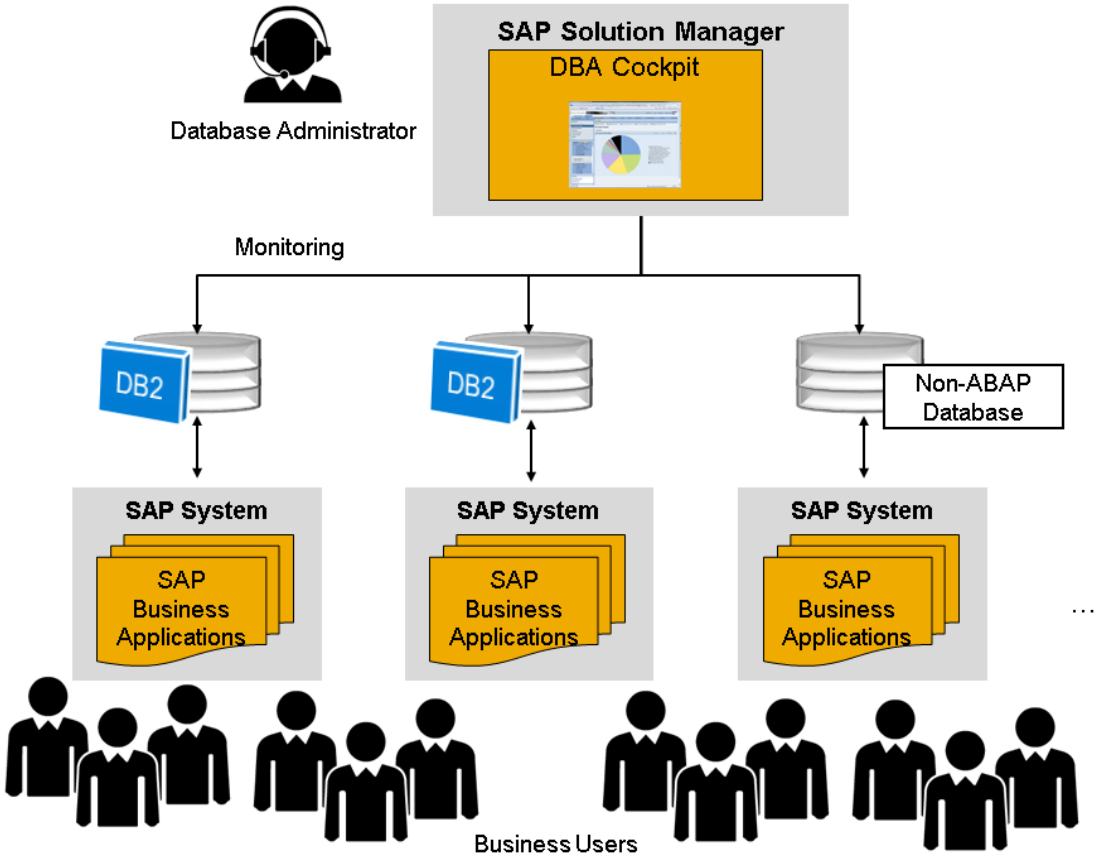
4.1 The DBA Cockpit on a Local System and on SAP Solution Manager

The DBA Cockpit is part of every SAP NetWeaver-based ABAP system. You can run the DBA Cockpit locally on an SAP NetWeaver-based system by calling the `DBACOCKPIT` transaction. Alternatively, you can run the DBA

Cockpit on your SAP Solution Manager system, where you can access all databases in your system landscape using remote connections. If you use the DBA Cockpit as part of the SAP Solution Manager system, this allows you to update and administrate all databases from a central system rather than logging on to each individual system separately. Using the DBA Cockpit centrally also allows you to administrate non-ABAP databases and ABAP-based systems with older versions of the DBA Cockpit.



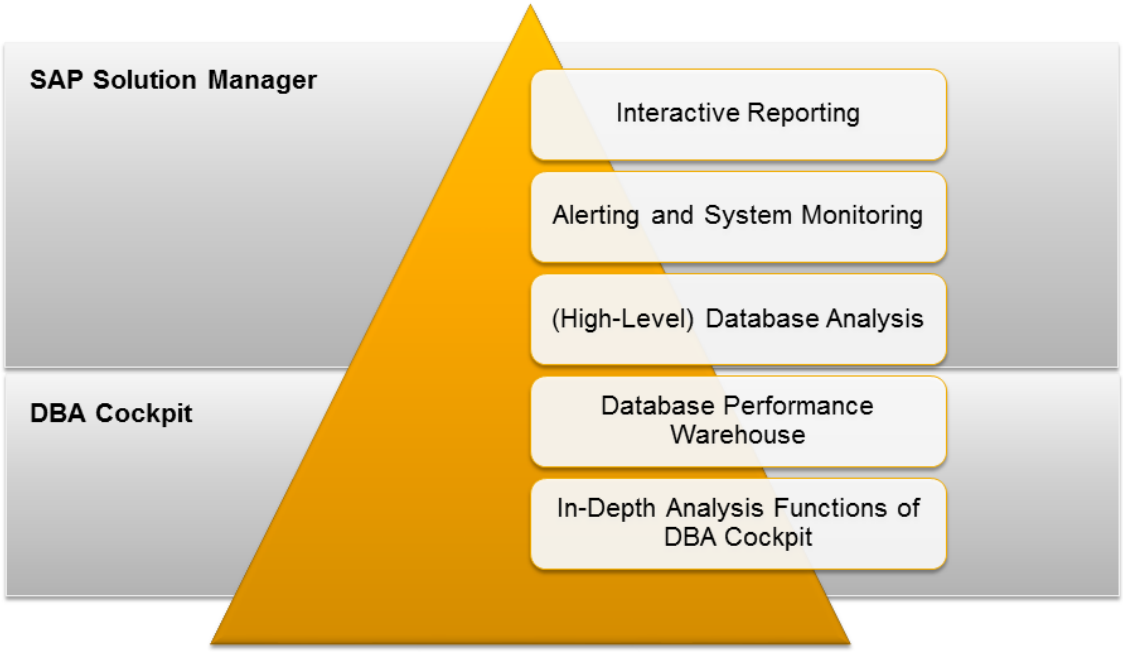
DBA Cockpit on a Local SAP System



DBA Cockpit on an SAP Solution Manager System

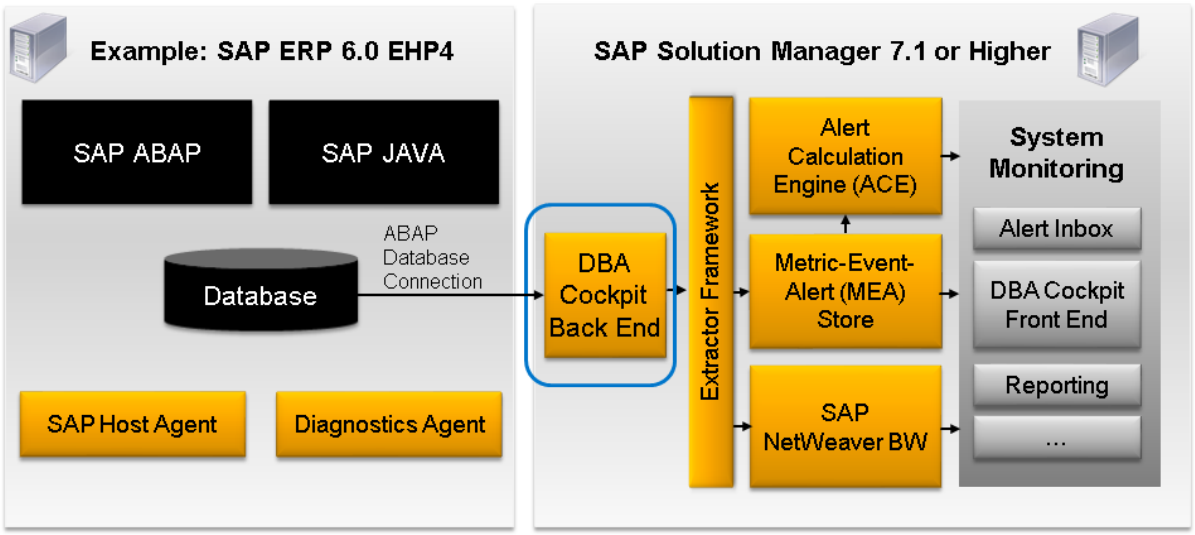
SAP Solution Manager is the free SAP toolset in your system landscape to monitor the full stack of an SAP system: from the operating system up to the business process. SAP Solution Manager is typically installed on a separate system, where it provides central access to tools, methods, and preconfigured contents that you can use during the evaluation, implementation, and operation of your systems. For database administrators, SAP Solution Manager offers a range of tools for root cause analysis, alerting, and reporting.

The tools of the DBA Cockpit complement the available SAP Solution Manager tools, which provide high-level overviews of possible database issues, including reporting and alerting functions. In addition to the high-level overviews of SAP Solution Manager, the DBA Cockpit serves as an in-depth analysis tool for database-related issues.



Available Database Monitoring in SAP Solution Manager and the DBA Cockpit

The DBA Cockpit not only provides more tools for the expert database administrator, but it also plays an important part in the SAP Solution Manager infrastructure. The remote database monitoring infrastructure of the DBA Cockpit is used by SAP Solution Manager to extract metrics from remote databases. These metrics are then passed on to the different applications in SAP Solution Manager.



The DBA Cockpit as Metrics Provider for SAP Solution Manager

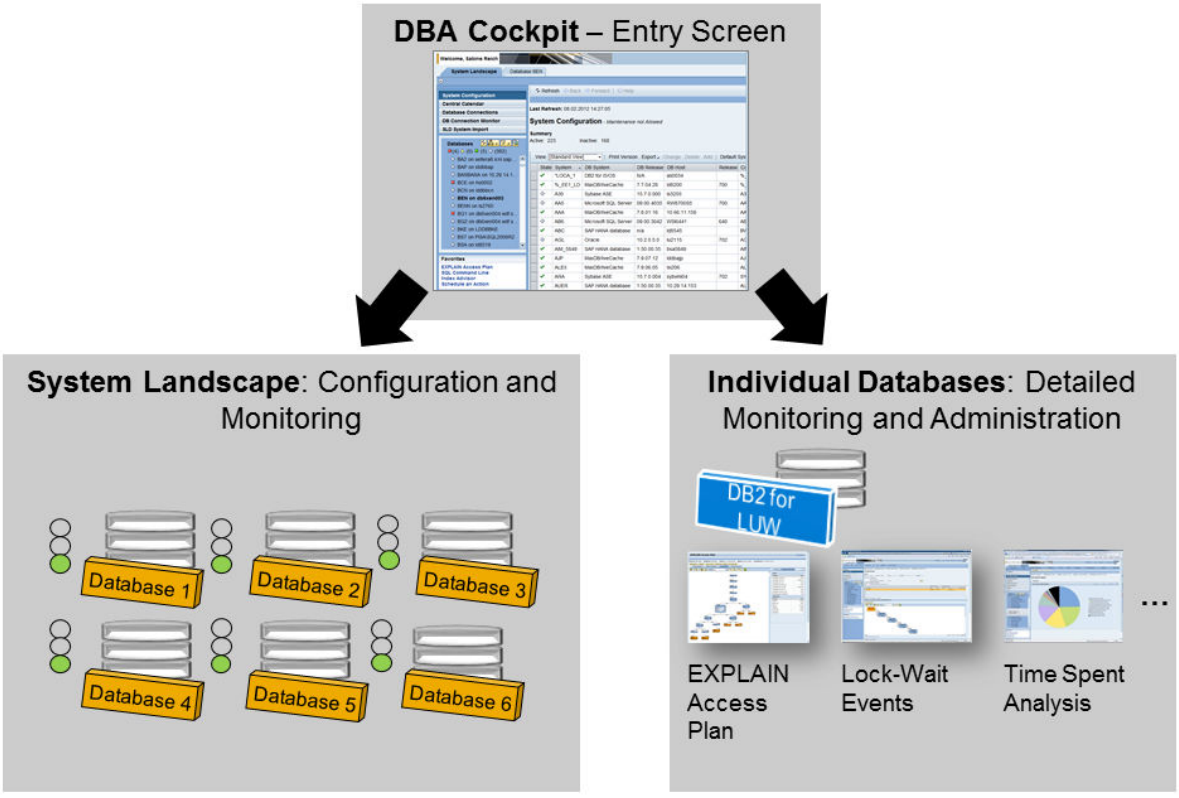
4.2 Entry Screen of the DBA Cockpit

In the DBA Cockpit, you can monitor the databases of your entire system landscape. The DBA Cockpit provides the following functions:

- Overview screens that allow you to check the status of your system landscape at one glance
- Database-specific screens that allow you to identify and analyze issues of individual databases in detail

When you start the DBA Cockpit, the system displays the *System Configuration* screen. On this screen, you can decide to proceed as follows:

- Configure and monitor your entire system landscape. To do this, you stay on the entry screen of the DBA Cockpit (this screen is the *System Landscape* tab page in the Web browser-based user interface). For more information, see System Landscape Monitoring and Configuration [page 57].
- Monitor and administer individual databases in detail in your system landscape. To do this, you choose the *Database <Your Database>* tab page (in the Web browser-based user interface) or the *System...* pushbutton (in the SAP GUI-based user interface).



DBA Cockpit: Entry Screen

4.3 Screen Layout and Navigation in the DBA Cockpit

The DBA Cockpit works with the following user interfaces:

- SAP GUI (recommended)
- A Web browser-based user interface (using the Web Dynpro UI technology)

By default, transaction `DBACOCKPIT` starts SAP GUI. You can switch to the Web browser-based user interface from SAP GUI or set the Web browser-based UI as default.

> **i** Note
>
> We recommend that you use SAP GUI as your default user interface. The Web browser-based version of the DBA Cockpit has been deprecated.

Most functions of the DBA Cockpit are available for both the Web browser-based and the SAP GUI-based user interface unless stated otherwise. If there are differences between the Web browser-based user interface and the SAP GUI, or if a feature is available for only one of the UIs, it's indicated in the documentation. Note that many procedures in this documentation describe how you navigate on the Web browser-based UI, which is slightly different from the SAP GUI navigation.

More Information

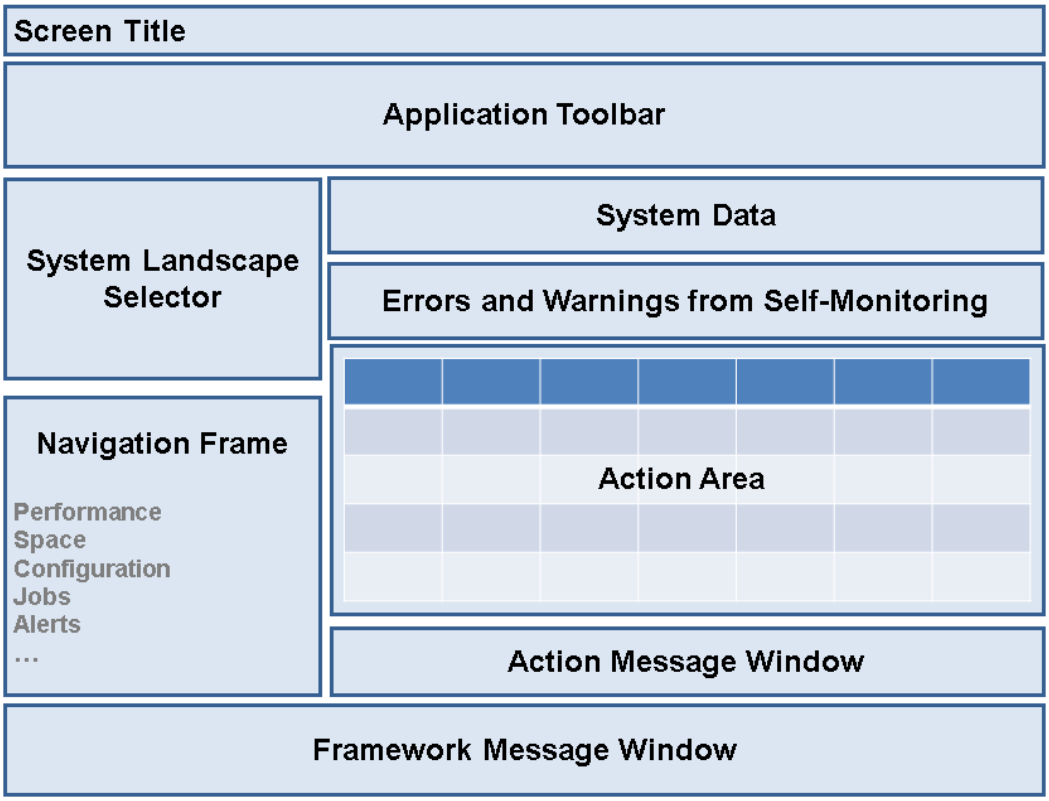
SAP GUI-Based User Interface [page 20]

Web Browser-Based User Interface (Web Dynpro) [page 23]

Switching Between the Web Browser-Based User Interface and SAP GUI [page 28]

4.3.1 SAP GUI-Based User Interface

SAP GUI is the recommended user interface for the DBA Cockpit. The screens of the DBA Cockpit using the SAP GUI are divided into the following areas:



Navigation and Screen Layout of the SAP GUI-Based User Interface

Application Toolbar

Provides basic functions, for example, displaying or hiding the areas on the left side by using the *Full Screen On/Off* pushbutton.

System Landscape Selector

Lets you choose the system to monitor. Additional information about a distributed database system is displayed if available.

Navigation Frame

Displays the main task areas of database administration in a tree structure, for example, performance monitoring, space management, and job scheduling. Within each task area, there is a set of related screens that you can choose and that correspond to the screens of the task areas on the *Database* tab page of the Web Dynpro user interface.

In addition, there is the *System Landscape* task area that provides access to all system landscape-relevant actions such as system configuration, database connection, or self-monitoring. This task area corresponds to the *System Landscape* tab page on the Web Dynpro UI.

System Data

This area is common to most actions. It provides, for example, the time of the last refresh, the database startup time, and the name of the database server.

Errors and Warnings from Self-Monitoring

If there are any issues with the infrastructure of the DBA Cockpit itself, the relevant message is displayed.

For more information, see Self-Monitoring of the DBA Cockpit [page 64].

Action Area

Displays the details of the currently selected action.

Action Message Window

Only appears with certain actions and displays additional information that is related to the selected action.

Framework Message Window

Displays the message window provided by the DBA Cockpit. It contains a complete history of all messages sent during the session.

In addition, you can:

- Clean up the window by choosing the *Clear Message Window* pushbutton.

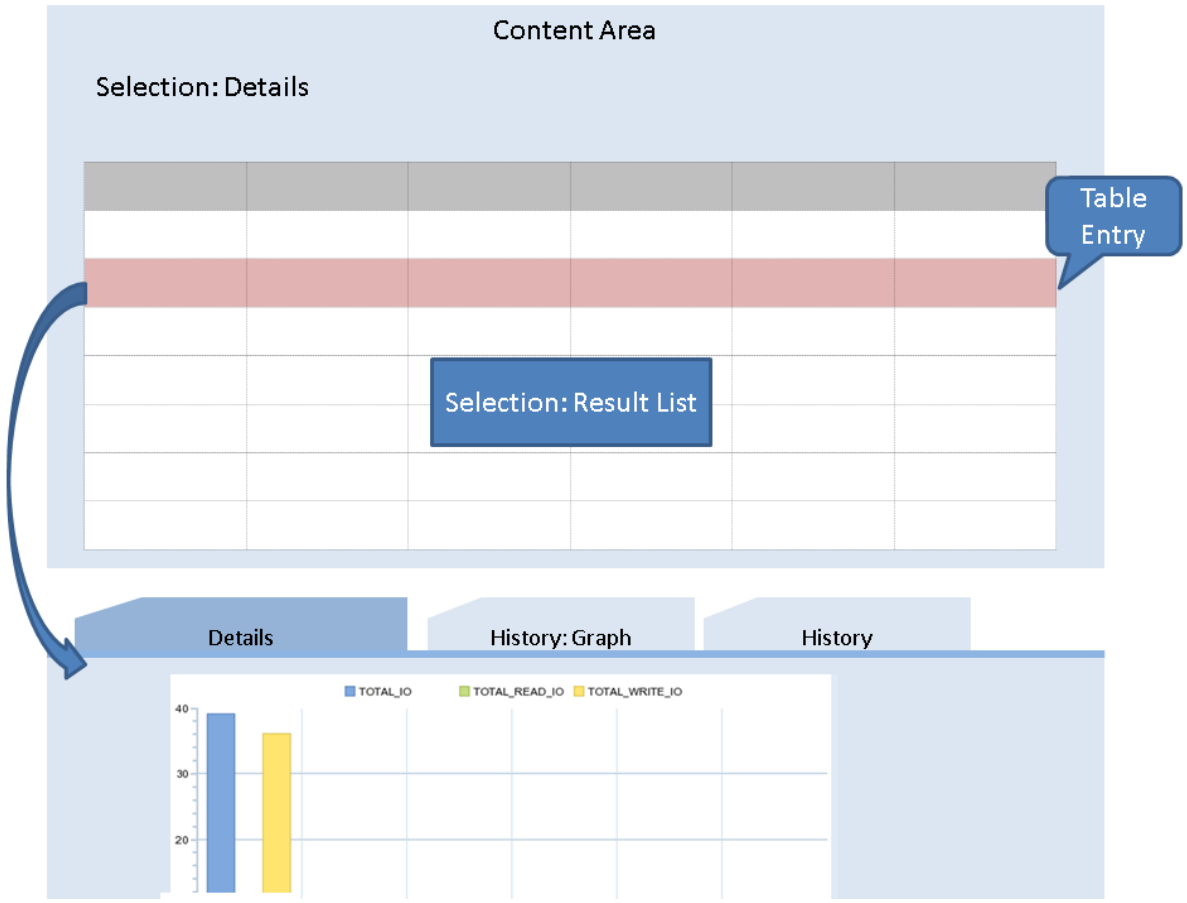
- Collapse or expand the window by choosing either the *Minimize Message Window* or the *Show Message Window* pushbutton.
- Check if a long text for a message is available by double-clicking the message or by choosing *Show Long Text*.
- Display previously shown messages by choosing the *Show History* pushbutton.
- Print the message text by choosing the *Print Version* pushbutton.

> **i** Note
>
> Changes to the screen area sizes are user-specific and are restored when you next start the DBA Cockpit.

4.3.2 Displaying Detailed Metrics, History, and History Graph

If the content in the content area is displayed as a table, you can usually display more details of this table entry by double-clicking or selecting a table row. Depending on the task area, the following is displayed in the content detail area:

- Current details of the table entry (*Details* tab page)
- History data of the table entry (*History* tab page)
- Graphical display of history data (*History: Graph* tab page)



Detail Area: Layout

The graphical view of the history complements the detail view of history data. The graphical view of the history is based on the time series that you can see in the history, which contains all data in table format.

> **i** Note
>
> You can switch off the display of the graphical view of history data using the *Personalize* option in the common header area of the DBA Cockpit screens.

4.3.3 Web Browser-Based User Interface (Web Dynpro)

> **i** Note
>
> The Web browser-based version of the DBA Cockpit is still available, but has been deprecated. We recommend that you use the DBA Cockpit with SAP GUI as your default user interface.

The Web browser-based user interface (also referred to as Web Dynpro UI) differs from the SAP GUI-based user interface with regard to the overall screen layout, navigation, and customizing of the user interface.

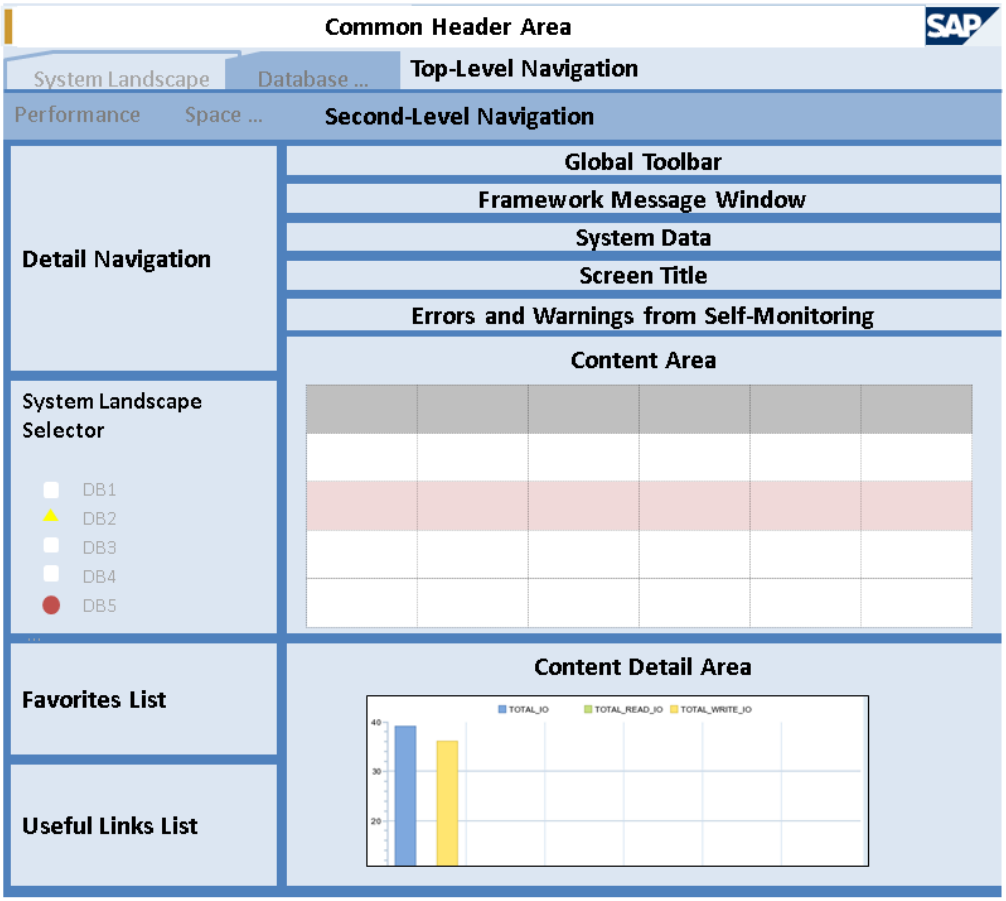
Prerequisites

In contrast to the standard SAP GUI, the Web browser-based UI additionally requires the configuration of the `DBA_COCKPIT` Web service. For more information, see **SAP Note** 1245200.

If you also want to display graphics, the Internet Graphics Server (IGS) is required, which is part of every SAP NetWeaver Application Server (AS) installation. For more information about the IGS, see SAP Note 458731 and the SAP Library for SAP NetWeaver on SAP Help Portal at http://help.sap.com/nw. Search for the keywords **`Internet Graphics Service`** and **`Installing and Updating the IGS`**.

Layout of the Web Browser-Based User Interface

The typical DBA Cockpit screen with the Web browser-based user interface is divided into areas as displayed in the following figure:



Navigation and Screen Layout of the Web Browser-Based User Interface

Area	Description
Common header area	Provides a standard set of functions, for example, to log off from the DBA Cockpit or to customize the layout.
Top level navigation including second-level navigation	In the top level navigation, you can switch between the following areas: • Cross-system area on the *System Landscape* tab page Provides information about the overall system landscape • Database-specific area on the *Database* tab page Provides information about the selected database In the second-level navigation, the main task areas of database administration are provided, for example, performance monitoring, space management, and job scheduling. For fast navigation, these main task areas provide pull-down menus corresponding to the related detail levels. You can hide the areas *Detail Navigation*, *System Landscape Selector*, and *Favorites* by choosing the *Expand or Collapse Launchpad* pushbutton on the left side of the top level navigation area. If the launchpad is collapsed, the second-level navigation part provides a simplified system selection field with an F4 help. Your chosen screen layout is stored in the user settings and restored at the next start of the DBA Cockpit.
Detail navigation	Displays the screen titles of the main task areas. **Example** If you choose *Performance* in the second-level navigation, the related screen titles are displayed in the detail navigation on the left, such as *Time Spent Analysis* and others.
System landscape selector	Provides a quick overview of all configured systems, including an overall status for each system (indicated by a traffic light icon). Depending on which alerting infrastructure you use, the overall status of each system is aggregated from the alert statuses coming from SAP Solution Manager or from the Computing Center Management System (CCMS). The system status in the system landscape selector is calculated as follows: • Alerts from SAP Solution Manager: The DBA Cockpit takes over the worst result of the main alert categories in SAP Solution Manager. For example, if the alert category *Performance* has the status *Red* in SAP Solution Manager, the system in the system landscape selector gets the status *Red*, too. • Alerts from the CCMS: The DBA Cockpit takes over the overall alert status of the system from the CCMS. For more information about customizing the system landscape selector, see Customizing the System Landscape Selector [page 27]. For more information about alerts in the DBA Cockpit, in the CCMS, and in SAP Solution Manager, see Alerts [page 282].

Area	Description
Favorites list	Provides quick access to important tools and actions. By default, the *Favorites* list contains a link to the *Schedule an Action* wizard, which cannot be removed. You can extend the *Favorites* list, that is, add and organize favorites, by choosing the *Personalize* drop-down menu icon in the common header area.
Useful links	Using the links in this area, you can go directly to documentation for SAP on Db2 for Linux, UNIX, and Windows.
Framework message window	Displays messages that are provided by the DBA Cockpit. This window contains a complete history of all messages that are sent during the session. In addition, you can: • Collapse or expand the window by choosing *Expand Message Window* or *Collapse Message Window*. • Check if a long text for a message is available by double-clicking the message or by choosing *Details*. **i Note** By default, the message window is collapsed. When a new message is generated, it is automatically expanded.
Errors and warnings from self-monitoring	If there are any problems with the infrastructure of the DBA Cockpit itself, the relevant errors and warnings are displayed. For more information, see Self-Monitoring of the DBA Cockpit [page 64].
Global toolbar	Provides a set of globally available functions for navigation and content-related functions like *Refresh*.
System data	This area is common to most actions providing, for example, the time of the last refresh, the startup time, and the name of the database server.
Content area	Displays details of the currently selected action. Depending on the chosen action, the content area is divided into the following areas: • A status area that provides information about limited functions if any technical prerequisites are not met. For more information, see *Diagnostics* > *Self-Monitoring* • The *Selection* area where you can enter selection criteria for the content to be displayed • The *Summary* area that provides summary views of data, for example, totals or execution times • The content, which depends on the screen and action you have chosen You can refresh the content by choosing the *Refresh* pushbutton in the global toolbar or by changing the selection criteria and then choosing the *Apply Selection* pushbutton in the *Selection* area.

Area	Description
Content detail area	Only appears with certain actions and displays additional information that is related to the selected action of the main content area. For more information, see Displaying Detailed Metrics, History, and History Graph [page 22].

4.3.3.1 Customizing the System Landscape Selector

By default, all systems are displayed without any grouping or filtering. For each configured system, the alert status, the name of the system and its database host is displayed. The following menu buttons are available for the list of systems:

- *Refresh System Landscape*
  You can refresh the information about the available systems in the list.
- *Group Systems by Selected Criteria*
  You can customize the displayed list of systems by grouping them according to the selected criteria:
  - *No Group*
  - *Database Platform*
  - *Name*
  - *Custom*
  - *Alerts*

  To use a custom grouping, you must first define and add a custom group to the list. To do so, choose *Add Group* from the pop-up menu of the menu button *Group Systems by Selected Criteria*. Specify a name for the custom group and assign the systems of your choice.
  As soon as you have added a custom group, the option *Organize Groups* becomes available in the pop-up menu of the menu button *Group Systems by Selected Criteria*, which lets you maintain an already existing group.
- *Filter Systems by Selected Criteria*
  You can filter the list of available systems to show only those systems that match the filter criteria. You filter, for example, by the alert status of the systems.
- *Search Systems*
  Provides an input field where you can search for a specific system in the list.

4.3.3.2 Displaying and Changing the Settings for Traffic Lights in the Dashboard

Context

> **i** Note
>
> Traffic lights in the dashboard are only available in the Web browser-based version of the DBA Cockpit.

If the Monitoring and Alerting Infrastructure (MAI) from SAP Solution Manager is enabled for the database, some metrics on the dashboard are rated with a traffic light based on settings in the MAI. You can display or change the thresholds for these traffic lights in SAP Solution Manager.

Procedure

1. In SAP Solution Manager, call up transaction `SOLMAN_SETUP`.
2. Choose *Technical Monitoring* > *Template Maintenance*.
3. Choose the DB2 template that is relevant for your database version. If you have created your own templates, choose the relevant template.
4. Choose the expert mode.
5. Choose the *Data Collection* tab.

   You can now display the thresholds for the traffic lights.
6. If you want to change the thresholds, create your own template, and change the thresholds according to your needs.

4.3.4 Switching Between the Web Browser-Based User Interface and SAP GUI

Background

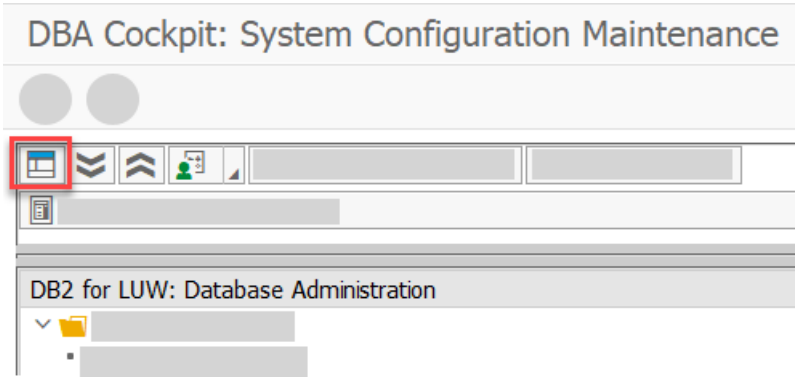
For a lot of your tasks you are free to choose the user interface. However, some functions are only available for the Web browser-based UI, others for SAP GUI only. Therefore, it might be necessary to switch between the UIs. You must use SAP GUI, for example, if in a heterogeneous system landscape, the database to be monitored does not fully support the Web browser-based user interface.

> **i** Note
>
> We recommend that you use the DBA Cockpit with SAP GUI as your default user interface. The Web browser-based version of the DBA Cockpit is still available, but has been deprecated.

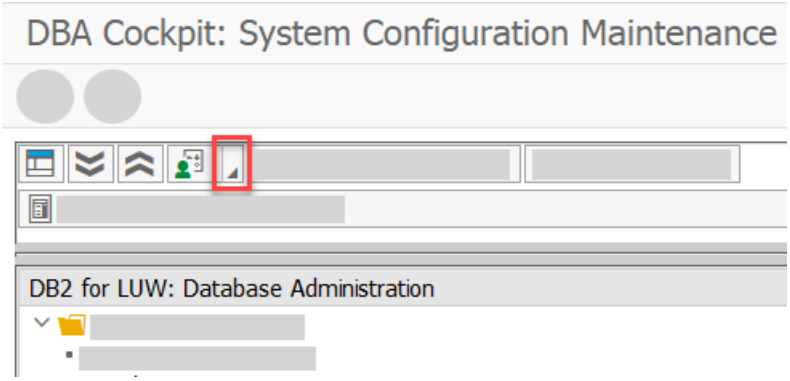
Procedure

To switch the user interface, perform the following steps:

1. In your SAP system, enter transaction `DBACOCKPIT`.
   Depending on your settings, the DBA Cockpit either starts with the Web browser-based user interface or with the SAP GUI.
2. Switch between the two user interfaces by choosing the *Switch to SAPGUI* button or the *Switch to Web Dynpro* button:



If you want the SAP GUI to be the default UI to start with, choose the option *Use SAP GUI as default UI* in the dropdown menu of the *Personal Settings* button in the SAP GUI. To set the Web Dynpro UI as default, choose the option *Use Web Dynpro as default UI*:



4.4 Locking and Auditing of Actions in the DBA Cockpit

The DBA Cockpit provides a set of actions to monitor and to maintain the database.

Locking of Maintenance Actions

For each maintenance action that you have selected using the DBA Cockpit, a lock is set for the system that is being monitored. All locks are released when you exit the DBA Cockpit or when you change to another system.

Auditing of Maintenance Actions

When you make changes that affect the system configuration of the DBA Cockpit or that affect database objects such as database configuration parameters or tablespaces, an audit log entry is written. You can display the audit log in the DBA Cockpit. Automatic changes like automatic updates for the data collection framework are also audited.

> **i** Note
>
> The audit log in the DBA Cockpit is **not** a certified tool for auditing.

Related Information

Diagnostics: Displaying the Audit Log [page 295]

4.5 Authorizations for the DBA Cockpit and Remote Monitoring

Use

The DBA Cockpit provides a set of actions to monitor and to maintain databases. To be able to perform these actions, certain authorizations in the SAP system are required. An SAP user must have the global authorization and the appropriate system-specific permission. For example, to administrate a system, the user must have global `S_RZL_ADM` authorization for maintenance and the system-specific authorization for maintenance. The following sections provide information about how global and system-specific authorizations are checked and what you need to do to gain the required authorizations.

Authorization Checks for the DBA Cockpit and for Monitored Systems

Global Authorization Check

When you start the DBA Cockpit or change to another system in the DBA Cockpit, an authorization check is performed.

You can enable or disable the database maintenance in general using the profile parameter `dbs/dba/ccms_maintenance`. If this profile parameter is not set in the instance profile, the default value *1* is used.

Depending on the setting of profile parameter `dbs/dba/ccms_maintenance`, the following authorization checks exist:

- If the profile parameter is set to *0*, SAP users cannot perform any maintenance actions, regardless of their personal permissions.

- If the profile parameter is set to *1*, SAP users can perform maintenance actions depending on their personal permission for the authorization object `S_RZL_ADM`. The attribute `ACTVT` of this authorization object defines whether a user may maintain or only monitor objects.

> **i Note**
>
> In older versions of the DBA Cockpit, the profile parameter was called `dbs/db6/ccms_maintenance`. The profile parameter has been renamed to `dbs/dba/ccms_maintenance` because it is relevant for all databases monitored by the DBA Cockpit. The two parameters can be used interchangeably. If you have already used `dbs/db6/ccms_maintenance`, you can continue using it.

System-Specific Authorization Check

In addition to the permissions that are globally granted, you can restrict access to specific systems that were configured in the DBA Cockpit. You enable or disable the system-specific permission checks using the profile parameter `dbs/dba/ccms_security_level`.

If this profile parameter is **not** set in the instance profile, the default value `0` is used. Depending on the setting of profile parameter `dbs/dba/ccms_security_level`, the following authorization checks are performed when you select a system in the DBA Cockpit:

- If parameter `dbs/dba/ccms_security_level` is set to `0`, no additional system-specific check is performed.
- If parameter `dbs/dba/ccms_security_level` is set to `1`, SAP system users can perform actions depending on their personal permission for the authorization object `S_DBCON`.
  The attributes `DBA_DBHOST`, `DBA_DBSID`, and `DBA_DBUSER` must match the corresponding attributes for the database connection that was assigned to the selected system. The special value `<LOCAL SYSTEM>` for the attribute `DBA_DBSID` is used to identify the local system itself.
  The attribute `ACTVT` of this `S_DBCON` authorization object defines the level of permitted actions and can have the following values:

Value	Description
03 Display	Enables read access to all screens of the DBA Cockpit except to those that only have a maintenance mode and no read-only mode
23 Maintain	Enables read and maintenance access to all screens of the DBA Cockpit except to those that require extended maintenance permissions
36 Extended maintenance	Enables read and maintenance access to all screens of the DBA Cockpit including special maintenance screens **i Note** The only screen for which extended maintenance permission is required is the *SQL Command Line* screen that you can access in the *Favorites* list of the DBA Cockpit.

You can grant authorizations for using the DBA Cockpit with the following roles:

- `SAP_BC_S_DBCON_USER`
  Read-only role that allows monitoring access to all systems configured within the DBA Cockpit

- `SAP_BC_S_DBCON_ADMIN`
  Additionally grants administration rights to the user for all systems. This role does **not** include the value *Extended maintenance*.

> **i** Note
>
> Make sure that you have maintained the authorizations for your DBA user and for all background users that either run jobs of the DBA Planning Calendar or the SAP standard jobs `SAP_COLLECTOR_FOR_PERFMONITOR` and `SAP_CCMS_MONI_BATCH_DP`. For the DBA Planning Calendar, check that you have authorization for DBA and background job scheduling, which is provided by profiles `S_RZL_ADMIN` and `S_BTCH_ALL`.

Database Authorizations

In addition to DBA Cockpit authorizations, the DBA user must also have sufficient database authorizations:

- Systems that are monitored via remote database connections use the user that is specified for the database connections. You can freely choose this user, but make sure that this user has sufficient authorizations so that monitoring and administration activities that are triggered in the DBA Cockpit can be performed on the database.
  - For remote systems running on IBM DB2 for Linux, UNIX, and Windows, you have full administrative authorizations if you use the DB2 instance owner `db2<dbsid>` for the administration connection.
  - If you want to connect to remote systems running on any other database platform, see the appropriate DBA Cockpit documentation for that database platform.
- Local systems use a special administration connection. This connection is called `+++DB6ADM` and is automatically generated. When you start the DBA Cockpit and no user has been assigned to the administration connection, you are asked for the user credentials.
  If you do not supply the correct user credentials, a standard connection with the SAP connect user is used instead of the administration connection. In this case all administrative actions of the DBA Cockpit are disabled. You can change the user and password for the administrative connection as described in Configuring Database Connections [page 45], which is mandatory for background tasks that require administrative permissions.

> → Recommendation
>
> You can also use the `SAPMON` and `SAPTOOLS` roles predefined by SAP to assign monitoring or administration authorizations to your database administrators. If the database user specified for the administrative database connection has an additional `SAPAPP` role, this user might be able to retrieve application data by using the *Diagnostics* > *SQLCommand Line* function of the DBA Cockpit.
>
> To protect business data in that scenario, you can perform the following steps:
>
> 1. Do **not** use the SAP standard connect user, but use a specific user with a dedicated `SAPTOOLS` or `SAPMON` role instead, for example, the DB2 instance owner.
> 2. Exclude this administrative user from access to application data.

For more information about the SAPMON, SAPAPP, and SAPTOOLS roles, see the *Database Administration Guide for SAP on IBM Db2 for Linux, UNIX, and Windows* at https://help.sap.com/viewer/db6_admin on SAP Help Portal.

5 Configuration and Management of the System Landscape

5.1 Setting Up the DBA Cockpit for Remote Monitoring

Use

To be able to use the DBA Cockpit to monitor remote systems, you have to configure a connection to those systems in the DBA Cockpit. If the DBA Cockpit is used as part of Solution Manager Diagnostics, the DBA Cockpit setup is part of the SAP Solution Manager setup. This means that when you integrate systems into the SAP Solution Manager landscape, the related databases are configured in the DBA Cockpit and no DBA Cockpit-specific setup is required anymore. For more information, see SAP Note 1267189 and Setting Up Database Monitoring and Alerting in SAP Solution Manager [page 37].

Alternatively, you can configure your database systems without using SAP Solution Manager, for example, by manually creating database connections and system configuration entries. For more information, see Setting Up Remote Monitoring Manually [page 38].

> → Recommendation
>
> We recommend that you use **SAP Solution Manager** for the setup and monitoring of your system landscape.

Implementation Considerations

To monitor a system remotely, you use the following methods:

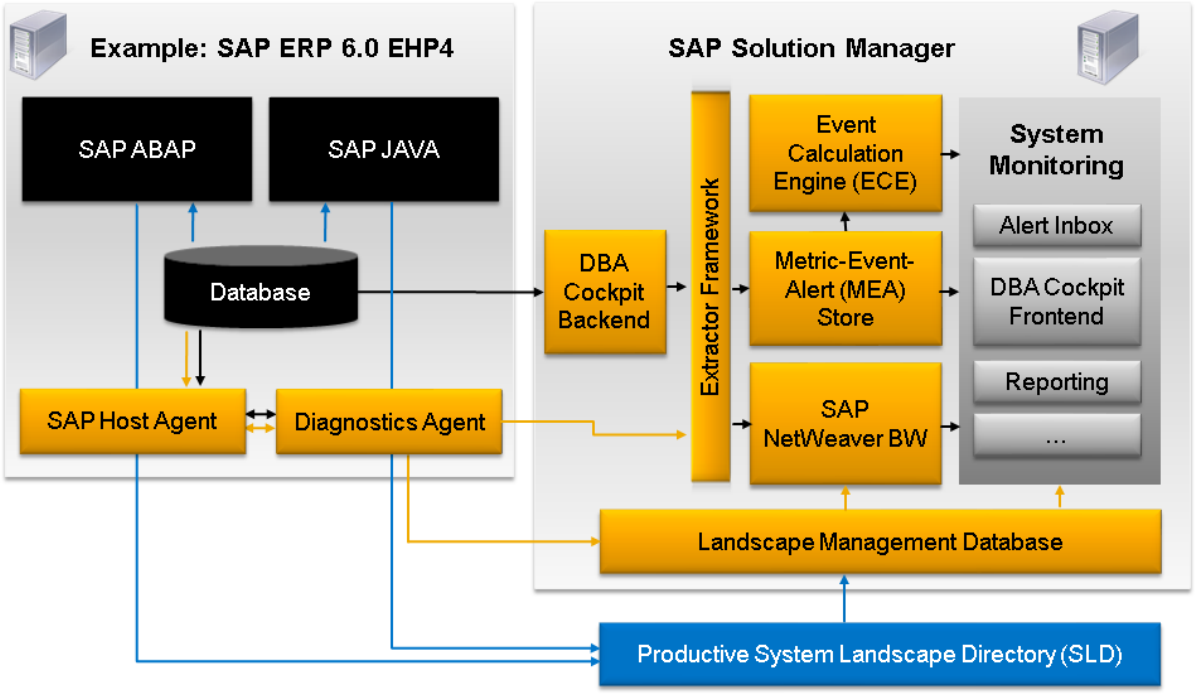
- Remote database connections **(mandatory)**
  This method uses additional connections. It is the main access method for monitoring and administration tasks and it is mandatory. You can specify remote connections for any database and maintain the connections using the DBA Cockpit.
- RFC connection **(additional option for SAP ABAP systems only)**
  For this method, you have to assign an RFC connection to your system. RFC connections are available for SAP ABAP systems only. You can use RFC connections as an optional access path for ABAP-related monitoring functions, for example, for the consistency check of the ABAP Dictionary. This means that the DBA Cockpit uses the RFC connection in parallel to the database connection for the same system.

  > i Note
  >
  > You can maintain RFC connections **only** with transaction SM59, **not** with the DBA Cockpit.

5.1.1 Architecture Overview: End-to-End Monitoring and Alerting in SAP Solution Manager and DBA Cockpit

Use

The architecture of SAP Solution Manager comprises a number of tools and frameworks that collect detail data about databases and their statuses automatically. The tools and frameworks for the automatic discovery of databases and database details include the landscape management database (LMDB), the Diagnostics Agent, and the extractor framework:



Data Collection and Database Monitoring and Alerting with SAP Solution Manager

Landscape Management Database (LMDB)

In SAP Solution Manager, all elements of a system landscape are modeled in the Landscape Management Database (LMDB). The core task of the LMDB is to provide information about the entire system landscape at a central location. The LMDB copies available data from the system landscape directory (SLD), but it enriches the data copied from SLD with additional information. For example, the SLD can provide the information which databases are available. For each database, the LMDB adds details such as the cluster topology of the database, including information about the use of pureScale or the database partitioning feature for IBM DB2 for Linux, UNIX, and Windows. All SAP Solution Manager applications are based on LMDB.

Diagnostics Agent

The Solution Manager Diagnostics Agent (Diagnostics Agent), which is installed on every local system of your system landscape, gathers information from the managed systems and reports the information to the SAP Solution Manager system. This includes information about the availability of the database and its host.

The Diagnostics Agent uses SAP Host Agent to discover all databases installed on a host. The Diagnostics Agent and the SAP Host Agent transfer some basic information, such as the database host, the database type and its name, to the SLD. In addition, the Diagnostics Agent reports technical attributes like cluster topology, database release, and properties required for a remote database connection to the LMDB. This kind of information is mapped in the LMDB with the data provided by the SLD. Information about databases that do not run with an SAP system are also pushed to the LMDB by the Diagnostics Agent.

DBA Cockpit Backend, Extractor Framework, and Data Flow

The DBA Cockpit backend performs the data collection via a remote database connection and calculates the metrics of the monitored databases, such as performance KPIs, space, or buffer pool quality. The DBA Cockpit takes snapshots of database statuses, and thus is able to generate historical data (time series), which allows database administrators to keep track of short- and mid-term developments in the database. The data collected by the DBA Cockpit backend is also fed into the extractor framework, the central infrastructure for SAP Solution Manager for data collection and distribution.

The extracted data is reused by the following engines and stores:

- Event calculation engine (ECE)
- Metric-event-alert (MEA) store
- SAP BW for reporting
- Data volume management (DVM)

These engines and stores further process the raw data taken from the databases and generate alerts and metrics for BW reporting, which are then passed on to the different SAP Solution Manager applications, such as the alert inbox, reporting functions and also the BW reporting screens in the DBA Cockpit.

Predefined Alerting Templates Shipped by SAP

During the SAP Solution Manager setup, you can find predefined monitoring and alerting templates that contain the definitions or values of metrics, events, and alerts (MEA). Depending on the defined threshold values, incidents and notifications are then automatically triggered. For IBM DB2 for LUW databases, predefined metrics, events, and alerts are available for the following task areas:

- Database availability
- Database exceptions, such as locks, missing backups, or error messages
- Database performance issues, such as bad buffer quality or problems with disk space usage

> **i** Note
>
> The availability of the end-to-end monitoring and alerting infrastructure in SAP Solution Manager depends on your support license.

5.1.2 Setting Up Database Monitoring and Alerting in SAP Solution Manager

Procedure

To set up the end-to-end monitoring and alerting infrastructure, you use the standard transactions for setting up SAP Solution Manager. You need to perform the following steps:

1. **Install the Diagnostics Agent on your database systems.**
   The Diagnostics Agent gathers information from the managed systems and reports them to the SAP Solution Manager system. This also includes information about the availability of the database and its host. You need to install a Diagnostics Agent on each server (virtual hostname) that you want to monitor. You can use the SAP installation tool to install the Diagnostics Agent.
2. **Configure data suppliers in the system landscape directory (SLD).**
   Use the transaction *System Landscape Directory: Local Administration* (`RZ70`) to configure SLD data suppliers in the system landscape. Here, you need to provide the port and the host of the SLD. The SLD data suppliers are programs that collect the database attributes *Database Host*, *Database Type* and *Database Name* at defined periods. Once the SLD data suppliers are up and running, they push database attributes from the database to the SLD.
3. **Connect the Diagnostics Agent to SAP Solution Manager.**
   Use the transaction *SAP Solution Manager Configuration (SOLMAN_SETUP)* > *System Preparation* to connect the SAP Diagnostics Agent to SAP Solution Manager.
4. **Configure the systems managed by SAP Solution Manager.**
   Use the transaction *SAP Solution Manager Configuration (SOLMAN_SETUP)* > *Managed System Configuration* to perform the following activities:
   - Connect the SAP Diagnostics Agent to the managed databases
   - Specify system parameters required to configure the managed databases
     Most of these parameters, such as hosts or ports, are automatically detected by the Diagnostics Agent, so only a password is required here.
   - Trigger automatic configuration activities
     This includes the setup of database extractors, which collect data for monitoring the databases in SAP Solution Manager.
5. **Set up technical monitoring, including alerting.**
   Use the transaction *SAP Solution Manager Configuration (SOLMAN_SETUP)* > *Technical Monitoring* to set up the technical monitoring of databases in SAP Solution Manager. You set up the following:
   - Activation or deactivation of auto-notifications (for example, e-mail notifications) about database alerts
     As a default, auto-notification is activated.
   - Recipients and recipient lists of auto-notifications

- Assignment of monitoring templates to selected systems in scope for monitoring

Monitoring templates contain the definitions or values of metrics, events, and alerts (MEA) that trigger incidents and notifications. The SAP templates have predefined settings, but you can also adapt the templates to your customer-specific needs.

More Information

- **Complete setup of SAP Solution Manager**: Documentation for SAP Solution Manager on SAP Help Portal at http://help.sap.com/solutionmanager
- **Additional database-specific setup steps**: SAP Note 1027146
- **Overall architecture**: Architecture Overview: End-to-End Monitoring and Alerting in SAP Solution Manager and DBA Cockpit [page 35]

5.1.3 Setting Up Remote Monitoring Manually

If you do **not** use SAP Solution Manager for monitoring, you can also set up the DBA Cockpit and databases for remote monitoring as follows:

- By manually creating database connections and system configuration entries in the DBA Cockpit
  This is relevant, for example, if you want to monitor a database using the DBA Cockpit and without configuring the database for monitoring in SAP Solution Manager. For more information, see Configuring Systems for Remote Monitoring Manually [page 39].
- By using database information that is stored in the system landscape directory (SLD) for automatic generation and update of system entries in the DBA Cockpit
  This option is deprecated and only available in the DBA Cockpit if you do **not** use the DBA Cockpit as part of an SAP Solution Manager system. For more information, see Configuring Systems for Remote Monitoring Using the System Landscape Directory [page 43].

> → Recommendation
>
> We recommend that you use SAP Solution Manager for the setup and monitoring of your system landscape (see Setting Up Database Monitoring and Alerting in SAP Solution Manager [page 37]).

5.1.3.1 Configuring Systems for Remote Monitoring Manually

Prerequisites

- The systems that you want to monitor must have a database release that is compatible with the database release of your local database.
- The user for the database connection must have sufficient database permissions. For more information, see Authorizations for the DBA Cockpit and Remote Monitoring [page 30].

Context

You use this procedure to configure systems that you want to monitor using remote database connections. A manual system configuration is only necessary if the monitored database has not been configured during the integration of a system using SAP Solution Manager.

The configuration does not include the setup of the monitoring infrastructure, but you perform only a basic setup that is necessary to connect to the monitored database. For subsequent configuration steps, see Enablement of Databases for the Data Collection Framework [page 48].

> **i Note**
>
> Depending on the database platform of the selected system, some options might not be available. In this case, you cannot enter any data in the corresponding fields.

Procedure

1. Call the DBA Cockpit.

   The *System Configuration* screen appears displaying a list of all available systems with a *Configuration Status* icon that indicates the current system status.

   > **i Note**
   >
   > When you start the DBA Cockpit for the first time, the local system is automatically added to the list of all available systems. At least one system entry is displayed.

2. Choose the *Add* button.

   The wizard *Integrate a System* appears.

The following table lists the steps and recommended actions:

Step Name	Action
Introduction	Provides an overview of the configuration steps No action to be taken

System Characteristics	Specify the following:
	• Name of the system that you want to monitor
	
	i Note
	This name is a unique ID and does not have to be the SAP system ID. You can choose any name **except** the SAP system ID of the **local system** that is reserved for the local system entry.
	• Description of the monitored system
	• Connection type
	Choose between the following connection types:
	  • Remote Database Connection
	  • Remote Database Connection and RFC Destination
	  • Remote Database Connection via RFC Destination

RFC Destination (**Optional** step that is only necessary if, in the previous step, you have chosen a connection type that requires an RFC destination.)	Specify the name of the RFC destination to be used **i Note** The specified RFC destination must already exist and be available. You can test the destination by choosing the *Test Connection* button. If the connection is not working, you can only proceed if you select the *Ignore Communication Errors* checkbox.

Step Name	Action
Database Connection	Specify the following: • Name of the database connection This is a unique name that you can freely choose. Alternatively, you can search for an already existing connection name using the related search help. In this case, confirm the selected name using the Enter key to prefill the connection attributes. • Database platform of the monitored database • Connection maximum that limits the parallel use of this connection by the kernel • Connection optimum that sets the optimum number of open connections → Recommendation You should set this value to *0* to ensure that the kernel closes connections if they are no longer used. • User name and password for the user that is used for monitoring • List of connection parameters that are required to identify the database

Step Name	Action
Monitoring Settings	Specify how you want to collect monitoring data: • If you want to use the Computing Center Management System (CCMS) and its alert monitor (transaction `RZ20`), select the *Activate Alert Monitoring* checkbox. **i Note** If you are using the DBA Cockpit in SAP Solution Manager as of release 7.1, do not select this option. Instead, use the end-to-end alerting and monitoring infrastructure (MAI) of SAP Solution Manager, which requires no specific setup in the DBA Cockpit. For more information, see Alerts [page 282]. • If the monitored database fulfills the requirements for using the data collection framework (DCF), a checkbox *Data Collection Framework (DCF) is Enabled* is displayed. This checkbox is read-only because the DCF is set up automatically. • If the monitoring data is to be collected by the remote system, select the *Data Collection by Remote System* checkbox. • If data for the Central Planning Calendar is to be provided, select the *Show Scheduled Jobs in Central Planning Calendar* checkbox. • Deprecated option: If data about the performance or the size of database objects is to be collected, select the checkbox *Enable Deprecated Data Collection Additionally (Not Recommended)*. **i Note** This option has been deprecated and should not be used anymore. Instead, proceed as described in Setting Up the Data Collection Framework (DCF) Manually [page 51]. Depending on the selected database platform, only a subset of options might be available.
Summary	Summarizes all actions to be performed To save your entries, choose the *Execute* button.

Step Name	Action
Execution Protocol	Summarizes all performed actions including error messages To exit the wizard, choose the *Finish* button and return to the *System Configuration* screen.

3. Optional:

   If you want to change an existing configuration entry, select the system entry in the overview list and choose the *Change* button.

   In the dialog box *Change System Configuration Entry*, enter and save your changes.
4. Optional:

   If you want to delete a configuration entry, select the system entry in the overview list, choose the *Delete* button and confirm the deletion.

5.1.3.2 Configuring Systems for Remote Monitoring Using the System Landscape Directory (SLD)

Context

The system landscape directory (SLD) contains data from all database systems that are available in your system landscape. You can use this data to set up the system configuration in the DBA Cockpit instead of setting it up manually or using SAP Solution Manager.

When you set up the DBA Cockpit for the first time, you use this procedure to import the appropriate data from the SLD. During production operation, you use the procedure to synchronize the data between the SLD and the DBA Cockpit periodically.

> **i** Note
>
> This function has been deprecated. We recommend that you use **SAP Solution Manager** for the setup and monitoring of your system landscape.

Procedure

1. To import database connection data from the SLD, call the DBA Cockpit.
2. On the *System Landscape* tab page, choose *SLD System Import*.

The *SLD System Import* screen appears. Depending on the system landscape, one or more of the following nodes are displayed:

- *New Database Systems in the SLD*
  All database systems that are registered in the SLD and that so far have been unknown to the DBA Cockpit are displayed.
- *Changed Systems From Earlier SLD Imports*
  All database systems for which the main data differs between the SLD and the DBA Cockpit are displayed.
- *Systems No Longer Registered in the SLD*
  All systems that were originally imported from the SLD into the DBA Cockpit but that are no longer registered in the SLD are displayed.
- *Systems Identical in the SLD and in the DBA Cockpit*
  All systems that are registered in the SLD and that are identical in the DBA Cockpit are displayed.
- *Unsupported Database Systems in the SLD*
  All database systems that are registered in the SLD but not supported by the DBA Cockpit are displayed.

> **i Note**
>
> Each database system is described as follows:
>
> ```
> <Name (system ID) of the database system> on <main database host>
> ( <database platform> )
> ```

3. To import database system data, choose the *Change* pushbutton.

   The actions allowed for each database system are displayed in the second column of the tree.

4. Select the actions that you want to execute for the selected database systems and choose the *Import* pushbutton. By default, only the import of new database systems is selected.

   The selected actions are executed. A short message for each executed action is displayed in the content detail area.

> **i Note**
>
> Connection data that is retrieved from the SLD might not be complete for one of the following reasons:
>
> - Depending on the data provided by a system to the SLD, some connection data can be incomplete.
> - User or password data is not available via SLD.
>
> When you establish the connection to an imported system for the first time, the DBA Cockpit checks the completeness of the configured system. This means that you are prompted for user, password, and connection information, if necessary. If additional connection information is required, enter the required data as described in Configuring Database Connections [page 45].

5.1.3.3 Configuring Database Connections

Use

This section describes how you set and maintain technical attributes for remote database connections on the *Database Connections* screen in the DBA Cockpit. Remote database connections are required by the DBA Cockpit itself for administration and monitoring or by application programs that use secondary connections to external databases.

Note, however, that new connections for database monitoring are created during the system configuration (see Setting Up Database Monitoring and Alerting in SAP Solution Manager [page 37] and Configuring Systems for Remote Monitoring Manually [page 39]). Therefore, such connections do not have to be created on the *Database Connections* screen.

You can also use the *Database Connections* screen to set up database connections that are used for non-monitoring components, for example, for the access to external data resources by BW.

Procedure

Calling Up the Database Connections Screen

Call the DBA Cockpit, and on the *System Landscape* tab page, choose *Database Connections*.

The *Database Connections* screen appears displaying a list of all available database connection definitions grouped by database platform:

Field	Description
Connection Name	Name of the database connection **i Note** This is a unique name that you can freely choose.
Database Name	Name of the database
Database Host	Name of the database host
DB Schema	Name of the database schema to be monitored
User Name	Name of the connect user
Permanent	Specifies whether the connect user must be permanently available
Max. Connections	Maximum allowed number of open connections
Opt. Connections	Optimal number of connections

Field	Description
SSL Encryption	Shows if the connection is SSL-encrypted (*Enabled*) or not (*Disabled*)

By default, the database connections that are defined in the local system are displayed.

Displaying a Database Connection

1. To display a database connection, expand the respective database platform in the *Remote Database Connection* column, select a connection, and choose the *Display* button.
2. For more information about the data shown, see the documentation under *Adding a Database Connection*.

Adding a Database Connection

1. To add a database connection, choose the *Add* button.
   The *Add Connection Entry* dialog box appears.
2. In the *Add Connection Entry* dialog box, enter the following:
   - In the *Connection Name* field, specify the name of the connection.

     > **i Note**
     >
     > This name is a unique ID that you can choose freely except for names that are reserved by SAP for generated connections. These names can be, for example, administrator connections or connections that are used by systems from the system landscape directory (SLD).

   - In the *Database System* field, select the name of the database platform from the dropdown list.
   - In the *Connection Maximum* field, enter an appropriate value. This value limits the number of database connections that are currently held by the SAP system. The SAP system does not let you exceed this limit.
   - In the *Connection Optimum* field, enter an appropriate value. This value is a more flexible limit that can be exceeded.
   - If you want the connection to be mandatory for the SAP system, select the *Permanent* checkbox. This parameter defines the availability of the database connection. It is then handled in the same way as the local default connection, that is, only if this database connection is available for a work process, the work process of the SAP system can run.

     > **⚠ Caution**
     >
     > Set this parameter only if the connection is absolutely required to run your SAP system.

   - In the *User Name* field, enter the name of the connect user. Make sure that you choose a user with the appropriate authorizations. For more information, see Authorizations for the DBA Cockpit and Remote Monitoring [page 30].
   - In the *Password* field, enter a password for the connect user and confirm it in the appropriate field.
   - In the parameter table, specify the following additional database-specific parameter values:

Attribute	Description
Database Host	Name of the remote database server

Attribute	Description
Database Name	Name of the database
Port Number	Enter the port number or the service name. The service name is the parameter `SVCENAME` of the database manager configuration (DBM) of the remote database. When you enter the service name, the entry is resolved as port number during runtime.
Schema Name	Name of the schema that is used as default schema when connecting to the database **i Note** If you omit this field, the name of the database connect user is used as schema.
Alternative Database Host	Parameter for client reroute if the original host does not answer
Alternative Host Number	Parameter for client reroute if the original host does not answer
SSL Server Certificate	If you are using SSL to encrypt your database connection, fill out this field. Enter the fully qualified name of the `.arm` file of the SSL server certificate.

3. To save your entries, choose the *Add* button.

Changing a Database Connection

1. To change a database connection, expand the respective database platform in the *Remote Database Connection* column, select a connection, and choose the *Change* button.
   The *Change Connection Entry* dialog box appears.
2. Enter your changes in the appropriate fields and save them.

Changing the User Name or Password of a Database Connection

1. To change the user name or password only for a connection, select a connection and choose the *Change User Credentials* button.
   The *Change User Credentials* dialog box appears.
2. Enter your changes and choose *Log On*.

> **⚠ Caution**
>
> If you change the connection of the currently monitored system, you must restart the DBA Cockpit. Otherwise the changes will not take effect.

Testing a Database Connection

To check if you have entered the correct user and password information as well as the correct technical connection data, you can test a database connection. To do so, select a connection and choose the *Test* button.

The test result is displayed in the *Connection Test Protocol* in the content detail area.

Deleting a Database Connection

To delete a connection, select a connection and choose the *Delete* button.

> ⚠ Caution
>
> If the selected database connection is still used by a system that is registered in the DBA Cockpit, you **cannot** delete it.

5.2 Enablement of Databases for the Data Collection Framework

The data collection framework (DCF) is part of the DBA Cockpit back end and is the infrastructure for collecting time-based database metrics. Regardless of whether the database has been configured for monitoring using SAP Solution Manager or using the DBA Cockpit, make sure that the DCF is set up correctly.

As a standard, the DCF is set up automatically (see Enabling the Database for the Data Collection Framework Automatically [page 50]). In some cases, however, you need to set up the DCF manually (see Setting Up the Data Collection Framework (DCF) Manually [page 51]).

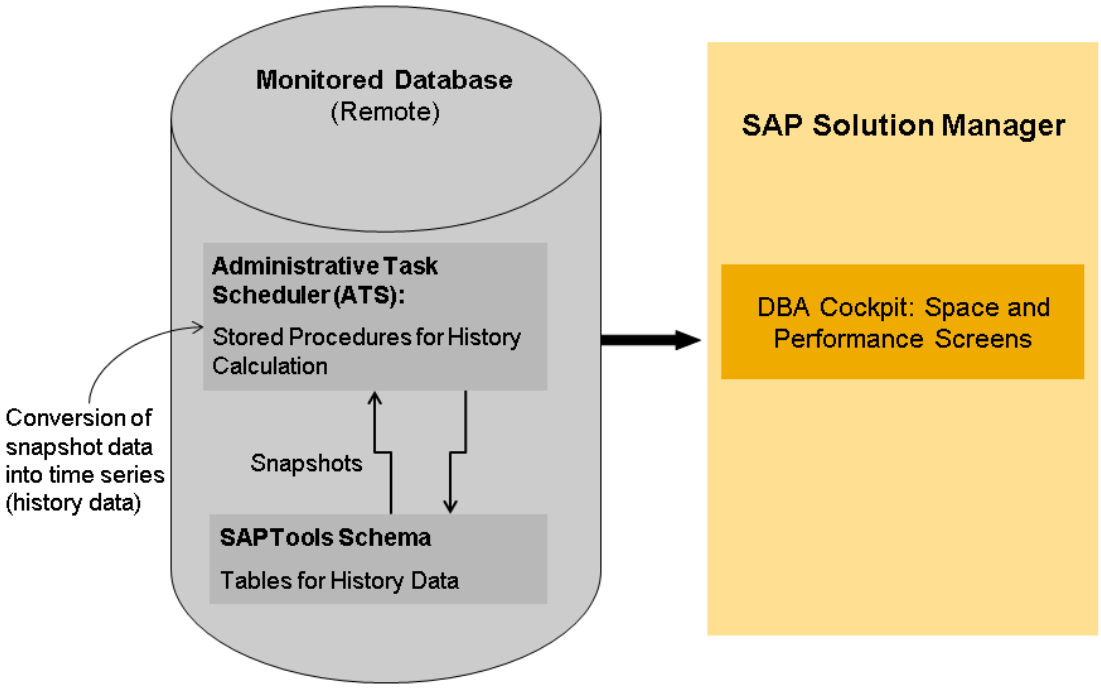
5.2.1 The Data Collection Framework

What Is the Data Collection Framework?

The data collection framework (DCF) is part of the DBA Cockpit back end and is the infrastructure for collecting time-based database metrics. The metrics include data about the performance, configuration, and space consumption of a database monitored by the DBA Cockpit. During the setup of the DCF, a separate schema (`SAPTOOLS`) is created in the monitored database that contains all relevant monitoring objects, such as:

- A set of stored procedures to collect data on a regular basis
  These stored procedures are scheduled by the DB2 administrative task scheduler (ATS).
- A set of tables that keep the history of the collected data
- Tablespaces to store these monitoring tables

In the case of an SAP system, a work process-based workload management configuration is applied.



Data Collection Framework: Architecture Overview

Benefits of the Data Collection Framework

The DCF provides a higher number of metrics than the older, deprecated collection of space and performance history data from releases lower than SAP enhancement package 2 for SAP NetWeaver 7.0. If you use the DCF, you can benefit from the following enhanced functions:

- More than 1000 metrics for performance and space
- Drill-down and aggregation functions for individual metrics
- Selection options to restrict the history data to a specific time period
- Graphics for metrics, history data, and diagnostics
- Time-based metrics

> **i** Note
>
> - The availability of these functions also depends on your database version.
> - The Web browser-based version and the SAP GUI version of the DBA Cockpit are enabled to process data provided by the DCF. However, there are a few screens that are still available for older DB2 releases that support the DCF and the older collection of space and performance history data. For more information, see Transition from the Deprecated Collection of Space and Performance History Data to the Data Collection Framework [page 55].

Setup of the Data Collection Framework

The DCF is set up during the setup of SAP Solution Manager. For more information, see Enablement of Databases for the Data Collection Framework [page 48].

Standard Templates for the DCF

All settings for the data collection, for example, recurrence patterns for the scheduled data collectors and the amount of data to be collected are determined by templates. SAP provides the following standard templates for the DCF:

- **SAP Default**
  Most of the data collectors run hourly. The sliding window is set to two weeks except for the most important tables and indexes.
- **SAP Detailed**
  Most of the data collectors run quarter-hourly. The sliding window is set to two weeks except for the most important tables and indexes.

If you need settings different from the standard setup, you can create your own template based on a standard template and adapt this template to your needs (see Maintaining Templates for the DCF [page 52]).

5.2.2 Enabling the Database for the Data Collection Framework Automatically

Use

Regardless of whether the database has been configured for monitoring using SAP Solution Manager or using the DBA Cockpit, the first time you select a database system for monitoring in the DBA Cockpit, the existence of the DCF is checked.

If the DCF does not yet exist, it is automatically set up if the following applies:

- The minimum database release level of the monitored database is at least DB2 V9.1.
- The database is enabled for automatic storage management or the required tablespaces have already been created.

If the DCF cannot automatically be set up because of missing tablespaces or insufficient privileges of the monitoring user, the DBA Cockpit initially displays the appropriate configuration screen. On this screen, you can perform the required setup steps as described in Setting Up the Data Collection Framework (DCF) Manually [page 51].

> **i Note**
>
> If you set up the DCF, the monitoring history function that has been deprecated as of enhancement package 2 for SAP NetWeaver 7.0 is switched off, that is:
>
> - The option *Enable Deprecated Data Collection Additionally (Not Recommended)* in the system configuration entry is disabled so that the related SAP background jobs do not collect history data anymore.
> - If the *REORGCHK for All Tables* job is scheduled, it is removed from the DBA Planning Calendar and the respective data is now collected by the DCF.
> - The special compression function, which was provided by the *REORGCHK for All Tables* job before, has been integrated as a data collector in the DCF. The compression results are available on the *Compression Status* and *Compression Candidates* screens under *Space*.

More Information

Data Collection Framework: Configuration [page 243]

Ownership of Data Collectors [page 247]

5.2.3 Setting Up the Data Collection Framework (DCF) Manually

Context

The DCF is set up individually for each database based on the database version, the usage type of the database, and the SAP coding. To set up the DCF manually, you can either use a template or activate and configure individual data collectors.

> **→ Recommendation**
>
> We recommend that you use templates to ensure that data is collected homogeneously in your system landscape. Only in rare cases, it might be advisable that you configure individual data collectors.

Procedure

1. Call the DBA Cockpit.
2. On the *Database* tab page (Web browser) or in the navigation frame (SAP GUI), choose *Configuration* ▸ *Data Collection Framework: Configuration*.

3. On the screen *Data Collection Framework: Configuration*, choose the *Implement Template* button.

   The *Implement Template* dialog box appears. The default template is marked with an asterisk (*).

   By default, deprecated functions are preselected for disabling. If you want to keep any old settings, deselect the relevant checkboxes.

   > **i** Note
   >
   > The option to disable deprecated functions or to keep them is only available if there are deprecated functions that are currently active but recommended for disabling.

4. To set up the template in the monitored database, choose the *Implement* button.

Next Steps

- Data Collection Framework: Configuration [page 243]
- Data Collection Framework: Template Definition [page 250]

5.2.4 Maintaining Templates for the DCF

The DCF is based on predefined templates. These templates ensure a homogeneous setup of the data collectors in your system landscape. You maintain these templates on the *Template Definition* screen in the DBA Cockpit.

For more information about creating and maintaining your own DCF templates, see Data Collection Framework: Template Definition [page 250].

SAP provides two default templates that cannot be modified. You can, however, save them as new templates and then modify those. For more information, see section *Standard Templates for the DCF* in The Data Collection Framework [page 48].

5.2.5 Collection of Space and Performance History Data (Deprecated)

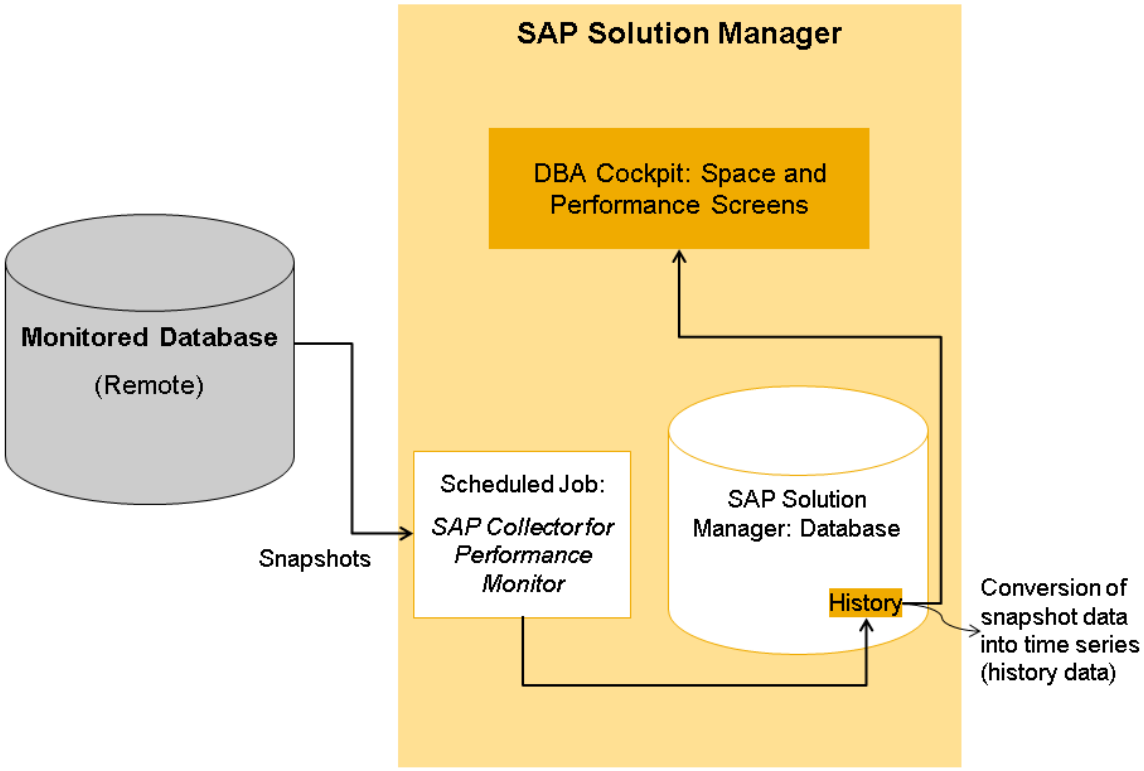
Use

For SAP releases lower than SAP enhancement package 2 for SAP NetWeaver 7.0, only a basic data collection mechanism is available that allows you to collect space and performance history data of your database. If you run the DBA Cockpit in such a system, this data is used by the DBA Cockpit on performance and space screens.

> **i** Note
>
> If you run the DBA Cockpit on a system based on SAP enhancement package 2 for SAP NetWeaver 7.0 or higher and your database release is DB2 V9.1 or higher, this collection of space and performance history

data has been replaced by the data collection framework (DCF) (see The Data Collection Framework [page 48]).

Architecture



Data Collection for Performance and Space Monitoring: Architecture Overview

The job *SAP Collector for Performance Monitor* collects snapshots of performance data of monitored databases. These snapshots are converted into time series (history data) by the back end of the DBA Cockpit. The collected history data is displayed on the space and performance screens of the DBA Cockpit.

> ### i Note
>
> Using the deprecated collection of space and performance history data results in additional workload and space consumption for the SAP Solution Manager system (or any other central monitoring system) because data is collected on the central system. Therefore, the deprecated collection of space and performance history data is not suitable for large system landscapes. The deprecated collection of space and performance history data differs here from the DCF, where data is collected in the local databases.
>
> Furthermore, the collected data is only a subset of the monitoring figures collected by the DCF. There is no separation between detailed short-term and long-term data, and long-term data is not transferred to the Performance Warehouse of SAP Solution Manager.

Setup

When you set up an SAP Solution Manager system (or any other system on which you run a DBA Cockpit) and you still want to use the deprecated collection of space and performance history data, proceed as follows:

> **i** Note
>
> Setting up this data collection is only necessary if the DCF is not supported by your DB2 release.

- Enable the data collection by selecting the *Enable Deprecated Data Collection Additionally (Not Recommended)* checkbox on the *System Configuration* screen (see also Editing the Monitored System Landscape on the System Configuration Screen [page 57]).
- Use the DBA Planning Calendar (transaction `DB13`) to schedule the job `REORGCK_ALL` once a week according to SAP Note 1268290.
- Use the *Define Job* (`SM36`) transaction to schedule the SAP standard job *SAP Collector for Performance Monitor* (`sap_collector_for_perfmonitor`) according to SAP Note 16083.

Configuration

You can configure the collection of space and performance history data in the DBA Cockpit under *Configuration* > *Monitoring Settings*. The configuration allows you, for example, to define the retention time of the collected data in the system. The default settings are: 20 days/12 weeks/12 months. Note that the values for the retention periods are only visible if the *Enable Deprecated Data Collection Additionally (Not Recommended)* checkbox is selected on the *System Configuration* screen.

Supported Screens

All SAP GUI-based screens that display data of the deprecated collection of space and performance history data have been enabled to process both old and new history data. This means that data collected by the old collection mechanism will be merged with data collected using the DCF (see Transition from the Deprecated Collection of Space and Performance History Data to the Data Collection Framework [page 55]).

For more information about the DCF, see The Data Collection Framework [page 48].

5.2.6 Transition from the Deprecated Collection of Space and Performance History Data to the Data Collection Framework

> **i** Note
>
> This section is only relevant if you have worked with the (deprecated) collection of space and performance history data in the DBA Cockpit and now set up the data collection framework (DCF) for the first time.

If you have used the (deprecated) collection of space and performance history data before and you now set up the DCF, this has an effect on the available functions and screens. Screens that run only with the deprecated collection of space and performance history data will be switched off, and new and enhanced screens and functions will become available.

New and Enhanced Screens After the Setup of the Data Collection Framework (DCF)

After the setup of the DCF, many screens in the SAP GUI-based DBA Cockpit get enhanced functions, for example, an integrated monitoring history and graphics.

The following screens become available in the SAP GUI-based DBA Cockpit:

- *Performance* > *Dashboard*
- *Performance* > *Containers*
- *Performance* > *Top SQL Statements*
- *Performance* > *Memory*
- *Performance* > *Transaction Log*
- *Performance* > *System Resources*
- *Performance* > *pureScale: Cluster Caching Facility* (on pureScale systems)
- *Jobs* > *Scheduled DB2 Tasks*
- *Space* > *Database*

Switch-Off of Deprecated Functions After the Setup of the Data Collection Framework (DCF)

If you set up the DCF, monitoring history functions that have been deprecated as of enhancement package 2 for SAP NetWeaver 7.0 are switched off:

- The option *Enable Deprecated Data Collection Additionally (Not Recommended)* on the *System Configuration* screen is disabled so that the related SAP background jobs do not collect history data anymore. The history data collection is taken over by the DCF.
- If the *REORGCHK for All Tables* job has been scheduled, it is removed from the DBA Planning Calendar and the respective data is now collected by the DCF.
- The special compression function, which was provided by the *REORGCHK for All Tables* job before, has been integrated in the DCF data collector *Compression*. You can see the collected compression data on the screens *Compression Status* and *Compression Candidates* under *Space*.

The following deprecated screens become unavailable in the SAP GUI-based DBA Cockpit:

- *Performance* > *History* > *Database*
- *Performance* > *History* > *Tables*
- *Performance* > *Schema*
- *Performance* > *Partitions*
- *Space* > *History* > *Overview*
- *Space* > *History* > *Database and Tablespaces*
- *Space* > *History* > *Tables and Indexes*
- *Space* > *Tables and Indexes*

We recommend that you use the more comprehensive functions on the new and enhanced screens after the setup of the DCF.

If you still want to use these disabled functions and screens, you need to run the deprecated collection of performance and history data in addition to the DCF, which is, however, not recommended. If you want to enable the deprecated collection of performance and history data again, select the *Enable Deprecated Data Collection Additionally (Not Recommended)* checkbox on the *System Configuration* screen. For more information, see Editing the Monitored System Landscape on the System Configuration Screen [page 57].

Checking the Scheduling and Execution of the Data Collection

If the DCF is set up, you can check the correct scheduling and execution of related collector jobs on the screen *Data Collection Framework: Configuration* under *Configuration*.

If the deprecated collection of performance and history data is used, the job to collect the database and tablespace history and the job to collect the history of tables and indexes are triggered by the standard performance collector job `SAP_COLLECTOR_FOR_PERFMONITOR`. You can display the schedule of these two jobs in the DBA Planning Calendar by choosing *Jobs* > *DBA Planning Calendar* on the *Database* tab page of the DBA Cockpit. From the *Category* dropdown list, you choose *DB Collectors*. The default setting is *All Actions*.

5.3 System Landscape Monitoring and Configuration

The overview screens for monitoring and administrating the system landscape are the following:

- *System Configuration*
  You can use this screen to change your system landscape that you have automatically set up using SAP Solution Manager or to set up your system landscape manually. For more information, see Editing the Monitored System Landscape on the System Configuration Screen [page 57].
- *Database Connections*
  You can use this screen to check whether your database connections are still up and running. For more information, see Configuring Database Connections [page 45].
- *DB Connection Monitor*
  You can use this screen to monitor the status of currently used database connections. For more information, see Monitoring of Database Connections [page 62].
- *Self-Monitoring*
  You can use the self-monitoring to identify issues that keep the DBA Cockpit from running properly. For more information, see Self-Monitoring of the DBA Cockpit [page 64].
- *Central Calendar*
  You can use the central calendar to monitor regular activities for your entire system landscape, such as backups. For more information, see Central Calendar [page 59].

These screens complement the functions for system monitoring available in SAP Solution Manager. The screens offer functions for database monitoring and administration that are not available in SAP Solution Manager, such as:

- Configuration of systems that are not visible in the landscape view of SAP Solution Manager
- Database connections for specific applications
- Infrastructure monitoring
- Monitoring of typical database administration tasks

5.3.1 Editing the Monitored System Landscape on the System Configuration Screen

Use

When you call up the DBA Cockpit, the *System Configuration* screen is displayed, which contains a list of all monitored systems with an icon showing the current status of a system. If you have used SAP Solution Manager to set up your system landscape for monitoring, all systems to be monitored are part of the displayed list on this screen.

Default System

When you start the DBA Cockpit for the first time, the local system is set as the default. To change this setting, select a system from the list and choose *Default System*.

> **i** Note
>
> This setting only applies to the user currently logged on to the system. It is not a system-wide setting.

Editing of the Monitored System Landscape

You can use the functions on this screen to add systems to your system landscape, to delete them, and to change the connection parameters of a system. If you use the DBA Cockpit as part of SAP Solution Manager, you do **not** need these functions to set up your monitored system landscape. You only need these functions if you want to set up the system landscape manually without using the automated setup of SAP Solution Manager or if you want to add a database that cannot be added to the system landscape using the SAP Solution Manager setup. For more information, see Configuring Systems for Remote Monitoring Manually [page 39].

Regardless of whether you have set up the system landscape for monitoring manually or by using SAP Solution Manager, you can also edit the monitored system landscape in the following cases:

- You want to enter an additional RFC connection for the database.
- You want to use the deprecated collection of space and performance history data. In this case, select the *Enable Deprecated Data Collection Additionally (Not Recommended)* checkbox.
  For more information, see Collection of Space and Performance History Data (Deprecated) [page 52].
- You want to select the monitored databases that appear in the central planning calendar. In this case, select the *Collect Central Planning Calendar* checkbox.

> **i** Note
>
> If you have set up the system landscape for monitoring with SAP Solution Manager, we recommend that you do not change any other settings.

Activation/Deactivation of a System

On the *System Configuration* screen of the DBA Cockpit, you can activate or deactivate a system. The deactivation of a system disables all background activities controlled by the ABAP server, which includes `RZ20` alert monitoring, the deprecated collection of space and performance history data, and jobs scheduled by the DBA Planning Calendar. Background activities of the data collection framework that are controlled by the database are not affected. You can switch them off separately on the screen *Configuration* > *Data Collection Framework: Configuration*.

5.3.2 Central Calendar

Use

You can use the Central Calendar in the DBA Cockpit to view actions on all the databases of your SAP systems.

> **i** Note
>
> The **Central Calendar** is only for **viewing** database administration (DBA) actions by system and for checking their results. If you need to schedule, change, delete, or execute actions, you have to switch to the **DBA Planning Calendar** for any SAP system registered in the DBA Cockpit.

The Central Calendar gives you a single point from which to manage DBA actions in an integrated SAP environment. The available actions differ according to the database platform but the method of use is the same.

Examples of actions are backups, database system checks, and so on.

The Central Calendar gives you a single point from which to manage:

- Databases of different types and versions on remote SAP systems
- Databases for different versions of the SAP system
- Databases of non-ABAP SAP systems

Using the Central Calendar, you can:

- Manage in real time systems directly administered from the system where the DBA Cockpit is running as well as remote systems, including non-ABAP systems.
- Quickly check the color-coded status for each system to see if actions have executed successfully.
- Quickly check the number of actions and number of actions with the highest status severity for each system.

More Information

- Managing the System Landscape Using the Central Calendar [page 60]
- DBA Planning Calendar [page 255]

5.3.2.1 Managing the System Landscape Using the Central Calendar

Prerequisites

- You have defined the systems to be displayed in the Central Calendar as follows:
  1. On the *System Landscape* tab page of the DBA Cockpit (Web browser) or in the navigation frame (SAP GUI), choose *System Configuration*.
  2. On the *System Configuration* screen, choose the required system.
  3. In the *System Configuration Details* area (Web browser) or on the *System Configuration Entry* screen (SAP GUI), under *Collector Options*, select the *Collect Central Planning Calendar Data* checkbox.
- In an ABAP system, make sure that you schedule the jobs for the remote database in the central monitoring system. Jobs that have been scheduled in the remote system are not displayed.
- In the local DBA Planning Calendar of the system where you want to use the Central Calendar, you have planned the action *Central Calendar Log Collector* to run regularly. This collects information from the defined remote systems for display in the Central Calendar.

For more information, see DBA Planning Calendar [page 255].

Procedure

1. Start the Central Calendar by choosing *System Landscape* > *Central Calendar*. The Central Calendar is displayed. If you have already run or planned actions, you see entries by day, one for each system, which provide a summary of the scheduled jobs.

   The following is an example of entries for *Thursday February, 08* affecting two systems, *FUD* and *FIB*:

   Example

3	*FUD*	*2*
4	*FIB*	*1*

   On system FUD for Thursday, February 8th, there were three actions planned, two of which had the highest status severity. For example, the highest status severity for FUD might be *Finished with warning*, in which case the entry for FUD is displayed with a yellow background. This means that two actions ended with a warning.

   On system FIB for the same day, there were four actions planned, one of which ended with the highest severity. For example, the highest severity for FIB might be *Finished with error*, in which case the entry for FIB is displayed with a red background. This means that one action ended with an error.

The following table shows the color-coded statuses in the Central Calendar, which you can also see by choosing the *Legend* button:

Color	Status
Light blue	Planned
Dark blue	Running
Green	Finished successfully
Yellow	Finished with warning
Red	Finished with error
Dark yellow	No longer available
Dark red	Scheduling failed

2. For a summary of the actions for a day, click the relevant day header.

   A summary of the actions and the status for each system on the day you selected are displayed in the *Details* area, as in the following example:

System	*Total*	*No longer available*	*Scheduled*	*Running*	*Finished*	*Warning*	*Error*	*Overdue*	*Not yet available*
FUD	*3*	*1*		*2*					
FIB	*4*		*3*				*1*		

3. To see the individual actions for a system, choose the relevant system entry on the required day. The relevant day is displayed in the DBA Planning Calendar. You can perform all the usual functions in the DBA Planning Calendar. (Note that if the selected system is not the currently selected system in the DBA Cockpit, a separate window opens for the local DBA Planning Calendar.)
4. To obtain new statistics of the scheduled jobs for all systems, choose the *Refresh in the Background* button.
5. In the *Schedule Refresh* dialog box, choose between the following options:

Option	Description
Run in Dialog	Refresh runs in dialog mode, which can take a long time. Therefore, we usually do not recommend this option.
Start Immediately in the Background	Refresh runs immediately as a background job
Schedule on <date> at <time>	Refresh runs in the background on the date and at the time that you specify

> → Recommendation
>
> We recommend that you schedule the action *Central Calendar Log Collector* to run regularly as described above in *Prerequisites*.

6. To refresh the display with the newly gathered statistics of the previous step, choose the *Refresh* button.
7. Optional step:

   In the *Selection* area, you can customize the calendar display as follows:

   - Specify dates or time periods to be displayed
   - Specify a factory calendar
     Holidays are displayed in the same background color as weekend days. This does not restrict the planning of actions in the DBA Planning Calendar.

5.3.3 Monitoring of Database Connections

Use

You can use the *DB Connection Monitor* screen to monitor all database connections from all application servers.

For more information about a database connection, select the relevant entry from the overview table. The *Database Connection Details* area appears displaying details about the database connection, such as technical settings, user credentials, and parameter values.

Features

To access the *DB Connection Monitor* screen, call the DBA Cockpit, and choose *System Landscape* > *DB Connection Monitor*.

The following information is displayed:

> ℹ Note
>
> By default, all disconnected database connections are hidden.

Column	Description
Server	Name of the application server
Work Process	Number of the work process

Column	Description
Handle	Internal number of each connection of a work process (each work process can have several connections) You can use this handle to analyze a specific connection.
Connection Name	Name of the connection
Connection ID	ID of the connection
Connection State	State of the connection Possible values are: • *ACTIVE* • *INACTIVE* • *CONNECTING* • *DISCONNECTED*
Changing Operation	Indicates if a changing operation was executed
Block Commit	Indicates if COMMIT statements are blocked (If a COMMIT is executed, a dump occurs.)
Cursor with Hold	Indicates that the connection has a cursor with hold
Permanent Connection	Indicates if the connection is permanent
Reconnect	Indicates if the connection automatically reconnects after it was closed
Force Reconnect	Indicates if the connection is forced to reconnect after it was closed
Timeout	Timeout when the connection is closed
Maximum	Maximum allowed number of open connections
Optimum	Optimal number of connections
Date	Date when the connection was established
Time	Time when the connection was established
DB Host	Host name
Program	Program that opened the connection

5.3.4 Self-Monitoring of the DBA Cockpit

Use

You can use the self-monitoring infrastructure of the DBA Cockpit to identify issues that keep the DBA Cockpit from running properly. Such issues can include the following, for example:

- Has the database configuration been set up correctly for monitoring?
- Has the monitoring infrastructure been set up for the DBA Cockpit (see also Setting Up the DBA Cockpit for Remote Monitoring [page 34])?
- Is the data collection framework set up, and are all data collectors set up and configured correctly (see also Enablement of Databases for the Data Collection Framework [page 48])?
- Do the data collectors of the data collection framework run without errors, and do they collect data (see also Data Collection Framework: Configuration [page 243])?
- Has a database upgrade or a Fix Pack update been performed successfully so that the DBA Cockpit can continue monitoring?
- Has the `db6_update_db` script run successfully for the database?

The self-monitoring infrastructure allows you to display a list of all database checks and to repair issues quickly.

> **i** Note
>
> The self-monitoring function only checks the monitoring infrastructure, not the database systems. For checking the database systems, you use the alerting infrastructure (see Alerts [page 282]).

Features

Predefined Checks

The self-monitoring infrastructure consists of a number of checks predefined by SAP that automatically run for each DB2 for LUW database monitored by the DBA Cockpit. There are two types of checks, which differ with respect to the level of detail of the check:

- **Fast**
  This check type can be executed at runtime. Its status can be updated by choosing the *Refresh* button on the screen.
- **Detailed**
  This check type is more detailed than a fast check, but only runs in batch mode.

Detail Information and Repair Function

On the *Self-Monitoring Details* screen area, the DBA Cockpit shows you detailed results of each check and advises you which actions you should take. To analyze and repair the issue identified by the check, you can proceed as follows:

- Use the *Quick Repair* button to let the DBA Cockpit repair the issue quickly in the background (if possible).
- Use the *Go to Repair Tool* button to navigate to the relevant tool that helps you analyze and repair the issue identified by the check.

- Use the *Go to SAP Note* button to display the content of the relevant SAP Note for the issue (if available).

Self-Monitoring Screens

The self-monitoring infrastructure is available in the DBA Cockpit as follows:

- **For your entire monitored system landscape**: Choose *System Landscape* > *Self-Monitoring*.
- **For a single database**: On the *Database* tab page (Web browser) or in the navigation frame (SAP GUI), choose *Diagnostics* > *Self-Monitoring*.

Messages from Self-Monitoring on Affected Screens

If there are issues with the setup and infrastructure of the DBA Cockpit, the relevant error messages are also displayed on the screens that are affected by the issues.

5.3.4.1 Displaying Check Results of Self-Monitoring

Use

You can use the self-monitoring infrastructure of the DBA Cockpit to identify and solve issues related to database configuration or monitoring setup that keep the DBA Cockpit from running properly. The *Self-Monitoring* screens display results of automated checks designed for the DBA Cockpit, from which you can navigate to individual check results and to repair tools.

Procedure

Displaying Check Results of Self-Monitoring for the System Landscape

1. Choose *System Landscape* > *Self-Monitoring*.
   The DBA Cockpit displays an overview of your system landscape, including information about the database release and whether the data collection framework is enabled. A red check status indicates that one of the self-monitoring checks resulted in an error.
2. To display details about the check status of a system, select a line of the overview table.
   On the *Self-Monitoring: System* screen area, the DBA Cockpit displays a list of all checks performed for a system, including a short description of each check and the check status.
3. To display details about a particular check, select a line in the table.
   On the *Self-Monitoring* screen area, the DBA Cockpit displays an information message or an error message. In the case of an error message, the DBA Cockpit displays instructions with recommended actions.
4. To navigate to the appropriate tool to repair an issue identified by a check, choose *Go to Repair Tool*.

Displaying Check Results of Self-Monitoring for a Database

1. On the *Database* tab page (Web browser) or in the navigation frame (SAP GUI), choose *Diagnostics* > *Self-Monitoring*.
   The DBA Cockpit displays a list of all checks performed for the database, including a short description of each check and the check status.

2. To display details about a particular check, select a line in the table.
   On the *Self-Monitoring* screen area, the DBA Cockpit displays an information message or an error message. In the case of an error message, the DBA Cockpit displays instructions with recommended actions.
3. To navigate to the appropriate tool to repair an issue identified by a check, choose *Go to Repair Tool*.

More Information

Self-Monitoring of the DBA Cockpit [page 64]

6 Performance

6.1 Performance: Dashboard

> **i** Note
>
> This screen is only available for DB2 versions as of V9.5.

The dashboard shows a selection of the most important metrics. You get information about properties of the database, host, cache qualities, memory sizes, locks and deadlocks, workload counters, backup, growth of the database, loggers, and physical I/O counters.

The screen is displayed when you switch from the *System Landscape* tab to the *Database* tab in the Web browser-based version of the DBA Cockpit. You can also call up the screen by choosing *Performance* ▸ *Dashboard* on the *Database* tab of the DBA Cockpit (Web browser) or in the menu of the DBA Cockpit (SAP GUI).

Web Browser-Based DBA Cockpit Only

If the Monitoring and Alerting Infrastructure (MAI) from SAP Solution Manager is enabled for the database, some metrics on the screen are rated with a traffic light. The traffic light depends on the value and the threshold of the corresponding metric in the MAI. The selection of metrics with a traffic light is predefined by SAP using the data provider *Database Alerts Generic*, but you can change the thresholds for the traffic lights in the MAI.

6.2 Performance - pureScale: Cluster Status

> **i** Note
>
> This screen is only available in the SAP GUI-based version of the DBA Cockpit.

The *pureScale: Cluster Status* screen provides an overview of the members or cluster caching facilities (CFs) that are part of the current DB2 pureScale cluster. In addition, information about current or home host names and alerts per member are displayed as well as the current state of a member.

6.3 Performance - pureScale: Page Reclaim

> **i** Note
>
> This screen is only available in the SAP GUI-based version of the DBA Cockpit.

The *pureScale: Page Reclaim* screen provides a detailed overview of events and statistics if competitive situations occur, for example, if different members try to access the same physical page at the same time. The overview table on the *Page Reclaim* screen displays the following key figures about page reclaims:

Column	Description
Schema	Name of the database schema
Table	Name of the table belonging to the schema
Reclaim Wait Time (sec)	Overall page-reclaim wait times that are aggregated for each table
Exclusive Page Reclaims	Total number of exclusive page reclaims per table
Shared Page Reclaims	Total number of shared page reclaims per table
SMP Exclusive Page Reclaims	Total number of exclusive space map page reclaims per table
SMP Shared Page Reclaims	Total number of shared space map page (SMP) reclaims per table

To access detailed statistics for each DB2 pureScale member, double-click a line in the overview table. To drill down the metrics for a specific member, double-click the member to be analyzed in the *Details of Table* area. In the *Table Page Contention* and *Index Page Contention* areas that appear, the number of shared and exclusive reclaims and currently initiated reclaims is displayed for the selected member and table.

6.4 Performance - pureScale: Cluster Caching Facility

Use

On the *pureScale: Cluster Caching Facility* screen, you can display status information about cluster caching facilities on the system.

Prerequisites

This screen is only available if the following applies:

- The currently monitored database is a Db2 for LUW database with the pureScale Feature.
- The data collection framework (DCF) is set up correctly (Data collector *Cluster Caching Facility*).
  If the DCF is not available or not correctly set up, a warning is displayed including a link to the DCF configuration screen where you can perform the necessary steps. For more information, see Data Collection Framework: Configuration [page 243].

Features

To access the *Cluster Caching Facility* screen, choose *Performance* > *pureScale: Cluster Caching Facility* on the *Database* tab page of the DBA Cockpit (Web browser) or in the navigation frame of the DBA Cockpit (SAP GUI).

Based on your time selection, the DBA Cockpit displays the following information for each cluster caching facility available on your system:

- Memory sizes and buffer pool sizes, such as the average configured global buffer pool size for the cluster caching facility
  The memory and buffer pool sizes are measured in pages with a page size of 4 KB.
- History data for each cluster caching facility
- A graphical view of the history data

For more information about the screen layout and navigation, see Web Browser-Based User Interface (Web Dynpro) [page 23].

6.5 Performance: Time Spent Analysis

Use

You can use this screen as the starting point for performance tuning and to identify time-based problems of your database. The *Time Spent Analysis* screen provides an aggregated and timeline-based view of the current or past performance situation of your database. You can analyze specific workload situations in real-time.

The advantage of time spent monitoring is that you do not only have to rely on standard database key performance indicators (such as the buffer pool hit-ratio) but you can also identify how much time is really spent on different kinds of database operations.

You can use time spent monitoring to answer questions such as the following:

- How do changes in system performance relate to changes in the total time spent in various Db2 components or wait situations?
- Where is most of the time spent in Db2 so that these areas can be tuned?

- What are the most active service classes and members on the database system, and are there any outliers?

Features

You can access the *Time Spent Analysis* screen by calling the DBA Cockpit and choosing *Performance* > *Time Spent Analysis* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

The *Time Spent Analysis* screen is divided into the following areas:

- *Selection* area
- Chart view *Time Spent in DB2 by <drill-down criteria>* with the following display options:
  - Pie chart
    The pie chart is the initial view of any time spent analysis.
    All occurrences of metrics or service classes are displayed as well as their relative values as a percentage of the total time consumption of the monitored system. This view shows the aggregated results of the drill-down dimension for the specified time frame.
  - Timeline histogram
    The timeline-based chart lets you analyze changes in workload performance for a given time frame. This chart type displays the cumulated occurrences of selected time-based metrics, members, or service classes into a histogram that consists of bars that are aligned to the timeline. The histogram bars are divided according to the time granularity you have specified in the *Selection* area.
- List view
  In the list view of the data displayed in the chart, you also get additional information like time averages, event counts, and so on. In the SAPGUI-based DBA Cockpit, the list view is always visible. In the Web browser-based DBA Cockpit, the list view is hidden. You can make the list view visible on the screen by choosing the *Toggle Table Display* button in the chart toolbar. Alternatively, you can display the data including additional information by hovering your mouse pointer over the corresponding area of the chart. In the Web browser-based version of the DBA Cockpit, you can click on the pie chart to navigate to detail information. In the SAP GUI version of the DBA Cockpit, double-click an entry in the list to display the details.

More Information

- Identifying and Analyzing Time-Based Database Problems [page 71]
- Example: Identifying and Analyzing I/O-Related Database Workload [page 75]
- Example: Identifying Reasons for a System Slowdown [page 76]
- Monitoring metrics for Db2 workload management in the IBM Db2 documentation

6.5.1 Identifying and Analyzing Time-Based Database Problems

Procedure

To access the *Time Spent Analysis* screen, call the DBA Cockpit and choose *Performance* > *Time Spent Analysis* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

Step 1: Specifying Filter and Drill-Down Criteria

In the *Selection* area, specify filter and drill-down criteria for the data to be displayed as follows:

Field	Description
Source	From the dropdown list, choose the source of the data to be analyzed: • *Periodic Data Collection* If you choose this option, the data that has already been collected by the data collector *WLM Statistics* is used for the time spent analysis. For more information, see Data Collection Framework: Configuration [page 243]. **i Note** The time granularity of the data analysis can only be as fine as specified in the collection interval. • *Ad-hoc Data* If you choose this option, real-time monitoring data is collected. This means that specific workload situations can be analyzed in real-time. To set a time interval for the data collection and the charts to be refreshed, choose an interval from the *Refresh Interval* dropdown list. To start the ad-hoc data collection, choose the *Resume* button. If you want to change the drill-down or filter criteria, you must first stop the current ad hoc data collection by choosing the *Pause* button.
Time Frame	Specify the time frame for which you want to analyze data.
Type of Display	• *Aggregate* If you choose the *Aggregate* option from the dropdown list, a pie chart is displayed according to your selected filter and drill-down criteria. • *Timeline* If you choose the *Timeline* option from the dropdown list, the monitoring data is displayed in a histogram according to your filter and drill-down criteria. You can change the time granularity of the displayed data from *As Collected* to *Hour*, *Day*, *Week*, or *Month*.

Field	Description
Members	**Note** You can **only** filter by members if you are using a **distributed database system**. From the dropdown list, choose the database member on which you want to apply the specific workload filter. You can choose all or single members. If you want to combine multiple members into a group (for example, by physical host), use the *Advanced Selection* option (Web browser-based version of the DBA Cockpit only).
Service Class(es)	From the dropdown list, choose the service classes that have been set up on the system. You can choose all or single service classes. If you want to combine multiple service classes into a group, choose *Advanced Selection* from the dropdown list (Web browser-based version of the DBA Cockpit only).

Field	Description
Metric(s)	From the dropdown list, choose between the following time-related WLM metrics: • *All* All known time-related metrics are included. • *All (Wait Times)* All wait time metrics are selected (W) • *All (Processing Times)* All processing times without wait times are selected (P). • Single metrics, for example, *(P) Commit Processing Time* or *(W) Buffer Pool Read Time* For more information, see Monitoring metrics for Db2 workload management in the IBM Db2 documentation. • *Advanced Selection* (Web browser only) You can choose *Advanced Selection* in the Web browser-based version of the DBA Cockpit. When you choose this option, all metrics including their parent metrics are displayed. You can now select those metrics that are relevant for your analysis. For example, you select all `REORG` and `RUNSTATS` processing times to drill down to specific members or member groups. In the SAP GUI-based version of the DBA Cockpit, the advanced selection is not available. However, you can display the metrics hierarchy. If you choose the *Choose Level of Detail* link, you can specify one of the following options in the *Metrics Hierarchy* dialog box: • *Total Request Time Only* Aggregates all metrics • *Wait vs. Processing vs. Unknown Times* Aggregates metric groups • *Detailed Wait and Processing Time* Includes only detailed wait and processing times • *Full Details* Includes all time metrics (default) **i Note** The metrics are mutually exclusive and are independent of each other. Since all metric metadata is retrieved dynamically from the database, the availability of specific metrics as filter criteria depends on the database release.

Field	Description
Drill-Down	Drill down into the display by choosing one of the following options from the dropdown list: • *Members* All results are distributed across the different members based on the specified filter criteria. **i Note** You can **only** drill down by members if you are using a **distributed database system**. • *Metrics* All results are displayed according to the different time-based monitoring metric occurrences and their relative values as a percentage of the total request time. • *Service Classes* Based on the specified filter criteria, all results are distributed across the service classes according to their occurrence and their relative percentage to the total request time. To avoid the chart becoming overloaded and unclear, the *Reduce Display to Most Significant* checkbox is by default selected. All metrics, members, and service classes with insignificantly small values are cumulated to give the value of *Other Known Times*. **i Note** Make sure that drill-down and filter criteria are not the same, that is, do not use, for example, non-aggregated single metrics combined with the option to drill down by metrics.

After applying your filter and drill-down criteria in the *Selection* area, the relevant time spent analysis chart is generated.

Depending on the specified display type, either a pie chart or timeline histogram is displayed.

For more information about the chart types, see Performance: Time Spent Analysis [page 69].

Step 2: Displaying Details and Analyzing the Charts

The analysis of a single pie chart or histogram item can vary due to the drill-down criteria you have specified (that is, metric vs. member or service class). To analyze a single item of a chart, click the item or caption in the Web browser-based version of the DBA Cockpit. In the SAP GUI version of the DBA Cockpit, double-click the relevant item in the list view under the chart.

The relevant details appear in the *<metric> Summary* area displaying all values and occurrences for the selected time frame or time interval as follows:

- Percentage of the total time related to the occurrence in the specified time interval or time frame
- Processing times, nominal and worst values per service class or member
- Total occurrences for the total time frame or the selected time interval
- Graphs of the absolute and average values (Web browser only)

Web browser only: In the *See Also* area, use links such as the following to navigate directly to areas of the DBA Cockpit that might help you analyze the problem:

- *Configuration Changes*
- *Workload Statistics*
- *Threshold Violations*

Previous filter and drill-down criteria are kept if possible.

Step 3: Customizing the Charts (Optional, Web browser-based version of the DBA Cockpit only)

You can change the chart type from *Stacked Columns* to *Stacked Area* or *Lines* by choosing the *Chart Type* menu button.

To specify whether you want to display the data in absolute or percentage values, use the *Values* menu button. If you choose the *Percentage* option, the values are displayed in relation to the total request time per time frame or per time interval.

By choosing the *Chart Size* menu button, you can specify the width and height of the chart to be displayed. If required, you can open a detailed table view by choosing the *Toggle Table Display* button.

More Information

- Performance: Time Spent Analysis [page 69]
- Example: Identifying and Analyzing I/O-Related Database Workload [page 75]
- Example: Identifying Reasons for a System Slowdown [page 76]
- Monitoring metrics for Db2 workload management in the IBM Db2 documentation

6.5.2 Example: Identifying and Analyzing I/O-Related Database Workload

Use

The following example illustrates how you can analyze and identify I/O-related workload on the database using the *Time Spent Analysis* screen of the DBA Cockpit.

Procedure

1. Set the metrics filter as follows:
   - **Web browser**: In the *Metric(s)* dropdown listbox, choose *Advanced Selection*, then select *Direct Read Time*, *Direct Write Time*, *BP Read Time*, and *BP Write Time*.
   - **SAP GUI**: In the *Metric(s)* dropdown listbox, select *Direct Read Time*.
2. As type of display, choose *Timeline*.

3. As drill-down option, choose *Service Classes*.
4. After applying your selection, you can identify time consumption peaks in the displayed charts. To identify the responsible workloads for those peak values, pay special attention to the nominal total time per service class versus the worst service class in the *Summary* area.
5. **SAP GUI only**: Repeat the procedure with *Direct Write Time*, *BP Read Time*, and *BP Write Time*.

6.5.3 Example: Identifying Reasons for a System Slowdown

Use

The following example illustrates how you can identify reasons for a general system slowdown using the *Time Spent Analysis* screen of the Web browser-based version of the DBA Cockpit.

Procedure

1. As *Type of Display*, choose *Timeline*, and as *Drill-Down* option, choose *Metrics*.
2. In the *Members* and *Service Class(es)* fields, choose *All* and apply your selection.
3. Identify the most significant time metrics in the displayed chart.
4. In the detailed *Summary* area of those metrics, check for outlier members or service classes to restrict the scope to a smaller group of members or service classes.
5. For a more detailed analysis of the system slowdown, keep your selection of members and service classes and navigate into other areas of the DBA Cockpit such as the following:
   - *Performance* > *SQL Cache*
     Search for statements with high values for the identified metrics.
   - *Diagnostics* > *Lock-Wait Events*
     Analyze lock-wait situations or a switch in data collection.
   - *Performance* > *Workload Statistics*
     Analyze statistical metrics for the identified members or service classes.
   - *Workload Management* > *Workloads and Service Classes*
     Check the priority settings of the involved workloads.

6.6 Performance: Database

You can access database performance data by calling the DBA Cockpit and choosing *Performance* > *Database* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

Depending on the setup of your DBA Cockpit, one of the following is displayed on the *Database* screen:

- Enhanced database performance information based on data from the data collection framework (DCF)

- Database performance data without DCF support

For more information about the DCF, see Enablement of Databases for the Data Collection Framework [page 48].

Enhanced Database Performance Information with DCF Support

To analyze system performance, it is not sufficient to review the key figures provided by the database from startup until the current time. This data has been cumulated over a long time period and, therefore, cannot reflect a short-term degradation of performance or an increasing or changing workload.

In addition, the interpretation of performance data mostly depends on configuration parameters that do not relate to a certain time frame either but describe **only** the current configuration of the database. Especially, if your database is using self-tuning memory management (STMM), you cannot correlate bad performance data with current storage parameters.

In contrast to the *Database* screen version without DCF support, the *Database* screen version with DCF support not only lets you view current global database performance data but also lets you monitor data that is periodically collected by the DCF.

> **i Note**
>
> To monitor data on the *Database* screen, you have to make sure that the DCF is set up correctly. If the DCF is not available or wrongly set up, an appropriate warning is displayed including a link to the screen *Data Collection Framework: Configuration* where you can perform the required steps.
>
> For more information, see Data Collection Framework: Configuration [page 243].

Features

The *Database* screen provides the following:

- A selection area where you specify filter criteria for the monitoring data (**only** available in a distributed database)
- An overview table displaying the retrieved monitoring data with the most important key figures that have been aggregated over the selected time period.
- A detail view of selected monitoring data of the result list
- A detail view of all history data that was collected for the selected monitoring data during the specified time frame.
- A graphical view of the history data

Process

To analyze database performance data, you proceed as follows:

1. You identify the time period where you experienced performance problems, such as, long-running background jobs, threshold violations detected by workload management, and bad user response time.
2. In the *Selection* area, you specify the appropriate time frame.

   > **i Note**
   >
   > If you are using a **distributed database system** and you want to retrieve a general overview, you set the member selection to *All* and you do **not** drill down by members. To retrieve a more detailed view

in a distributed database system, you can further drill down the selection by members. If a single member or a subset of members shows deviations compared to other members, you can also restrict the member selection.

3. To refresh the monitoring data, you choose the *Apply Selection* button.
   The following information is displayed for each partition:
   - Number and total size of buffer pools
   - Number of data and index logical reads
   - Number of data and index physical reads
   - Average physical read and write time that is required to read from and write data into the buffer pool
   - Executed SQL statements
   - Package cache size
   - Package cache quality
4. To display detailed information or history data, you select a line in the overview table.
   The system then displays additional tab pages for details, history data, and a graphical view of the history.
   For more information, see Displaying Detailed Database Performance Data [page 79] and Database Performance Data: History Graph [page 79].
5. To isolate the time frame during which the performance problem occurred, you use the data in the overview table on the *History* tab page.

Database Performance Information Without DCF Support

This *Database* screen provides a selection of current performance data that is related to each partition. You can use the information to identify performance-critical partitions before starting a more detailed analysis of your database.

For each partition of your database system, information about the following performance indicators is displayed:

- Number and total size of buffer pools
- Number of data and index logical reads
- Number of data and index physical reads
- Average physical read and write time that is required to read from and write data into the buffer pool
- Executed SQL statements
- Package cache size
- Package cache quality

> **i Note**
>
> If you choose a line in the overview table of the *Database* screen, detailed database performance data is retrieved and displayed in the content detail area below the overview table.

6.6.1 Displaying Detailed Database Performance Data

1. On the *Database* screen, choose a line in the overview table of the *Database* screen. Depending on the setup, data has been collected as follows:
   - **With DCF support**, the displayed monitoring values refer to the time frame that was specified in the *Selection* area. The configuration values show the configuration that was valid during the specified time frame. All values are retrieved from the history that is stored by the DCF.
   - **Without DCF support**, data has been collected since the database was started. The configuration values show the current configuration.
2. In the content detail area of the *Database* screen, choose the *Details* tab page.
3. Choose the appropriate tab page to display an overview of the following critical database performance indicators:
   - Buffer Pool [page 347]
   - Group Buffer Pool (GBP) I/O [page 349]
   - Cache [page 351]
   - Asynchronous I/O [page 353]
   - Direct I/O [page 358]
   - Real-Time Statistics [page 359]
   - Locks and Deadlocks [page 359]
   - Logging [page 361]
   - Calls [page 363]
   - Sorts [page 365]
   - XML Storage [page 366]

   The data displayed on these tab pages is described in more detail in the following sections.

6.6.2 Database Performance Data: History Graph

Use

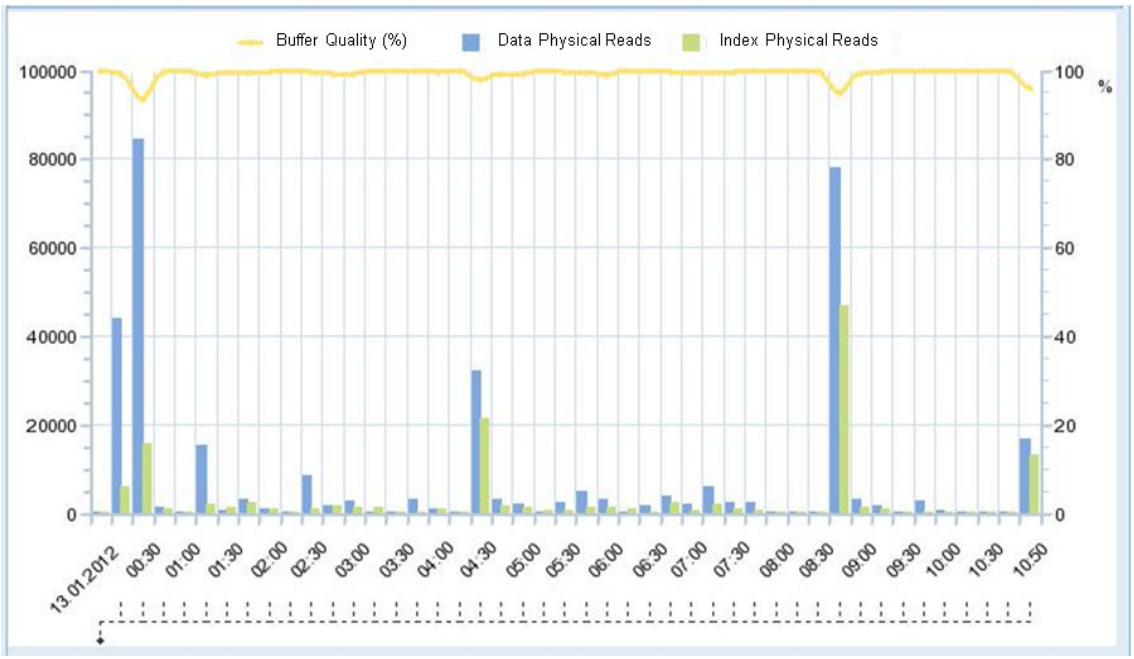
The history graph of the database performance data allows you to analyze performance changes of your database quickly. Depending on the selected time period, you can either monitor long-term trends or perform a short-term analysis, such as “Has the buffer quality dramatically gone down since this morning?”. This might be the starting point, for example, to check whether the sizes of the buffer pools are still adequate (if you have manually maintained them) or whether performance-intensive jobs have run.

Available History Graphs

The DBA Cockpit displays the following graphs for database performance data:

- Buffer quality and buffer pool size
  The graph shows the development of the buffer quality in comparison to the buffer pool size for the selected time period.

- Buffer quality and the number of physical reads of data and indexes
  The graph shows the development of the buffer quality in comparison to the number of physical reads of data and indexes for the selected time period.
- Average direct read/write time and the number direct reads/writes
  The graph shows the development of the average direct read time and write time in comparison to the number of direct reads and writes for the selected time period.



Example Graph: Buffer Quality and Number of Physical Reads of Data and Indexes

> **i** Note
>
> The y-axis on the left side refers to the number of physical reads, whereas the y-axis on the right side refers to the buffer quality (in percent).

6.7 Performance: Buffer Pools

You can access performance data for buffer pools by calling the DBA Cockpit and choosing *Performance* ▸ *Buffer Pools* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

Depending on the setup of your DBA Cockpit, one of the following is displayed on the *Buffer Pools* screen:

- Enhanced buffer pool performance information based on data from the data collection framework (DCF)
- Buffer pool performance information without DCF support

For more information about the DCF, see Enablement of Databases for the Data Collection Framework [page 48].

Enhanced Buffer Pool Performance Information with DCF Support

The *Buffer Pools* screen does not only display current global buffer pool performance data but also monitoring data that is periodically collected by the DCF.

> **i** Note
>
> To monitor data on the *Buffer pools* screen, you have to make sure that the DCF is set up correctly. If the DCF is not available or wrongly set up, an appropriate warning is displayed including a link to the screen *Data Collection Framework: Configuration* screen where you can perform the required steps.

For more information, see Data Collection Framework: Configuration [page 243].

Features

The enhanced *Buffer Pool* screen provides the following:

- A selection area where you specify filter criteria for the monitoring data (*only* available in a distributed database)
- An overview table displaying the retrieved monitoring data with the most important key figures that have been aggregated over the selected time period.
- A detail view of selected monitoring data of the result list
- A detail view of all history data that was collected for the selected monitoring data during the specified time frame.
- A graphical view of the history data

Process

To analyze buffer pool performance data, you proceed as follows:

1. You identify the time period where you experienced performance problems, for example, long running background jobs, threshold violations detected by workload management, and bad user response time
2. In the *Selection* area, you specify the appropriate time frame.

   > **i** Note
   >
   > If you are using a **distributed database system** and you want to retrieve a general overview, you set the member selection to *All* and you do **not** drill down by members. To retrieve more detailed information in a distributed database system, you can further drill down the selection by members. If a single member or a subset of members shows deviations compared to other members, you can also restrict the member selection.

3. To refresh buffer pool performance data, you choose the *Apply Selection* pushbutton.
   An overview table provides information about important key figures of the buffer pool activity of your database, for example:
   - Buffer pool size
   - Members
   - Percentage at which the data is read from the buffer pool, rather than directly from the hard disk
   - Whether the buffer pool is enabled for STMM
   - Number of data and index logical reads
   - Number of data and index physical reads

- Number of logical and physical read requests that required I/O to get data pages into the temporary tablespace

4. To display detailed information about a buffer pool, you select a line in the overview table.
   The system then displays additional tab pages for details, performance history data, and a graphical view of the performance history. For more information, see Displaying Detailed Buffer Pool Performance Data [page 82] and Buffer Pool Performance Data: History Graph [page 83].
5. To isolate the time frame during which the performance problem occurred, you use the data in the overview table on the *History* tab page.

Buffer Pool Performance Information Without DCF Support

The *Buffer Pool* screen provides an overview of the following important key indicators of the buffer pool activity of your database and enables you to compare these key indicators:

- *Buffer Pool Name*
- *Buffer Quality (%)*
- *Current Size (KB)*
- *Automatic Size (Yes or No)*
- *Data Logical Reads*
- *Data Physical Reads*
- *Index Logical Reads*
- *Index Physical Reads*
- *Temporary Data Logical Reads*
- *Temporary Data Physical Reads*
- *Temporary Index Logical Reads*
- *Temporary Index Physical Reads*

6.7.1 Displaying Detailed Buffer Pool Performance Data

Procedure

1. On the *Buffer Pools* screen, choose a line in the overview table. Depending on the setup, data has been collected as follows:
   - **With DCF support**, the displayed monitoring values refer to the time frame that was specified in the *Selection* area. The configuration values show the configuration that was valid during the specified time frame. All values are retrieved from the history that is stored by the DCF.
   - **Without DCF support**, data has been collected since the database was started. The configuration values show the current configuration.
2. In the content detail area of the *Buffer Pools* screen, choose the *Details* tab page.

3. Choose the appropriate tab page to display an overview of the following critical database performance indicators:
   - Buffer Pool [page 367]
   - Group Buffer Pool (GBP) I/O [page 349]
   - Asynchronous I/O [page 370]
   - Direct I/O [page 374]
   - XML Storage [page 375]

   The data displayed on these tab pages is described in more detail in the following sections.

6.7.2 Buffer Pool Performance Data: History Graph

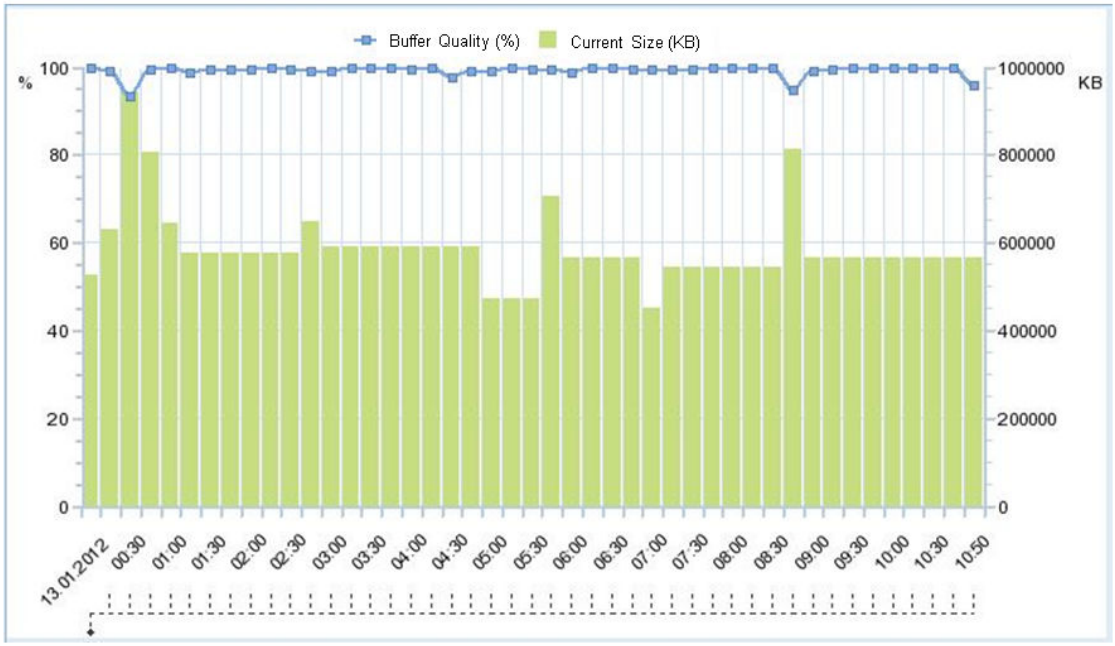
Use

The history graph of the performance data allows you to quickly analyze the buffer quality of your database. Depending on the selected time period, you can either monitor long-term trends or perform a short-term analysis, such as “Has the buffer pool size changed dramatically since this morning?”.

Available Graphs

The DBA Cockpit displays the following history graphs of the performance data for buffer pools:

- Buffer quality and buffer pool size
  The graph shows the development of the buffer quality in comparison to the buffer pool size for the selected time period.
- Buffer quality and the number of physical reads of data and indexes
  The graph shows the development of the buffer quality in comparison to the number of physical reads of data and indexes for the selected time period.
- Average direct read/write time and the number of direct reads/writes
  The graph shows the development of the average direct read time and write time in comparison to the number of direct reads and writes for the selected time period.



Example Graph: Buffer Quality and Buffer Pool Size

6.8 Performance: Tablespaces

You can access performance data for tablespaces by calling the DBA Cockpit and choosing *Performance* > *Tablespaces* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI). The *Tablespaces* screen displays important information for each tablespace in your database.

Depending on the setup of your DBA Cockpit, one of the following is displayed on the *Tablespaces* screen:

- Enhanced tablespace performance information based on data from the data collection framework (DCF)
- Tablespace performance information without DCF support

For more information, see Enablement of Databases for the Data Collection Framework [page 48].

Enhanced Tablespace Performance Information With DCF Support

The *Tablespaces* screen does not only display current global tablespace data but also monitoring data that is periodically collected by the DCF.

> **i Note**
>
> To monitor data on the *Tablespaces* screen, you have to make sure that the DCF is set up correctly. If the DCF is not available or wrongly set up, an appropriate warning is displayed including a link to the screen *Data Collection Framework: Configuration* screen where you can perform the required steps.
>
> For more information, see Data Collection Framework: Configuration [page 243].

Features

The *Tablespaces* screen provides the following:

- A selection area where you specify filter criteria for the monitoring data (only available in a distributed database)
- An overview table displaying the retrieved monitoring data with the most important key figures that have been aggregated over the selected time period.
- A detail view of selected monitoring data of the result list
- A detail view of all history data that was collected for the selected monitoring data during the specified time frame.
- A graphical view of the history data

Process Flow

To analyze tablespace performance data, you proceed as follows:

1. You identify the time period where you experienced performance problems, for example, long running batch jobs, threshold violations detected by workload management, and bad user response time
2. In the *Selection* area, you specify the appropriate time frame.

   > **i** Note
   >
   > If you are using a **distributed database system** and you want to retrieve a general overview, you set the member selection to *All* and you do **not** drill down by members.
   >
   > To retrieve more detailed information in a distributed database system, you can further drill down the selection by members. If a single member or a subset of members shows deviations compared to other members, you can also restrict the member selection.

3. To refresh tablespace performance data, you choose the *Apply Selection* button.
   An overview table provides information about important key figures of the tablespace activity, for example:
   - Tablespace name
   - Members
   - Buffer quality
   - Average physical read and write times
   - The number of data and index logical reads
   - The number of data and index physical reads
   - Name of the buffer pool the tablespace is assigned to
4. To display detailed data about a tablespace, you select a line in the overview table.
   The system then displays additional tab pages for details, history data, and a history graph of the performance data. For more information, see Displaying Detailed Tablespace Performance Data [page 86] and Tablespace Performance Data: History Graph [page 86].
5. To isolate the time frame during which the performance problem occurred, you use the data in the overview table on the *History* tab page.

Tablespace Performance Information Without DCF Support

The *Tablespaces* screen without DCF support provides an overview of the following tablespace activities of your database:

- *Tablespace Name*
- *Member* (distributed systems only)
- *Buffer Quality (%)*
- *Avg. Physical Read Times (ms)*
- *Avg. Physical Write Time (ms)*
- *Data Logical Reads*
- *Data Physical Reads*
- *Index Logical Reads*
- *Index Physical Reads*
- *Buffer Pool Name*

6.8.1 Displaying Detailed Tablespace Performance Data

1. On the *Tablespaces* screen, choose a line in the overview table. Depending on the setup, data has been collected as follows:
   - **With DCF support**, the displayed monitoring values refer to the time frame that was specified in the *Selection* area. The configuration values show the configuration that was valid during the specified time frame. All values are retrieved from the history that is stored by the DCF.
   - **Without DCF support**, data has been collected since the database was started. The configuration values show the current configuration.
2. In the content detail area of the *Tablespaces* screen, choose the *Details* tab page.
3. Choose the appropriate tab page to get an overview of the following critical database performance indicators:
   - Buffer Pool [page 376]
   - Group Buffer Pool (GBP) I/O [page 349]
   - Asynchronous I/O [page 378]
   - Direct I/O [page 383]
   - XML Storage [page 383]

   Buffer Pool and *Asynchronous I/O* provide information about buffer pool access.
   Direct I/O and *XML Storage* provide information about direct accesses, that is, I/O activity that does not use the buffer pool (for example, access to LONG VARCHAR columns or backup).

6.8.2 Tablespace Performance Data: History Graph

Use

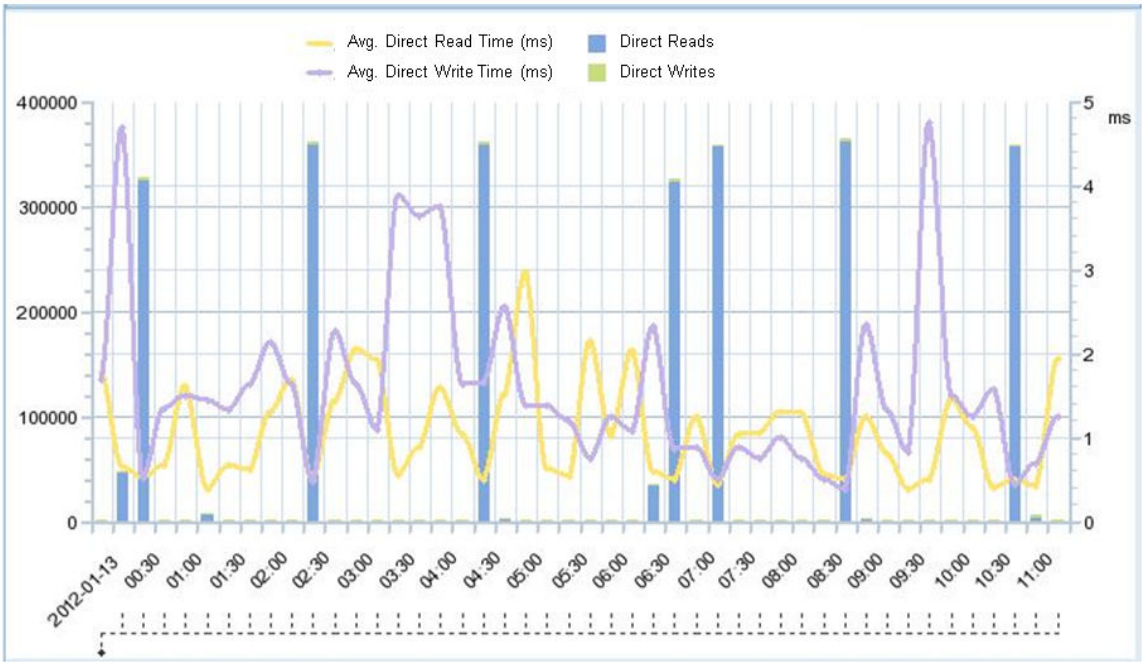
The history graph of the performance data allows you to quickly analyze the performance of tablespaces. Depending on the selected time period, you can either monitor long-term trends or perform a short-term

analysis. This includes questions such as “Has the average read/write time increased since this morning, even though the number of direct reads and writes has remained the same?”. This might be the starting point, for example, to investigate whether the hard drive on which the tablespace is stored is also used elsewhere, which affects performance.

Available History Graphs

The DBA Cockpit displays the following history graphs of the performance data for tablespaces:

- Buffer quality and the number of physical reads of data and indexes
  The graph shows the development of the buffer quality in comparison to the number of physical reads of data and indexes for the selected time period.
- Average direct read/write time and the number of direct reads/writes
  The graph shows the development of the average direct read time and write time in comparison to the number of direct reads and writes for the selected time period.



Example Graph: Average Direct Read/Write Time and the Number of Direct Reads/Writes

6.9 Performance: Network Statistics

> **i** Note
>
> The *Performance: Networks Statistics* screen is available as of Db2 9.7 FP4. For information about the required minimum SAP NetWeaver support packages to use this screen, see SAP Note 2208447.

Use

On this screen, you can analyze network statistics and get answers to the following questions:

- Are the long database response times due to long processing times on the database server or to slow communication between the database server and the SAP application server?
- How are the network statistics distributed among the SAP systems, members, and the application servers?
- How long is the network time compared to the server time?
- How long is the average network time per roundtrip and how does this metric react to changing network load?

Features

The *Network Statistics* screen provides different views of network characteristics and metrics that you can display in several ways according to the filters and drilldown options you select.

To access the screen, call the DBA Cockpit and choose *Performance* > *Network Statistics* on the *Database* tab page (web browser) or in the navigation frame (SAP GUI).

The screen is divided into a selection area and a graphic view followed by a list view.

In the *Selection* area, you specify the time frame and select filter and drilldown options for the data to be displayed. These filter and drilldown selections are valid for both graphic and list view.

In the *Graphic View*, the data is displayed in two charts according to your selected characteristics. The first chart shows the average network time per roundtrip together with the underlying network load (bytes sent and bytes received). The length of the network time compared to the server time is displayed in the second chart. Each graphic is a combined curve diagram/bar chart aligned to the timeline. The curve diagrams are displayed depending on your chosen graphic drilldown options whereas the bar charts always show the aggregate values.

In the *List View*, the data is displayed in a table according to your selected filter and drilldown options.

More Information

SAP Note 1625157

SAP Note 2208447

6.10 Performance: Costly Transactions

Use

You can use this screen to identify database transactions that are costly in terms of:

- Lock wait time
- Number of locks held
- Runtime while holding log space
- Amount of log space used

By selecting one or all of the above cost categories, you can identify units of work (UOWs) that potentially impact system performance, and obtain information such as the name of a program being executed by the UOW.

Features

To access the *Costly Transactions* screen, call the DBA Cockpit and choose *Performance* > *Costly Transactions* on the *Database* tab page (web browser) or in the navigation frame (SAP GUI).

In the *Selection* area, you can specify the time frame, display mode (aggregate or timeline), and granularity for the data to be displayed. In addition, you select one or all of the cost categories that you want to be taken into consideration. Choose *Apply Selection*, and in the list view, the data is displayed in a table according to your selection.

Which and how many transactions are considered costly depends on the collection options and threshold values of the corresponding data collector. If you want to edit these data collector options, go to the screen *Configuration* > *Data Collection Framework: Configuration* and choose the *Costly Transactions* collector.

More Information

Data Collection Framework: Configuration [page 243]

6.11 Performance: Tables

Use

You can access performance data for tables by calling the DBA Cockpit and choosing *Performance* > *Tables* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

Depending on the setup of your DBA Cockpit, one of the following is displayed on the *Tables* screen:

- Enhanced table performance information based on data from the data collection framework (DCF)
- Table performance information without DCF support

For more information about the DCF, see Enablement of Databases for the Data Collection Framework [page 48].

The *Tables* screen provides information that you can use to analyze tables under the following aspects:

1. Which tables are accessed the most and require extra tuning?
2. Which tables have frequent update operations and are, therefore, potential candidates for new statistics?
3. Which tables are candidates for reorganization due to space fragmentation?

Enhanced Table Performance Information with DCF Support

The *Tables* screen does not only display current global table data but also monitoring data that is periodically collected by the DCF.

> **i Note**
>
> To monitor data on the *Tables* screen, you have to make sure that the DCF is set up correctly. If the DCF is not available or wrongly set up, an appropriate warning is displayed including a link to the screen *Data Collection Framework: Configuration* where you can perform the required steps.
>
> For more information, see Data Collection Framework: Configuration [page 243].

Features

The *Tables* screen provides the following areas:

- A selection area where you specify filter criteria for the monitoring data (**only** available in a distributed database system)
- A result list of the retrieved monitoring data providing the most important key figures that have been aggregated over the selected time period
- A detail view of all history data that was collected for the selected monitoring data during the specified time frame
- A graphical view of the history data

Process

To analyze table performance data, you proceed as follows:

1. You identify the time period where you experienced performance problems, for example, long running background jobs, threshold violations detected by workload management, and bad user response time
2. In the *Selection* area, you specify the appropriate time frame.

> **Note**
>
> If you are using a **distributed database system** and you want to retrieve a general overview, you set the member selection to *All* and you do **not** drill down by members. To retrieve more detailed information in a distributed database system, you can further drill down the selection by members. If a single member or a subset of members shows deviations compared to other members, you can also restrict the member selection.

3. You restrict the result set in the overview table, for example, by limiting the maximum number of rows, by using wildcards, or by specifying lower limits for I/O counters.

> **Example**
>
> By using the appropriate wildcard in the *Table Name* input field, you can view special BW tables with a specific naming convention.

4. To refresh table performance data, you choose the *Apply Selection* button.
5. To analyze a single table in more detail, you select a line in the overview table and choose the *Detailed Table Analysis* button.
   You are redirected to the *Single Table Analysis* screen where detailed table analysis data is displayed.
6. To display detailed history data, you select a line in the overview table.
   The *History* tab page and the *History: Graph* tab pages appear. For more information, see Table Performance Data: History Graph [page 92].
7. To isolate the time frame during which the performance problem occurred, you use the data in the overview table on the *History* tab page.

Table Snapshot Without DCF Support

The following information is displayed:

Column	Description
Table Schema	Name of the schema
Table Name	Name of the table
Partition	Number of partition (only displayed in a multi-partition database)
Rows Written	Number of rows changed (inserted, deleted, or updated) in the table
Rows Read	Number of rows read from the table

Column	Description
Overflow Access	Number of accesses (reads and writes) to overflowed rows of the table Overflowed rows indicate that data fragmentation has occurred. If this number is high, you might be able to improve table performance by reorganizing the table using the `REORG` utility, which cleans up this fragmentation. **i Note** Pay particular attention to this column. If the value in this column is very high, consider reorganizing the table.
Page REORGs	Number of page reorganizations executed for the table Too many page reorganizations can result in less than optimal insert performance. You can use the `REORG TABLE` utility to reorganize a table and eliminate fragmentation.
Overflows Created	Number of page overflows that were automatically created A high number of overflows might become critical if there is also a high number of overflow accesses (see *Overflow Access* column).
Table Scans	Number of scans executed for the table. A high number of scans might indicate a missing index for the table or a bad access plan.

i Note

If you select a line, you are redirected to the *Single Table Analysis* screen where detailed table analysis data is displayed.

More Information

Analyzing and Maintaining a Single Table [page 152]

6.11.1 Table Performance Data: History Graph

Use

The history graph of the performance data allows you to analyze the table changes that occur over time quickly. Depending on the selected time period, you can either monitor long-term trends or perform a short-

term analysis, such as “Have there been many read accesses to the table recently?”. This might be the starting point to check whether a table reorganization, database partitioning, or a new index are necessary.

Available Graphs

The DBA Cockpit displays the following history graphs of the performance data for tables:

- Number of rows read and of rows written
  The graph shows the development of the number of rows read and of rows written for the selected time period.
- Number of rows updated, inserted, and deleted
  The graph shows the development of the number of rows updated, inserted, and deleted for the selected time period.

Example Graph: Number of Rows Read and Written

6.12 Performance: Applications

Use

The *Applications* screen provides information about applications that are currently connected to the database. The metrics and information displayed for each application only reflect the most recent unit of work of the selected application but not the metrics since the selected application connected to the database.

You can use the *Applications* screen to:

- Monitor the number and details of active and inactive applications that are connected to the database

- Analyze lock wait situations by monitoring the progress of the application that is holding the lock

Prerequisites

This screen is only available if the database configuration parameter `MON_REQ_METRICS` is set to a value other than *NONE* and the database manager configuration parameter `DFT_MON_UOW` is set to *ON*.

Features

If the currently monitored database comprises more than one member, you can specify in the *Selection* area whether you want to retrieve information from applications that are connected to any member or only to certain members by setting the appropriate filter criteria.

> **i** Note
>
> At the beginning of a unit of work (UOW), each application is assigned to a workload and therefore, also to a service class. This assignment is reevaluated at the beginning of the next UOW. If you want to monitor applications that are assigned to a specific service class, for example, the ABAP dialog service class, set a filter accordingly in the *Selection* area.

In addition, you can specify if you want to display information about Db2 autonomic processes (for example, STMM or Event Monitor) by selecting the *Show DB2 Autonomics and Utilities* checkbox.

If you apply your selections, the monitoring data is refreshed and the following information is displayed:

Column	Description
Application Handle	A system-wide unique ID for the application in a distributed database system This ID is the same on every member where the application can span sub-agents.
Coord. Member	ID of the coordinating member of the application
Application State	State of the application
Application Name	Name of application running at the client as known to the database manager or DB2 Connect
UOW Runtime	Total runtime of the current unit of work (UOW) in milliseconds
Entity	Type of the entity (`db2agent`, `db2fmp`, or name of the system entity)
Client PID	Process ID of the client application that made the connection to the database
Client Workstation	Client workstation as specified in the TPMON special register

Column	Description
Client User ID	Client user ID as specified in the TPMON special register. In an SAP ABAP system, the client user ID corresponds to the SAP user name.
Client Application	Client application as specified in the TPMON special register In an SAP ABAP system, the client application is specified as follows: • `ptype BTC` for SAP batch work processes • `ptype SPO` for SAP spool work processes • `ptype UPD` or `ptype UPD2` for SAP update work processes • the SAP transaction code for SAP dialog work processes
Service Superclass	Name of the service superclass to which the coordinating agent is currently assigned
Service Subclass	Name of the service subclass to which the coordinating agent is currently assigned

You can set a filter on the retrieved performance data by specifying the following in the *Details* area:

> **i** Note
>
> In the web browser-based version of the DBA Cockpit, the numbers displayed behind the selectable characteristics indicate the number of applications, agents, or activities that are currently in the respective state.

Displaying Details About Applications

If you choose an entry from the result list, more detailed information about an application is displayed on the following tab pages in the content detail area:

- *Unit of Work*
- *Current Activities*
- *Assigned Agents*
- *Group Buffer Pool (GBP) IO*

Unit of Work

On the *Unit of Work* tab page, you can find details about the unit of work (UOW) that the application is currently executing on the database system. The pie chart indicates where this UOW has spent most of its time since it was started by the respective application.

> **i** Note
>
> If the selected UOW has just started and therefore no metrics have yet been rolled up into the UOW, the following message is displayed instead of the pie chart:
>
> `No data is available`

You can find more details in the chart area, for example, detailed aggregate metrics about all activities that were or are just running as part of this unit of work as well as details on the UOW itself and its WLM context.

Current Activities

The *Current Activities* tab page displays the activities that are currently processed as part of the current UOW. In most cases, only a single activity (identified by an activity ID) is displayed. Only in certain cases – for example, if the application has opened a cursor that uses the `WITH_HOLD` attribute – more than one activity might be displayed. If you are using a distributed database or if the activity uses intra-parallelism, you can review the status of each activity on individual members by selecting the relevant row in the table. The details shown also include the SQL statement text of the activity.

In the SQL statement display area, you can get two types of access plan for the statement: You can either trigger a new calculation of the access plan by choosing the *EXPLAIN* pushbutton, or you trigger a lookup in the SQL cache by choosing the *EXPLAIN from Cache* pushbutton.

> **i** Note
>
> If the UOW is inactive, the name of the tab page *Current Activities* changes to *Last Activities*.

Assigned Agents

The *Assigned Agents* tab page displays details about all database agents that are currently assigned to the selected application. In a distributed database or if the activity uses intra-parallelism, you can check the status of each individual agent that is currently assigned to the selected application.

Group Buffer Pool (GBP) I/O

The *Group Buffer Pool I/O* tab page displays details about the I/O and buffer pool ratios of a database with the Db2 pureScale Feature. You can use this information to tune the local and global buffer pools.

More Information

Tracing, Capturing, or Canceling an Activity [page 96]

6.12.1 Tracing, Capturing, or Canceling an Activity

Use

On the *Applications* screen, you can trace, capture, or cancel an activity if you discover, for example, that the activity has been running for a long time.

> **i** Note
>
> These functions are only available for the Web browser-based version of the DBA Cockpit.

Procedure

Tracing an Activity

To trace and analyze the progress of a unit of work, select the application handle in the overview table and choose the *Trace Activity* pushbutton.

The *Application Activity Trace* starts in a separate Web browser providing detailed information about the selected application handle. By default, the screen is refreshed every 5 seconds. You can choose different refreshing intervals by pausing the trace, specifying a different time interval, and resuming the trace.

After the screen has been refreshed, all agents that are currently assigned to the respective application handle are displayed in the *Assigned Agents* table. Especially in a distributed database, or if the activity uses intra-parallelism, you can find out whether the progress of an activity is slowed down by a certain subagent. To identify the reason for this slowdown, check the event-related and request-related information in the *Assigned Agents* table.

The *Unit of Work Progress* table grows as the trace continues to collect information about activities of the respective application handle.

> **i** Note
>
> - The *Unit of Work Progress* table is sorted by time in descending order, that is, new data is inserted at the top of the table.
> - You can configure metrics of interest to be displayed by choosing the *Open Dialog Settings* pushbutton.
> - In a distributed database environment or if the activity uses intra-parallelism, you can filter data by member.

To review the status of all assigned agents at a certain point in time, you have to pause the trace. After selecting the respecting point in time in the *Unit of Work Progress* table, details about the assigned agents of that point in time is again displayed in *Assigned Agents* table. The trace is automatically paused as soon as the application disconnects from the database system or as soon as this window is closed.

Capturing an Activity

Capture detailed information about the currently executed activity of an application as follows:

1. Select the application and choose the *Capture Activity* pushbutton.
2. In the top-level navigation bar, choose *Performance* > *Critical Activities* > *Threshold Violations*.
   On the *Threshold Violations* screen, activities that have been captured manually are displayed with the value *MANUAL* in the *Predicate* column.

   > **i** Note
   >
   > To find manually captured activities more easily, use the filter function of the list viewer.

Canceling an Activity

To cancel an activity, select the application and choose the *Cancel Activity* pushbutton.

If the activity is successfully canceled, the SQL error `SQL4725N` with status `SQLSTATE 57014` is returned to the canceled application.

6.13 Performance: Containers

Use

You can use the *Containers* screen to gain an overview of the performance of containers in your database.

Prerequisites

This screen is only available if the following applies:

- The database configuration parameter `MON_OBJ_METRICS` is set to a value other than *NONE*.
- The data collection framework (DCF) is set up correctly. If the DCF is not available or not correctly set up, a warning is displayed including a link to the DCF configuration screen where you can perform the necessary steps.
  For more information, see Data Collection Framework: Configuration [page 243].

Features

To access the container snapshot, call the DBA Cockpit and choose *Performance* > *Containers* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

In the *Selection* area, you can specify filter criteria for the collected data. Using the *Sort By* field, you can choose if the containers are sorted by direct reads or by accessibility.

The retrieved history data is displayed in an overview table according to your selection criteria.

The following table provides explanations for some columns and values on the *Containers* screen (self-explanatory columns are not included):

Column	Description
Availability	Indicates whether the container is available Possible values are: • *1*: Container is available. • *0*: Container is not available.
Direct Writes	Number of write operations that do not use the buffer pool for the container
Direct Reads	Number of read operations that do not use the buffer pool for the container

Column	Description
Direct Read Time (ms)	Elapsed time for direct reads in milliseconds
Direct Write Time (ms)	Elapsed time for direct writes in milliseconds
Est. Time per Write (ms)	Estimated time per write operation (direct writes and direct write time) in milliseconds
Pages Read	Number of pages read from the container

6.14 Performance: Top SQL Statements

> **i Note**
>
> This screen is only available if the database configuration parameter `MON_ACT_METRICS` is set to a value other than *NONE* and if the monitored database is Db2 9.7 or higher.

You can access the *Top SQL Statements* screen by calling the DBA Cockpit and choosing *Performance* > *Top SQL Statements* on the *Database* tab page (Web browser) or in the navigation frame of the DBA Cockpit (SAP GUI).

The *Top SQL Statements* screen displays information about SQL statements that are executed very often or that are expensive with regard to specific metrics. This information helps you identify those SQL statements that consume many resources. You can also determine whether tuning of those statements is necessary to improve the performance of the database.

> **i Note**
>
> The data for this screen is captured by the *Package Cache* data collector of the DCF. This data collector periodically dumps the most expensive SQL statements and their metrics into the history tables. To reduce the amount of data stored in the history tables, the data collector performs a top n analysis and only stores the data of those statements that fall into the top n category with respect to a variety of important metrics.
>
> Therefore, the data provided on this screen, does **not** represent a 100% view of the SQL statements that are executed on the database server.

In the *Selection* area, you specify the appropriate time frame. The result is an aggregated view of the metrics for the SQL statements captured in the specified time frame.

To limit the result set, you can specify ranking or filter criteria in the *Selection* area.

In the *Selection* area, you can also define the maximum number of returned SQL statements. The default limit is 25 SQL statements. By default, static packages are not displayed as the SAP system uses dynamic SQL only. If you want to analyze static packages, you have to select the *Display Static Packages* checkbox.

If you choose the *Apply Selection* pushbutton, the result set is determined according to your selection criteria.

The following information is displayed:

Column	Description
Statement Text	Text of the SQL statement
<ranking metric>	The first metric column is used for ranking as chosen in the *Selection* area from the *Top SQL Statements by* dropdown list. By default, *Coord. Statement Exec. Time* is used.
No. of Coord. Executions	Number of executions by coordinator agent
Avg. Coord. Stmt. Exec. Time	Average execution time in milliseconds for statement by coordinator agent
Total CPU Time	Total CPU time in milliseconds
Rows Read / Rows Processed	Ratio of rows read from the base table compared to rows processed, which can be either rows returned to the application (`SELECT` statements) or rows written (`UPDATE`, `INSERT`, `DELETE` statements) A value of *1* indicates optimal access to the requested data. High values indicate statements with inefficient access.
BP Gets / Rows Processed	Average number of pages read from the buffer pool per rows processed, which can be either rows returned to the application (`SELECT` statements) or rows written (`UPDATE`, `INSERT`, `DELETE` statements)

Displaying Details About SQL Statements

To display detailed information about an SQL statement, select a line from the result list. The *SQL Statement Details* area appears displaying the following tab pages:

- *SQL Statement*
- *Metrics Summary*
- *Metrics Details*
- *Access Plan Versions*

SQL Statement

On the *SQL Statement* tab page, the complete SQL statement text is displayed. To show the current access plan of the statement, you can choose the *EXPLAIN* pushbutton.

Alternatively, you can choose the *EXPLAIN from Cache* pushbutton to try a lookup of the access plan in the current SQL cache (package cache). This will only be successful if the corresponding entry can still be found in the SQL cache.

For more information, see EXPLAIN Function [page 297]. To display the ABAP source code from where the SQL statement was executed, you can choose the *Show Source* pushbutton.

> **i Note**
>
> The *Show Source* pushbutton is only available if the following applies:
>
> - The SAP system on which the currently monitored database is running is an **ABAP** system.
> - The SAP system has the minimum Support Package level as described in **SAP Note** 1493490.

In contrast to the analysis of the SQL cache, the compilation environment is **not** available at this level because you see an aggregation of metrics over a time frame that could have multiple compilations of the SQL statement.

For more information about the SQL cache, see Performance: SQL Cache (DB2 V9.7 or Higher) [page 103].

Metrics Summary

The *Metrics Summary* tab page provides the chart view *Time Spent in SQL Statement*, which is an aggregated view of the time frame defined in the *Selection* area. In addition, detailed information about the following is displayed:

- Statement metadata
- Miscellaneous metrics
- Sorts
- Non-prefetch I/O
- Prefetcher I/O
- Locking
- Logger
- Buffer pool
- Direct I/O
- Fast communication manager

Metrics Details

The *Metrics Details* tab page provides the selection of specific time slices. For each time slice, the same detailed information is displayed.

Access Plan Versions

The *Access Plan Versions* tab page provides an overview of the different access plans generated for the SQL statement in the history. The first number in the brackets represents the number of access plan versions in the selected time frame. The second number in the brackets is the number of access plan versions in the complete history.

> **i Note**
>
> On this tab page, different compiled access plans to different points in time are displayed. The access plan version is a virtual version number that you use to distinguish the versions in the complete collected history. If an SQL statement is flushed from the package cache and is recompiled at a later point in time, a new access plan version is considered even if DB2 has generated exactly the same access plan. You can use the estimated query cost as an indicator for similar access plans.

Displaying Details About Access Plan Versions

To display detailed information about an access plan, select a line from the list on the *Access Plan Versions* tab page. The *Access Plan Details* area appears displaying the following tab pages:

- *SQL Statement*
- *Metrics Summary*
- *Metrics Details*

SQL Statement

On the *SQL Statement* tab page, the SQL statement text and the compilation environment are displayed. To show the access plan of the statement, you can choose the *Historic EXPLAIN* pushbutton. “Historic” here means that it shows the access plan captured when the access plan was generated. For more information, see *The EXPLAIN Function*.

> **i** Note
>
> The *Historic EXPLAIN* is only available if the corresponding option *Collect Section EXPLAIN* is enabled for the DCF data collector *Package Cache*. In the SAP Default or SAP Detailed templates this option is not enabled.

The *Compilation Environment* area displays information, for example, about the following:

- Isolation
- Query optimization
- SQL rules
- Refresh age
- Resolution time stamp

Metrics Summary

The *Metrics Summary* tab page provides the chart view *Time Spent in SQL Statement*. It is an aggregated view over the timeframe defined in the selection area and the access plan version selected. In addition, detailed information about the following is displayed:

- Statement metadata
- Miscellaneous metrics
- Sorts
- Non-prefetch I/O
- Prefetcher I/O
- Locking
- Logger
- Buffer pool
- Direct I/O

Metrics Details

The *Metrics Details* tab page provides the selection of specific time slices for the access plan version. For each time slice, the same detailed information is displayed.

6.15 Performance: SQL Cache

Use

The *SQL Cache* screen displays information about SQL statements that are executed very often and stored in the SQL cache of your system. This information helps you identify those SQL statements that consume many resources. You can also determine whether fine-tuning of those statements is necessary to improve the performance of the database.

Prerequisites

This screen is only available if the database configuration parameter `MON_ACT_METRICS` is set to a value other than *NONE*.

Features

Depending on your system, the screen can give you a wide range of information, which can lead to a large result set.

In the *Selection* area, you can specify ranking and filter criteria and define the maximum number of returned SQL statements to limit the result set. By default, static packages are not displayed as the SAP system uses dynamic SQL only. If you want to analyze static packages, select the *Display Static Packages* checkbox.

If you apply your selection, the result set is displayed in a table as follows:

> **i** Note
>
> Self-explanatory columns are not described in the following.

Column	Description
<ranking metric>	The first metric column is used for ranking as chosen in the *Selection* area from the *Top SQL Statements by* dropdown list. By default, *Coord. Statement Exec. Time* is used.
No. of Coord. Executions	Number of executions by coordinator agent
Avg. Coord. Stmt. Exec. Time	Average execution time in milliseconds for statement by coordinator agent

Column	Description
Rows Read / Rows Processed	Ratio of rows read from the base table compared to rows processed, which can be either rows returned to the application (`SELECT` statements) or rows written (`UPDATE`, `INSERT`, `DELETE` statements) A value of *1* indicates optimal access to the requested data. High values indicate statements with inefficient access.
BP Gets / Rows Processed	Average number of pages read from the buffer pool per rows processed, which can be either rows returned to the application (`SELECT` statements) or rows written (`UPDATE`, `INSERT`, `DELETE` statements)

You can display detailed information about an SQL statement by selecting a line from the result list. The *SQL Statement Details* area appears with the following tab pages:

- *SQL Statement*
  On this tab page, the SQL statement text and the compilation environment are displayed. To show the access plan of the statement, you can choose the *EXPLAIN* button to trigger a new calculation, or choose the *EXPLAIN from Cache* button to trigger a lookup from the SQL cache. For more information, see EXPLAIN Function [page 297].
  To display the ABAP source code from where the SQL statement was executed, you can choose the *Show Source* button.

  > **i Note**
  >
  > The *Show Source* button is only available if the following applies:
  >
  > - The SAP system on which the currently monitored database is running is an ABAP system.
  > - The SAP system has the minimum Support Package level as described in **SAP Note** 1493490.

  In the *Compilation Environment* area, information about, for example, the resolution time stamp, isolation, query optimization, SQL rules, and refresh age is displayed.
- *Statement Metrics*
  This tab page provides the chart view *Time Spent in SQL Statement* (only in the Web browser-based version of the DBA Cockpit).
  In addition, detailed information about the following is displayed:
  - Statement metadata
  - Miscellaneous metrics
  - Sorts
  - Locking
  - Logger
  - Buffer pool
  - Direct I/O

  Additional metrics for prefetcher I/O and non-prefetch I/O are available. For more information, see the documentation for the screen *Performance* > *Applications*, where the same metrics for prefetcher I/O and non-prefetch I/O are displayed (see Application Details: Unit of Work [page 398]).
- *Group Buffer Pool (GBP) I/O*
  The *Group Buffer Pool I/O* tab page displays details about the I/O and buffer pool ratios of a database with the pureScale Feature. You can use this information to tune the local and global buffer pools.

6.16 Performance: Memory

Use

The *Memory* screen provides you with an overview of the memory consumers and their distribution in the database. You can use the *Memory* screen to track and analyze memory consumption.

Prerequisites

Make sure that in the DCF, the data collector *Memory* is set up correctly.

Features

You can access the *Memory* screen by choosing *Performance* > *Memory* on the *Database* tab page of the DBA Cockpit (Web browser) or in the navigation frame of the DBA Cockpit (SAP GUI).

Based on your time frame selection, the following information is displayed about the memory consumers:

- Configured size
- Current size
- High-water mark

To identify the consumers, use the columns *Memory Group*, *Memory Pool*, and *Memory Level*.

If you choose a memory consumer from the list, the *Memory Details* area appears. Here, you can obtain information about the history of a memory consumer.

6.17 Performance: Transaction Log

Use

On the *Transaction Log* screen, you can display key figures related to transaction logging for a monitored database.

Features

You can access the *Transaction Log* screen by choosing *Performance* > *Transaction Log* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

You can enter a time frame and select one or multiple database members. Based on your selection, the DBA Cockpit displays key figures related to transaction logging such as information about log space, log reads, or log writes.

To display transaction log details for a member, you can select a line in the displayed list.

The *Details* area appears displaying the following:

- Additional key figures related to transaction logging on the *Details* tab page
- History data for these key figures on the *History* tab page
- A graphical view of the history data for selected key figures on the *History: Graph* tab page

Advanced Log Space Management (ALSM)

For the monitoring of the Db2 Advanced Log Space Management (ALSM), which is available as of Db2 11.5 Mod Pack 4 Fix Pack 0, there's a dedicated screen area on the *Transaction Log* screen. For more information, see the section on ALSM in our Database Administration Guide for SAP on Db2 for Linux, UNIX, and Windows and the SAP blog post on ALSM on SAP Community.

More Information

For more information about the individual key figures displayed on the screen, see Transaction Log: Field Help [page 106].

For more information about the screen layout and navigation, see Web Browser-Based User Interface (Web Dynpro) [page 23].

6.17.1 Transaction Log: Field Help

The following table gives you an overview and explanation of fields and field values that are available on the *Transaction Log* screen.

> **i Note**
>
> Self-explanatory fields are omitted from the table.

Field Labels, Field Values, and Their Explanations

Field Label	Explanation
No. of Indoubt Transactions	The number of outstanding indoubt transactions in the database. Indoubt transactions hold log space for uncommitted transactions, which can cause the logs to become full.
Available Log Space	The amount of active log space in the database that is not being used by uncommitted transactions
Log Read Time	The total elapsed time spent by the logger reading log data from the disk
Log Write Time	The total elapsed time spent by the logger writing log data to the disk
First Active Log File	The file number of the first active log file. Use the number of the first active log file, the last active log file, and the currently active log file to determine the range of active log files. Knowing the range of active log files helps you determine the disk space required for log files.
Last Active Log File	The file number of the last active log file
Currently Active Log File	The file number of the active log file the DB2 database system is currently writing.
Method 1: Status of Most Recent Archiving Attempt	The result of the most recent log archive attempt. Possible values are: • `1` for success • `0` for failure • `N/A` if not applicable
Method 2: Status of Most Recent Archiving Attempt	The result of the most recent secondary log archive attempt. Possible values are: • `1` for success • `0` for failure • `N/A` if not applicable

For more information about the key figures used for the *Transaction Log* screen, see the IBM Db2 documentation for the MON_GET_TRANSACTION_LOG table function .

6.18 Performance: Active Utilities

On the *Active Utilities* screen, you can check if the database performance is currently influenced by utilities such as `RUNSTATS` or `REORG`.

Depending on the utility, performance can decrease due to one of the following reasons:

- Reduced I/O performance because a utility performs much physical I/O.
- Locking situations on database resources such as tables that are locked by an offline `REORG`.

The following information is displayed for all running utilities:

Column	Description
Partition	Number of the partition on which the utility is running
Type	Type of utility that is running, for example: • `ASYNC_INDEX_CLEANUP` • `BACKUP` • `CRASH_RECOVERY` • `LOAD` • `REBALANCE` • `REDISTRIBUTE` • `REORG` • `RESTART_RECREATE_INDEX` • `RESTORE` • `ROLLFORWARD_RECOVERY` • `RUNSTATS`
Description	Describes the utility and what it is working on, for example, whether a table is being processed by a `RUNSTATS`.
Start Time	Date and time when the utility started.

Column	Description
Processing Mode	Defines how to interpret the progress list that is shown in the content detail area after you have selected a line in the overview table. The following values are possible: • *Concurrent* Due to concurrent processing, any element in the progress list can be updated at any time. • *Serial* Interprets the elements in the list as a set of serial phases. This means that completed work must equal the total work for element n before the completed work of element n+1 is first updated. This attribute is used to describe the progress of a task that consists of a set of serial phases where a phase must fully complete before the next phase begins. • *<space>* The utility is running as one single phase without parallelism.
Sequence Number	Current phase the utility is working on
Invoker	Indicates how the utility was started. The following values are possible: • *AUTO* The utility was automatically started by DB2. • *USER* The utility was started manually.
Priority	Specifies the relative importance of a throttled utility. The higher the priority, the more you might experience an impact on overall database performance.
State	Shows the current execution state of the utility: • *ERROR* The utility is stopped due to an unexpected error situation. • *EXECUTE* The utility is executing. • *WAIT* The utility is waiting, for example, due to blocked resources or synchronous I/O requests.

Displaying Details About Utilities

If a utility is in concurrent or serial processing mode, you can choose the utility in the overview table. The *Utility Details* area appears providing information about all phases of the selected utility. The *Utility Data* subarea provides a summary of important indicators of the overview table, such as *Utility ID* or *Invoker*.

The *Running Phases* table provides the following information:

Column	Description
Partition	Number of the partition where the utility phase is running
Start Time	Date and time when the utility phase started
Sequence Number	Number of the utility phase
Description	Description of the action performed in this phase
Progress (%)	Work progress in percent (that is, the percentage of completed units compared to the total number of units)
Completed	Number of units that have been processed
Total	Total number of units that need to be processed by the utility phase
Work Metrics	Metrics of the work units The following values are possible: • *NOT_SUPPORT* • *BYTES* • *EXTENTS* • *INDEXES* • *PAGES* • *ROWS* • *TABLES*
State	Shows the current execution state of the utility phase

6.19 Performance: Utility History (Db2 10.1 or Higher)

Use

The *Utility History* screen provides an overview of past database utility runs such as reorganizations, backups, restores, or `RUNSTATS`. You can use the information on this screen to find out, for example, which database utility was executed on your database during a specific time period.

You can monitor the execution of the following utility types:

- `BACKUP`
- `LOAD`
- `MOVETABLE`
- `REDISTRIBUTE`

- `REORG`
- `RESTORE`
- `ROLLFORWARD`
- `RUNSTATS`

More Information

- Displaying Past Utility Runs Using the Utility History [page 111]
- Utility History: Field Help [page 112]

6.19.1 Displaying Past Utility Runs Using the Utility History

Prerequisites

This function is only available if the monitored database is Db2 10.1 or higher and the data collection framework (DCF) has been set up correctly (`Change History` data collector). If the DCF is not available or wrongly set up, a warning is displayed including a link to the DCF configuration screen, where you can perform the required steps. For more information, see Data Collection Framework: Configuration [page 243].

Procedure

1. To call up the utility history, choose *Performance* > *Utilities* > *Utility History* on the *Database* tab page (Web browser) or in the navigation frame of the DBA Cockpit (SAP GUI).
2. In the selection area, you can enter a time frame, a utility, and an object name.

   > **Example**
   >
   > You can choose **`Last 7 Days`** as time frame, **`RUNSTATS`** as utility type, and the **`EXAMPLETABLE`** table as object name. The DBA Cockpit then populates the overview table according to your selection criteria.

   For more information about the fields and the field values in the overview table, see Utility History: Field Help [page 112].
3. To display the Db2 command that was used to call up the utility, select a line in the overview table. The DBA Cockpit displays the Db2 command in the *Utility History Details* screen area.
4. You can display additional columns by customizing your layout using the *Open Settings Dialog* button.

6.19.2 Utility History: Field Help

Use

The following table gives an overview and an explanation of fields and field values that are available on the *Utility History* screen.

> **i** Note
>
> Self-explanatory fields are omitted in the table.

Features

Column	Description
Utility Type	Type of utility that ran on the database during the selected time frame. Possible values are: • `BACKUP` • `LOAD` • `MOVETABLE` • `REDISTRIBUTE` • `REORG` • `RESTORE` • `ROLLFORWARD` • `RUNSTATS`
Object Type	The type of object for which the utility is executed. Possible values are: • `DATABASE` • `INDEX` • `PARTITIONGROUP` • `TABLE` • `TABLESPACE`

Column	Description
Utility Operation	Details about the types of operations that the utility performs. Possible values for `BACKUP` are: • `D`: Delta • `I`: Incremental • `F`: Full Possible values for `LOAD` are: • `I`: Insert • `R`: Replace • `S`: Restart • `T`: Terminate Possible values for `MOVETABLE` are: • `A`: Cancel • `C`: Copy • `I`: Init • `L`: Cleanup • `M`: Move • `R`: Replay • `S`: Swap • `V`: Verify Possible values for `REDISTRIBUTE` are: • `A`: Abort • `C`: Continue • `D`: Default • `T`: Target Map Possible values for `REORG` are: • `A`: Reorganize all table indexes • `I`: Index reorganization • `N`: Inplace table reorganization • `R`: Reorganize table reclaim extents • `T`: Classic table reorganization Possible values for `RESTORE` are: • `A`: Incremental automatic • `B`: Incremental abort • `F`: Full • `M`: Incremental manual

Column	Description
	Possible values for `ROLLFORWARD` are: • `E`: End of logs • `P`: Point in time Possible values for `RUNSTATS` are: • `A`: All indexes on a table • `I`: Index • `T`: Table
Invoker	Indicates whether the utility was started by a user or automatically. Possible values are: • `AUTO` • `USER`
Details	Details about what the utility performs, including some options specified for the utility or partial command strings

6.20 Performance: Utility History (Db2 9.7 or Lower)

Use

> **i** Note
>
> This function is only available for the Web browser-based version of the DBA Cockpit.

The *Utility History* screen provides an overview of logged database utility operations for a specific time frame that were recorded in the database history file (`DB_HISTORY`). Using the data provided on this screen, you can analyze operations that were performed on your database system for a specific time period.

> **i** Note
>
> The *Utility History* screen is **only** available if the DCF has been set up correctly. If the DCF is **not** available or wrongly set up, an appropriate warning is displayed including a link to the DCF configuration screen where you can perform the required steps.
>
> For more information, see Data Collection Framework: Configuration [page 243].
>
> As of Db2 10.1 for LUW, this screen is obsolete and has been replaced by a new *Utility History* screen (see Performance: Utility History (Db2 10.1 or Higher) [page 110]).

To retrieve the required data, specify a time frame in the *Selection* area and apply your selection. The following information is displayed:

Column	Description
Start Time	Start time of a logged event
End Time	End time of a logged event
Operation	Identifier of the logged operation. For more information, see DB_HISTORY administrative view - Retrieve history file information , *Table 2. OPERATION and OPERATIONTYPE values* in the IBM Db2 documentation.
Operation Type	Action identifier for a logged operation that depends on the operation run For more information, see DB_HISTORY administrative view - Retrieve history file information , *Table 2. OPERATION and OPERATIONTYPE values* in the IBM Db2 documentation.
Object Type	Identifier for the type of the target object. The following values are possible: • *D* Stands for full database • *P* Stands for tablespace • *T* Stands for table
Schema	Schema of the target object
Object	Name of the target object
Event Info	Additional event information for the operation
Event ID	Number that uniquely identifies an entry in the history file

Displaying Details About Logged Events

To display detail information about a logged event, select a line in the overview table. In the *Utility History Details* area, additional event attributes are indicated as a formatted record string. This detail area also provides further internal records, such as the event duration or utility ID (if available). You can display additional fields by customizing your layout using the *Open Settings Dialog* pushbutton.

More Information

For more information about logged events in the database history file, see DB_HISTORY administrative view - Retrieve history file information in the IBM Db2 documentation.

6.21 Performance: Inplace Table Reorganization

Use

The *Inplace Table Reorganization* screen provides an overview of inplace table reorganizations that are currently running or that have been interrupted. Inplace table reorganization allows you to access tables while they are being reorganized.

> **i** Note
>
> In contrast to the *Active Utilities* screen, the *Inplace Table Reorganization* screen provides a more detailed view of reorganizations. In addition, not only currently active reorganizations are displayed but also the DB2 history file for finished reorganizations is retrieved.

The information is displayed in the following table:

Column	Description
Table Schema	Table schema of the table that is currently being reorganized
Table Name	Name of the table that is currently being reorganized
Partition	Number of the partition (only displayed in a multi-partition database)
REORG Status	Status of the inplace table reorganization Possible values are: • *Running* • *Paused* • *Suspended* • *Completed*
Progress %	Progress of the reorganization
Start Date	Start date of the inplace table reorganization
Start Time	Start time of the inplace table reorganization
Access Mode	Access mode for other users while the table reorganization is running The following access modes are possible: • *READ* • *WRITE* • *NO ACCESS*

Column	Description
Tablespace	Name of the tablespace where the reorganization is performed

> **i Note**
>
> If no active inplace table reorganization was found, the system displays the following message: `No Inplace Table Reorganizations are running`. Only the `REORG` activities since the database start are displayed. `REORG`s that were active before the database start are not displayed.

Depending on your requirements, you can customize the view on the *Inplace Table Reorganizations* screen using the following functions:

Pushbutton	Function
Choose View	• *Active Only* Only inplace table reorganizations with the status *Started* or *Paused* are displayed. • *All* All inplace table reorganizations are displayed including those with status *Completed* or *Suspended*.
Choose Data Source	• *Since DB Start* Only inplace table reorganization that have been started after the last restart of the database manager are displayed. • *Incl. History File* This option additionally reads the DB2 history file. Thus, the data of inplace table reorganizations that were started before the last restart of the database manager is also retrieved.

Activities

You can perform the following actions for an inplace reorganization:

- *Pause*
  Select a running inplace reorganization and choose *Pause*.
- *Resume*
  Select a paused inplace table reorganization and choose *Resume*.
- *Suspend*
  Select any inplace table reorganization and choose *Suspend*.

As a result of any of these actions, the list of active inplace table reorganizations is refreshed.

6.22 Performance: System Resources

The *System Resources* screen provides you with an overview of all resources that are available for each member of the currently monitored database system.

You can access information about system resources by calling the DBA Cockpit and choosing *Performance* > *System Resources* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

On the *System Resources* screen, the following information is displayed:

Column	Description
Host Name	Name of the host
OS	Operating system of the host
Members	Members located on this host
CPU Usage (%)	Current overall CPU usage of the host in percent
CPU Load Short	**i Note** This information is **only** available on UNIX operating system platforms. Short-period CPU load, for example, load samples over the last 5 minutes
CPU Load Medium	**i Note** This information is **only** available on UNIX operating system platforms. Medium-period CPU load, for example, load samples over the last 10 minutes
CPU Load Long	**i Note** This information is **only** available on UNIX operating system platforms. Long-period CPU load, for example, load samples over the last 15 minutes
CPUs Online	Number of CPUs online
Memory Total (MB)	Total size of physical memory in MB
Memory Free (MB)	Amount of free physical memory in MB

6.23 Performance: Performance Warehouse

Use

> **i** Note
>
> This function is only available for the Web browser-based version of the DBA Cockpit.

You can analyze performance data of your database system using the Performance Warehouse. To access the Performance Warehouse, call the DBA Cockpit and choose *Performance* > *Performance Warehouse*.

The following content areas are available in the Performance Warehouse:

- *Reporting*
  By default, the *Reporting* content area is displayed.
- *Configuration*

Prerequisites

An SAP Solution Manager system with Solution Manager Diagnostics (SMD) enabled is required.

Integration

The Performance Warehouse is part of the DBA Cockpit.

Features

In the Performance Warehouse, all relevant performance indicators that are collected by the DBA Cockpit are stored in an SAP Business Warehouse system. This SAP BW system is used by the Solution Manager Diagnostics (SMD) back end of an SAP Solution Manager system. SMD already uses this SAP BW to store workload data of SAP applications. To configure the extraction of data into the SMD BI, you use the SMD Setup Wizard.

Based on this architecture, the DBA Cockpit uses SAP BW technology to provide reports for performance analysis, which you can customize according to your needs. All collected data has a time dimension, so you can analyze the database performance for any point in time or over a specified time frame.

Almost all reports are displayed as a chart to visualize the key performance indicators (KPIs). In addition, there is also a detailed table view. To navigate within these reports, you can use the SAP BW drilldown feature. Violations to performance thresholds are highlighted based on predefined BW exceptions to make you immediately aware of performance issues.

By default, the Performance Warehouse is delivered with predefined content that you can use to create your own reports according to your needs.

More Information

- *Performance Warehouse: Reporting*
- *Performance Warehouse: Configuration*

6.23.1 Performance Warehouse: Reporting

> **i** Note
>
> This function is only available for the Web browser-based version of the DBA Cockpit.

You use the data provided on the *Reporting* screen to analyze database performance problems in the present or the past. To access the *Reporting* screen of the Performance Warehouse, call the DBA Cockpit and choose *Performance* > *Performance Warehouse* > *Reporting* on the *Database* tab page of the DBA Cockpit.

Specifying the Time Frame

To display detailed reports, you first have to specify the time frame for which you want to analyze data by defining the following:

- *Granularity*
  You can choose between *Minute*, *Hour*, *Day*, or *Month*. Depending on your selection, the values for your time frame might change.
- *Time Frame*
  If you choose *Custom Selection* from the dropdown list, you can manually enter the starting and ending time for your analysis. To activate your custom selection, choose *Apply Filter*. For any other selection from the dropdown list, the reports are automatically refreshed.

The reports are categorized and for each category there is one tab page. On every tab page, you find a button row for the reports. Every button in the button row represents a specific view on the database performance, for example, *I/O*, *Prefetcher*, *Sort Heap*, and so on.

Displaying a Report

To display a report, choose the appropriate view button on the respective tab page.

> **Note**
>
> The availability of the tab pages and of the buttons on each tab page can vary depending on the selected system. Some reports are only available if special database features are enabled.

The reports consist of two sections:

- In the upper section, a chart is displayed to visualize the key performance indicators. The chart provides a subset of the key columns from the detail table view.

  > **Note**
  >
  > The chart display is optional and not available for all views.

- In the lower section, a detailed table view is available.

You can drill down your reports by either using the context menu of a column header in the *Detail: <Category – View>* screen area or by specifying the respective value using the buttons in the *Detail: Navigation* screen area. Here, you can also add and remove columns or key figures, or you can set filters on columns.

In addition, there are predefined exceptions (for example, *Chart: Exceptions* or *Details: Exceptions*) for almost all reports on key performance indicators. The used thresholds are based on Early Watch Alerts and each violation to these thresholds is displayed in red.

> **Note**
>
> If you want to reset a report to its initial state, choose *Reset Report* in the central system area.

6.23.2 Performance Warehouse: Configuration

> **Note**
>
> This function is only available for the Web browser-based version of the DBA Cockpit.

On the *Configuration* screen, you configure all configuration parameters that are related to the performance warehouse. You can configure, for example, the framework, the templates used for the reports, and the report categories.

The DBA Cockpit uses BI Business Explorer (BEx) Web templates to analyze the performance data that is stored in the Solution Manager Diagnostics (SMD) BI. You can create your own BI BEx Web templates based on this data and integrate new BI BEx Web templates into the performance warehouse.

You can access the *Configuration* screen of the performance warehouse by calling the DBA Cockpit and choosing *Performance* > *Performance Warehouse* > *Configuration*.

On the *Performance Warehouse: Configuration* screen, the following tab pages are available:

- *Configuration*
- *Web Reports*
- *Report Categories*

Configuration

Here, you can view or modify the configuration parameters of the performance warehouse for the monitored system. To modify some of these parameters, use the *Edit*, *Save*, and *Cancel* buttons in the toolbar.

Depending on your database platform, the displayed selection of values can vary. The following parameters are displayed for all database platforms:

Parameter	Description
BI Server	BI server where the database performance data is located
Reporting Time Zone	The performance data time-stamps are converted to one global time zone for all reports in SMD BI.

> **i** Note
>
> The *Default* checkbox is selected if the default value for your complete landscape is the same as the one specified for your system.

Web Reports

Here, you can configure the display on the *Reporting* screen. That is, you can view and modify the integrated BI BEx Web templates in the tree table. To modify some of these parameters, use the *Edit*, *Add*, and *Delete* buttons in the toolbar.

The main report categories appear and for each report category, you can view or modify the views by expanding the appropriate report category. These views appear as buttons on the respective category tab page on the *Reporting* screen. To change the sequence within a category, use the *Up* or *Down* buttons.

To display details about a view, simply select it in the table. The following parameters are displayed in the *Details for Web Reports* area below the table view:

Parameter	Description
Report	Specifies the name of the report This text appears on the view button on a category tab page.
Description	Detailed description for the report This text appears as a tooltip for the button of the key indicator on a category tab page.
Category	Specifies the report category Each category is represented on a separate tab page.

Parameter	Description
Web Report (Default)	Technical name of the BI BEx Web templates
Web Report (Day)	Technical name of the BI BEx Web templates for granularity *Day*
Web Report (Month)	Technical name of the BI BEx Web templates for granularity *Month*
Data Providers (Time)	Specifies the data provider of the BI BEx Web templates with a time dimension The drilldown of the time dimension is changed according to the selected granularity.
Active	If selected, the report is available for performance analysis.
Default	If selected, the report is executed as soon as the tab page is selected.
Database	Name of database platform
Release (min)	Minimum database release for this report
Release (max)	Maximum database release for this report

> **i** Note
>
> Depending on your database platform, there might be more checkboxes available for database-specific features. If you select these checkboxes, the report is only displayed if the specific database features have been set up in the monitored database system.

Report Categories

Here, you can view and modify the categories for BI BEx Web templates of the reports that are displayed the *Reporting* screen. To modify some of these parameters, use the *Edit*, *Add*, and *Delete* buttons in the toolbar. To change the sequence of the categories on the *Reporting* screen, use the *Up* or *Down* buttons.

The following parameters are displayed:

Parameter	Description
Category	Name of the category
Description	Detailed description for the category

6.24 Performance: Workload Statistics

> **i** Note
>
> This screen is only available if the currently monitored database version is at least DB2 V9.5 and if you are using the Web browser-based version of the DBA Cockpit.

You can access information about workload statistics by calling the DBA Cockpit and choosing *Performance* ▸ *Workload Statistics* on the *Database* tab page of the DBA Cockpit.

On the *Workload Statistics* screen, basic statistics are displayed that describe the overall behavior of the database system. This information helps you to:

- Understand the type of a system slowdown. For example, most of the queries are running slower versus only certain queries are running slower.
- Monitor service level agreements (SLAs)
- Search for time frames, members, or service classes in which long-running queries are executed on the database system
- Define meaningful workload management thresholds using the high-water mark metrics

In the *Selection* area, you can specify the appropriate time frame for which you want data to be displayed.

> **i** Note
>
> In a distributed database system, you can also filter by members and service classes.

After having made your selection using the *Apply Selection* pushbutton, the *Activity Lifetimes* and *Statistics* tab pages are displayed.

Activity Lifetimes

The histograms that you can choose from the *Histogram Type* dropdown list give you a better understanding of your database workload.

The following histogram types are available:

- *Activity Execution Time Histogram*
  Displays the history of the histogram for the execution times of all activities that were executed in the selected service classes and members for the selected time frame
  The execution time corresponds to the time an activity has spent executing on the coordinator partition. Execution time does not include the time spent during initialization, queuing, or between cursor operations.
- *Activity Lifetime Histogram*
  Displays the history of the histogram for the lifetimes of all activities that were executed in the selected service classes and members for the selected time frame
  The lifetime corresponds to the total elapse time of an activity, as measured on the coordinator partition. Lifetime does include the time spent during initialization, queuing, or between cursor operations.

- *Activity Queue Time Histogram*
  Displays the history of the histogram for the queue times of all activities that were executed in the selected service classes and members for the selected time frame
  The queue time corresponds to the time an activity has spend in a workload management (WLM) queue.

The average execution, life, or queue time for all activities is displayed in a dotted line. The total number of activities that have completed, were rejected, or aborted by WLM in the selected time frame is displayed in an additional chart below the histogram chart.

For more information about the histogram types and all other monitoring counters provided here, see the IBM Db2 documentation at https://www.ibm.com/docs/en/db2 .

Statistics

The *Statistics* tab page provides you with a range of high-watermark and average metrics that are separated by service class, member, and time. By default, the following information is displayed:

Column	Description
Time from	Begin of statistics interval timestamp
Time to	End of statistics interval timestamp
Member	Member that has captured the statistics
Service Superclass	Name of the service superclass for which the statistics were captured
Service Subclass	Name of the service subclass for which the statistics were captured
Coord. Act. Lifetime Top	High-water mark in milliseconds for coordinator activity lifetime
Coord. Act. Lifetime Avg.	Arithmetic mean of lifetime in milliseconds for coordinator activities
UOW Total Time Top	High-water mark in milliseconds for unit of work lifetime
Act. Temp. Tablespace Top	High-water mark in KB for the temporary tablespace usage of a single DML activity
Agg. Temp. Tablespace Top	High-water mark in KB for the aggregate temporary tablespace usage across all DML activities
Activity CPU Time Top	High-water mark for processor time used by activities at all nesting levels, in milliseconds
Rows Read Top	High-water mark for the number of rows read by activities on all nesting levels

Column	Description
Rows Returned Top	High-water mark for the number of rows returned by activities on all nesting levels

For more Information about all available statistics, see the IBM Db2 documentation at https://www.ibm.com/docs/en/db2 .

6.25 Performance: HADR

Use

You can use the *HADR* screen to monitor the current and past status of standby databases including multiple standby databases with replay delay. This screen is relevant if you have a high availability disaster recovery (HADR) strategy in place and you want to monitor the status of your standby databases.

Prerequisites

This screen is only available if the following applies:

- The currently monitored database is DB2 for LUW version 10.1 or higher, and it runs in HADR mode.
- The data collection framework (DCF) is set up correctly.
  If the DCF is not available or not correctly set up, a warning is displayed including a link to the DCF configuration screen where you can perform the necessary steps. For more information, see Data Collection Framework: Configuration [page 243].

Features

To access the *HADR* screen, choose *Performance* > *HADR* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

Based on your time selection, the DBA Cockpit displays information for each standby database such as the following, for example:

- Synchronization mode
- HADR log wait times
- Standby replay delay
- Standby replay gap time and volume

In the *HADR Details* area, you can display the following:

- More HADR details on the *Summary* tab page
- History data on the *History* tab page
- A graphical view of the history data

For more information about the screen layout and navigation, see Web Browser-Based User Interface (Web Dynpro) [page 23] and SAP GUI-Based User Interface [page 20].

6.25.1 HADR: Field Help

Metrics on the *HADR* screen and their descriptions are listed in the following table.

> **i Note**
>
> Self-explanatory metrics are not listed.

Metric	Description
Avg. HADR Log Gap	Average log gap in bytes between the primary database and the standby database
Max. HADR Log Gap	Maximum log gap in bytes between the primary database and the standby database
Avg. Standby Receive Replay Gap	Recent average gap (in bytes) between the standby log receive position and the standby log replay position. It generally does not exceed the sum of `STANDBY_RECV_BUF_SIZE` plus `STANDBY_SPOOL_LIMIT`. A slightly higher value is possible due to the flexibility in buffer and spool management. When the gap reaches the combined buffer and spool limit, the standby receive buffer stops receiving logs, which will block the primary database in peer state. The standby database may also run out of buffer and spool space if the reported receive-replay gap is smaller than the sum of buffer and spool because a partial page can be sent multiple times and occupy multiple pages of space in the buffer (but always one page in the spool). However, the log gap calculation does not take multiple send operations into account.
Send Buffer Requested	Requested socket send buffer size in bytes. If there is no request, the value is *0*. We recommend that you use the system default.
Send Buffer Actual	Actual socket send buffer size in bytes, which can differ from the requested size.

Metric	Description
Receive Buffer Requested	Requested socket receive buffer size (registry variable `DB2_HADR_SORCVBUF`) in bytes. If there is no request, the value is *0*. We recommend that you use the system default.
Receive Buffer Actual	Actual socket receive buffer size in bytes, which can differ from the requested size.
Avg. Standby Receive Buffer Used	Average use of the standby receive buffer in percent. If spooling is enabled, the standby receive buffer can continue to receive logs even if it is fully (that is, 100%) used.
Max. Standby Receive Buffer Used	Maximum use of the standby receive buffer in percent. If spooling is enabled, the standby receive buffer can continue to receive logs even if it is fully (that is, 100%) used.
Heartbeat Interval	Indicates how often the primary and standby exchange monitor information. The interval is indicated in milliseconds and computed from various factors like `HADR_TIMEOUT` and `PEER_WINDOW`.
Peer Wait Limit	Peer wait limit (in milliseconds) that is set using the registry variable `DB2_HADR_PEER_WAIT_LIMIT`.
Peer Window	Value of the `HADR_PEER_WINDOW` database configuration parameter in milliseconds
Reads on Standby Enabled	Indicates whether read operations on the standby database are enabled.
Standby Spool Limit	Indicates the maximum number of pages to spool that are set using the `HADR_SPOOL_LIMIT` configuration parameter on the standby database. The value *0* means that spooling is disabled and *-1* stands for *no limit*.

6.26 History: Database

> **i Note**
>
> This screen has been superseded by the enhanced Performance: Database [page 76] screen under *Performance* > *Database*.

Prerequisites

This screen is only available if you have selected the *Enable Deprecated Data Collection Additionally (Not Recommended)* checkbox on the *System Configuration* screen (see Editing the Monitored System Landscape on the System Configuration Screen [page 57]).

For more information about how history data is evaluated, see Transition from the Deprecated Collection of Space and Performance History Data to the Data Collection Framework [page 55].

Features

The system provides a day-by-day trend analysis of database activity. You can check the workload of the days and display the workload peak of a single day.

To access the *History: Database* screen, call the DBA Cockpit and choose *Performance* > *History* > *Database* in the navigation frame of SAP GUI-based user interface of the DBA Cockpit.

An overview of the following performance key figures for all days monitored is displayed:

Column	Description
Partition	Monitored partition (only displayed if you are using a multi-partition database)
Date	Day when monitoring was performed
Avg. Phys. Read Time (ms)	Average physical read time
Avg. Phys. Write Time (ms)	Average physical write time
Data Logical Reads	Number of read accesses to data in the buffer pool
Data Physical Reads	Number of read accesses to data on disk: The value includes the number of physical reads that were performed synchronously (by the database manager agents) and asynchronously (by the I/O servers).
Data Physical Writes	Number of write accesses to data on disk: The value includes the number of physical writes that were performed synchronously (by the database manager agents) and asynchronously (by the page cleaners).
Index Logical Reads	Number of read accesses to index data in the buffer pool

Column	Description
Index Physical Reads	Number of read accesses to index data on disk: Since index data is read only by the database manager agents, this value includes the number of synchronously read index pages.
Index Physical Writes	Number of write accesses to index data on disk: The value includes the number of physical writes that were performed synchronously (by the database manager agents) and asynchronously (by the page cleaners).
COMMIT Statements	Total number of COMMIT statements that have been attempted
ROLLBACK Statements	Total number of ROLLBACK statements that have been attempted: Automatic ROLLBACKs caused by error situations or deadlocks are not included.
Lock Waits	Total number of times that applications or connections waited for locks
Lock Wait Time (ms)	Total elapsed time in milliseconds waited for a lock
Deadlocks	Total number of deadlocks that have occurred
Lock Escalations	Number of times that locks have been escalated from several row locks to a table lock
Exclusive Lock Escalations	Number of times that locks have been escalated from several row locks to one exclusive table lock or that an exclusive lock on a row caused the table lock to become an exclusive lock

Note that all but the first four figures listed in the table result from counting actions or events in the database. The physical read and write times are averages.

If you choose *Total Day* in the *Workload* field in the *Selection* area, the total workload of the relevant day is displayed. The values displayed are calculated as follows:

- For the average physical read/write times, the average of all measurements taken on that day is calculated.
- For all other figures, the difference between the lowest and the highest value of one day is calculated (the delta value), except if there is a database restart. If the database was restarted during a day, the absolute value of the last measurement is displayed for that day.

Database restarts are taken into consideration.

If you choose *Peak* in the *Workload* field in the *Selection* area, the value is calculated as follows:

- For average read/write times, this is the worst measured read/write time within the chosen day.

- For all other figures, the maximum delta value of all measured values is displayed.

You can display details for specific time intervals by clicking a field or selecting a row. A detail area appears with the following tab pages:

- *Snapshot*
  The measured values of the selected day are displayed.
- *Interval*
  The delta values of the measurements that are provided under *Snapshot* are displayed. (This only applies for the figures that are the result from counting actions or events in the database.)

> **i** Note
>
> If you select a day with a database restart, however, the time interval containing the restart is treated differently. For this interval, the absolute values of the first measurements after the restart are displayed.

If you select a particular day, snapshots of the database activity are displayed in a two-hour cycle.

6.27 History: Tables

> **i** Note
>
> This screen has been superseded by the enhanced Performance: Tables [page 90] under *Performance* > *Tables*.

Prerequisites

This screen is only available if you have selected the *Enable Deprecated Data Collection Additionally (Not Recommended)* checkbox on the *System Configuration* screen (see Editing the Monitored System Landscape on the System Configuration Screen [page 57]).

For more information about how history data is evaluated, see Transition from the Deprecated Collection of Space and Performance History Data to the Data Collection Framework [page 55].

Features

The system provides a day-by-day trend analysis of table activity. You can access the *History: Tables* screen by calling the DBA Cockpit and choosing *Performance* > *History* > *Tables* in the navigation frame of the SAP GUI-based user interface of the DBA Cockpit.

An overview of the monitored days is displayed:

Column	Description
Table Schema	Name of the schema to which the table belongs
Table Name	Name of the table
Partition	Partition number
Rows Written	Number of rows written
Rows Read	Number of rows read
Overflow Accesses	Number of read accesses to tables that resulted in overflow pages, that is, to records that were swapped from their original page. **i Note** If there is a high number of overflow accesses in comparison to total read accesses, the table is a candidate for reorganization.
Page REORGs	Number of internal page reorganizations during INSERT operations

i Note

If you want to display history data that is cumulated by day, week, or month, choose the relevant option from the dropdown list in the *Statistics* field in the *Selection* area.

6.28 Monitoring Data on Performance Screens Using the Deprecated Data Collection

Context

The following comments only apply to screens in the *Performance* task area that use the deprecated collection of space and performance history data instead of the data collection framework (DCF) as source of the displayed metrics. In these cases, screens using the deprecated collection of space and performance history data are displayed by the DBA Cockpit (see also Collection of Space and Performance History Data (Deprecated) [page 52]).

Issue and Suggested Solution

In the *Performance* task area, with the deprecated collection of space and performance history data, the various DB2 snapshots provide monitoring data in both unfiltered and raw format. Snapshot data is provided

on different levels (for example, database, buffer pools, or SQL statements) and is collected by the database beginning from startup of the database instance until the current time. That is, data is accumulated over a long time period and, therefore, cannot reflect short-term decreases in performance or increasing or changing workloads over a certain time period.

Furthermore, for the interpretation of snapshot data, it is necessary to refer to related configuration parameters. Configuration parameters, however, do **not** relate to a certain time frame either but only describe the current configuration of the database. Especially if your database is enabled for self-tuning memory management (STMM), there is no way of correlating bad performance data with current storage parameters.

To overcome these problems, you can observe monitoring data over a certain time period as follows:

You can explicitly set a starting point for measurements using the *Reset* and *Since Reset* pushbuttons that are available for some snapshot options. As a result, a set of monitoring data is displayed that is related to the time frame defined by the reset operation.

This applies to the following screens:

- *Database*
- *Buffer Pools*
- *Tablespaces*
- *Applications*

In parallel, you can check related configuration parameters for changes by choosing *Configuration* > *Configuration History*.

> **i Note**
>
> If the relevant screen supports the evaluation of data that is periodically collected by the data collection framework (DCF), these problems do not arise: You can specify any time interval for analysis as long as the relevant data is kept in the DCF history. Since the displayed configuration data is also retrieved from the DCF, there is a consistent, time-related view of both the snapshot and the configuration.

7 Space

7.1 Space: Automatic Storage

Use

You can access information about automatic storage file systems of the database by calling the DBA Cockpit and choosing *Space* > *Automatic Storage* on the *Database* tab page of the DBA Cockpit (Web browser) or in the SAP GUI menu of the DBA Cockpit. The *Automatic Storage* screen appears displaying all the storage groups and storage paths that are available for the storage management of the database.

With Db2 10.1, the concept of multiple storage groups was introduced that allows for multi-temperature storage. Storage groups are named sets of storage paths where data can be stored according to its temperature. The temperature of data is based on its relevance and on how often it is accessed. Hot data is frequently accessed and updated, cold data is rarely accessed and updated. Subsequently, you can use faster, more expensive storage devices for hot data and slower, less expensive storage devices for cold data, which reduces your overall storage costs and increases system performance. To move the hot, warm, and cold data automatically to the appropriate storage device, you can create storage groups and assign storage paths to the relevant storage group. Depending on your database version, the following information is displayed in a table:

Column	Description
Storage Group Name	Name of the storage group This information is only displayed if your Db2 version is at least 10.1. For databases up to and including Db2 9.7, there is only one implicit storage group that is not externalized by Db2.
Partition	Number of database partition
Storage Path	Full path name
State	State of the path The following values are possible: • *Not in Use* • *In Use* • *DROP Pending*
FS ID	ID of the related file system
Available Size (GB)	Free size in GB that is available in the file system

Column	Description
Total Size (GB)	Total size in GB that is available in the file system

If you choose an entry from the list, you can get more details about the technical attributes and the storage paths in the *Automatic Storage Details* area.

More Information

- Automatic Storage: Maintaining the List of Storage Paths [page 135]
- Automatic Storage: Maintaining Storage Groups [page 136]

7.1.1 Automatic Storage: Maintaining the List of Storage Paths

Procedure

If you want to add or delete a storage path to or from a storage group, choose the *Change* pushbutton. The *Change Automatic Storage* dialog window appears.

You can do one of the following:

- Add one or more storage path(s) as follows:
  1. Enter the complete name of each storage path that you want to add and press `Enter` or choose the *Add* pushbutton.
  2. To verify your entries, you can use the *Check* pushbutton. In case of errors, the relevant message or SQL statement is displayed.
  3. Save your entries.
- Delete a storage path as follows:
  1. Select a storage path and choose the *Delete* pushbutton.
  2. Save your changes.

In the *Generated SQL Statement* area, all the generated SQL statements to be executed are displayed. This area is automatically filled and refreshed as soon as you apply changes to the storage path list.

> **i** Note
>
> You can enable a database, which was originally created without automatic storage, for automatic storage management by adding storage paths on the *Automatic Storage* screen. This means that after having created the storage path(s), you can convert non–automatic storage tablespaces to automatic storage tablespaces as described in Space: Tablespaces [page 138].

More Information

- Space: Automatic Storage [page 134]

7.1.2 Automatic Storage: Maintaining Storage Groups

On the *Automatic Storage* screen, you can add, change, or delete storage groups.

Procedure

Adding Storage Groups

1. Choose the *Add* pushbutton.
2. In the *Add Storage Group* dialog window, fill in the name of the new storage group and add or delete the required storage paths as needed. If you want this storage group to be the default storage group, select the relevant checkbox.
3. To verify your entries, you can use the *Check* pushbutton. In case of errors, the relevant message or SQL statement is displayed.
4. To save the storage group, choose the *Execute* pushbutton.

Changing Storage Groups

1. Choose the *Change* pushbutton.
2. In the *Change Automatic Storage* dialog window, add or delete storage paths as needed.
3. To verify your entries, choose the *Check* pushbutton. In case of errors, the relevant message or SQL statement is displayed.
4. Save your changes.

Deleting Storage Groups

Select the storage group to be deleted, choose the *Delete* pushbutton and confirm the deletion.

More Information

Space: Automatic Storage [page 134]

7.2 Space: Database

Use

You can use the *Database* screen to access information about space consumption of your database. Monitoring the space that is consumed by the database does not only require that you check the current space consumption but also that you consider the progress of the space growth to answer questions such as the following:

- Is data continuously growing and is it predictable that the database runs out of space?
- Did archiving operations or reorganizations successfully reduce the space consumption?
- Can I identify any short-term growth that is related to specific activities of an application, such as client copy or mass inserts of data to SAP Business Warehouse?

Prerequisites

The data collector framework (DCF) is set up correctly. If the DCF is not available or wrongly set up, a warning is displayed including a link to the *Data Collection Framework: Configuration* screen where you can perform the required steps.

For more information, see Data Collection Framework: Configuration [page 243].

Features

You can access information about space consumption of your database by calling the DBA Cockpit and choosing *Space* > *Database* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

The *Database* screen provides the following:

- A selection area where you specify filter criteria for the monitoring data
- An overview table displaying the retrieved monitoring data and providing the space consumption at the end of the selected time period
- An option to break down all history data that was collected for the selected monitoring data during the specified time frame

Process

To analyze the data provided on this screen, you proceed as follows:

1. You identify the time period where you expect major changes in space consumption.

2. In the *Selection* area, you specify the appropriate time frame.
3. In case of a multi partitioned system, you can do one of the following:
   - To retrieve a **general snapshot**, you set the partition selection to *All* and you do **not** drill down by partitions or tablespaces.
   - To retrieve a more **detailed snapshot**, you drill down the selection by partition or by tablespace.

   > **i** Note
   >
   > If you want to drill drown by tablespace, you can restrict the result set by limiting it to special tablespaces.
4. To refresh the monitoring data, you choose the *Apply Selection* pushbutton.
5. To display detailed history data, you select a line in the overview table. The data is displayed in the *History Details* content area.
6. To isolate the time frame during which the problem occurred and to analyze the space growth in more detail, you use the data displayed in the *History Details* content area.

7.3 Space: Tablespaces

You can access information about space for tablespaces by calling the DBA Cockpit and choosing *Space* > *Tablespaces* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

During the installation of your SAP system, you specified one of the following options for the maintenance of tablespaces:

- *Automatic Storage*
  DB2 automatically allocates and extends tablespace containers in the file system.
- *DMS/SMS Tablespaces*
  You manually allocate containers for tablespaces. The extension of the corresponding containers can be performed either manually or automatically.

Depending on your choice, the corresponding tab page is available on the *Tablespaces* screen.

If you are using tablespace pools, all tablespaces that belong to a tablespace pool are displayed on the *Automatic Storage* tab page. For more information about tablespace pools, see SAP Note 2267446.

Automatic Storage Tablespaces

If your database is enabled for automatic storage management, the following information is displayed about all tablespaces that are part of automatic storage management:

Column	Description
Tablespace Name	Name of the tablespace

Column	Description
Partition	Number of the database partition (only displayed in a multi partition database)
Contents	Contents of tablespace, for example, any data or temporary data
TS State	Status of tablespace, for example, *Normal* or *Load Pending*
KB Total	Total space in KB used by the tablespace
Page Size	Size of a page in bytes
No. Containers	Number of containers
KB Free	Total amount of free space
High-Water Mark (KB)	Indicates the maximum value of used pages reached
Percent Used	Used space in relation to available space
Pending Free Pages	Number of free pages that are pending

DMS/SMS Tablespaces

Regardless whether you have chosen automatic storage management tablespaces or manual maintenance of DMS/SMS tablespaces during the SAP system installation, the following information is displayed for all DMS/SMS tablespaces that are maintained manually:

Column	Description
Tablespace Name	Name of the tablespace
Partition	Number of the partition (only displayed in a multi partition database)
TS Type	Type of tablespace, for example, DMS or SMS
Contents	Contents of tablespace, for example, any data or temporary data
TS State	Status of tablespace, for example, *Normal* or *Load Pending*
KB Total	Total space in KB used by the tablespace
Page Size	Size of a page in bytes
No. Containers	Number of containers
KB Free	Total amount of free space

Column	Description
High-Water Mark (KB)	Indicates the maximum value of used pages reached
Percent Used	Used space in relation to the available space
AUTORESIZE	Indicates if the tablespace is enabled for automatic resizing
Pending Free Pages	Number of free pages that are pending

Displaying Technical Details of a Tablespace

> **i** Note
>
> The following information applies to automatic storage management and DMS/SMS tablespaces.

To display detail information about the tables or indexes of a tablespace, choose a line in the overview table. In the *Tablespace Details* area of the *Tablespaces* screen, the following tab pages are displayed:

- *Technical Settings*
- *Storage Parameters*
- *Containers*

In addition, the following information is displayed for each selected tablespace:

Field	Description
Tablespace Maintenance	
Name	Name of the tablespace
Database Partition Group	Name of the partition group where the selected tablespace is defined A partition group defines a set of partitions.
Space	
Total	Total space in KB This information is not displayed when creating tablespaces.
Used	Fill level of the selected tablespace as a percentage This information is not displayed when creating tablespaces.
Free	Free space in KB This information is not displayed when creating tablespaces.

Technical Settings

The following information is displayed on the *Technical Settings* tab page:

Field	Description
Contents	Type of data that is stored in the tablespace, for example, regular data, large objects, temporary user objects, or temporary system objects
Size of I/O Units	Page and extent size in KB, and the number of pages to be prefetched
Space Management by	Displays if the space of the tablespace containers is managed by, for example, the database (DMS), the file system (SMS), or automatic storage management **i Note** • If the *AUTORESIZE Enabled* checkbox is selected, tablespace containers are automatically extended using the file systems where the containers are located. If you are using automatic storage tablespaces, the *AUTORESIZE Enabled* checkbox is selected by default and **cannot** be deselected. • If the *Reclaimable Storage Enabled* checkbox is selected, the tablespace is enabled for reclaimable storage. This means that unused extents can be released to the system for reuse. Tablespaces that have been created with DB2 V9.7 or higher do have this property. The storage group of the tablespace is displayed if the space of the tablespace is managed by automatic storage management.
Disk Performance	Information about disk performance, such as: • *Overhead* Displays I/O controller overhead and disk seek and latency time in milliseconds. This value is used to determine the cost of I/O during query optimization. • *Transfer Rate* Time to read one page into memory in milliseconds. This value is used to determine the cost of I/O during query optimization. • *File System Caching* Indicates the use of file system caching as follows: • *YES* File system caching has been explicitly enabled using the `FILE SYSTEM CACHING` clause of the `CREATE` or `ALTER TABLESPACE` statement. • *NO* File system caching has been explicitly disabled using the `FILE SYSTEM CACHING` clause of the `CREATE` or `ALTER TABLESPACE` statement. • *OS* No explicit file system caching clause has been specified. Therefore, the default of the file system is used.
Recovery	Dropped tables in the specified tablespace may be recovered using the `RECOVER TABLE ON` option of the `ROLLFORWARD` command.

Field	Description
Buffer Pool	By default, the buffer pools are displayed that match the page size of the tablespace. If required, you can add a new buffer pool. For more information, see Maintaining Buffer Pools [page 226].

For more information about technical settings, see the IBM documentation *SQL Reference*.

Storage Parameters

For tablespaces that are completely managed by automatic storage management or that have at least *AUTORESIZE* enabled, the following values are displayed on the *Storage Parameters* tab page:

Field	Description
Settings	
Initial Size	Initial space allocated when a tablespace is created
Size	
Current Size	Displays the current size
Last Resize	Date and time of last automatic resize operation
Last Resize Terminated with SQL Error	Date and time when the automatic resize operation failed **i Note** This field **only** appears if the last automatic resize operation failed. The SQL error is displayed in the lower half of the *Tablespaces* screen.

Containers

This tab page provides information about the containers of the selected tablespace.

Tablespace Map (Web Browser-Based Version of the DBA Cockpit Only)

The *Tablespace Map* tab page provides a graphical overview of the tablespace, including its stripe sets and the containers that belong to a stripe set. The tablespace map offers you a simple overview of the size of all containers and how containers are allocated to stripe sets. Such a graphical overview is useful in the following cases, for example:

- Is data evenly distributed, and are all containers well balanced?
- How can storage space be expanded? Where can a container or a stripe set be added without incurring the cost of rebalancing the containers?

When you move the mouse cursor over a container, the DBA Cockpit displays details such as the lowest and highest extent of the container. When you click a container in the tablespace map, the DBA Cockpit displays additional details of the container, such as the container name and the range numbers for the container in the stripe set.

You can use the *Display/Hide Table* button to hide or display a table view of the table space. For each range of the table space, the following details are displayed:

- Stripe set number
- Extent number of the start stripe of the range
- Extent number of the end stripe of the range
- Number of containers covered by the range

For more information about using the tablespace map, see Example: Using the Tablespace Map to Check the Balancing of Storage Space [page 143].

Displaying the Content of a Tablespace

To display detail information about the tables or indexes of a tablespace, select a line in the overview table and choose the *Contents* pushbutton. The following information is displayed in the *Tables of a Tablespace* content detail area:

Column	Description
Tablespace Name	Name of the tablespace
Schema	Name of the schema
Name	Name of the table or index
Type	Type of object, for example, index, primary index, or table

Maintaining Tablespaces

You can maintain tablespaces, for example, *Change*, *Add*, or *Delete* tablespaces.

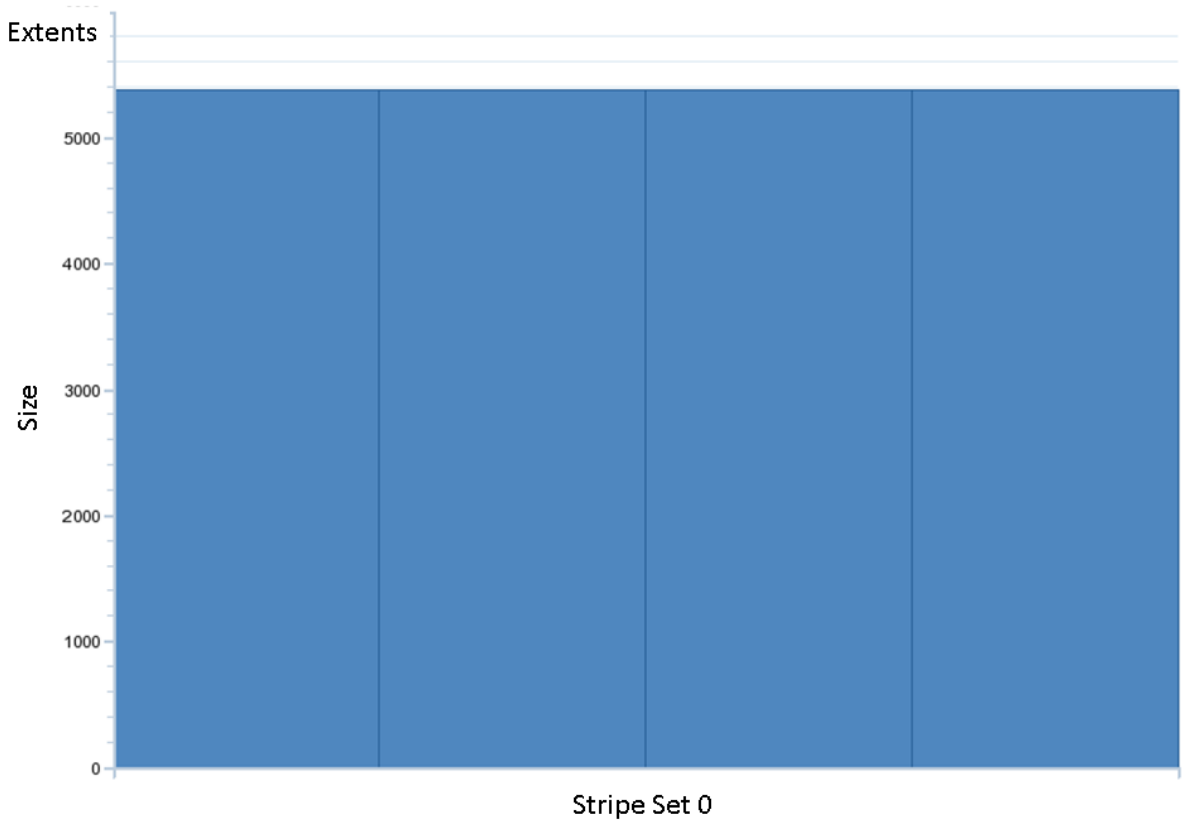
For more information, see Maintaining Tablespaces [page 145].

7.3.1 Example: Using the Tablespace Map to Check the Balancing of Storage Space

You can use the tablespace map to check whether storage space is well balanced in a tablespace.

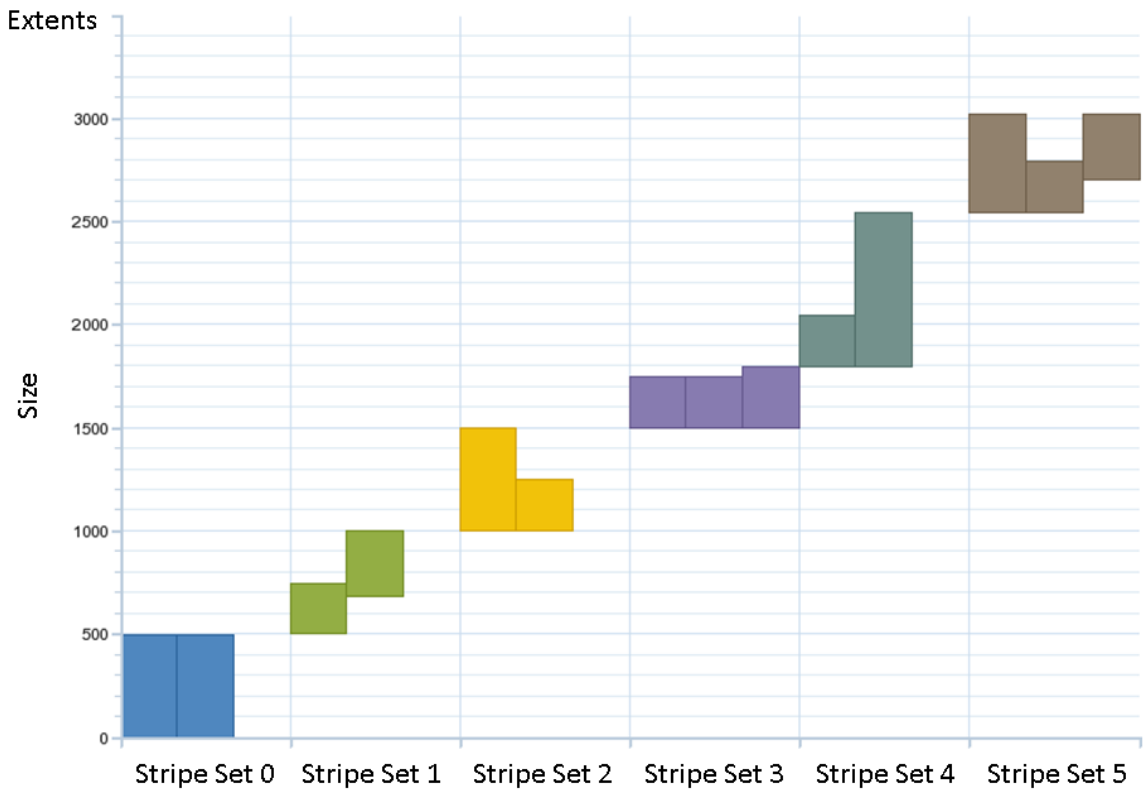
> **i Note**
>
> The tablespace map is only available for the Web browser-based version of the DBA Cockpit.

The following tablespace maps are examples of well-balanced and problematic storage distributions in tablespaces:



Example of a Well-Balanced Tablespace

In this example, you can clearly see that all storage space is evenly distributed across four containers in one stripe set.



Example of a Tablespace With Possible Issues

In this example, the display of the tablespace map indicates the following issues:

- The containers have varying sizes.
- Containers in one stripe set start at different ranges.
- Some stripe sets contains more containers than others.

7.3.2 Maintaining Tablespaces

Using the tablespace list on the *Tablespace* screen, you can maintain tablespaces as follows:

- Change the settings of tablespaces, tablespace pools, or containers
- Add new tablespaces or tablespace pools
- Delete tablespaces or tablespace pools
- Convert a regular tablespace to tablespace of type LARGE
- Start and stop the reduction of the high-water mark as well as rebalance the tablespace.

Tablespace Pools

You can add a new tablespace pool by choosing the *Add* pushbutton and entering a name that meets the naming conventions for tablespace pools as described in SAP Note 2267446. For a SAP system with SID *ABC*, a valid tablespace pool name would be, for example, `ABC#DATA{25}`. The DBA Cockpit will recognize a valid tablespace pool name and generate the necessary set of SQL statements to create all tablespaces of the new tablespace pool.

For tablespace pools, change and delete operations always apply to all tablespaces that belong to a tablespace pool . To initiate the desired operation, you can select any tablespace of a tablespace pool from the tablespace list on the *Space* > *Tablespaces* screen and choose the relevant pushbutton. In the subsequent dialog window, you will see that the tablespace pool name is used. For all other maintenance options for automatic storage tablespaces, the operation will only be on the selected tablespace of a tablespace pool.

Procedure

Changing a Tablespace

1. On the *Tablespace* screen, select a tablespace and choose the *Change* pushbutton. The *Change Tablespace* dialog box appears where you can change the following settings:

Field	Description
Technical Settings	
Prefetch Size	Number of pages to be prefetched
Overhead	I/O controller overhead and disk seek and latency time in milliseconds You can use this value to determine the cost of I/O during query optimization.
Transfer Rate	Time to read one page in milliseconds You can use this value to determine the cost of I/O during query optimization.
Buffer Pool	Name of the allocated buffer pool **i Note** You can enter only buffer pools that match the page size of the tablespace.

Field	Description
File System Caching	You can activate or deactivate file system caching. **i Note** As long as neither the `FILE SYSTEM CACHING` nor the `NO FILE SYSTEM CACHING` clause has been specified, the default caching options of the underlying file system type are used. The first use of any of these options **explicitly** sets this value. You **cannot** fall back to the default behavior.
Dropped Tables	Selecting this checkbox enables to you recover dropped tables using the `RECOVER TABLE ON` option of the **ROLLFORWARD** command.
Storage Parameters	
Space Management by AutoStorage	Select this checkbox if you want to enable a tablespace that is not yet managed by automatic storage for automatic storage.
Storage Group Name	For tablespaces that are enabled for automatic storage, you can select the storage group to which the tablespace will be allocated.
AUTORESIZE Enabled	Enable this option if you want to allow Db2 to automatically enlarge tablespace containers using the file system where the containers are located.
Maximum Size	You can enter one of the following values: • *NONE* There is no maximum size limit. In this case, you allow Db2 to extend containers until they occupy all file systems where the containers are located • An absolute value in KB This value defines an upper threshold that shall not be exceeded by automatic extensions **i Note** This field is valid only if the *AUTORESIZE Enabled* option is selected.
Increase Size	Size in KB or in percent by which the tablespace is extended if it has become full You can specify an absolute value in KB or a relative percentage. **i Note** This field is valid only if the *AUTORESIZE Enabled* option is selected.

Field	Description
Containers (Optionally available)	If a tablespace is **not** managed by automatic storage management, you can add or delete containers as follows: • To add containers, choose the *Add Container* pushbutton. The system automatically suggests a default path where the container is located. However, you can modify that path by manually editing the line. **⚠ Caution** Adding or changing containers might result in rebalancing, which has a heavy impact on system performance. At least one container must be available for each partition. If you are using a multi partition database, you need to add containers for all partitions of the corresponding partition group. If you have to change container sizes, we recommend that you enter an appropriate value in the *Resize all containers to* field to ensure a balanced distribution of data on the different containers. **⚠ Caution** Different container sizes might result in bad performance of the database. • To delete containers, select one or more lines in the table and choose *Delete*.

2. To confirm your entries, choose the *Execute* pushbutton.

Adding a Tablespace

1. On the *Tablespace* screen, choose the *Add* pushbutton.
2. In the *Add Tablespace* dialog box, specify a name and a partition group.

> **→ Recommendation**
>
> We recommend that you use uppercase letters for the tablespace name. Using lowercase letters or special characters makes accessing the selected tablespace with the Db2 command line processor less comfortable.

3. Enter the technical settings. By default, the system displays SAP's recommendations.
4. Add containers.
   If you are using a multi partition database, you must add containers for all partitions of the corresponding database partition group.

> **⚠ Caution**
>
> This step does **not** apply to tablespaces managed by **automatic storage management**.

5. To confirm your entries, choose the *Execute* pushbutton.

Deleting a Tablespace

On the *Tablespace* screen, select a tablespace and choose the *Delete* pushbutton.

> ⚠ Caution
>
> You **cannot** delete tablespaces that are still used by the SAP system, that is, if they are related to some data class. You must delete the data class before deleting the tablespace.

Converting a Regular Tablespace to LARGE

Db2 databases use large tablespaces by default. If you still have have regular tablespaces in your database, you can convert these regular tablespaces to large tablespaces. To do so, select a tablespace in the overview table of the *Tablespace* screen and choose the *Convert to Large* pushbutton. After you have confirmed the conversion, the conversion job is scheduled as a background job in the DBA Planning Calendar.

Reducing the High-Water Mark (HWM) of a Tablespace

After major table reorganizations or deleting and archiving operations, free space in a tablespace is not necessarily released directly to the file system.

To release free space, you can lower the high-water mark as follows:

1. On the *Tablespace* screen, select a line in the overview table and choose the *Reduce HWM* pushbutton.
2. In the *Reduce High-Water Mark* dialog box, you can do one of the following:
   - To automatically determine the minimum size that can be reached, choose the *Reduce Until Minimum* radio button.
   - To manually specify the size of the free space of a tablespace to be released, choose the *Reduce* radio button.
     By default, the unit of reduction is KB. You can, however, switch to MB, GB or to %.
3. Choose the *Execute* pushbutton.
   The reduction process starts asynchronously in the background.

To monitor the progress of reduction, you can check the *TS State* column in the overview table. If the column shows the value *High-water mark reduction in progress*, select the column to see a detail view of the reduction progress.

For the automatic reduction of high-water marks, you can use a job in the DBA Planning Calendar. For more information, see Reducing Tablespace High-Water Marks on a Regular Basis [page 275].

Stopping the High-Water Mark Reduction

To stop an ongoing high-water mark reduction, select the appropriate tablespace in the overview table of the *Tablespace* screen and choose the *Stop HWM* pushbutton.

Rebalancing a Tablespace

Regular and large tablespaces that are managed by automatic storage management have to be rebalanced if the following applies:

- You created new containers on recently added storage paths.
- A container and its content has to be distributed to the remaining containers.
- If a database was converted to automatic storage by adding storage paths, the tablespaces still reside in the old containers and do not yet have containers in the automatic storage database.

For the first two cases, the rebalancing process is **automatically** started. If a database was converted to automatic storage, however, you have to **manually** start the rebalancing process by selecting a tablespace in the overview table and choosing the *Rebalance* pushbutton.

After the rebalancing process has been started, data is moved from the containers of the dropped storage paths to the containers on new storage paths. The rebalancing process runs asynchronously in the background and does not affect the availability of data.

To monitor the rebalancing process, check the *TS State* column in the overview table. If this column shows any of the values *Reverse Rebalance*, *Forward Rebalance* or *DMS rebalancer is active*, select the *TS State* column to see a detail view of the rebalancing progress.

For each partition, the following data is displayed:

Column	Description
Partition	Partition number
Tablespace Name	Name of the tablespace
Mode	The rebalancer mode can be one the following: • *No Rebalance* The rebalancer process has finished on this partition. • *Reverse Rebalance* Data movement starts at the high-water mark extent and moves in reverse order through the tablespace, ending with the first extent in the tablespace. • *Forward Rebalance* Data movement starts with the first extent in the tablespace and ends with the high-water mark extent.
Extents Processed	Number of tablespace extents that have already been moved to the new location
Extents Remaining	Number of tablespace extents that still need to be moved
Last Extent Moved	Number of the last extent that was moved
Priority	Priority with which the rebalancing process is running in the database
Restart Time	Time when the rebalancing process was restarted after it had been paused or stopped
Start Time	Time when the rebalancing process was first started
Progress (%)	Displays the progress of the reduction process in percent

More Information

Configuration: Data Classes [page 229]

SAP Note 2267446

7.4 Space: Containers

You can access information on containers by calling the DBA Cockpit and choosing *Space* > *Containers* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

The following information is displayed on the *Containers* screen:

Column	Description
Tablespace Name	Name of the tablespace
Partition	Number of the partition (only displayed in a multi-partition database)
Stripe Set	Number of the strip set the container belongs to
Container Name	Name of the container in which the tablespace is located
Type	Type of the container, for example, disk or file
KB Total	Total size of the container in KB
Pages Total	Total amount of pages
Accessible	Indicates whether the container is accessible (*YES*) or not (*NO*)
FS ID	File system ID
FS Free Size (GB)	Free space in the file system in GB

To display detail information about a container, choose a line in the overview table. For more information, see Space: Tablespaces [page 138].

Maintaining Containers

You can maintain tablespace containers by selecting a line in the table on the *Container* screen and choosing the *Change*, *Add*, or *Delete* pushbutton. For more information about how to maintain container settings, see Maintaining Tablespaces [page 145].

7.5 Space: File Systems

The *File Systems* screen helps you determine how much free space is available in your file systems to extend tablespaces.

You can access information about your file systems by calling the DBA Cockpit and choosing *Space* > *File Systems* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

The following information is displayed:

Column	Description
Partition	Number of the partition (only displayed if you are using a multi partition database)
Mount Point	Location in the OS directory structure where the file system appears
KB Total	Total size of the file system in KB
KB Used	Total amount used of the file system in KB
Percentage Used	Used percentage of total size of the file system
KB Free	Total amount free of the file system in KB
Percentage Free	Free percentage of total size of the file system in KB
File System Type	File system type
Device Name	Name of the device
Block Size	Block size of the file system

7.6 Analyzing and Maintaining a Single Table

Use

By analyzing and maintaining single tables, you can optimize tables and indexes to:

- Save disk space by releasing unused space or by activating the DB2 compression features
- Improve access to a table by keeping statistics up-to-date or by reorganizing fragmented tables or indexes

You can access detailed information about a single table and maintain table statistics by calling the DBA Cockpit and choosing *Space* > *Single Table Analysis* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

Features

The *Single Table Analysis* screen provides the following areas:

- A selection area, where you specify the schema and name of the table to be analyzed

- A summary area that provides an extract of important key figures and status information of the specified table
- Tab pages with detail information that is relevant for tables or indexes

In the summary area, the following information is displayed:

Field	Description
Size	
Total Table Size	Total table size This is the summary of all `DATA`, `LONG`, `LOB`, `XML`, and columnar data pages over all database partitions
Total Index Size	Total size of all indexes over all database partitions
Number of Indexes	Number of indexes defined on the selected table
Administrative Actions	
RUNSTATS Status	Status of the statistics The following values are possible: • *Statistics available* • *No Statistics available* • *VOLATILE Table Without Statistics* • *Unwanted Statistics for VOLATILE Table*
REORGCHK Status	Indicates whether the `DB2 REORGCHK` had findings that need to be analyzed further
Compression Status	Indicates whether and how the table is compressed **i Note** A table or index is assumed to be compressed if the following applies: • In the system catalog, the table or index is enabled for compression. • The data or index pages have been compressed by performing an appropriate reorganization.

Field	Description
Size	
Savings by Compression **i Note** The following fields are **only** relevant if the compression status is accordingly. That is, if the *Compression Status* tab page displays that a table or at least its index is compressed.	
Current Compression Ratio	Current percentage of all savings for row and index compression compared to the total size of all `DATA`, `LONG`,`LOB`, `XML`, and index objects
Savings with Current Compression	Current total savings of space This value includes all savings regardless whether they were caused by data or index compression.
Estimated Compression Ratio	Compression ratio that would be reached after compression **i Note** This value is **only** displayed if a compression check has already been performed.
Estimated Savings After New Compression	Savings that would be reached after a compression
Additional Savings After Compression	Additional savings after compression for tables that have already been compressed

The following tab pages are available on the *Single Table Analysis* screen providing detail information about tables or indexes:

- *System Catalog*
- *Table Structure*
- *Indexes*
  This tab page contains an overview list of all indexes and provides integrated tab pages with detail information about, for example, system catalog, structure and status of the index, as well as information about the compression status and `REORGCHK`.
- *Table Status*
- *Compression Status*
- *REORGCHK*
- *CREATE Statement*

Activities

To optimize tables and indexes, you can use the following pushbuttons on the *Single Table Analysis* screen:

- *RUNSTATS*
  Schedules a `RUNSTATS` job for a single table using the DBA Planning Calendar. For more information, see Updating Statistics [page 268].

  > **i Note**
  >
  > To decide if statistics are out-of-date, you can check the statistics time. Alternatively, you can count the number of entries in the table by choosing the *Count* pushbutton to compare the real number of records with the *Cardinality* field on the *System Catalog* tab page.

- *REORG*
  Schedules a `REORG` job for the table using the DBA Planning Calendar. For more information, see Reorganizing Tables [page 270].
- *Compression Check*
  Schedules a job that checks whether a table would benefit from compression

  > **⚠ Caution**
  >
  > The *Compression Check* job is expensive and can considerably decrease the system performance. We, therefore, recommend that you schedule this job at a time with a low system workload and **not** in dialog mode.

- *Compression On/Off*
  Switches the compression status of a table or index.

  > **i Note**
  >
  > The term compression is used here as a synonym for both data and index compression.

  The *Compression Options* dialog box appears displaying the current compression status of the table, that is, if compression is enabled or disabled, and if a compression dictionary exists. Depending on the status, you can choose between the appropriate actions, such as:

Action	Description
Enable Compression	Switches the compression attributes for the table and its indexes on. As of DB2 version 10, you can also specify the type of data compression.
Enable Compression and Run REORG	Switches the compression attributes on and schedules a `REORG` job that compresses the table data and its indexes. As of DB2 version 10, you can also specify the type of data compression.
Run REORG to Remove Dictionary	Schedules a `REORG` job with the `RESET DICTIONARY` option for a table for which the compression attribute was switched off

Action	Description
Disable Compression	Switches the compression attributes off
Disable Compression and Run REORG	Switches the compression attributes off and schedules a `REORG` that uncompresses table and index data
Run REORG to Rebuild Dictionary	Schedules a `REORG` job that rebuilds a dictionary on which compression is based for an already compressed table This is recommended if a new compression check indicates a better compression ratio or if the compression attributes were switched on but the table had not yet been reorganized.
Run REORG to Build Dictionary	Schedules a `REORG` job for a table where the compression attributes were switched on but the table and indexes have not yet been compressed.

> **i Note**
>
> Before you enable compression, we recommend that you check the possible savings by starting a compression check using the *Compression Check* pushbutton.

- *VOLATILE On/Off*
  Changes the `VOLATILE` attribute of a table

> **i Note**
>
> If you set the `VOLATILE` attribute, the table is no longer processed by automatic `RUNSTATS`.

7.6.1 Single Table Analysis: System Catalog

The *System Catalog* tab page provides the following information that is available from the system catalog entry of the selected table:

Field	Description
Space Management	
Data Tablespace	Name of the tablespace where the data pages of the table are stored
Index Tablespace	Name of the tablespace where the index pages of the table are stored
Free Space Reserved	Percentage of each page to be reserved for future inserts A value of *-1* indicates that the DB2 default value is used.

Field	Description
Table Organization	Indicates how the table is organized Possible values are: • *Column-Organized* • *Row-Organized* This information is available only as of database release 10.5 or higher.
Technical Attributes	
VOLATILE	Indicates whether the table is flagged as `VOLATILE` in the system catalog If the table is flagged as `VOLATILE`, statistics are not gathered by DB2's automatic `RUNSTATS`. In addition, statistics data (if available) is not used by the DB2 optimizer. **i Note** Newly created tables and tables that were dropped or re-created during an upgrade or a table conversion are always marked as `VOLATILE` as long as valid statistics available are not yet available.
Pooled, Clustered, or Import Table	**⚠ Caution** The information displayed **only** applies to SAP ABAP systems only. Indicates whether the table is defined as a pooled table, a cluster or, an import or export table in the ABAP Dictionary
Distribution Statistics	Indicates whether the table has distributed statistics If this value is selected, further details about column distribution values are available on the tab page *Table Structure*.
Statistics Data	
Last RUNSTATS	Date and time of the `RUNSTATS`
Cardinality	Number of data records in the table
Counted Rows	Number of rows that have been counted by a `SELECT(*)` statement This information is **only** displayed if you choose the *Count* pushbutton.

Field	Description
Deviation	Deviation of the number of rows provided by `RUNSTATS` in the system catalog from the number of rows provided by a `SELECT COUNT(*)` statement This information is **only** displayed if you choose the *Count* pushbutton.
Overflow Records	Number of records that have overflowed Records overflow when a data record is updated and the new data record is larger than the old one, or when a column is added to a table. The percentage of overflow records is used as a decision criterion for `REORG` (see F1 formula on the *REORGCHK* tab page)
No. of Pages With Data:	Number of pages containing data
Total Number of Pages	Total number of pages in the table
RUNSTATS Profile	If a `RUNSTATS` was executed using the `PROFILE` option, this profile is displayed. The profile is the same as the `RUNSTATS` command that was executed with the `SET PROFILE` option. **i Note** If a profile exists, it is used by automatic `RUNSTATS` and therefore the resulting statistics can deviate from the default setting for statistics. In addition, if you manually execute a `RUNSTATS`, you have to make sure that you update statistics by using the profile or by explicitly choosing another type of statistics.
Compression	
Value Compression	Indicates whether value compression is enabled for the table
Row Compression	Indicates whether the table is enabled for row compression
Compression Mode	Type of row compression (adaptive or static)
Average Length of Compressed Rows	Average length of compressed records in bytes
Average Compression Ratio by Row	Average compression ratio by row
Average Length of Compressed and Uncompressed Rows From Total Number of Rows	Average length of all rows (compressed and uncompressed) in bytes
Percentage of Compressed Rows From the Total Number of Rows	Compressed rows as a percentage of the total number of rows in the table

Field	Description
Approximate Percentage of Pages Saved	Percentage of pages saved by compression **i Note** Only data pages are taken into account.

7.6.2 Single Table Analysis: Table Structure

The *Table Structure* tab page provides the following information:

Column	Description
DB Column No.	Number of the column in the database
DB Column Name	Name of the column in the database
DB Type	Data type of the column in the database
DB Length	Length of the column in the database
Inline Length	For LOB or XML data, the maximum size of the field that can be inlined in the base table instead of storing it separately
Hidden Column	Indicates whether this is a hidden column or not

The following information is **only** displayed for the **local system** and if the table is defined as a transparent table in the ABAP Dictionary:

Column	Description
SAP Column Name	Name of the column defined in the ABAP Dictionary
SAP Key	Column is part of the primary key defined in the ABAP Dictionary.
SAP Type	Data type of the column defined in the ABAP Dictionary
SAP Length	Length of the column defined in the ABAP Dictionary
If the statistics for the table have been gathered using the `WITH DISTRIBUTION` option, the following information is displayed:	
Column Name	Name of the column to which the statistics apply

Column	Description
Seq. No	If the *Type* column contains the value *F*, the value *n* in this column identifies the `nth` most frequent value. If the *Type* column contains the value *Q*, *n* in this column identifies the `nth` quantile value.
Value	Data value as a character literal
Value Count	If the *Type* column contains the value *F*, this is the number of occurrences of the *Value* column. If the *Type* column contains the value *Q*, this is the number of rows whose value is less than or equal to the value in the *Value* column.
No. of Distinct Values	If the *Type* column contains the value *Q*, this is the number of distinct values that are less than or equal to the value in the *Value* column.
Type	Specifies how to interpret the displayed values The following values are possible: • *F* Indicates the frequency value • *Q* Quantile value

7.6.3 Single Table Analysis: Indexes

The *Indexes* tab page provides a list of all indexes that are defined on the table summarizing the following information:

Field	Description
Index Schema	Name of the index schema
Index Name	Name of the index
Unique Rule	Rules for index uniqueness The following values are possible: • *D* Permits duplicates • *U* Indicates index uniqueness • *P* Implements the primary key

Field	Description
Index Type	Type of index The following values are possible: • *BLOK* Block index • *CLUS* Clustering index that controls the physical placement of newly inserted rows • *DIM* Dimension block index • *REG* Regular index • *XPTH* XML path index • *XRGN* XML region index • *XVIL* Index over XML column (logical) *XVIP* Index over XML column (physical)
Statistics Time	Date and time of last `RUNSTATS`

Together with the summary information, there are detail tab pages displaying all information that is related to the currently selected index. By default, the data of the first index is displayed. To view the details of another index, select a table from the list.

> **i** Note
>
> If there is only one index, the summary list does not appear.

7.6.3.1 Indexes: System Catalog

The integrated *System Catalog* detail tab page on the *Index* tab page provides the following information about the selected index(es):

Field	Description
Index	
Name	Name of the index

Field	Description
Schema	Schema of the index, which is usually the user who created the index
Type	Type of index For more information, see Single Tables Analysis: Indexes [page 160].
Unique Rule	Rules for index uniqueness
Last Used	Date when the index was last used by any DML statement or when it was used to enforce referential integrity constraints **i Note** If the index is used on an HADR standby database, this column is **not** updated. The default value is *0001-01-01*. This value is updated asynchronously.
Technical Attributes	
Free Space Reserved	Percentage of free space reserved in the index pages This free space is taken into account by the `LOAD` and `REORG` utilities of DB2.
Compression	Indicates whether the index is enabled for compression
Reverse Scans Supported	Indicates whether the index supports reverse scans
Statistics Data	
Last RUNSTATS	Date and time of last `RUNSTATS`
Number of Leaves	Number of index leaves
Number of Levels	Number of index levels
Sequential Pages	Number of index leaves that are physically located on the hard disk and that are sorted by index without large intervals between them
Density	Relative density of the sequential pages as a proportion of the total number of index pages 100% is the optimum value.
Cluster Ratio	Degree of index fragmentation A value of *100%* means no fragmentation and this is the optimal value.

Field	Description
Cluster Factor	Finer measurement of the degree of clustering If statistics are not collected or if the index is defined on a nick-name, a value of *-1* is displayed.
Cardinality	Cardinality of the index This value might be different from the cardinality of the table for indexes that do not have a one-to-one relationship between the table rows and the index entries.
First Key Cardinality	Number of different values in the first column of the index
First 2 Key Cardinality	Number of different values in the first two columns of the index
First 3 Key Cardinality	Number of different values in the first three columns of the index
First 4 Key Cardinality	Number of different values in the first four columns of the index
Full Key Cardinality	Number of different values in all columns of the index

7.6.3.2 Indexes: Index Structure

The integrated *Index Structure* detail tab page on the *Index* tab page provides the following information:

Column	Description
Position	Position of the column in the index
Order	Sorting order of the column The following values are possible: • *A* Data is sorted in ascending order • *D* Data is sorted in descending order
DB Column Number	Number of the column in the database
DB Column Name	Name of the column in the database
DB Type	Data type of the column in the database
DB Length	Length of the column in the database

> **i Note**
>
> The following information is **only** displayed for the **local system** and if the table is defined as a transparent table in the ABAP Dictionary:

Column	*Description*
SAP Column Name	Name of the column defined in the ABAP Dictionary
SAP Type	Data type of the column defined in the ABAP Dictionary
SAP Length	Length of the column defined in the ABAP Dictionary

7.6.3.3 Indexes: Index Status

The integrated *Index Status* detail tab page on the *Index* tab page provides the following information:

Field	Description
Partition	Database partition number
Data Partition ID	> **i Note** > This field is **only** available if the related table is using data partitioning. Data partition ID
Compression Attribute	State of the `COMPRESSION` attribute of the index The following values are possible: • *Yes* Index compression is enabled • *No* Index compression is not enabled
Index Partitioning	Partitioning characteristic of the index The following values are possible: • *N* Non partitioned index • *P* Partitioned index • *Blank* Index does not reside on a partitioned table.

Field	Description
Index Requires Rebuild	Indicates whether the index requires to be rebuilt
Large RIDs	Indicates whether the index is using large row IDs (RIDs), that is, 4-byte page number, 2-byte slot number The following values are possible: • *Yes* The index is using large RIDs. • *No* The index is not using large RIDs • *Pending* The table where the index is defined supports large RIDs, that is, the table resides in a large tablespace. The index for the table or data partition, however, has not yet been reorganized or rebuilt. Therefore, the table is still using 4-byte RIDs, and the table or index has to be converted to use large RIDs.
Index Compressed	Physical index format The following values are possible: • *Yes* Index is in compressed format • *No* Index is in uncompressed format **i Note** If the physical index format does not match the compression attribute, you have to reorganize the index to convert the index to the defined format.
Logical Size	Amount of disk space in KB that is logically allocated for the index
Physical Size	Amount of disk space in KB that is physically allocated for the index
Index ID	Identifier for the index

i Note

If the index is defined on a table that is partitioned or using data partitioning, the index status is separately available for each partition or data partition.

7.6.3.4 Indexes: Compression Status

The integrated *Compression Status* detail tab page on the *Index* tab page provides the following information:

Field	Partition
Partition	Database partition number
Data Partition ID	Data partition ID **i Note** This field is **only** available if the related table is using data partitioning.
Compression	State of the `COMPRESSION` attribute on the index • *Yes* Index compression is enabled. *No* Index compression is not enabled.
Index Compressed	Physical index format • *Yes* Index is in compressed format. • *No* Index is in uncompressed format. If the physical index format does not match the compression attribute, an index reorganization is required to convert the index.
Saved Pages	**i Note** This value is the same for each entry of an index or index partition for each database partition in a DPF environment. If the index is not physically compressed (that is, `INDEX_COMPRESSED` is *N*), the displayed value represents the estimated percentage of leaf pages saved, as if the index were actually compressed. If the index is physically compressed (that is, `INDEX_COMPRESSED` is *Y*), the displayed value reports the `PCTPAGESSAVED` value from the system catalog view (either `SYSCAT.INDEXES` or `SYSCAT.INDEXPARTITIONS`). For more information, see Indexes: System Catalog [page 161].

Field	Partition
Saved Leaf Pages	**i Note** This value is the same for each entry of an index or index partition for each database partition in a DPF environment. If the index is not physically compressed (that is, `INDEX_COMPRESSED` is *N*), the displayed value represents the estimated number of leaf pages saved as if the index were actually compressed. If the index is physically compressed (that is, `INDEX_COMPRESSED` is *Y*), the displayed value reports the calculated number of leaf pages saved. This calculation is based on the `PCTPAGESSAVED` and `NLEAF` values from the system catalog view (either `SYSCAT.INDEXES` or `SYSCAT.INDEXPARTITIONS`). If either `PCTPAGESSAVED` or `NLEAF` are invalid values (*-1*), this value is also set to *-1*. For more information, see Indexes: System Catalog [page 161].
Time of Collection	Date and time of the last compression check for this index

i Note

If the index is defined on a table that is partitioned or using data partitioning the index status is available for each partition or data partition separately.

7.6.3.5 Indexes: REORGCHK

The integrated *REORGCHK* detail tab page on the *Index* tab page provides the following information:

Field	Description
Summary	Brief description which kind of `REORG` is recommended according to the recommendations for this index
Data Partition Name	**i Note** This field is available *only* if the related table is using data partitioning. Name of the data partition
F4: Cluster Ratio	Index size divided by allocated space
F5: Index Size / Allocated Space	Index size divided by allocated space as a percentage

Field	Description
F6: No. Entries / No. Poss. Entries	Number of deleted entries in relation to total entries in index
F7: Ratio of Deleted Index Entries	Number of deleted entries in relation to total entries in index
F8: Ratio of Deleted Index Leafs	Number of deleted tree leafs in relation to total tree leafs of index tree
Cardinality	Number of index entries in the index For some indexes, this value can be different from table cardinality. For example, the index cardinality on XML columns might be higher than the table cardinality.
Number of Pseudo-Deleted RIDs	Number of pseudo-deleted RIDs.

The `REORGCHK` information for indexes of tables with data partitioning is available per data partition. In this case, the data is displayed in list format.

7.6.4 Single Table Analysis: Table Status

The *Table Status* tab page provides the following information:

Field	Description
Partition	Database partition number
Data Partition ID	Data partition ID **i Note** This field is only available if the related table is using the data partitioning feature (DPF).
Physical Size	
Data Objects	Amount of disk space in KB that is physically allocated for the table For tables using multi dimensional clustering (MDC), the value displayed includes the size of the block map object. The displayed value represents the physical size of the base table only. Space that is consumed by LOB data, long data, indexes, and XML objects is reported by other fields as described in the following.
Long Objects	Amount of disk space in KB that is physically allocated for long field data in a table

Field	Description
LOB Objects	Amount of disk space in KB that is physically allocated for long field data in a table
XML Objects	Amount of disk space in KB that is physically allocated for XML data in a table
Index Objects	Amount of disk space in KB that is physically allocated for the indexes
Columnar Objects	Amount of disk space in KB that is physically allocated for columnar data. This information is only available as of DB2 10.5 or higher.
Logical Size	
Data Objects	Amount of disk space in KB that is logically allocated for the table For MDC tables, this size includes the logical size of the block map object. This size represents the logical size of the base table only. Space that is consumed by LOB data, long data, indexes, and XML objects is reported by other fields described in the following.
Long Objects	Amount of disk space in KB that is logically allocated for long field data in a table
LOB Objects	Amount of disk space in KB that is logically allocated for long field data in a table
XML Objects	Amount of disk space in KB that is logically allocated for XML data in a table
Index Objects	Amount of disk space in KB that is logically allocated for the indexes
Columnar Objects	Amount of disk space in KB that is logically allocated for columnar data. This information is only available as of DB2 10.5 or higher.
REORG	

Field	Description
Inplace REORG Status	Current status of an inplace table reorganization on the table The following values are possible: • *ABORTED* The inplace table reorganization has status *PAUSED* but is unable to resume. A `STOP` is required. For more information, see Performance: Inplace Table Reorganization [page 116]. • *EXECUTING* The inplace table reorganization is currently being executed. • *NULL* This value only appears if **no** inplace reorganization has been performed on the table. • *PAUSED* The inplace table reorganization currently pauses and can be resumed again. For more information, see Performance: Inplace Table Reorganization [page 116].
REORG Pending	Indicates whether a `REORG` is pending for the table This state is possible for `ALTER` operations like `DROP COLUMN`that require an offline reorganization of the table
Number of ALTER Operations	Number of `ALTER` operations after which a reorganization is recommended and that have been performed against this table since the last reorganization. If the number of operations reaches a limit of 3, a reorganization is required.
Reclaimable MDC Space	For an MDC table in a DMS tablespace, this value indicates the amount of disk space that can be reclaimed by running the `REORG` command with the `RECLAIM` option. For any other table, the value is zero.
Indexes Require Rebuild	Indicates whether an index requires rebuilding
Other Technical Attributes	

Field	Description
Large RIDs	Indicates if a table is using large row IDs (RIDs) (4-byte page number, 2-byte slot number) The following values are possible: • *YES* The table is using large RIDs. • *NO* The table is not using large RIDs. • *PENDING* The table supports large RIDs (that is, the table is in a large tablespace) but at least one of the indexes for the table has not yet been reorganized or rebuilt. Therefore, the table is still using 4 byte RIDs, which means that the table or indexes have to be converted.
Large Slots	Indicates whether the table is using large slots (which allows more than 255 rows per page) The following values are possible: • *YES* The table is using large slots. • *NO* The table is not using large slots. • *PENDING* The table supports large slots (that is, the table resides in a large table-space) but an offline table reorganization or a table truncation operation has not yet been performed. Therefore, the table is still using a maxi-mum of 255 rows per page.
Blocks Pending Cleanup	Indicates the number of blocks pending cleanup for MDC tables. For tables that do not use MDC, this value is always *0*.

Field	Description
Type of Statistics	Indicates how the statistics were generated The following values are possible: • *System fabricated* Statistics are gathered by the system without a table or an index scan. These statistics are stored in-memory and are different from the statistics that are stored in the system catalog. This is a temporary state and finally full statistics are gathered by DB2 and are stored in the system catalog. • *System asynchronously gathered* Statistics are gathered asynchronously by the system. The statistics have been collected automatically by DB2 by a background process and stored in the system catalogs. • *System synchronously gathered* Statistics are gathered synchronously by the system. • *User gathered* Statistics are gathered by the user. • *Undef* Unknown type of statistics or information that is not available for the current database release
Current Dictionary Size	Current size of compression dictionary in bytes
Index Type	Indicates the type of indexes that are currently in use for the table The following values are possible: • *1* Appears if `type-1` indexes are being used • 2 Appears if `type-2` indexes are being used
Availability	
Available	Describes the status of the table The following values are possible: • *No* The table is not available and all other output information that relates to the size and state is *0*. • *YES* The table is available. **i Note** Rollforward through an unrecoverable load makes a table unavailable.

Field	Description
Read Access Only	If the table is read-only, the value is *YES*. Otherwise, the value is *NO*.
No Load Restart	The value *YES* indicates that the table is in partially loaded state that does not allow a load restart. Otherwise, the value *NO* is returned.

If the table is partitioned or using data partitioning the table status is available for each partition or data partition separately. In this case, an overview list that contains a subset of the information described before, is displayed first. To view the full details, you can select an entry in the list. By default the first entry is displayed.

> **i** Note
>
> The physical sizes returned consider full extents allocated for the appropriate object and include the Extent Map Page (EMP) extents for objects created in DMS tablespaces.
>
> The logical size is the amount of space that is known for this table. It might be less than the amount of space that is physically allocated to hold object data for the table, for example, in case of a logical table truncation. The logical size returned considers full extents that are logically allocated for the object and, for objects created in DMS tablespaces, an estimate of the EMP extents.

7.6.5 Single Table Analysis: Compression Status

The *Compression Status* tab page provides the following information:

Field	Description
Partition	Database partition number
Data Partition ID	**i** Note This field is available **only** if the related table is using data partitioning Data partition ID
Dictionary	
Compression	State of the `COMPRESS` attribute on the table The following values are possible: • *Y* Row compression attribute is set to yes. • *N* Row compression attribute is set to no.

Field	Description
Built By	Code path taken to build the dictionary The following values are possible: • *INSPECT* The dictionary was built using the **INSPECT ROWCOMPESTIMATE** command. • *LOAD* The dictionary was built using the **LOAD INSERT** or **REPLACE** commands • *NOT BUILT* The dictionary was not built and is, therefore, not available. • *REDISTRIBUTE* The dictionary was automatically built during a `REDISTRIBUTE` operation. • *REORG* The dictionary was automatically built by the `REORG RESETDICTIONARY` utility. • *TABLE GROWTH* The dictionary was built in the course of `INSERT` operations.
Build Time	Date and time when the dictionary was built
Rows Sampled	Rows that have been sampled to calculate the compression results
Estimated Compression	
Saved Pages	Estimated amount of pages in percent that will be saved after compression
Saved Bytes	Estimated amount of bytes in percent that will be saved after compression
Dictionary Size	Estimated size of compression dictionary in bytes if the table will be compressed
Expansion Dictionary Size	Size of the expansion dictionary measured in bytes If a historical dictionary exists, this value is the sum of the current and historical dictionary sizes.
Average Length of Compressed Rows	Average length of rows if the table will be compressed

In the following cases, compression information is available more than once:

- The table is distributed across several database partitions.
- The table uses data partitioning
- The table contains XML data that is compressed separately from normal `DATA` pages.

If any of these conditions is fulfilled, an overview list containing a subset of the information described before, is displayed first. To view the full details, you can select an entry in the list. By default, the first entry is displayed.

7.6.6 Single Table Analysis: REORGCHK

The *REORGCHK* tab page provides the following information:

Field	Description
Summary	Provides a brief description which kind of `REORG` is recommended according to the recommendations for the table
Data Partition Name	Name of the data partition **i** Note This field is **only** available if the related table is using data partitioning.
F1: Overflow Rows	Overflow rows as a percentage
F2: Table Size / Allocated Space	Table size divided by allocated space as a percentage
F3: Full Pages / Allocated Pages	Full pages divided by allocated pages as a percentage
Number of Overflow Pages	Number of records that have overflowed Records overflow when a data record is updated and the new data record is larger than the old one, or when a column is added to a table.

The `REORGCHK` information for tables with data partitioning is available per data partition. In this case, the data is displayed in list format.

7.6.7 Space: Overview

Use

> **i** Note
>
> This screen is only available if you have activated the deprecated space and performance history data collection (see *Prerequisites*). This screen is only available in the Web browser-based version of the DBA Cockpit.

You can access general information about space for your database by calling the DBA Cockpit and choosing *Space* > *Overview* on the *Database* tab page of the DBA Cockpit.

Prerequisites

You have selected the *Enable Deprecated Data Collection Additionally (Not Recommended)* checkbox on the *System Configuration* screen (see Editing the Monitored System Landscape on the System Configuration Screen [page 57]).

Features

On the *Overview* screen, the following tab pages are available:

- *Databases and Tablespaces*
- *Tables and Indexes*

Databases and Tablespaces

Field	Description
Tablespaces	
Last Analysis	Date and time of the last analysis
Total Number	Total number of tablespaces in the database
Total Size	Total size of all tablespaces in KB
Free Space	Free space in all tablespaces in KB
Used Space	Used space of all tablespaces as a percentage
Minimum Free Space in a Tablespace	Free space of the tablespace with the lowest amount of free space in KB
Maximum Used Space in a Tablespace	Used space of the tablespace with the highest fill level as a percentage
Database Partitions	
Total Number	Number of database partitions The value displayed is only higher than 1 if you are using a multi-partition database.

Tables and Indexes

Field	Description
Last Analysis	Date and time of the last analysis

Field	Description
Total Number of Tables	Total number of tables defined in the database
Total Size of Tables	Total amount of used space of all tables defined in the database
Total Number of Indexes	Total number of indexes defined in the database
Total Size of Indexes	Total amount of used space of all indexes defined in the database
Oldest REORG Check	Date and time of the oldest execution of the job *REORGCHK for All Tables*
Latest REORG Check	Date and time of the latest execution of the job *REORGCHK for All Tables*

7.7 Space: Indexes

Use

On the *Indexes* screen, you can access information about all indexes in your database.

Prerequisites

To use the *Indexes* screen, the data collection framework (DCF) must be set up correctly. If the DCF is not available or not set up correctly, a warning is displayed including a link to the DCF configuration screen where you can perform the necessary steps.

For more information, see Data Collection Framework: Configuration [page 243].

Features

The *Indexes* screen provides the following:

- A *Selection* area where you specify a time frame and filter criteria for the collected data
  In the *Sort Tables by* field, you can choose if the indexes are sorted by size or by last used date.
- An overview table displaying the retrieved history data according to your selection criteria

In the overview table, the following information is displayed:

Column	Description
Index Schema	Name of the index schema
Index Name	Name of the index
Table Schema	Name of the table schema
Table Name	Name of the table
Number of Leaves	Number of leaves in the index
Cardinality	Cardinality of the index
Est. Phys. Index Size (KB)	Estimated physical index size in KB
Est. Log. Index Size (KB)	Estimated logical index size in KB
Last Used	Date when the index was last used

7.8 Space: Virtual Tables

Use

You can use the information on this screen to check whether a table can be virtualized or materialized to save space.

> ⚠ Caution
>
> Before you drop tables and re-create them as virtual tables, make sure that you have read **SAP Note** 1151343.

Features

An SAP system contains thousands of empty tables consuming much space in the DB2 tablespaces. These empty tables also generate an additional load on database administration tasks and autonomic features, for example, automatic `RUNSTATS` and automatic `REORG`.

Each empty table occupies several extents of storage space. For example, an empty table with a primary key requires 5 to 11 extents, which translates into 160 KB to 352 KB on a tablespace with a page size of 16 KB and with an extent size of 2. To save this unnecessary allocated space, you can replace these empty tables with views, which are called virtual tables in this context. On the first WRITE operation on such a virtual table, this virtual table is automatically replaced with a table by the SAP system.

On the *Virtual Tables* screen, the following tab pages are available:

- *Virtual Tables*
  Contains a list of all virtual tables that exist in your SAP system. To materialize a single or multiple tables, select one or more tables and choose the *Materialize* pushbutton.
- *Candidates for Virtualization*
  Displays a list of tables that are candidates for being dropped and re-created as virtual tables. If you choose the *Convert Empty Tables* pushbutton, a background job is scheduled that checks each table for the following:
  - Whether it is empty
  - Whether it is not volatile
  - Whether it does not have a partitioning key
  - Whether it does not use MDC tables
  - Whether it has no triggers defined
  - Whether it does not depend on triggers
  - Whether it does not participate in data replication

  Tables that meet these conditions are dropped and re-created as virtual tables.

> **i Note**
>
> The use of virtual tables is transparent to the ABAP Dictionary.
>
> The *Convert Empty Tables* pushbutton is only available for the local system.

7.9 Space: Top Space Consumers

Use

On the *Top Space Consumers* screen, you can access information about the largest or fastest growing tables of your database.

Prerequisites

The *Top Space Consumers* screen requires the data collection framework (DCF) to be set up correctly. If the DCF is not available or wrongly set up, an appropriate warning is displayed including a link to the DCF configuration screen where you can perform the required steps. For more information, see Data Collection Framework: Configuration [page 243].

Features

The *Top Space Consumers* screen provides the following:

- A selection area where you specify filter criteria for the collected data
- An overview table displaying the retrieved history data
  This overview table also includes a subset of space-relevant data reflecting the last set of values that was collected during the selected time period.
- An option to break down all history data that was collected for the selected monitoring data during the specified time frame.

Process

To be able to analyze the space situation in your database, you proceed as follows:

1. You identify the time period where you expected a major impact on space consumption of your tables, for example:
   - Time periods with archiving and `REORG` activities
   - Mass-processing of application data
   - Installation of new application functions
2. In the *Selection* area, you specify the appropriate time frame.

   > **i** Note
   >
   > If you are using a multi partition database system and you want to retrieve a general snapshot, you set the partition selection to *All* and you do **not** filter by partition. To retrieve a more detailed snapshot, you can drill down the selection by partitions.
   >
   > If a single partition or a subset of partitions shows deviations compared to other members, you can also restrict the partition selection. Deviations are usually an indication for a bad partitioning key.
3. To refresh the result set, you choose the *Apply Selection* pushbutton.
4. To display detail information about the growth of a table over time, you select the appropriate entry in the overview table. The data is displayed in the *History Details* area.
5. To display further details of the selected table, you select a line in the overview table and choose the *Detailed Table Analysis* pushbutton. You are directed to the *Space* > *Single Table Analysis* screen.
   For more information about the data provided on the *Single Table Analysis* screen and how to evaluate it, see Analyzing and Maintaining a Single Table [page 152].

7.10 Space: REORGCHK Results

> **i** Note
>
> This screen is deprecated.

The *REORGCHK Results* screen displays the findings of the `REORGCHK` tool. By default, all tables with `REORG` results are displayed. You can, however, filter the data to be displayed by specifying the schema name and the table name.

> **⚠** Caution
>
> The `REORGCHK` results do not necessarily indicate a need for reorganization. For more information, see SAP Note 975352.

As of Db2 10.5, this screen does not show any `REORGCHK` recommendations.

We recommend that you use **Db2 automatic reorganization** (`AUTO REORG`) to automate your regular maintenance activity of reclaiming space and cleaning up indexes. To see what is currently in the queue for automatic maintenance, use the screen Space: Automatic Maintenance Queue/RTS Requests [page 181].

7.11 Space: Automatic Maintenance Queue/RTS Requests

> **i** Note
>
> This screen is available as of Db2 10.1. If you are monitoring a database with Db2 version 9.7 or lower, you can use the Space: REORGCHK Results screen.

On the *Automatic Maintenance Queue/RTS Requests* screen, you can display information about objects that are in the queue for automatic maintenance, about RUNSTATS candidates, and about requests for real-time statistics.

7.11.1 Displaying the Queue for Automatic Maintenance Jobs

Use

When you run the DB2 database with automatic maintenance, jobs like reorganizations or `RUNSTATS` runs are scheduled automatically by DB2. However, you might sometimes want to know the position of a maintenance job in the queue for automatic maintenance, and what other jobs are ahead of the job in the queue. The

Automatic Maintenance Queue tab page on the *Automatic Maintenance Queue/RTS Requests* screen displays details of all queued and running automatic maintenance jobs.

> **i Note**
>
> Requests for real-time statistics are not displayed on the *Automatic Maintenance Queue* tab page because these jobs are not scheduled in the queue for automatic maintenance. Use the *Requests for Real-Time Statistics* tab page to display requests for real-time statistics (see also Displaying Requests for Real-Time Statistics [page 185]).

Procedure

To display the automatic maintenance queue in the DBA Cockpit, choose *Space* > *Automatic Maintenance Queue/RTS Requests* and then the *Automatic Maintenance Queue* tab page.

On the *Automatic Maintenance Queue* tab page, the DBA Cockpit displays the following fields:

Column	Description/Comments
Queue Position	The position of the job in the automatic maintenance queue. Jobs in the automatic maintenance queue are ordered first by earliest start time. Entries with the same earliest start time are ordered by priority. Entries with the same earliest start time and priority are ordered by queue entry time.
Database Name	Name of the database
Member	Member where the automatic maintenance queue runs. There is one automatic maintenance queue per member.
Object Type	Type of object in the maintenance queue (database, table, nickname, or view)
Object Schema	If the object is the database itself, this field is empty.
Object Name	If the object is the database itself, this field is empty.
Job Type	Type of automatic maintenance job: • `RUNSTATS` • `REORG` • `BACKUP` • `STATSPROFILE`

Column	Description/Comments
Job Details	Details about the maintenance job if the job type is `RUNSTATS` or `REORG`: • For `RUNSTATS`, the job details indicate whether the `RUNSTATS` job is doing a full runstats or just sampling. • For `REORG`, the job details list the keywords that will be applied to modify the behavior of the `REORG` utility (for example, `INDEXES`, `CLEANUP`, and so on). • For an index reorganization, if the keywords in the job details include `CLEANUP` and `RECLAIM EXTENTS` separated by a semicolon, an index reorganization cleanup is done, followed by the evaluation of and potential run of index reclaim extents.
Job Status	Current status of the job (queued or executing)
Job Priority	Priority of the job in the queue
Maintenance Window Type	Type of maintenance window that will be used for the job (online or offline)
Queue Entry Time	Time when the job was added to the automatic maintenance queue
Execution Start Time	Time when the job started execution (only if its status is `EXECUTING`)
Earliest Start Time	Start time of next maintenance window where the job is eligible to run

7.11.2 Displaying the RUNSTATS Candidates for the Automatic Maintenance Queue

Use

When you run the DB2 database with automatic maintenance, statistics collections (`RUNSTATS`) are scheduled automatically by DB2. However, you might sometimes want information about the `RUNSTATS` candidates that are queued for evaluation by the automatic statistics collection.

Candidates that are eligible for `RUNSTATS` are moved to the automatic maintenance queue (see also Displaying the Queue for Automatic Maintenance Jobs [page 181]).

Procedure

To display the RUNSTATS candidates for the automatic maintenance queue, in the DBA Cockpit, choose *Space* > *Automatic Maintenance Queue/RTS Requests* and then the *Runstats Candidates* tab page.

On the *Runstats Candidates* tab page, the DBA Cockpit displays the following fields:

Column	Description
Queue Position	The position of the RUNSTATS candidate in the queue for the evaluation of the automatic statistics collection
Member	Member where the automatic evaluation runs
Object Type	Type of object queued for evaluation by the automatic statistics collection (table, nickname, or view)
Object Schema	-
Object Name	-
Object Status	Status of the queued object: • EVALUATION_PENDING The automatic statistics collection is still evaluating whether a statistics update for the object is required. • JOB_SUBMITTED The automatic statistics collection has determined that a statistics update for the object is required and has submitted a job to the automatic maintenance scheduler. The automatic statistics collection is waiting for the job to complete. **i Note** RUNSTATS candidates whose evaluation has been finished are removed from the evaluation queue for automatic statistics collection. They will not be added to the queue again until the next evaluation interval for the automatic statistics collection.
Job Submission Time	Time at which the automatic statistics collection job was submitted to the automatic maintenance scheduler (only if the status is JOB_SUBMITTED)
Queue Entry Time	Time when the job was added to the evaluation queue of the automatic statistics collection

i Note

The table on the *Runstats Candidates* tab page might be empty if there are no objects currently queued for evaluation by the automatic statistics collection. This can be the case, for example, if the automatic

statistics collection has processed all the tables that were queued for evaluation in the current evaluation interval.

7.11.3 Displaying Requests for Real-Time Statistics

Use

When you run the DB2 database with automatic maintenance, updates of real-time statistics are scheduled automatically by DB2. However, there might be situations where you want the following information:

- Which real-time statistics requests are pending in the system?
- Which real-time statistics requests are currently being processed?
- Where is a real-time statistics request queued, and what other requests are ahead of this request in the queue?

Procedure

To display the requests for real-time statistics, in the DBA Cockpit, choose *Space* > *Tables and Indexes* > *Automatic Maintenance Queue/RTS Requests* and then the *Requests for Real-Time Statistics (RTS)* tab page.

On the *Requests for Real-Time Statistics (RTS)* tab page, the DBA Cockpit displays the following fields:

Column	Description
Queue Position	The position of the job in the queue of the real-time statistics requests
Member	Member that requested the real-time statistics
Object Type	-
Object Schema	-
Object Name	-
Request Type	Type of request: • `WRITE_STATS`: Request is to write statistics already collected to the catalogs • `COLLECT_STATS`: Request is to perform full statistics collections

Column	Description
Request Status	Request status: • `PENDING`: Request is waiting to be picked up by the real-time statistics daemon • `QUEUED`: Request has been gathered by the real-time statistics daemon and is awaiting processing • `EXECUTING`: Request is currently being processed by the real-time statistics daemon
Request Time	Time when the request was submitted
Queue Entry Time	Time when the request was added to the processing queue for real-time statistics
Execution Start Time	Time when the request was executed (only if its status is `EXECUTING`)

7.12 Space: Compression Status

Use

The *Compression Status* screen provides information about tables that are already compressed, including the sum of the current savings and the total number of compressed tables. In addition, you can view and change the current value of the database parameter *Global Compression Option*.

> **i** Note
>
> Depending on your database version, the DBA Cockpit displays additional metrics for index compression and adaptive compression. As of Db2 9.7, index compression is available. As of Db2 10.1, adaptive compression for data is available.

Features

You can call up the *Compression Status* screen by choosing ▶ *Space* ▶ *Compression Status* ▶ on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

Global Compression Option

The global compression option is the database variable `SAP<SID>.GLOBAL_COMPRESSION_OPTION`, which defines whether newly created database tables are compressed after creation. For more information, see SAP Note 1690077.

The global compression option can be set as follows:

Option	Description
NO	New tables in the database are **not** compressed.
YES	New tables in the database are compressed using the default compression mode.
YES STATIC	New tables in the database are compressed using the static compression mode.
YES ADAPTIVE	New tables in the database are compressed using the adaptive compression mode.

If the database variable for the global compression option is not defined in the database, the global compression option is displayed as *Not set* in the DBA Cockpit.

Summary

In the screen area above the list of compressed tables, you can see the current savings resulting from compression and the number of compressed tables.

Compressed Tables

In this screen area, all tables meeting at least one of the following requirements are displayed:

- The table is enabled for data compression and contains compressed records.
- The indexes of a table are enabled for index compression and are compressed.

The following table provides explanations for some columns and values on the *Compressed Tables* screen. Self-explanatory columns are not included.

Compressed Tables: Field Help

Field	Description
Savings (MB)	Total savings of space This includes all savings regardless of whether they are caused by data or index compression
Total Size (MB)	Total size of the table The value displayed is the sum of `DATA`, `LONG`, `LOB`, `XML`, columnar data, and index pages over all database partitions.

Details About the Compression Status of a Table

You can display details about the compression status of a table by choosing a line in the overview table.

> **i** Note
>
> The compression results are based on checks that have been performed before in the background. These checks evaluate statistics data that was gathered by the `RUNSTATS` utility. In the case of missing or incomplete statistics data, the calculated check results can be inaccurate. In this case, an appropriate

message is displayed. To display more details, choose *Show Table Details*. You are redirected to the *Single Table Analysis* screen where you can do further analysis.

For more information, see Analyzing and Maintaining a Single Table [page 152].

More Information

Space: Details for Compression Status and Compression Candidates [page 191]

7.13 Space: Compression Candidates

Use

On the *Compression Candidates* screen, you can see information about tables that are candidates for compression (or recompression) based on selection criteria that you can modify. You can also see the estimated total number of selected tables and the corresponding amount of possible savings if the selected tables have been compressed according to the recommended compression action.

Features

You can call up the *Compression Candidates* screen by choosing *Space* > *Compression Candidates* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

Selection Area

In the selection area, you can enter the following criteria for compression candidates:

- Absolute and relative savings of space after compression or recompression exceed specified values.
- The ratio of update operations in comparison to read operations of a table is lower than a specified percentage.

In addition, you can specify whether adaptive or static compression should be used to calculate the savings for compression candidates. Based on your input, the DBA Cockpit calculates which tables (including their indexes) fulfill the selection criteria.

Summary of Compression Savings

In the *Summary of Compression Savings* screen area, you can see a summary of possible savings as follows:

Field	Description
Current Savings	Current total savings based on the existing compression in the database (without additional savings from the compression of the selected compression candidates)
Selected Additional Savings (Estimate)	Additional savings from the compression of the selected compression candidates
All Additional Savings (Estimate)	All additional savings that are possible using compression for all database tables (estimate)

Selected Compression Candidates

In the *Selected Compression Candidates* screen area, tables that meet the selection criteria are displayed. Compression candidates can be the following:

- A table has already been compressed but more space could be saved by recompressing it based on a better compression dictionary.
- A table has been enabled for data compression but still contains records that are not yet compressed.
- The indexes of a table are not yet enabled for index compression
- The indexes of a table have been compressed, but more space could be saved by recompressing the table
- A table has already been compressed, but more space could be saved by using another compression mode for data (static vs. adaptive compression).

The following table provides explanations for some columns and values in the *Selected Compression Candidates* screen area. Self-explanatory columns are not included.

Compression Candidates: Field Help

Column	Description
Total Size (MB)	Total size of table if the compression was performed
Estimated Additional Savings (MB)	Estimated additional savings of space that can be achieved for the table in MB. This includes all savings regardless of whether they are caused by data or index compression. The DBA Cockpit calculates the savings based on the compression option displayed in the *Recommended Compression* column. If the table has already been compressed, then the value displayed does not represent the new total savings but the additional savings.
Estimated Additional Savings (%)	Percentage of estimated additional savings

Details About Compression Candidates

You can display details about a compression candidate by choosing a line in the overview table.

More Information

Space: Details for Compression Status and Compression Candidates [page 191]

7.13.1 Compressing Tables

Context

To implement the estimated space savings displayed on the *Compression Candidates* screen, you must schedule a compression job to reorganize the tables that you want to compress.

> **i** Note
>
> An offline REORG job is triggered by this procedure.

Procedure

1. On the *Compression Candidates* screen, select all tables that you want to compress and choose the *Compress* pushbutton.
2. In the *Schedule Job* dialog box, you specify when you want the compression job to run:
   - To run the job immediately, choose the *Start Immediately in the Background* radio button.
   - To schedule the job at a certain day and time, choose the *Schedule* radio button and specify an appropriate date and time.
3. To schedule the compression job, choose *OK*.

Results

The appropriate **REORG** jobs are scheduled in the DBA Planning Calendar where you can view the progress of the compression job. Depending on the required actions to perform the compression, there can be separate **REORG** jobs for tables and for indexes.

7.13.2 Updating the Screen for Compression Candidates Manually

Context

To get an up-to-date overview of all compression candidates on the *Compression Candidates* screen, you must ensure that compression information is collected. One option is to use the data collection framework (DCF), which can collect compression information regularly. If you want to use the DCF, you need to activate the data collector *Compression*, which is inactive by default.

If you do not want to use the DCF for collecting compression information regularly, you can trigger a compression check manually using the procedure below.

Procedure

1. On the *Compression Candidates* screen, choose the *Start Compression Check...* button.
2. In the *Schedule Compression Check* dialog box, specify when you want the compression check job to run.

Results

A job is scheduled in the DB2 administrative task scheduler (ATS).

You can view the progress of the compression check job either under *Jobs* > *Scheduled DB2 Tasks* (Web browser) or in the DBA Planning Calendar, where you choose *DB Collectors* from the *Category* dropdown list.

7.14 Space: Details for Compression Status and Compression Candidates

The following table provides explanations for some columns and values in the detail screen areas on the *Compression Status* and *Compression Candidates* screens. Self-explanatory fields are not included.

Summary

Field	Description
Uncompressed Size	Total size in MB of the table (including indexes) without compression

Field	Description
Current Total Size	Total size in MB of the table (including indexes) with current compression
Current Total Table Size	Total size in MB of the table (without indexes) with current compression
Current Total Index Size	Total size in MB for all indexes of the table with current compression

Savings with Current Compression

Field	Description
Total	Total amount of savings in MB for this table (including indexes) with the current compression
Data	Amount of savings in MB for data with the current compression
Index	Amount of savings in MB for all indexes with the current compression

Estimated Savings with Recommended Compression (Compression Candidates Details Only)

Field	Description
Total	Total amount of estimated savings in MB for this table (including indexes) after the new recommended compression
Total Compression Ratio	Amount of estimated savings in percent for this table (including indexes) after the new recommended compression
Data (Static/Adaptive)	Amount of estimated savings in MB for data after the new recommended compression.
Data Compression Ratio (Static/Adaptive)	Amount of estimated savings in percent for data after the new recommended compression.
Index	Amount of estimated savings in MB for all indexes after the new recommended compression
Index Compression Ratio	Amount of estimated savings in percent for indexes after the new recommended compression

Estimated Additional Savings After Recommended Compression (Compression Candidates Details Only)

Field	Description
Check Date, *Check Time*	Date and time when the compression status was checked last

Field	Description
Saved Size	Estimated additional saved space after compression using the recommended compression type (in comparison to the savings with the current compression)
Saved Percentage	Percentage of estimated additional saved space after compression using the recommended compression type (in comparison to the savings with the current compression)
Recommended Compression	Recommended compression type (static or adaptive compression, index compression).

7.15 Space: Tables and Indexes (Deprecated)

> **i Note**
>
> This screen has been replaced by the following enhanced screens under *Space*:
>
> - Space: Indexes [page 177]
> - Space: Virtual Tables [page 178]
> - Space: Top Space Consumers [page 179]
> - Space: REORGCHK Results [page 181]
> - Space: Compression Status [page 186]
> - Space: Compression Candidates [page 188]

Prerequisites

- You have selected the *Enable Deprecated Data Collection Additionally (Not Recommended)* checkbox on the *System Configuration* screen (see Editing the Monitored System Landscape on the System Configuration Screen [page 57]).
- The data displayed is based on a set of database tables that have been filled by the job *REORGCHK for all Tables*. This job must have been scheduled using the DBA Planning Calendar. If the job is not running, no up-to-date data is available.

Features

You can access information about space for tables and indexes by calling the DBA Cockpit and choosing *Space* > *Tables and Indexes (Deprecated)* in the navigation frame of the DBA Cockpit.

The *Selection Criteria* dialog window appears in which you can limit the result set by choosing from the following filter and display options:

Field	Description
Filters	
Tablespace Name	Indicates the location of the table
Table Name	Name of the table
Table Size	Size of the table
Flagged Tables	If this flag is not set, only tables are displayed that have a recommendation for table or index reorganization.
Large RIDs	If this flag is set, only tables are displayed that are located in large RID tablespaces but that have not been enabled for large RIDs.
Not Available	If this flag is set, only tables are displayed that have the status `not available`.
REORG Pending	If this flag is set, only tables are displayed that have the status `REORG PENDING`.
Index Type-1	If this flag is set, only tables are displayed that still have *Type–1* indexes.
Load Status	If this flag is set, only tables are displayed that have the status `LOAD PENDING`.
Row Compression	If this flag is set, only tables are displayed that have been recommended for row compression.
Display Options	
Sort by	Sorts the tables by *Size* or *Name*
Maximum Number of Rows	Number of rows to be displayed

Example

To display the first 100 tables with the largest size, choose *Size* in the *Display Options* group box and enter *100* in the *Maximum Number of Rows* field.

When you have made your selections and chosen *OK*, the *Space: Table and Indexes* screen appears with the following information:

Column	Description
Schema	Schema of the table, usually the user who created the table
Table Name	Name of the table
Tablespace Name	Tablespace to which the table currently belongs
F1	Overflows rows as a percentage
F2	Table size divided by allocated space as a percentage
F3	Full pages divided by allocated pages as a percentage
Table Flagged	Indicates that table reorganization is recommended
Index Flagged	Indicates that table reorganization is recommended because of `REORGCHK` recommendations for at least one of the table indexes
Size (KB)	Table size
REORG Check Date	Date of the last `REORG` check, for example, the date when `RUNSTATS` ran using program `dmdb6srp`
REORG Check Time	Time of the last `REORG` check, for example, the time when `RUNSTATS` ran using program `dmdb6srp`

If you want to change the selection, choose the *Set Selection Criteria* pushbutton in the application toolbar. The *Selection Criteria* dialog box appears and you can make a new selection.

To display detailed information about tables and indexes, double-click a table or choose *Details*. A detail screen is displayed with information about tables, indexes, and table structures. You can directly access this screen by choosing *Space* > *Single Table Analysis* in the navigation frame of the DBA Cockpit.

7.16 Space: History - Database and Tablespaces (Deprecated)

> **i Note**
>
> This screen, which is only available for the SAP GUI-based user interface of the DBA Cockpit, has been superseded by the enhanced database screen under *Space* > *Database*. For more information, see Space: Database [page 137].

Prerequisites

This screen is only available if you have selected the *Enable Deprecated Data Collection Additionally (Not Recommended)* checkbox on the *System Configuration* screen (see Editing the Monitored System Landscape on the System Configuration Screen [page 57]).

For more information about how history data is evaluated, see Transition from the Deprecated Collection of Space and Performance History Data to the Data Collection Framework [page 55].

Features

You can access history data of the database and tablespaces by calling the DBA Cockpit and choosing *Space* > *History* > *Database and Tablespaces* in the navigation frame of the DBA Cockpit. The *History – Database and Tablespaces* screen appears. By default, the database history is displayed.

To switch to the tablespace history, select *Tablespaces* in the *Object Selection* field.

The following information is displayed on the tab pages:

Tab Page: Space

Column	Description
Tablespace Name	Name of the tablespace (only displayed if you have selected *Tablespaces* in the *Object Selection* field)
Partition	Monitored partition - displayed only if you are using a multi-partition database
KB Total	Amount of space in KB allocated
Changes (KB Total)	Average change of *KB Total* The average value depends on your selection in the *Statistics* field.
KB Used	Used space in KB of the allocated space
Changes (KB Used)	Average change of used space
% Used	Percentage used of allocated space
KB Free	Free space in KB of allocated space
Containers	Number of containers belonging to the tablespace
Changes Containers	Average change of number of containers

Tab Page: Tables and Indexes

Column	Description
Tablespace Name	Name of the tablespace (only displayed if you have selected *Tablespaces* in the *Object Selection* field)
Tables	Number of tables
Changes Tables	Average change of number of tables
Table (KB)	Space used by tables
Changes Table (KB)	Average change of space used by tables
Indexes	Number of indexes
Changes Indexes	Average change of number of indexes
Index (KB)	Space used by indexes
Changes Index (KB)	Average change of space used by indexes

If you want to display delta values between available measurements, select a row and choose *Details*. Alternatively, you can double-click the selected row. The table is displayed again with the following difference: Columns with the heading *Changes...* are renamed with *Delta...*.

7.17 Space: History - Tables and Indexes (Deprecated)

> **i Note**
>
> This screen, which was only available for the SAP GUI-based user interface, has been replaced by the following enhanced screens under *Space*:
>
> - Space: Top Space Consumers [page 179]
> - Space: Indexes [page 177]
> - Space: Virtual Tables [page 178]
> - Space: Compression Status [page 186]
> - Space: Compression Candidates [page 188]
> - Space: REORGCHK Results [page 181]

Prerequisites

This screen is only available if you have selected the *Enable Deprecated Data Collection Additionally (Not Recommended)* checkbox on the *System Configuration* screen (see Editing the Monitored System Landscape on the System Configuration Screen [page 57]).

For more information about how history data is evaluated, see Transition from the Deprecated Collection of Space and Performance History Data to the Data Collection Framework [page 55].

Features

You can access history data on tables and indexes by calling the DBA Cockpit and choosing *Space* > *History* > *Tables and Indexes* in the navigation frame of the DBA Cockpit (SAP GUI).

The *Set Selection Criteria* dialog window appears in which you can limit the result set by choosing certain filter criteria (for example, table size) and display options (maximum number of rows to be displayed and sort by growth, size, or name).

> ### Example
>
> To display the first 100 tables or indexes with the highest growth, choose the display option *Growth* in the *Sort by* field and enter *100* in the *Maximum number of rows* field.

If you confirm your selection, the *History – Tables and Indexes* screen appears with the following information:

Column	Description
Object Name	Name of the table or index
Object Type	Table or index
Tablespace Name	Tablespace to which the objects belong
Size (KB)	Size of the table or index
Changes Size (KB)	Average change of space used by table or indexes
REORG Check Date	Date of the last `REORG` check, for example, the date when `RUNSTATS` ran using program `dmdb6srp`
REORG Check Time	Time of the last `REORG` check, for example, the time when `RUNSTATS` ran using program `dmdb6srp`

If you want to display delta values between available measurements, select a row and choose the *Details* button. Alternatively, you can double-click the selected row. The table will be displayed again with the following difference: Columns with the heading *Changes...* are renamed to *Delta...*.

If you want to change the selection criteria, choose the *Set Selection Criteria* button.

8 Backup and Recovery

8.1 Backup and Recovery: Overview

You can access the *Overview* screen by calling the DBA Cockpit and choosing *Backup and Recovery* > *Overview* on the *Database* tab page (Web browser) or in the navigation frame of the DBA Cockpit (SAP GUI).

The following information is displayed on tab pages on the *Overview* screen:

Database Backup

In the *Selection* area of the *Database Backup* tab page, you can specify a time range. After applying your selection, a list with information about backups in this time range is displayed. On the browser-based user interface, there is an additional graphic that shows the backup runtime and the backup size for your selected time range. For each backup, the Db2 backup type, and whether it finished successfully or failed, is indicated.

You can modify the graphic by selecting a granularity other than *As Collected* on the time axis. Then, the displayed time range is subdivided into time slots of the selected granularity and backup sizes within the same time slot are added up. This allows you to get a clear overview of larger time ranges.

> **i** Note
>
> Backup size information is unavailable in database versions before Db2 9.7 and readily available as of Db2 10.1 FP2. For Db2 9.7 up to and including Db2 10.1, the registry variable DB2_BAR_STATS must be set to ON to enable the recording of backup size information.

If backup size information is unavailable, the value *0* (zero) is displayed in the list. For presentational purposes, this is shown as an arbitrary negative size in the graphic to keep the runtime information visible.

On the browser-based user interface, an additional graphic shows the log volume for the selected time range (if the data is available).

For more information about a database backup, you can choose an entry from the list. The details are displayed on the tab pages *Backup Details*, *Tablespaces*, and *Backup Sequences* in the content detail area.

Log Files

The *Log Files* tab page contains information about log files that have been moved from the log directory to the log archive or to a storage product, such as Tivoli Storage Manager (TSM)

If a log file is missing, it still appears in the list of log files. Instead of the file location, the text `Missing in History File` is displayed. In the SAP GUI-based version of the DBA Cockpit, these entries are also highlighted in red.

8.2 Backup and Recovery: Logging Parameters

You can access information about logging parameters by calling the DBA Cockpit and choosing *Backup and Recovery* > *Logging Parameters* on the *Database* tab page (Web browser) or in the navigation frame of the DBA Cockpit (SAP GUI).

The *Logging Parameters* screen appears, displaying the *Log Directory* tab page and, depending on the setting of the DB2 logging parameters, further directories for log files on separate tab pages:

- *Mirror Log* (`MIRRORLOGPATH`)
- *Overflow* (`OVERFLOWLOGPATH`)
- *Failover* (`FAILARCHPATH`)
- *ARCHMETH1* (`LOGARCHMETH1`)
- *ARCHMETH2* (`LOGARCHMETH2`)
- *New Log Directory* (`NEWLOGPATH`)

Only if the corresponding configuration parameter is set, the related tab page becomes visible on the screen.

Each tab page provides information about the related logging parameter and information about the file system. This information enables you to check the available space of the file systems where your database logs and the archived database logs are stored.

> ⚠ Caution
>
> In a production system, the *Rollforward Recovery Enabled* field must be set to *YES*.
>
> If this is not the case, you risk losing data and the ability to roll forward your database if serious database problems occur.

9 Configuration

9.1 Configuration: Overview

You can access general information about the database instance by calling the DBA Cockpit and choosing *Configuration* > *Overview* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI) of the DBA Cockpit.

On the *Overview* screen, the following information is displayed:

Database Instance	Description
Name	Name of the database instance
Partitionable	Indicates whether or not the current instance is a partitionable database server instance
Number of Partitions	Number of database partitions If the database environment is not partitioned, the value is *1*.
Number of Members	Number of pureScale members If the database is not installed as a pureScale system, the value is 1.
Address Space	Size of the current database instance in Bit (32 or 64)
Database Release	DB2-internal release number, as it is returned if you use the **`db2level`** command, for example, `03030106`
Service Level	Service level, as it is returned if you use the **`db2level`** command, for example, `DB2 v10.5.0.3`
Build Level	Build level, as is returned if you use the **`db2level`** command, for example, `n041021`
PTF	Program temporary fix (PTF) identifier, as it is returned if you use the **`db2level`** command, for example, `U498350`
Fix Pack	Fix Pack number, as it is returned if you use the **`db2level`** command, for example, `9`

For the operating system, the following information is displayed:

Operating System	Description
Host Name	Name of the system
Partition	Partition number
OS Name	Name of the operating system
OS Version	Version number of the operating system
OS Release	Release number of the operating system
Total CPUs	Total number of physical CPUs of the system
Configured CPUs	Number of configured physical CPUs of the system
Total Memory (MB)	Total amount of memory in the system in MB

> **i Note**
>
> If the database is distributed over several partitions, the operating system data is displayed in a table.

If the system has been installed as a high-availability disaster recovery (HADR) system, the following additional information is displayed:

HADR Information	Description
Connect Status	Current HADR connection status of the database The following values are possible: • *CONGESTED* • *CONNECTED* • *DISCONNECTED*
Local Host	Local HADR host name The value is displayed as a host name string or an IP address string, for example, `1.2.3.4`.
Local Service	Local HADR TCP service The value is displayed as a service name string or a port number string.
Log Gap	Average of the gap between the primary log sequence number (LSN) and the standby log LSN The gap is measured in bytes.
Primary Log File	Name of the current log file on the primary HADR database

HADR Information	Description
Primary Log LSN	Current log position of the primary HADR database The log sequence number (LSN) is a byte offset in the log stream of the database.
Primary Log Page	Page number in the current log file indicating the current log position on the primary HADR database The page number is relative to the log file, for example, page zero is the beginning of the file.
HADR Syncmode	Current HADR synchronization mode of the database The following values are possible: • *ASYNC* • *NEARSYNC* • *SYNC*
HADR Timeout	Number of seconds without any communication from its partner server after which an HADR database server considers that the connection between them has failed
Heartbeat	Number of missed heartbeats on the HADR connection If the database is in HADR primary or standby role, this field indicates the health of the HADR connection.
Connect Time	If the database is in HADR primary or standby role, the meaning of this field depends on the value of the *Connect Status* field. The following values are possible: • *CONNECTED* Displays the connection time • *CONGESTED* Displays the time when the congestion began • *DISCONNECTED* Displays the disconnection time If there has been no connection since the HADR engine dispatchable unit (EDU) was started, the connection status is reported as *Disconnected*, and the HADR EDU startup time is used for the disconnection time. Since HADR connect and disconnect events occur relatively seldom, the time is collected and reported even if the `DFT_MON_TIMESTAMP` switch is off. This element should be ignored if the database's HADR role is *STANDARD*.

HADR Information	Description
Remote Host	Host name of the HADR remote host The value is displayed as a host name string or an IP address string, for example, `1.2.3.4`.
Remote Instance	Name of the HADR remote instance
Remote Service	Remote HADR TCP service This value is displayed as a service name string or a port number string.
HADR Role	Current HADR role of the database The following values are possible: • *PRIMARY* • *STANDARD* • *STANDBY*
HADR State	Current HADR state of the database The following values are possible: • *DISCONNECTED* • *LOCAL_CATCHUP* • *PEER* • *REM_CATCH_PEN* • *REM_CATCHUP*
Standby Log File	Name of the current log file on the standby HADR database
Standby Log LSN	Current log position of the standby HADR database Log sequence number (LSN) is a byte offset in the log stream of the database.
Standby Log Page	Page number in the current log file indicating the current log position on the standby HADR database The page number is relative to the log file, for example, page zero is the beginning of the file.

9.2 Configuration: Database Manager

You can access information about the configuration of the database manager by calling the DBA Cockpit and choosing *Configuration* > *Database Manager* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI) of the DBA Cockpit.

On the *Configuration: Database Manager* screen, the parameter information is displayed as a list with a hierarchy column with the following first-level nodes:

Tree Node	Description
Common	Common information about the database manager, for example, release level and CPU speed
Diagnostics	Information about diagnostics
Default Monitor Switches	Information about the default monitor switches of the database
Security > *Groups*	Information about user groups of the database manager
Security > *Authentication*	Information about authentications of the database manager and about clients
Memory	Information about the memory of the database manager
Agents	Information about agents
Application Remote Interface	Information about the database application remote interface (DARI)
Sync Point Manager	Information about the configuration of the synchronization manager and the transaction manager
Transaction Manager	Information about the transaction manager
Network	Information about network characteristics such as communication protocols
Fast Communication Manager	Information about the Fast Communication Manager (FCM), that is, the configured communication in a multi-partition database
DB2 Discovery	Information about the configuration of the discovery mode
Others	Single parameters that are not accessible to the groups described above as well as parameters that are not known by the DBA Cockpit, for example, those of a new database release

For each node, the following information about the respective parameters is displayed:

Column	Description
Configuration Parameter	Description of the parameter
Technical Name	DB2 technical name of the parameter

Column	Description
Current Value	Currently active value of the parameter
Deferred Value	Value that becomes active after a restart of the database instance

For more information about these parameters, see the IBM DB2 online documentation.

You can switch off the hierarchy column by choosing the *Off* option of the *Hierarchy View* dropdown list. As a result, the list of parameters is displayed as a flat list in alphabetical order. The setting is saved as personal setting.

For information about maintaining the database configuration parameters, see Maintaining the Database Configuration [page 208].

9.3 Configuration: Database

You can access information about the database configuration by calling the DBA Cockpit and choosing *Configuration* > *Database* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI) of the DBA Cockpit.

On the *Database* screen, the parameter information is displayed as a list with a hierarchy column. In a distributed database, the number of partitions are displayed on the first level. Otherwise, the first-level tree nodes are as follows:

Tree Node	Description
Common	Common information about the database, for example, release level and country code
Automatic Maintenance	Information about the automatic maintenance switches
Optimization	Information about optimization
I/O	Information about I/O
Self-Tuning Memory Manager	Information about the self-tuning memory manager
Database Shared Memory	Information about the memory that is available for the database
Application Memory	Information about the memory that is available for the application
Logging	Information about log files and logging parameters

Tree Node	Description
Log File Management	Information about log file management parameters
Backup & Recovery	Information about recovery availability and backups
TSM	Information about Tivoli Storage Management (TSM)
Locks	Information about locks, for example, the percentage of lock lists per application
Space	Information about containers and tablespaces
Applications	Information about applications that connect to the database
High Availability	Information about the system configuration is only displayed if you are running a high availability system.
Others	Single parameters that are not accessible to the groups described above as well as parameters that are not known by the DBA Cockpit, for example, those of a new database release

For each node, the following information about the respective parameters is displayed:

Column	Description
Configuration Parameter	Description of the parameter
Technical Name	DB2 technical name of the parameter
Current Value	Currently active value of the parameter
Deferred Value	Value that becomes active after a restart of the database instance

> **i** Note
>
> - In a multi partition environment, the parameters can vary for each partition.
>   For more information about how to compare the configuration of several database partitions, see Comparing Database Configuration Parameters For Several Database Partitions [page 208].
> - Depending on your database release level, the tree nodes that are displayed can vary, that is, some might not be visible or others might be added.

You can switch off the hierarchy column by choosing the *Off* option of the *Hierarchy View* dropdown list. As a result, the list of parameters is displayed as a flat list in alphabetical order. The setting is saved as personal setting.

For information about how to maintain the database configuration parameters, see Maintaining the Database Configuration [page 208].

For more information about these parameters, see the IBM Db2 documentation .

9.3.1 Maintaining the Database Configuration

Context

You use the following procedure to maintain configuration parameters either on the *Database* or the *Database Manager* screen in the *Configuration* area.

Procedure

1. Select the parameter that you want to change and choose the *Change* pushbutton.

   The *Change Database Configuration Parameter* dialog box appears.
2. Enter the new configuration parameter values.

   > **i** Note
   >
   > Some configuration parameters are enabled for automatic value adjustment. In this case, the *AUTOMATIC* checkbox is displayed. If you select *AUTOMATIC*, the value is automatically maintained by DB2.
3. To check your entries, choose the *Check* pushbutton.

   The generated CLP commands that are based on your input are displayed. This area is automatically filled and refreshed whenever you choose the *Check* pushbutton.
4. To confirm your entries, choose the *Execute* pushbutton.

9.3.2 Comparing Database Configuration Parameters for Several Database Partitions

Procedure

1. On the *Database* screen in the *Configuration* area, choose the *Compare* button.
2. On the *Choose Database Partitions* dialog box, select the database partitions that you want to compare.
3. Choose the *OK* button.

Results

The *Database Configuration Adjustment* screen appears displaying the database configuration parameters for the selected database partitions in a table. The values that differ are marked blue.

> **i** Note
>
> By default, only the parameters that differ from one another are displayed. If you want to display all parameters, choose *All Parameters* from the *Filter* dropdown list.

9.4 Configuration: Registry Variables

You can access information about DB2 registry variables by calling the DBA Cockpit and choosing *Configuration* > *Registry Variables* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI) of the DBA Cockpit. The *Registry Variables* screen appears displaying the information as a list with a hierarchy column. Aggregate variables are displayed as a node that contains all the registry variables affected by the aggregate variable.

> **i** Note
>
> The variables that are affected when setting an aggregate variable, such as `DB2_WORKLOAD`, are grouped in folders. If the value of such a variable has been manually overwritten, it is marked yellow.

You can switch off the hierarchy column by choosing the *Off* option of the *Hierarchy View* dropdown list. As a result, the list of variables is displayed as a flat list in alphabetical order. The setting is saved as personal setting.

The *Scope* variable indicates the level at which the DB2 registry variable acquires its value. These levels are as follows:

- *Instance*
- *Global*
- *Environment*

9.5 Configuration: Configuration History

You use the *Configuration History* screen to keep track of database, instance, or registry configuration changes.

> **i** Note
>
> Depending on the version of your monitored database and depending on the setup of the data collection for your DBA Cockpit, the available metrics on the *Configuration History* screen may vary.

Prerequisites

You have set up one of the following data collections for the DBA Cockpit:

- **Data collection framework (DCF)**
  If the DCF is not available or not set up correctly, a warning is displayed including a link to the screen *Data Collection Framework: Configuration* where you can perform the necessary steps as described in Data Collection Framework: Configuration [page 243].
- **Old collection of space and performance history data**
  For more information, see Collection of Space and Performance History Data (Deprecated) [page 52].

Features

Depending on the version of your monitored database and the setup of the data collection, the layout and the available metrics on the *Configuration History* screen vary as follows:

Database Version	Available Metrics	Data Collection	User Interface Type	Comments
DB2 V9.7	Records of configuration events that are parsed from `db2diag.log` and the DB2 history file	DCF with data collector *Diag Log*	SAP GUI and Web Dynpro	-
DB2 10.1 and higher	Records of configuration events captured by the DB2 change history event monitor: Changes to the database configuration, the database manager configuration, and the registry	DCF with data collector *Change History*	SAP GUI and Web Dynpro	As of DB2 10.1, with the reuse of the DB2 change history event monitor, the DBA Cockpit can capture configuration changes continuously and with improved system performance.

More Information

Keeping Track of Configuration Changes Using the Configuration History [page 211]

Configuration History: Field Help [page 211]

9.5.1 Keeping Track of Configuration Changes Using the Configuration History

Procedure

1. To access information about the configuration history, call the DBA Cockpit and choose *Configuration* > *Configuration History* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI) of the DBA Cockpit.
2. In the *Selection* area, proceed as follows:
   - Specify the time frame for which you want to display the configuration history details.
   - Choose an option in the *Event Type* field to restrict the result list. By default, all event types are displayed.
   - Set additional filter criteria by using the keyword search in the *Object Name* field. You can use the asterisk (*) character in this field.

   The DBA Cockpit then displays the configuration changes for the selected time period, including details about the type of configuration change, the changed object, and whether the configuration was changed automatically or manually.
3. To display more details about a configuration history entry, use the *Open Settings Dialog* button to make more columns visible.

9.5.2 Configuration History: Field Help

The following table provides you with an overview and an explanation of fields and field values that are available on the *Configuration History* Web Dynpro screen.

> **i** Note
>
> Self-explanatory fields are omitted in the table.

Column	Description
Time of Collection	Time when the event (configuration change) was logged

Column	Description
Event Type	Event type that is associated with the configuration change As of Db2 10.1, possible values are: • *DBCFG*: Database parameter configuration changes • *DBCFGVALUES*: Initial values for all database configuration parameters at event monitor startup time if any database configuration parameter update was not captured by the data collector • *DBMCFG*: Database manager configuration parameter changes • *DBMCFGVALUES*: Initial values for all database manager configuration parameters at event monitor startup time if any database manager configuration parameter update was not captured by the data collector • *DDLSTMSTEXEC*: DDL storage events For Db2 9.7 and lower, possible values are: • *DB_CFG*: Database configuration parameter changes • *DB_CFG_STMM*: Changes of database configurations that are modified by STMM • *DBM_CFG*: Database manager configuration parameter changes • *REGISTRY*: Registry configuration parameter changes • *DDL_BP*: DDL statements for buffer pools • *DDL_BP_STMM*: DDL statements for buffer pools by STMM • *DDL_TBSP*: DDL statements for tablespaces • *DDL_WLM*: DDL statements for WLM • *DDL_THRES*: DDL statements for threshold violations • *HADR*: Changes of HADR
Object Name	Name of the object that the event relates to
Old Value	If available, the old state or value of the configuration object
Old (Value) Flag	If available, the old flag of the configuration change Possible values are: • *COMPUTED* • *AUTOMATIC* • *NONE* If the value is `NONE`, this means that the configuration was entered manually by a user.
New Value	If available, the new state or value of the configuration object

Column	Description
New (Value) Flag	If available, the new flag of the configuration change Possible values are: • *COMPUTED* • *AUTOMATIC* • *NONE* If the value is *NONE*, this means that the configuration was entered manually by a user.
Collection Type	Time when the configuration parameter value was collected. Possible values are: • *Initial*: The initial value that was captured when the configuration history was activated • *Updated*: Updated value
Configuration Deferred	Indicates whether a change to a configuration parameter value is deferred. Possible values are: • Y: Change deferred until next database activation • N: Change takes effect immediately
Statement Text	This column is only filled if you have selected the event type of DDL storage events (*DDLSTMSTEXEC*). The column contains the statement text of the DDL statement that was executed. To display the full text of the DDL statement more easily, select the relevant line in the table. The full text appears in the Statement Text screen area below the table.

9.6 Configuration: Parameter Check

You use the *Parameter Check* screen to check the existing parameterization of a DB2 database against SAP recommendations.

9.6.1 Comparing Your Database Parameterization Against SAP Recommendations

Context

SAP publishes database parameter recommendations for SAP systems running on IBM for DB2 for Linux, UNIX, and Windows in SAP Notes. For each DB2 version, one SAP Note is available.

In the DBA Cockpit, you can use the *Parameter Check* screen to compare the existing parameterization of a DB2 database against these SAP recommendations. When you access the *Parameter Check* screen, the SAP parameter recommendations are automatically downloaded and checked for the following against the SAP recommendations:

- DB2 database manager configuration
- DB2 configuration
- DB2 registry variables
- Buffer pool settings

The checks are performed depending on the current system characteristics, such as the application type, the operating system, or the usage of STMM. The parameter check reports deviations from the SAP standard recommendations. To correct these deviations, you can either change the respective parameters or declare the deviations as intended.

> **i** Note
>
> A parameter is deviating if its current or its deferred value is different from the SAP recommendations. A deferred parameter value is a value that comes into effect after the database has been restarted.

Procedure

1. On the *Database* tab page (Web browser) or in the navigation frame (SAP GUI), choose *Configuration* *Parameter Check*.

   The DBA Cockpit automatically downloads the SAP parameter recommendations and compares them with the parameterization of your database.
2. To display the parameters that deviate from the standard recommendation, choose the *Deviations Found* tab page.
3. To display information about which parameters are intended to deviate from the standard recommendation, choose the *Intended Deviations* tab page.

   On this tab page, you can find information about when the deviation was declared as intended and by whom. To display the comments in full length, click the relevant text in the *Comment* column.
4. To display the system attributes and attribute values that are relevant for the parameter check, choose the *Check Environment* tab page (see also System Attributes in the Check Environment [page 217]).

9.6.2 Declaring Parameter Deviations as Intended

Use

If a parameter value in the SAP recommendations (that is, in the SAP parameter notes) is not valid for a specific customer situation, you can declare the parameter deviation as intended. By doing so, you avoid that a specific deviation is displayed every time you run the parameter check tool.

Procedure

1. On the *Deviations Found* tab page, select a row from the table and choose the *Declare as Intended* menu button.

   > **i Note**
   >
   > In a database environment with multiple database partitions, the deviations of database configuration parameters and registry variables can occur on several partitions. When choosing the *Declare as Intended* menu button, you have to specify whether this operation applies to a single or to all partitions where this parameter deviation was found.

2. In the *Comment* dialog box, enter a reason why you consider this deviation as intended and choose the *Declare as Intended* pushbutton.
   The relevant parameter deviation is moved to the *Intended Deviations* tab page.

> **i Note**
>
> Since the list of intended deviations is stored in a table of the SAPTOOLS schema, the data collection framework (DCF) must be set up correctly. Otherwise, the *Declared as Intended* pushbutton is disabled.

9.6.3 Reversing the “Declared as Intended” Status

Context

During a parameter check, it can occur that a deviation that you formerly declared as intended is no longer valid. The status of intended deviations is automatically reversed to *not intended* in the following situations:

- The current parameter value in the monitored system has changed from the value at the moment when the deviation was declared as intended.
- The current deferred parameter value in the monitored system has changed from the value at the moment when the deviation was declared as intended.
- The SAP recommendation has changed, that is, the check has been changed since the parameter was declared as intended.

- The DB2 version or the Fix Pack have changed since the parameter was declared as intended.

Procedure

1. Choose the *Intended Deviations* tab page, where all parameter deviations that were declared as intended are displayed.
2. If you want to reverse the status of a single parameter deviation, select a deviation from the list and choose the *Declare as Not Intended* pushbutton.

   As a result, the selected parameter deviation is removed from the *Intended Deviations* tab page and appears again on the *Deviations Found* tab page.

   > **i** Note
   >
   > In a database environment with multiple database partitions, the deviations of database configuration parameters and registry variables can occur on several partitions. When choosing the *Declare as Not Intended* menu button, specify whether this operation applies to a single or to all partitions where this parameter deviation was found.

3. If you want to reverse the status of all deviations on the *Intended Deviations* tab page to not intended, you can clear the complete list by choosing the *Clear List of Intended Deviations* pushbutton.

   As a result, the deviations appear again on the *Deviations Found* tab page.

9.6.4 Correcting Parameter Deviations

1. On the *Deviations Found* tab page, select a deviation from the list and choose the *Correct this Parameter* pushbutton.

   > **i** Note
   >
   > This function is **not** available for buffer pools and registry variables. If you want to correct buffer pool parameters, use the *Configuration* > *Buffer Pools* screen. To correct parameter settings for registry variables, see the relevant section below.
   >
   > In a database environment with multiple database partitions, the deviations of database configuration parameters and registry variables can occur on several partitions. When choosing the *Correct This Parameter* menu button, you have to specify whether this operation applies to a single or to all partitions where this parameter deviation was found.

   The dialog box *Change Database Configuration Parameters* or *Change Database Manager Configuration Parameters* appears displaying the SAP-recommended value as default suggestion.
2. Enter the required value and choose the *Execute* pushbutton.

Correcting Registry Variables

Registry variables must be corrected by using the native DB2 command `db2set`. For registry variables, there are the following special recommendations in the case of deviations:

- Recommended value *Do not overwrite*:
  This applies to registry variables that are normally implicitly set as part of the `DB2_WORKLAD=SAP` aggregate registry variable. They should never be set explicitly, even if the current setting is the same as the implicit setting.
- Recommended value *Do not set*:
  This applies to registry variables that you should not set in an SAP environment.

To implement these two recommendations, run the following command:

```
db2set <registry_variable>=
```

To implement other deviations, use the following command:

```
db2set <registry_variable>=<recommended_value>
```

Result

The parameter changes are tracked in the DBA Cockpit and you can analyze them in the *Configuration History* task area under *Configuration*.

9.6.5 System Attributes in the Check Environment

On the *Check Environment* tab page of the *Parameter Check* screen, the following system attributes and attribute values that are relevant for the parameter check are displayed in the *System Characteristics* area:

Attribute	Description
DPF	Indicates if the database is using the database partitioning feature (DPF) The following values are possible: • *NO* In the case of a single-partition system • *YES* In the case of more than one partition

Attribute	Description
BI	Indicates whether the database system appears to be of BW type The following values are possible: • *NO* There are no active InfoCubes and DataStore objects in the database. Therefore, the DBA Cockpit does not consider the database system as a BW system. • *YES* There are active InfoCubes or DataStore objects in the database. Therefore, the DBA Cockpit considers the database system as a BW system. **i Note** You might have a BW system that is still empty, that is, no active InfoCubes or DataStore objects are in the database. Since the DBA Cockpit uses active InfoCubes and DataStore objects as a criterion to identify the database as a BW system, it might happen that an empty BW system is considered as not of BW type. If so, the parameter recommendations in the DBA Cockpit are not fully applicable for the BW system. In this case, make sure that you define the appropriate BW parameters as indicated in the relevant SAP Note for Db2 standard parameters for your database version.
STMM	Indicates if the database system is using STMM The following values are possible: • *NO* STMM is not enabled for all partitions. • *YES* STMM is enabled for all partitions. • *MIXED* STMM is only enabled for some partitions.
OS	Indicates the operating system platform
BIT	Indicates if the database instances are 32 bit or 64 bit instances
PSCALE	Indicates whether the database uses the Db2 pureScale Feature. The following values are possible: *NO*: For single-member systems *YES*: For systems with more than one member
NW	SAP NetWeaver release

Attribute	Description
CDE	Indicates whether there are any tables in the database that are column-organized. The following values are possible: • *NO*: All tables of the database are row-organized • *YES*: There is at least one table that is column-organized
HADR	Indicates whether your database system is configured for high availability and disaster recovery (HADR) The following values are possible: • *YES*: Database system is configured for HADR. • *NO*: Database system is not configured for HADR.

> **i** Note
>
> In the *Check Configuration* area, you are also provided with the number of the SAP parameter note of the underlying Db2 release. You can download the latest version of this SAP Note by choosing the *Download SAP Note* pushbutton.

9.7 Configuration: Db2 Fix Pack Level Check

By calling up this screen, you can run checks on your installed Db2 Fix Pack level. The results contain information about your Db2 Fix Pack level and suggestions on what to do if your system fails one of the checks.

Prerequisites

You have implemented the correction instruction attached to SAP Note 2989894.

When Do You Need This Data?

Use the *Db2 Fix Pack Level Check* screen to get answers to the following questions:

- Is the installed software level of your Db2 for LUW database an SAP-certified Db2 Fix Pack level according to SAP Note 101809?
- Is the committed software level identical to the installed software level of your database?
- Is a new Db2 Fix Pack available that contains important fixes?

How Do You Start the Checks?

You automatically initiate the run of the checks by going to the *Db2 Fix Pack Level Check* screen under *Configuration* for the first time, or when you change the monitored database in the DBA Cockpit.

More Information

SAP Note 2989894

9.8 Configuration: Client Configuration

> **i** Note
>
> The *Client Configuration* screen is only available in the SAP GUI-based version of the DBA Cockpit.

You can access information about the client configuration by choosing *Configuration* > *Client Configuration* in the navigation frame of the DBA Cockpit.

The *Client Configuration* screen provides an overview of the database connections that are currently open on a specific member. Furthermore, the configuration of the `db2cli.ini` file and the `db2dsdriver.cfg` file is displayed on tab pages.

On the `db2dsdriver.cfg` tab page, you can analyze the client affinity file by reloading it on all application servers. The files cannot be displayed for remote database connections, but only if a local database connection exists (local monitoring).

9.9 Configuration: Database Partition Groups

You can access information about available database partition groups by calling the DBA Cockpit and choosing *Configuration* > *Database Partition Groups* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI) of the DBA Cockpit.

On the *Database Partition Groups* screen, the following information is displayed:

Column	Description
Database Partition Group	Name of the database partition group
Partition	Number of the partition

Column	Description
Status	Current status of the partition The following values are possible: • *Status information not available* • *Partition not in partitioning map; containers not yet created* • *Partition not in partitioning map; containers created* • *Partition in partitioning map; containers created* • *Partition will be dropped after next redistribution*

You can display details about a database partition group by selecting it from the list. The following tab pages are displayed in the *Database Partition Group Details* area:

- *Partitions*
- *Buffer Pools*
- *Tablespaces*

Partitions

The *Partitions* tab page displays the following information:

Column	Description
Partition	Number of the partition
Status	Current status of the partition The following values are possible: • *Status information not available* • *Partition not in partitioning map; containers not yet created* • *Partition not in partitioning map; containers created* • *Partition in partitioning map; containers created* • *Partition will be dropped after next redistribution*

Buffer Pools

The *Buffer Pools* tab page displays a list of all buffer pools that have been defined for the selected database partition group as follows:

Column	Description
Buffer Pool Name	Total space in KB If you create tablespaces, this information is not displayed.
Buffer Pool Size (KB)	Size of the buffer pool in KB Value *–1* indicates that the default buffer pool size parameter from the database configuration is used (parameter BUFFPAGE).
Page Size	Size of one buffer pool page in bytes

You can display more details about a buffer pool by selecting it from the list. The detail information is displayed in the *Buffer Pool Details* area below the *Database Partition Group Details* area.

Tablespaces

The *Tablespaces* tab page displays a list of all tablespaces that have been defined for the selected database partition group including their page size in bytes.

You can display more details about a tablespace of a buffer pool by selecting the tablespace from the list. The detail information is displayed in the *Tablespace Details* area that appears below the *Database Partition Group Details* area.

Maintaining Database Partition Groups

You can maintain database partition groups, that is, change, add, redistribute, or delete them as described in the following section *Maintaining Database Partition Groups*.

9.9.1 Maintaining Database Partition Groups

Use

Using the list of database partition groups on the *Database Partition Groups* screen, you can maintain database partition groups as follows:

- Change database partition groups, that is, add or remove partitions

- Add new database partition groups
- Delete database partition groups
- Redistribute database partition groups

Procedure

Changing Database Partition Groups

1. On the *Database Partition Groups* screen, select a database partition group from the list and choose the *Edit* pushbutton.
2. In the *Change Database Partition Group* dialog box, select new database partitions to be added, or deselect existing database partitions to be dropped.
3. Confirm your selection by choosing the *Execute* pushbutton.

Adding New Database Partition Groups

1. On the *Database Partition Groups* screen, choose the *Add* pushbutton.
2. In the *Add Database Partition Group* dialog box, specify a name for the new database partition group.

   > → Recommendation
   >
   > We recommend that you use uppercase letters for the database partition group name. Using lowercase letters or special characters makes it less comfortable to access the selected database partition group with the DB2 command line processor.

3. From the list of available database partitions, select one or more partitions that you want to add to the new database partition group.
4. Confirm your entries by choosing the *Execute* pushbutton.

Deleting Database Partition Groups

1. On the *Database Partition Groups* screen, select a database partition group and choose the *Delete* pushbutton.

   > ⚠ Caution
   >
   > You **cannot** delete database partition groups that contain tablespaces that are still being used by the SAP system.
   >
   > You **must** delete the tablespaces first.

2. Confirm the deletion.

Redistributing Database Partition Groups

> i Note
>
> You can only redistribute database partition groups that have the status `Partition not in partitioning map; containers created`.

1. On the *Database Partition Groups* screen, select a database partition group and choose the *Redistribute* pushbutton.

2. In the *Schedule Job* dialog box, define if you want to redistribute the database partition group immediately or at a later point in time.

SQL Statements

In the *Change Database Partition Group* and *Add Database Partition Group* dialog boxes, the generated SQL statement(s) to be executed are displayed. This area is automatically filled and refreshed whenever changes are correctly entered.

9.10 Configuration: Buffer Pools

You can access information about available buffer pools by calling the DBA Cockpit and choosing *Configuration* > *Buffer Pools* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI) of the DBA Cockpit.

The following information is displayed on the *Buffer Pools* screen:

Column	Description
Buffer Pool Name	Name of the buffer pool
Partition	Number of the partition (only displayed if you are using a multi partition database)
Size (Pages)	Size of the buffer pool in KB A value of *–1* indicates that the default buffer pool size parameter from the database configuration is used (parameter `BUFFPAGE`). `AUTOMATIC` indicates that the selected buffer pool is tuned by the DB2 self–tuning memory management (STMM). If one of these special values is displayed and you want to see the real size of the buffer pool, you can use the buffer pool snapshot [page 80].
Page Size (Byte)	Size of one buffer pool page in bytes

To display details about a buffer pool, choose a buffer pool from the list. The following information is displayed in the *Buffer Pool Details* area:

- *Technical Settings*
- *Database Partition Groups*
- *Tablespaces*

Technical Settings

The *Technical Settings* tab page displays the following technical attributes:

Field	Description
Size	
Default (BUFFPAGE)	The size of the buffer pool has not been set at creation time, and the default size as defined in the database configuration parameter BUFFPAGE is used.
Automatic	The size of the buffer pool is managed by STMM.
Fixed Standard Size	The size of the buffer pool is set to a fixed value. This means that this size is set for all partitions except for those with exception entries.
I/O Units	
Page Size	Size of one page in the buffer pool
Number of Block Pages	Number of pages that are reserved for block I/O usage (only displayed if you are using DB2 UDB Version 8)

For all partitions of the selected buffer pool, the following information is additionally displayed:

Column	Description
Partition	Database partition number
Buffer Pool Size	Size of the buffer pool in pages
Exception	Indicates whether an exception entry has been defined for this partition that specifies a size different from the default size in the *Fixed Standard Size* field

Database Partition Groups

The *Database Partition Groups* tab page displays a list of all database partition groups to which the buffer pool is related. A buffer pool can be related to all available partitions or to a set of partitions defined by database partition groups. If the buffer pool is not already defined on all partitions, you can select further database partition groups.

Tablespaces

The *Tablespaces* tab page displays a list of all tablespaces that use this buffer pool as well as the page size of these tablespaces.

Maintaining Buffer Pools

You can maintain buffer pools, that is, change, add, or delete them as described in the following section *Maintaining Buffer Pools*.

9.10.1 Maintaining Buffer Pools

Use

Using the buffer pool list on the *Buffer Pools* screen, you can maintain buffer pool as follows:

- Change buffer pools, that is, add or remove partitions, resize or control the use of extended storage
- Add new buffer pools
- Delete buffer pools

Procedure

Changing Buffer Pools

1. On the *Buffer Pools* screen, select a buffer pool from the list and choose the *Edit* pushbutton.
2. In the *Change Buffer Pool* dialog box, you can modify the buffer pool attributes as follows:
   - Change the buffer pool size from *Automatic* to *Fixed* and vice versa.
     - If you select the *Automatic* radio button or if the *Automatic* attribute is already set for the buffer pool, you can set the initial size for STMM.
     - If you select the *Fixed* radio button or if the buffer pool is already using a fixed size, you can set or change the standard size for all partitions except for those partitions with exception entries.
     - If you want to create or delete an exception entry for a partition, select or deselect the relevant *Exception* checkbox in the partition list. After you selected the *Exception* checkbox for a partition, you can change the buffer pool size (pages).
   - If the buffer pool is specified for a set of database partition groups only and not yet for all database partition groups, you can add database partition groups by selecting them in the *Database Partition Groups* list.
   - You can change the block size and the number of block pages.
3. If you want the changes take effect immediately, select the *Change Immediately* checkbox. Otherwise, the changes will take effect after the next restart of the database.

4. To confirm your entries, choose the *Execute* pushbutton.

Adding Buffer Pools

1. On the *Buffer Pools* screen, select a buffer pool from the list and choose the *Add* pushbutton.
   The *Add Buffer Pool* dialog box appears.

   > **i** Note
   >
   > By default, all available partitions that relate to the new buffer pool are listed. To reduce this list, you can choose the *On Selected Database Partition Groups* radio button in the *Database Partition Groups* area, and select single database partition groups.

2. Specify a name for the new buffer pool.

   > **→** Recommendation
   >
   > We recommend that you use uppercase letters for the buffer pool name. Using lowercase letters or special characters makes it less comfortable to access the selected database partition group with the DB2 command line processor.

3. In the *Size* and *I/O Units* areas, enter the relevant technical settings, such as the page size.
4. If the buffer pool is specified for a set of database partition groups only and not yet for all database partition groups, you can add database partition groups by selecting them in the *Database Partition Groups* list.
5. If you want the new buffer pool to be created immediately, select the *Create Immediately* checkbox. Otherwise, the newly created buffer pool is created after the next restart of the database.
6. To confirm your entries, choose the *Execute* pushbutton.

Deleting Buffer Pools

1. On the *Buffer Pools* screen, select a buffer pool from the list and choose the *Delete* pushbutton.
2. Confirm the deletion.

SQL Statements

In the *Change Buffer Pool* and *Add Buffer Pool* dialog boxes, the generated SQL statement(s) to be executed are displayed. This area is automatically filled and refreshed as soon as any changes are entered correctly.

9.11 Configuration: Special Tables Regarding RUNSTATS

You can access information about tables with special `RUNSTATS` characteristics by calling the DBA Cockpit and choosing *Configuration* > *Special Tables Regarding RUNSTATS* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI) of the DBA Cockpit.

There are two categories of tables that are treated by the optimizer in a special way due to their characteristics:

- Tables marked as `VOLATILE` in the system catalog
  A volatile table is a table whose content can vary from a few entries to a very large amount of entries at lifetime, that is, statistics data is often out-of-date and may result in wrong access plans by the optimizer. These tables should be marked as `VOLATILE` and should have no statistics at all.

- Tables with RUNSTATS control parameters that are not in accordance with the default settings of the DBA Planning Calendar, for example, special scheduling patterns, different kinds of RUNSTATS, or tables that have a profile that may influence automatic RUNSTATS

On the basis of the list displayed, you can check system catalog-related information against the DBSTATC control table.

On the *Special Tables Regarding RUNSTATS* screen, the following information is displayed:

Column	Description
Table Schema	Name of the schema to which the table belongs
Table Name	Name of the database table
VOLATILE	Indicates whether the table is flagged as VOLATILE in the system catalog If the table is flagged as VOLATILE, statistics are not used by the optimizer.
Active	Type of entry in control table DBSTATC The following values are possible: • *N* No RUNSTATS is run by any DBA Planning Calendar job. This status corresponds to the VOLATILE attribute of a database table, which prevents the query optimizer from using statistics. • *R* No RUNSTATS is run automatically by any DBA Planning Calendar job. The only exception is that you explicitly schedule a RUNSTATS for the relevant table. • A RUNSTATS may be run by DBA Planning Calendar jobs. This information is only relevant if RUNSTATS is performed by DBA Planning Calendar jobs and not by DB2 automatic RUNSTATS.
Profile	Indicates whether a RUNSTATS profile was set for the table
RUNSTATS Date	Date of the last RUNSTATS in the system catalog table
RUNSTATS Time	Time of the last RUNSTATS in the system catalog table
Cardinality	Number of rows as calculated by the last RUNSTATS (*–1* indicates that there are no statistics available.)

For more details about a table, choose a table from the list. The *Single Table Analysis* screen appears providing detailed information about the selected table. For more information, see Analyzing and Maintaining a Single Table [page 152].

9.12 Configuration: Data Classes

> **i Note**
>
> This function is only available for SAP ABAP systems.

The technical settings of SAP tables define data classes that need to be related to database tablespaces or tablespace pools. For more information about tablespace pools, see SAP Note 2267446.

You can access the list of available data classes by calling the DBA Cockpit and choosing *Configuration* > *Data Classes* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI) of the DBA Cockpit.

On the *Data Classes* screen, the following information is displayed:

Column	Description
State	• Green: No action required • Yellow: Indicates warnings > **→ Recommendation** > We strongly recommend that you take immediate action. • Red: Indicates errors, immediate action required The following errors are checked: • Is there a related data tablespace or tablespace pool? • Does the data tablespace or tablespace pool exist in the database? • Is there a related index tablespace or tablespace pool? • Does the index tablespace or tablespace pool exist in the database? • Does the name of the tablespace or tablespace pool comply with the naming conventions for the customer namespace? • Is there a description for the data class?
Data Class	Name of the data class known to the ABAB Dictionary
Data Tablespace	Name of the tablespace or tablespace pool where table data is stored
Index Tablespace	Name of the tablespace or tablespace pool where table indexes are stored
No. of Tables	Number of tables within the related data tablespace or tablespace pool

Column	Description
No. of Indexes	Number of indexes within the related index tablespace or tablespace pool
Category	Category of the data class
Description	Description of the data class

In addition, you can maintain data classes, that is, change, add, or delete them as described in *Maintaining Data Classes*.

9.12.1 Maintaining Data Classes

Use

Using the data class list on the *Data Classes* screen, you can maintain data classes as follows:

- Change data classes
- Add new data classes
- Delete data classes

> **i** Note
>
> The configurations that store the data class information are buffered by the table buffer of the SAP application server. If you perform any changes, these buffers must be synchronized. Since this is possible only for the local system, changes might not become automatically visible in the remote system.

Procedure

Changing Data Classes

1. On the *Data Classes* screen, select a data class and choose the *Edit* pushbutton.
2. If required, change the description in the *Change Data Class* dialog box.
3. Change the tablespace assignment by selecting an entry from the dropdown list in the *Data Tablespace* field. Tablespace pools will also appear in this list.
4. Save your changes.

> **⚠** Caution
>
> Changing the related tablespaces does not affect already existing tables. It only has an impact on new tables.

Adding Data Classes

1. On the *Data Classes* screen, choose the *Add* pushbutton.
2. In the *Add Data Class* dialog box, specify a unique name for the data class while making sure that you are following the naming conventions for customer-defined data classes: Names with the prefixes *Z*, *Y*, or *USR*, as well as the names *USER2* to *USER9* are permitted. If you do not follow these naming conventions, you might get an error message.

   > ⚠ Caution
   >
   > If you do not define data classes according to these naming conventions, it will have an impact on future upgrades of your system. Such entries are not recognized as customer entries and will be lost during the upgrade.

   > ℹ Note
   >
   > You cannot enter a value in the *Category* field. It always has the value *USR*.
3. Enter a description.
4. Select a data and index tablespace from the dropdown list.
5. Save your entries.

Deleting Data Classes

1. On the *Data Classes* screen, select a data class and choose the *Delete* pushbutton.
2. Confirm the deletion.

9.13 Configuration: Monitoring Settings

Use

On the *Monitoring Settings* screen, you can perform the following actions to configure the monitoring settings:

- Change the retention periods for history data
  This applies only to systems where you have selected the *Enable Deprecated Data Collection Additionally (Not Recommended)* checkbox on the *System Configuration* screen.

  > ℹ Note
  >
  > This function has been deprecated. We recommend that you use the data collection framework (DCF) to collect history data.

  For more information, see Configuring Systems for Remote Monitoring Manually [page 39].
- Activate the automatic switching of the `db2diag.log` file
- Specify a dedicated server for the background processing

Procedure

1. Call the DBA Cockpit and choose *Configuration* > *Monitoring Settings* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI) of the DBA Cockpit.
2. On the *Monitoring Settings* screen, choose the *Change* pushbutton.
3. Enter your changes in retention time of the statistics in the relevant fields under *Database and Tablespace Size*, *Table and Index Size*, and *Database Performance*.
4. If you want to switch the `db2diag.log` file to restrict its size to a manageable value automatically, select the *Switch Weekly* checkbox.
   This means that the `db2diag.log` file is saved under a new name with a time stamp and a new `db2diag.log` file is created.
5. If you want to collect history data on a dedicated background server, specify a server in the *Server for Data Collection* field by choosing a name from the dropdown list.
6. Save your changes.

9.14 Configuring Automatic Maintenance Functions

Use

We recommend that you automate maintenance tasks for your database by configuring DB2 automatic maintenance functions in the DBA Cockpit. You should check the settings from time to time to make sure that they meet the requirements of your production system.

You can configure the following automatic maintenance functions on the *Automatic Maintenance* screen:

- *Automatic Backup*
- *Automatic RUNSTATS*
- *Automatic REORG*

> **i** Note
>
> If you enter this task area for the first time, certain settings are preselected based on the DB2 default policy.

Procedure

1. Call the DBA Cockpit and choose *Configuration* > *Automatic Maintenance* on the *Database* tab page (Web Dynpro) or in the navigation frame (SAP GUI) of the DBA Cockpit.
   The *Automatic Maintenance* screen appears displaying the following tab pages:
   - *General*
   - *Automatic Backup*
   - *Automatic RUNSTATS*
   - *Automatic REORG*

   Each of these tab pages provides detailed information about the currently selected settings for each of the functions as well as general information applying to all functions. These tab pages are described in more detail later in this document.

2. To configure the automatic maintenance functions in the web browser-based DBA Cockpit, choose the *Configure* pushbutton. The *Configure Automatic Maintenance Functions* wizard appears. In the SAP GUI, there is no *Configure* pushbutton and no wizard, but you select your configuration options directly on the tab pages that correspond to the wizard steps. The wizard guides you through the following configuration steps:

Step	Action
Introduction	Provides an overview of the available configuration steps

General	**i Note**
	To be able to use the automatic maintenance functions, you **must** select the checkbox *Automatic maintenance is switched on*.
	• Specify the maintenance windows during which automatic maintenance is performed by DB2:
	  • *Online Maintenance Window*
	  Time frame with only low activity on the database
	  • *Offline Maintenance Window*
	  Time frame with no activity on the database
	• Select the required maintenance task, for example, *Automatic REORG*, for one of the maintenance windows. DB2 then decides if any action is required and automatically triggers the correct action.

Online Window / *Offline Window*	**i Note** Since the tab pages for specifying the online and the offline maintenance windows are identical, they are only described once. Specify the following parameters: • *Time* • *Day of Week* • *Day of Month* • *Month of Year* **⚠ Caution** The definition of all time-related parameters is combined by AND. Therefore, a valid maintenance window must meet all definitions.
Backup	Specify the following: • *Backup Operation Mode* • *Start Conditions* • *Backup Media*

Step	Action
RUNSTATS	Specify the following: • *Configuration Profile* • *Table Filter* **i Note** You can **only** specify a table filter if you select *Custom* as *Configuration Profile*.
REORG	Specify the following: • *Configuration Profile* Choose either the *SAP Default* policy or *Custom* mode. • *Index Reorganization Mode* • Parameters such as the *Maximum Table Size*, *Reclaim Size for MDC/ITC/BLU Tables*, *Reclaim Size for Indexes*, *Number of Pseudo-Empty Index Pages for Volatile Tables*, whether a temporary table-space should be used, or whether you want to keep the compression dictionary. **i Note** You can only specify the index reorganization mode and the above mentioned parameters if, for your configuration profile, you do not select the SAP default policy, but choose to configure in custom mode.
Summary	Summarizes all actions to be performed To apply your configuration entries and to exit the wizard, choose the *Execute* pushbutton.

More Information

For more information about reorganizations using the automatic maintenance functions, see SAP Note 975352 and Automatic Maintenance Settings: Automatic REORG [page 239].

9.14.1 Automatic Maintenance Settings: General Settings

When you open the *Automatic Maintenance* screen, the *General* tab page is displayed by default providing information about the maintenance windows during which automatic maintenance is performed by DB2.

The maintenance windows are as follows:

- *Online Maintenance Window*
  Time frame with only low activity on the database. For example, during an online maintenance window, you can still be connected to the database.
- *Offline Maintenance Window*
  Time frame with no activity on the database. For example, during an offline maintenance window, neither connections to the database are allowed nor updates for tables and indexes while they are being reorganized.

> **i Note**
>
> In this context, online and offline does not mean the state of the database itself but the time frame with only low activity (online) or no activity (offline) on the database. Since the tab pages specifying the online and the offline maintenance windows are identical, they are only described once.

On the *Online Maintenance Window* and *Offline Maintenance Window* tab pages, the following information is displayed:

Parameter	Description
Automatic maintenance is switched on	Enables or disables automatic maintenance To be able to use the automatic maintenance function, this checkbox must be selected.
Online maintenance window is enabled / *Offline maintenance windows is enabled*	Indicates if the online or offline maintenance window was enabled
Time of Automatic Maintenance	Indicates if the maintenance window was specified directly or inverted
Time	Time frame when the automatic maintenance function is to be performed
Day of Week	Days of the week during which the automatic maintenance function is to be performed
Day of Month	Days of months during which the automatic maintenance function is to be performed You can specify days or day ranges that are separated by comma, for example, *1, 7–10*.
Month of Year	Months of the year during which the automatic maintenance function is to be performed

> **i Note**
>
> To change the settings, choose the *Configure* pushbutton. For more information, see Configuring Automatic Maintenance Functions [page 232].

9.14.2 Automatic Maintenance Settings: Automatic Backup

The *Automatic Backup* tab page provides the following information:

Parameter	Description
General	
Automatic backup is switched on	Enables or disables the automatic backup function
Backup Operation Type	Specifies the type of backup (online or offline) When an online backup is performed, you are still able to access the database during the online maintenance window. When an offline backup is performed within the offline maintenance window, you cannot access the database.
Priority	Specifies the priority of the automatic backup over the other automatic maintenance features such as *Automatic RUNSTATS* or *Automatic REORG* **i Note** `1` means highest priority.
Starting Conditions	
Optimize for Database Recoverability	Backups are created more frequently. Therefore, less time is required to recover the database. The following limits apply: • Maximum time between backups: 1 day • Maximum log space used between backups: 10 MB
Balance Between Recoverability and Performance	Indicates the balance between the number of backups and the time for recovery. The following limits apply: • Maximum time between backups: 7 days • Maximum log space used between backups: 25 MB
Optimize for Database Performance	Fewer backups are created. Therefore, more time to recover the database is required. The following limits apply: • Maximum time between backups: 30 days • Maximum log space used between backups: 50 MB

Parameter	Description
Customize	If you choose this option, you can customize the following parameters: • *Number of full backups is less than* Minimum number of backups • *Time since last full backup exceeds <value> days* Maximum time between backups • *Log space used since last full backup is more than <value> MB* Maximum log space between two backups
Number of full backups is less than	If the number of backups is less than the specified value, a backup is created.
Time since last full backup exceeds	If the time since the last backup exceeds the specified value, a backup is created.
Log space used since last full backup is more than	If the log space exceeds the specified value, a backup is created.
Backup Media	
File System	The backup is created in the specified file systems. If you choose *File System*, you also have to specify the *File Paths* where the backup is to be created.
Tape Device	The backup is created on tape. If you choose *Tape Device*, you also have to specify the *Number of Parallel Sessions*.
TSM	The backup is created and stored in IBM Tivoli Storage Manager (TSM). If you choose *TSM*, you also have to specify the *File Paths*.
XBSA	The backup is created using the XBSA API for storing the data. If you choose *XBSA*, you also have to specify the *Number of Parallel Sessions*.
Vendor Library	The backup is created and data is stored using a vendor library. If you choose *Vendor Library*, you also have to specify the *Location* (that is, a path and file name of the library) and the *Options*.

> **i Note**
>
> To change the settings, choose the *Configure* pushbutton. For more information, see Configuring Automatic Maintenance Functions [page 232].

9.14.3 Automatic Maintenance Settings: Automatic RUNSTATS

The *Automatic RUNSTATS* tab page provides the following information:

Parameter	Description
General	
Automatic RUNSTATS is switched on	Enables or disables the automatic `RUNSTATS` function
Maintenance Window	Specifies that automatic `RUNSTATS` can only be performed in the online maintenance window
Parameters	
Configuration Profile	If you select the value *SAP Default*, the table filter editor is pre-filled with SAP-specific filter conditions. If you select the value *Custom*, you can define your own filter conditions in the table filter editor.
Table Filter	Defines a filter condition for tables to be excluded from `RUNSTATS`

In addition, the following information is displayed for tables that are excluded from `RUNSTATS`:

Column	Description
Tables Excluded from RUNSTATS	The excluded tables are divided into the following categories: • *Volatile Tables* Volatile tables are always excluded from automatic `RUNSTATS`. If you expand this node, the volatile tables are displayed. • *Tables Excluded by Policy* The tables that have been excluded from automatic `RUNSTATS` according to the table filters. If you expand this node, the excluded tables are displayed.
Schema	Name of the schema to which the table belongs
Volatile	Indicates whether the table is flagged as `VOLATILE` in the system catalog If the table is flagged as `VOLATILE`, statistics are not used by the optimizer.
Active	A value of *N* indicates whether the table is marked as VOLATILE by SAP
Profile	Indicates whether a `RUNSTATS` profile was set for the table
RUNSTATS Date	Date of the last `RUNSTATS` in the system catalog table
RUNSTATS Time	Time of the last `RUNSTATS` in the system catalog table

Column	Description
Cardinality	Number of rows as calculated by the last `RUNSTATS` (*–1* indicates that no statistics are available.)

> **i** Note
>
> To change the settings, choose the *Configure* pushbutton. For more information, see Configuring Automatic Maintenance Functions [page 232].

9.14.4 Automatic Maintenance Settings: Automatic REORG

Use

To automate the regular maintenance activity of reclaiming space and cleaning up indexes, we recommend that you use `DB2 AUTO REORG` (see also SAP Note 975352).

Features

On the *Automatic REORG* tab page, the following information is displayed:

Parameter	Description
General	
Automatic REORG is switched on	Enables or disables the automatic `REORG` function
Index Reorganization Mode	Specifies a maintenance window for index reorganization → Recommendation We recommend that you reorganize indexes during the online maintenance window.
Parameters	**Description**
Configuration Profile (Web Dynpro) or *SAP Default Policy* (SAP GUI)	If you choose the *SAP Default* policy, all configuration parameters are prefilled with SAP-specific default values. If you choose *Custom* mode, you can define your own configuration settings.

Parameter	Description
Use a System Temporary Tablespace with Compatible Page Size	If you select this option, a copy of the table or index is created in the temporary tablespace, and the table or index is copied to the original tablespace. Since temporary tablespaces in SAP systems are SMS tablespaces, the required space for defragmentation will be available after the reorganization. With the SAP default policy, this option is selected, but an offline `REORG` is not performed. **→ Recommendation** If you want to execute an offline table reorganization, we recommend that you use a system temporary tablespace.
Maximum Table Size (Offline Table and Online Index REORG)	Specifies that tables that are larger than your defined size are excluded from offline `REORG`, even if an offline window is defined. The SAP default setting excludes all tables from offline `REORG`. In addition, the following applies for Db2 9.5 as of FP7, Db2 9.7 as of FP4, and Db2 10.1 and higher: Indexes on tables that are larger than your specified size are excluded from full online index `REORG`. The SAP default setting excludes all indexes from full online index `REORG`. Note that an index `CLEANUP` will still be executed if needed. **→ Recommendation** We recommend a maximum table size filter of 1 KB.
Reclaim Size for MDC/ITC/BLU Tables	Free space reclaim for MDC tables, ITC tables, and column-organized tables (BLU tables) is executed in the online window if your specified value or more free space can be reclaimed. **i Note** This option is only available for Db2 10.1 and higher. It replaces the `reclaimExtentsSizeForMDCTables` policy attribute of Db2 9.7. **→ Recommendation** We recommend a minimum reclaim size of 100,000 KB.
No. of Pseudo-Empty Index Pages for Volatile Tables	Specifies that an index cleanup is executed for indexes on volatile tables if at least 20 index pages are filled with pseudo-deleted keys. **→ Recommendation** We recommend a minimum number of 20 pages. **i Note** This option is only available as of Db2 9.7 FP4.

Parameter	Description
Reclaim Size for Indexes	Free space reclaim for indexes is executed in the online window if your specified value or more free space can be reclaimed. **→ Recommendation** We recommend a minimum value of 50,000 KB. **i Note** This option is only available as of Db2 10.1.
Compression Dictionary	Specifies whether you want to keep or rebuild the compression dictionary during offline reorganizations. With the SAP default policy, this parameter is set to *Rebuild*, but an offline `REORG` is not performed. A rebuild of the compression dictionary can lead to a better compression ratio but implicates a longer `REORG` runtime. **→ Recommendation** We recommend that you rebuild the compression dictionary.
Table Filter	Specifies filter criteria for tables to be excluded from the automatic `REORG`.

In addition, the following information is displayed for tables that are excluded from automatic `REORG`:

Setting	Description
Tables Excluded from REORG by Table Filter	Specifies tables to be excluded from automatic `REORG` by your chosen table filter. We recommend that you use the following clause: `TABSCHEMA NOT LIKE 'SYS%' AND (TABSCHEMA, TABNAME) NOT IN (SELECT TABSCHEMA, TABNAME FROM SYSCAT.EVENTTABLES)` This clause excludes the DB2 event monitor tables from any `AUTO REORG` activity (see SAP Note 1687901). If you maintain an `AUTO REORG` policy manually, you must specify this clause in every case.
Schema	Name of database schema to which the table belongs
Table Flagged	Table to be reorganized
Index Flagged	Indexes to be reorganized
Table Size (KB)	Size of the table in KB

More Information

- SAP Note 975352
- Reorganization of Database Objects in our Database Administration Guide for SAP on IBM Db2 for Linux, UNIX, and Windows on SAP Help Portal

9.15 Configuration: Encryption

> **i** Note
>
> This screen is only available as of DB2 10.5 Fix Pack 5.

You can use DB2 native encryption to protect the data of your SAP system by creating an encrypted database or by performing encrypted database backups. On the *Encryption* screen, you can display information about the current encryption settings of your database. The *Encryption* screen is available under *Configuration* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI) of the DBA Cockpit.

If you want to make encryption settings, you can do so on the following, different DBA Cockpit screens:

- To display and, if permitted, change values for the database parameters `ENCRYPTED_DATABASE`, `ENCRLIB`, and `ENCROPTS`, go to the *Configuration* > *Database* screen.
- To display and, if permitted, change values for the database manager parameters `KEYSTORE_TYPE` and `KEYSTORE_LOCATION`, go to the *Configuration* > *Database Manager* screen.
- To schedule DB2 backup jobs with encryption, use the `DBA Planning Calendar` (under *Jobs*). You can schedule backup jobs using the DB2 default libraries or a specified custom library. You can also specify library options.

More Information

Configuration: Database [page 206]

Configuration: Database Manager [page 204]

DBA Planning Calendar [page 255]

Scheduling Database Backups [page 263]

9.16 Data Collection Framework: Configuration

Use

The *Data Collection Framework: Configuration* screen provides an overview of the state of the installed history data collectors. You can use this screen to change the configuration of specific data collectors. Furthermore, you can set up the data collection framework (DCF) using templates as described in Setting Up the Data Collection Framework (DCF) Manually [page 51].

You can access information about the data collectors by calling the DBA Cockpit and choosing *Configuration* > *Data Collection Framework: Configuration* on the *Database* tab page of the web browser-based version of the DBA Cockpit or in the navigation frame of the SAP GUI-based version of the DBA Cockpit.

Prerequisites

If you want to use the maintenance functions of the *Data Collection Framework: Configuration* screen in the SAP GUI-based version of the DBA Cockpit, you must implement SAP Note 1806649.

Features

Error Messages About Tablespace Status

If the database is not enabled for automatic storage and the `SAPTOOLS` or `SAPEVENTMON` tablespaces are not available, the data collection framework is not properly set up. In this case, the appropriate messages with detailed setup information are displayed.

System Settings and Ownership

In the *System Settings* and *Ownership* sections in the upper area of the *Data Collection Framework: Configuration* screen, status data such as the following is displayed:

- **Whether the automatic back-end update function is enabled or disabled**
  By default, the automatic back-end update function is enabled. This means that once per day, the installed data collection framework is checked. If it needs to be updated due to SAP code changes or database changes, these updates are performed automatically.
- **Which task scheduler is used**
  By default, the Db2 Administrative Task Scheduler is used to execute stored procedures for the data collection framework regularly. If for some rare individual reason you have been using an external scheduler instead of the Db2 Administrative Task Scheduler, you can migrate the tasks to the Db2 Administrative Task Scheduler using the*Migrate to DB2 Admin Task Scheduler* link.
- **Owners and ownership**
  The DBA Cockpit displays information about the ownership of the data collectors. The owner is the system from which a database administrator can set up, update, patch, remove, or drop data collectors. If you are in a single ownership mode, only the primary owner can maintain data collectors. If you are in a

dual ownership mode, the supplementary owner can maintain the data collectors that the primary owner cannot cover. For more information, see Ownership of Data Collectors [page 247].

Individual Collector Settings

Below the status data, a list of data collectors and their individual settings is displayed providing the following information:

Column	Description
Data Collector	Name of the data collector Typically, the name refers to the data that is collected, for example, the data collector *Tables* stores table-related metrics in the history tables.
Type	Type of data collector The following collector types are possible: • *SAP Data Collection and Preparation (STP)* These data collectors have their own mechanism to calculate the delta of the different snapshots. • *DB2 Event Monitor and Preparation (STP)* A DB2 event monitor is used to store the data in tables. • *Cleanup of Message Logs (STP)* The *Framework* data collector is not a typical data collector type, but a cleanup job for the messages generated by the above-mentioned data collector types.
Setup	Indicates the setup status The following values are possible: • Green: The data collector is set up as intended. • Yellow: The data collector is not set up properly. We recommend that you repair the data collector. • Red: The data collector is not set up. Set up the data collector according to the assigned template. • Gray: The data collector is not set up and the assigned template also marks this data collector to be deactivated.
Collection Interval	Schedule of the data collector A data collector should always remain scheduled. Otherwise, the housekeeping of event monitor-based data collectors or of the *Network Statistics* data collector is disabled which would lead to perpetually growing tables.

Event Monitor Active	State of the DB2 event monitor The following values are possible: • Green: The DB2 event monitor is set up as intended. • Yellow: The DB2 event monitor is not set up properly. We recommend that you repair the data collector. • Red: The DB2 event monitor is not set up. Set up the data collector according to the assigned template. • Gray: No DB2 event monitor is used by the data collector.
Available Days	Availability of the collected data for the given number of days as of now
Days Kept in History	Number of days of the sliding history window The data in the history is deleted after this number of days.
Size on Disk (MB)	Space consumption in the local history of the data that is stored by the data collector
Errors	Number of errors that the data collector has captured and that the scheduler has recorded
Warnings	Number of warnings that the data collector has captured and that the scheduler has recorded
Owner	SAP system ID of the owning DBA Cockpit of that data collector
Version	SAP system version of the owning DBA Cockpit of that data collector

When you double-click a list entry, more information about the selected data collector is displayed:

- On the *Configuration* tab page, details of the data collector such as version, recurrence pattern, retention period of historical data, and collector options are displayed.
- On the *Object Log* tab page, you find the data collector log where you can check for each collector run if there are errors, warnings, or info messages.
- On the *Execution Statistics* tab page (SAP GUI only), you get statistics on the runtime execution and can check the performance of the data collection and its impact on system performance.

9.16.1 Maintaining Data Collectors

1. Call up the DBA Cockpit and choose *Configuration* > *Data Collection Framework: Configuration* on the *Database* tab page (Web browser) or in the navigation frame of the DBA Cockpit (SAP GUI).

2. To maintain data collectors, that is, for example, to set them up, change, or analyze them, or to remove data from them, choose a data collector from the list.
   In the content detail area of the *Configuration* screen, the *Configuration*, the *Data Collector Log*, and the *Execution Statistics* tab pages appear, displaying the relevant data collector details.
3. Depending on your setup status and on the ownership status of your DBA Cockpit system, you can switch the screen to change mode and proceed with the following maintenance actions:

Action	Setup Status	Steps to Be Performed
Setting up a single data collector	Gray or red as the data collector has not been set up yet	On the *Configuration* tab page, choose the *Set Up* pushbutton. **i Note** You must set up the *Framework* collector before all other data collectors.
Repairing a single data collector	Yellow as the data collector is set up but should be updated or needs to be repaired	On the *Configuration* tab page, choose the *Repair* pushbutton.
Changing a single data collector (in the web browser-based version of the DBA Cockpit)	Green	1. On the *Configuration* tab page, choose the *Change Settings* pushbutton and enter your changes. 2. To apply your changes, choose the *Apply Changes* pushbutton. If you want to discard your changes, choose the *Reset Changes* pushbutton.
Changing a single data collector (in SAP GUI)	Green	1. Toggle the screen from display to change. 2. On the *Configuration* tab page, enter your changes. 3. Save your changes.
Removing data of a single data collector	Yellow or green	On the *Configuration* tab page, choose the *Prune Data* pushbutton. By doing so, the collected data and the generated messages of the data collector are removed.
Dropping a single data collector	Yellow or green	On the *Configuration* tab page, choose the *Drop* pushbutton.

Action	Setup Status	Steps to Be Performed
Analyzing a single data collector	Yellow or green	On the *Data Collector Log* tab page, you can choose between the following push-buttons: • *Show Error Messages* • *Show Warnings* • *Show Information Messages*

9.16.2 Ownership of Data Collectors

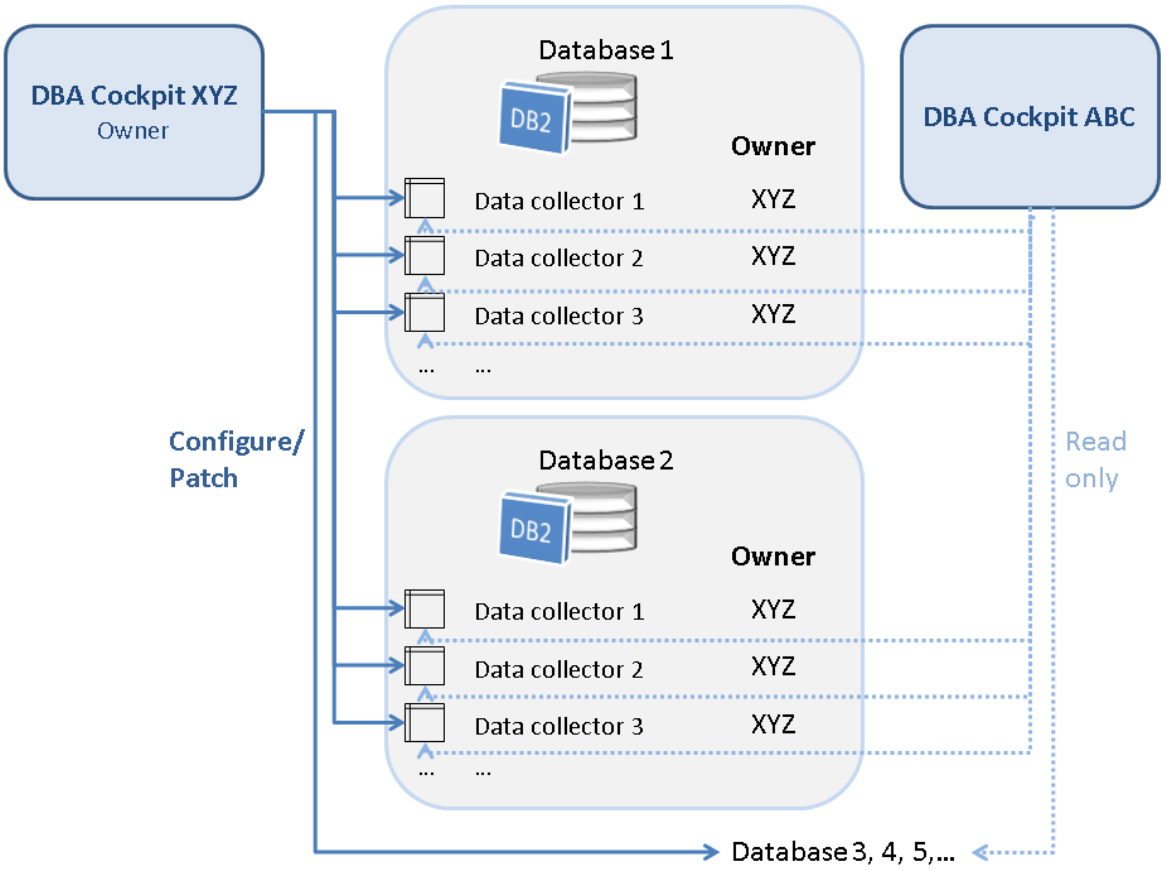
Concept

The data collection framework serves to collect data from the databases monitored by the DBA Cockpit. The DBA Cockpit provides an ownership concept that allows you to maintain data collectors for the IBM DB2 databases in your system landscape centrally from an SAP Solution Manager system or a system where the DBA Cockpit is running. With this ownership concept, you can have one system that owns the data collectors of the DB2 databases. This means that this is the only system from which a database administrator can set up, update, patch, remove, or drop data collectors. This system is referred to as the owner system.

The idea of this ownership concept is that you can roll out data collectors in a system landscape using one dedicated SAP Solution Manager system. This way, you can distribute patches, updates, or configurations in the system landscape using the template mechanism in the DBA Cockpit. When you work with a single owner system, other systems running the DBA Cockpit can also access the data collectors, but only in read-only mode, so they cannot overwrite changes made to data collectors by the SAP Solution Manager system. The advantage of having a primary owner of data collectors is that you can centrally provide the most up-to-date data collectors for DB2 databases that SAP Solution Manager needs for monitoring. This saves you applying SAP Notes individually for each managed system.

Single Ownership

The following graphic shows an example of a system landscape with two DBA Cockpits, with DBA Cockpit XYZ as the single owner of the data collectors and DBA Cockpit ABC with read-only authorizations:



System Landscape with Single Ownership

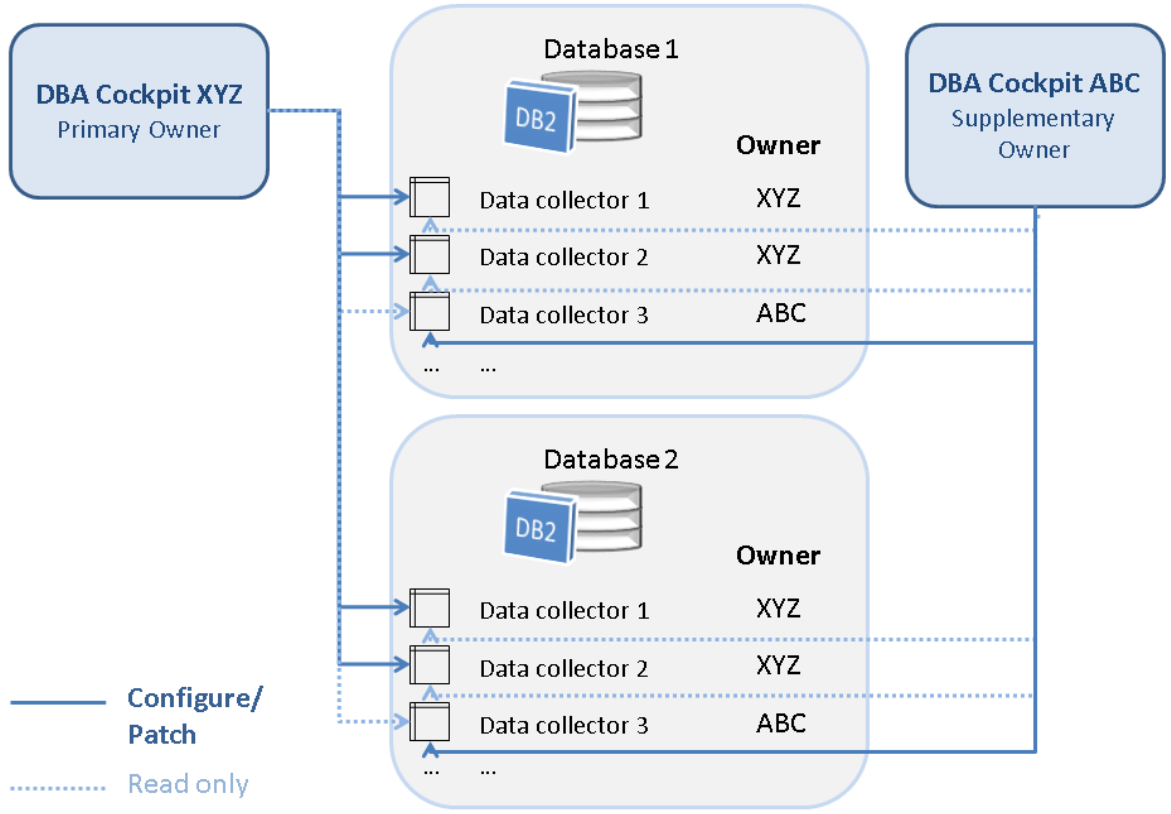
Dual Ownership

It might be the case that the owner system of the data collectors is on a lower release or support package level than another SAP Solution Manager system. Then, the more up-to-date SAP Solution Manager system is able to provide and patch data collectors that the current owner system cannot cover because its release or support package level is too low. This is the case, for example, when you want to test a new SAP Solution Manager version on another installation in parallel to your productive SAP Solution Manager system. In this case, you can switch to a dual ownership: In addition to the primary owner, you define the more up-to-date SAP Solution Manager system as a supplementary owner. In the supplementary owner system, the database administrator can set up, update, patch, remove, or drop only those data collectors that the primary owner cannot cover.

> **i** Note
>
> You can allow a dual ownership only if the two owner systems are based at least on the following releases and support package levels:
>
> - SAP NetWeaver 7.02 with Support Package 10 or higher

- SAP NetWeaver 7.03 with Support Package 02 or higher
- SAP NetWeaver 7.30 with Support Package 05 or higher
- SAP NetWeaver 7.31 with Support Package 02 or higher
- SAP NetWeaver 7.40 with Support Package 02 or higher

The following graphic shows an example of a system landscape with two DBA Cockpits, with DBA Cockpit XYZ as the primary owner of the data collectors and DBA Cockpit ABC as the supplementary owner:



System Landscape with Dual Ownership

9.16.3 Changing the Ownership of Data Collectors

Procedure

1. In the DBA Cockpit, choose *Configuration* > *Data Collection Framework: Configuration*.
   In the system data area of the *Data Collection Framework: Configuration* screen, the ownership status is displayed.
2. To change the ownership details, switch to change mode (in the SAP GUI) or click *Transfer Ownership* (in the Web Dynpro) and choose the appropriate transfer ownership operation from the dropdown list.
3. After the ownership change, choose *Repair All Collectors*.

→ Recommendation

We recommend that you assign the ownership of data collectors to the SAP Solution Manager system that you use for monitoring your system landscape. An SAP Solution Manager system can always take over the ownership of the data collectors of a monitored database. When an SAP Solution Manager system is the primary owner of the data collectors, only another SAP Solution Manager system can take over this primary ownership. This might be necessary, for example, when the original primary owner experiences severe problems or has a longer downtime. Note, however, that when an SAP Solution Manager system owns data collectors, another DBA Cockpit **not** running on an SAP Solution Manager system cannot take over the primary data collector ownership. It can only run in read-only mode or take over supplementary ownership.

More Information

Data Collection Framework: Configuration [page 243]

9.17 Data Collection Framework: Template Definition

Use

A template in the data collection framework is a set of predefined data collector settings that you can use for different usage scenarios. On the screen *Data Collection Framework: Template Definition*, you can perform the following actions:

- Create new data collection templates
- Change individual data collectors in a template
- Set a default template
- Delete a template
- Edit a template's name and description
- Show where a template is used for data collection in your system landscape

SAP provides two default templates that cannot be modified. You can, however, save them as a new template and then modify them. For more information, see section *Standard Templates for the DCF* in The Data Collection Framework [page 48].

ℹ Note

In the SAP Solution Manager setup wizard for managed systems for the database, the templates are available for use but cannot be maintained.

Features

You can access the screen *Data Collection Framework: Template Definition* by calling the DBA Cockpit and choosing *Configuration* > *Data Collection Framework: Template Definition* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

From the *Template* dropdown list, choose the appropriate template for your setup.

The following information is displayed:

Column	Description
Object	Name of the data collector Typically, the name refers to the data that is collected, for example, the data collector *Tables* stores table-related metrics in the history tables.
Object Type	The following data collector types are available: • *SAP Data Collection and Preparation (STP)* These data collectors have their own mechanism for delta calculation of the different snapshots. • *DB2 Event Monitor and Preparation (STP)* A DB2 event monitor is used to store the data in tables. • *Cleanup of Message Logs (STP)* The *Framework* data collector is not a typical data collector type, but a cleanup job for the messages generated by the above-mentioned data collector types.
Release (min)	Minimum database release required for the data collector
Release (max)	Maximum database release supported by the data collector
Collection Interval	Schedule of the data collector
Days Kept in History	Number of days the collected data is kept before it will be deleted

More Information

Creating a New Data Collector Template [page 252]

Changing an Individual Data Collector in a Template [page 252]

Setting the Default Template [page 253]

Deleting a Template [page 254]

9.17.1 Creating a New Data Collector Template

Use

You can create a new data collector template either by copying an existing template or by generating a template that is based on the configuration of an individual system.

Procedure

Copying an Existing Template

1. On the *Template Definition* screen, choose a template from the *Template* dropdown list and choose the *Save As New Template* button.
   The *Save As New Template* dialog box appears.
2. Enter a unique name and a description of the new template.
3. Save the template.

Creating a Template Based on a System Configuration

1. On the *Data Collection Framework: Configuration* screen, choose the *Save as New Template* button.

   > **i** Note
   >
   > The *Save As New Template* button is only enabled if the respective system configuration fulfills the minimum template requirements of SAP that are defined by the *SAP Default* template. This means that every new template must have a finer time granularity or a longer retention time for the history data than the *SAP Default* template does.

2. In the *Save As New Template* dialog box, enter a unique name and a description of the new template.
3. Save the template.

9.17.2 Changing an Individual Data Collector in a Template

Use

You **cannot** change an SAP template. Instead, you can create a custom template based on an SAP template and change this newly created template.

> → Recommendation
>
> In some areas, the DBA Cockpit joins data that is meant to be collected at the same time by different data collectors. Therefore, we recommend that you set the same schedules for the following two groups of data collectors:
>
> - General data collectors
>   - Buffer Pools

- Containers
- Configuration
- Database
- Tablespaces

- Database Objects
  - Indexes
  - Tables

Prerequisites

The *Setup* status of the data collector must be green.

Procedure

1. On the *Template Definition* screen, choose a template that is not an SAP template from the *Template* dropdown list.
2. On the *Configuration* tab page in the content detail area, choose the *Change Settings* button.
3. In the content detail area, enter your changes in the respective fields.
4. Apply your changes.

> **i** Note
>
> If you want to discard your changes, choose the *Reset Changes* button.

9.17.3 Setting the Default Template

Use

The default template is preselected in the template-based setup and in the SAP Solution Manager setup wizard for managed systems, but you can change it manually.

Procedure

To change the default template, choose a template from the *Template* dropdown list and choose the *Set As Default* button.

> **i** Note
>
> The current default template is marked with an asterisk (*). If the default template is currently selected, the *Set As Default* button is disabled.

9.17.4 Deleting a Template

Procedure

1. On the *Template Definition* screen, choose a template that is not an SAP template.

   > **i** Note
   >
   > You **cannot** delete SAP templates or the current default template. If you want to delete the template that is currently the default, you first have to unmark it as the default by setting a different template as the new default.

2. To delete the template, choose it from the *Template* dropdown list and choose the *Delete* button.

   A list of all systems is displayed that are assigned to this template. If there are entries in the list, the default template is assigned to these systems before the template is deleted.

10 Jobs

10.1 DBA Planning Calendar

Use

You use the DBA Planning Calendar to automate database administration actions that have to be performed regularly. You can schedule operations such as online backups, which are then automatically performed, and you can check if the operations were successful.

The main function of the DBA Planning Calendar is to define the start times and parameters for such database actions. Since these actions run without administrator interaction, you have to make sure in advance that the necessary resources are available.

You can start the DBA Planning Calendar by calling the DBA Cockpit and choosing *Jobs* > *DBA Planning Calendar* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

Features

The initial screen of the DBA Planning Calendar consists of the *Selection* area and the calendar.

The Selection Area

The *Selection* area contains all information and parameters to define the set of actions that you want to display. You can:

- Select the category of an action:
  - *DBA Actions*
    These are plannable actions.
  - *External Actions*
    These are plannable actions that have not been started via the DBA Planning Calendar but that have been started manually or by external job schedulers.
  - *All Actions*
    These are all plannable actions, regardless how they have been scheduled.
  - *DB Collectors*
    These are actions that are automatically selected by the system to collect data, for example, on performance or history and that are only available for RFC-monitored systems.
- Select the week to be displayed
  By default, the current week is displayed.
- Select a factory calendar
  Specifying a factory calendar only has an impact on the calendar display. Holidays are the same color as weekend days. It does not result in any restrictions on planned actions.

The Calendar

The calendar displays all scheduled jobs. The status of each job is indicated by different colors. To display the meaning of each color, choose the *Legend* pushbutton. The following information is displayed:

Color	Status	Description	Action
Light blue	*Planned*	The action has not yet started.	–
Dark blue	*Running*	The action has not yet finished.	–
Green	*Finished successfully*	The action has run successfully.	–
Yellow	*Finished with warning*	The action has finished with a warning.	Check the job log for details.
Red	*Finished with error*	An error has occurred and the action was interrupted.	Check the job log for details and reschedule the action.
Dark yellow	*No longer available*	Information about the action is not available anymore because the maximum retention period of the protocol records has been exceeded.	–
Dark red	*Scheduling failed*	Scheduling failed, that is, there is no status available and the action is overdue.	–

> **i** Note
>
> By default, the DBA Planning Calendar appears in display mode. To switch to editing mode, choose the *Change* pushbutton in the global toolbar.

More Information

- Setup Considerations and Prerequisites for the DBA Planning Calendar [page 256]
- Scheduling Typical DBA Actions [page 262]

10.1.1 Setup Considerations and Prerequisites for the DBA Planning Calendar

Setup Considerations

Since there are no planned actions in the DBA Planning Calendar when you start your system for the first time, you need to initially set up the DBA Planning Calendar. The most important thing when setting up the DBA

Planning Calendar is to choose a pattern of actions that covers your regular database administration (DBA) needs and to specify all required action parameters that take account of any dependencies between actions.

You must also consider that there is a number of database-related jobs (for example, performed by the DB2 scheduler) that are not controlled by the DBA Planning Calendar but that you must take into account when scheduling regular actions.

Depending on the underlying DB2 version, some of these jobs are automatically performed by data collectors of the data collection framework (DCF) or by automatic maintenance settings (for example, automatic `RUNSTATS` or automatic `REORG`). If the DCF has been enabled and correctly set up, you can view these data collectors on the screen *Configuration* > *Data Collection Framework: Configuration*.

For more information, see Data Collection Framework: Configuration [page 243].

If the DCF is not enabled and the SAP history collection is used instead, the following jobs run automatically:

- Collection of database performance history data every two hours starting at 00:00
- Monitoring of database and database manager configuration changes daily at 8:00 a.m., 01:00 p.m., and 7:00 p.m.
- Collection of database and tablespace history data daily at 7:00 a.m. and 8:00 p.m.
- Collection of space history data for tables and indexes weekly on Sundays at 12:00 p.m.

> ⚠ Caution
>
> - Some of the available actions have an impact on database performance and availability. In this context, also consider if systems in different time zones are affected. Check the start and end times of scheduled actions to make sure that they do not interfere with each other and that they do not impact other system activities.
> - You **cannot** perform all required DBA actions from the DBA Planning Calendar or the DBA Cockpit. For more information about actions that you must perform during an SAP system downtime, such as an offline database backup, see the *SAP Database Administration Guide* for your database version.

Optionally, you can configure the back end of the DBA Planning Calendar to be able to control the execution of these background jobs.

For more information, see Controlling the Execution of Background Jobs [page 277].

Prerequisites

Before you start using the DBA Planning Calendar, you have to check the following:

- Hardware and backup media
  Check if you have enough hardware (such as tape drives) and backup media (such as tapes) for the backup strategy you intend to use.
  Make sure that the media are initialized and ready so that the operator does not have to interrupt scheduled backup runs. For example, check if tapes are already in the specified tape drive or tape changer. You might require different tape drives for database backups and log backups.
- SAP system authorizations
  Make sure that you have authorization for the DBA and background job scheduling that is provided by profiles `S_RZL_ADMIN` and `S_BTCH_ALL`.

Check if external programs are able to run on the database server so that actions affecting the database can be performed from other application servers.

- Database system authorizations
  Check if actions requiring authorization to connect to the database are permitted. Some actions, such as online backups, require higher levels of authorization.
  Before scheduling database backups, see the information about backing up the database in the *SAP Database Administration Guide* for your database.

For more information, see Authorizations for the DBA Cockpit and Remote Monitoring [page 30].

Next Steps

The following sections provide information about setting up the DBA Planning Calendar initially and performing standard actions:

- *Creating a Pattern of Actions in the DBA Planning Calendar*
- *Scheduling and Displaying an Action*
- *Changing or Deleting an Action*
- *Checking the Execution of Scheduled Actions*
- *Re-Executing an Action*

More Information

Scheduling Typical DBA Actions [page 262]

10.1.2 Creating an Action Pattern in the DBA Planning Calendar

1. On the *Database* tab page of the DBA Cockpit, choose *Jobs* > *DBA Planning Calendar*.

   > **i Note**
   >
   > By default, the DBA Planning Calendar appears in display mode. To switch to editing mode, choose the *Change* pushbutton in the global toolbar.

2. Choose the *Pattern Setup* pushbutton.
   The *Setup Planning Pattern* wizard appears guiding you through the procedure of setting up a pattern of recurring actions that cover your regular DBA needs.

   > **i Note**
   >
   > Depending on your database version and configuration, a set of recommended actions is available and some of these actions might be preselected by default. You may change this selection set, for example,

if you want to use IBM Tivoli Storage Manager (TSM) or backup and archiving instead of backup and archiving to devices.

3. To enter the pattern into the DBA Planning Calendar, choose the *Execute* pushbutton on the *Summary* screen.
4. Exit the wizard.

> ⚠ Caution
>
> When you start using the DBA Calendar in production operation, you **must** check **daily** if scheduled actions have been executed correctly.
>
> For more information, see Checking the Execution of Scheduled Actions [page 261].

10.1.3 Scheduling and Displaying an Action

Scheduling an Action

1. To add new actions to the DBA Planning Calendar, you can do one of the following:
   - Click a calendar cell.
   - Position the cursor on a calendar cell and choose the *Add* pushbutton.
   - Choose *Schedule an Action* in your *Favorites*.

   The *Schedule an Action* wizard appears.
2. In the *Job Selection* step, do the following:
   1. Choose the job that you want to schedule from the *Action* dropdown list.
   2. Select one of the following options:
      - *Start Immediately*
        The job starts immediately in the background after you have completed the required steps in the wizard.
      - *Start on*
        Specify the day and time when the action is to start.
      - *Start as Recurring Action*
        You must specify the recurrence pattern in a later step of the wizard.
3. In the *Parameter* step, change or enter the basic parameters for the action.
4. If you previously selected the *Start as Recurring Action* option in the *Job Selection* step, you now have to enter a *Recurrence Pattern* (that is, an action interval in weeks, days, or hours) and a *Recurrence Range* (that is, a time interval or limited number of occurrences) in the *Recurrence* step.

   > ⚠ Caution
   >
   > The system warns you if there is a conflict with an existing action, but it does not prevent you from inserting the new action.
   >
   > You must decide whether the actions might conflict in terms of database access or performance. The system does not check for conflicts between actions with identical start times but checks for actions within a range of approximately 30 minutes.

5. On the *Summary* screen, review the specified parameters and schedule the job by choosing the *Execute* pushbutton.
6. Exit the *Schedule an Action* wizard.

Displaying a Scheduled Action

The status of an action is indicated by the color of the calendar cell where the action was inserted.

To display details of a scheduled action, click the relevant calendar cell. The *Action Description* detail area appears displaying scheduling information or the return status of a finished action.

The following tab pages are displayed:

- *Action Parameters*
  Displays the parameters of the action
- *Recurrence*
  Displays the recurrence patterns and only appears if the action is part of a recurring action
- *Job Log*
  Displays the background processing job log that has been generated by the action and all messages that have been written by the background job. If available, you can display long texts by double-clicking a message.
- *Program Log*
  Some actions write log files onto the database server. If such a program log exists, it is displayed on this tab page.

10.1.4 Changing or Deleting an Action

Prerequisites

The action to be changed or deleted must have the status *Planned* (that is, not yet executed).

> **i** Note
>
> If an action has already been executed, you can only display it. For more information, see Scheduling and Displaying an Action [page 259].
>
> If you want to change a recurring action, the changes only affect recurrences of the action in the future. The action is split into two actions, one describing the old action and one the new action.

Procedure

1. In the DBA Planning Calendar, click a calendar cell that contains an action.
   The *Action Description* detail area appears displaying the action parameters and, if available, the recurrence pattern.

2. From the *Change* menu button (respectively, the *Delete* menu button), choose one of the following options:
   - If you want to edit or delete the currently selected occurrence only, choose *Selected Occurrence*.
   - If you want to edit or delete all occurrences of a recurring job, choose *All Occurrences*.

   If you have chosen one of the options from the *Delete* menu button, the action is removed from the Planning Calendar.
   If you have chosen one of the *Change* options, the *Change Scheduled Action* dialog box appears displaying the action parameters and, if available, the recurrence patterns.
3. Apply and save your changes.

> **i** Note
>
> If you want to clean up all your scheduled actions completely, that is, delete all jobs, all scheduling data, and all related protocol records, choose the *Cleanup* pushbutton. By doing so, you also reset the DBA Planning Calendar to its initial state.
>
> We recommend that you clean up the DBA Planning Calendar after an SAP system upgrade or if jobs have become corrupt.

10.1.5 Checking the Execution of Scheduled Actions

Context

Since actions that were scheduled in the DBA Planning Calendar might fail, you must check if critical actions, for example, database backups were executed successfully. The following steps describe how to check if an action was executed correctly and how to proceed if it failed.

Procedure

1. Check the status of a scheduled action in the DBA Planning Calendar by its color: If the action was executed successfully, the calendar cell is marked green.
2. If the action terminated with a warning or an error, consult the job log.

   > **i** Note
   >
   > If there is no job log available, the background job was probably not started. In this case, proceed as follows:
   >
   > 1. In your SAP system, call transaction SM37 and display the job overview.
   >   The jobs related to the DBA Planning Calendar start with the prefix “DBA:”. The complete name of such a job in SM37 contains a short description of what the job is about. The following is an example of a `REORG` job:
   >   `DBA:REORG_TABLE_____@095534/4000`.

2. To search for information about possible errors that prevented the job from being executed, double-click the respective background job in the overview list.

3. Correct the error using the information in the job log or in transaction SM37, and re–execute the action in the DBA Planning Calendar.

   Make sure there are no conflicts with other scheduled actions before you re-execute the action.

10.1.6 Re-Executing an Action

Context

If a scheduled action failed or if there is a resource bottleneck that needs immediate attention, you might have to re-execute an action.

> **i** Note
>
> When executing actions immediately, consider the possible impact on system performance (including affected systems that are used, for example, in different time zones). Make sure there are no conflicts with other scheduled actions.

Procedure

1. Click the action that you want to re-execute.

   The *Action Description* detail area of the action appears where you can check the action parameters.

2. Choose the *Re-Execute* pushbutton.

   The action is re-executed immediately.

10.1.7 Scheduling Typical DBA Actions

You can use the DBA Planning Calendar to automate database administration tasks that have to be performed on a regular basis, such as:

- Scheduling database backups
- Archiving log files to tape
- Updating statistics
- Reorganizing tables
- Scheduling a *REORGCHK* for all tables

- Scheduling scripts
- Running the NLS cleanup job

The following sections describe these tasks in more detail.

> **i** Note
>
> Although some of the tasks are already performed by DB2's automatic maintenance functions (for example, automatic `RUNSTATS` or automatic `REORG`), there may be situations where you have to explicitly update statistics or perform a `REORGCHK` for all tables.
>
> For more information, see Configuring Automatic Maintenance Functions [page 232].

10.1.7.1 Scheduling Database Backups

A database backup is a complete copy of your database. To be able to restore the database to a consistent state that is as up-to-date as possible, you have to perform database backups on a regular basis.

You can perform database backups using the DBA Planning Calendar in the DBA Cockpit. Depending on the storage device that you are using, you can choose one of the following jobs:

- *Database Backup into TSM*
  You back up the database to Tivoli Storage Manager (TSM).
- *Database Backup to Device*
  You back up the database to a specified tape or disk.
- *Database Backup with Vendor Library*
  You back up the database to any other vendor storage management product.
- *Snapshot Backup*
  You back up the database using the fast copying technology of a storage device.

  > ⚠ Caution
  >
  > This job is **only** available as of DB2 V9.5.

Procedure

1. To schedule a database backup in the DBA Planning Calendar, choose the *Add* pushbutton or the *Schedule an Action* link in your *Favorites*.
   The *Schedule a New Action* dialog window (in the SAP GUI) or the *Schedule an Action* wizard (in the Web Dynpro) appears. Specify the following parameters:

Parameter	Description
Backup Mode	

Parameter	Description
Online	Access to the database is not blocked. The users can continue to work normally during the backup.
Offline	**i Note** This option is no longer supported and is displayed only for upward compatibility reasons. Backup jobs that have this option are automatically performed as online backups with the *Include Logs* option.
Backup Type	
Full	The complete database is backed up.
Incremental	Cumulative (that is, incremental) backup image An incremental backup image is a copy of all database data that has changed since the most recent successful full backup operation.
Incremental Delta	Non-cumulative (that is, delta) backup image A delta backup image is a copy of all database data that has changed since the most recent successful backup operation.
Compression and Encryption Options	
Compress	The backup is to be compressed.
Encrypt	The backup is to be encrypted.
Use DB2 Default Library or *Use this Library:*	The backup with compression and/or encryption is done using either the DB2 default library or the library that you specify in the relevant input field.
Use Library Options	The backup with compression and/or encryption is done using the library options that you specify in the relevant input field.
Exclude Library from Backup	The backup is performed without storing the compression or encryption library in the backup.
Additional Options	

Parameter	Description
Include Logs	**i Note** Only choose this option if you want to perform an online backup. Only those log files are included in the backup that are required to get a consistent database. Any further log files are not taken into consideration.

⚠ Caution

The following options are only available for downward-compatibility reasons and we strongly recommend that you do **not** set them:

- *Number of Buffers*
- *Buffer Size*
- *Parallelism*

2. Finish scheduling the job by choosing the *Execute* pushbutton on the *Summary* screen of the wizard (Web Dynpro) or by choosing the *Add* (or *Execute Immediately*) pushbutton (SAP GUI).

10.1.7.1.1 Backup Considerations

Use

When performing a backup, you should consider the following:

- Regardless of the selected backup mode, you can only restore the database if you have at least one valid full backup.
- To restore the database completely and to bring the system up-to-date, you have to roll in the log files that were generated after the backup was performed.
- The database may be local or remote. The backup, however, remains on the database server unless a storage management product, for example, Tivoli Storage Manager (TSM), is used.
- After an online backup, DB2 forces the currently active log files to be closed and as a result they are archived. Therefore, an online backup has a complete set of archived log files that are available for database recovery.

Backup of a Multi–Partition Database

You have to back up partition by partition. Therefore, you have to schedule backup jobs for each partition.

In a multi–partition database system, keep a copy of file `db2nodes.cfg` with any backup copy that you take. This copy of file `db2nodes.cfg` is used as a safety copy in case of possible damage to the original file.

> **Note**
>
> As of DB2 V9.5, a “single system view” backup is available that performs the backup for all database partitions in one job. You can use this option by choosing the value *All* for the database partition in the *Schedule an Action* wizard.

Frequency of Backups and Time Required

You should take full database backups on a regular basis, regardless of how often log files are archived. A current full backup means that there are fewer archived log files that you have to apply in case of a database recovery. Therefore, the amount of time that is required by the `ROLLFORWARD` utility to recover the database decreases. In addition, the chance of a log file not being available (for example, due to data corruption or data loss) also decreases.

To reduce the amount of time that the database is not available, we recommend that you consider performing online backups.

> **Note**
>
> You can only use an online backup for recovery if the database log files that were created during the online backup are available.

Advanced Backup Techniques

The following list provides information on advanced backup techniques:

- Incremental or delta backups
  To reduce the backup and restore time, you can use incremental or delta backups. For more information, see the IBM manual *Data Recovery and High Availability Guide and Reference*.
- Backup of a file system copy using the `db2inidb` tool
  For more information about the `db2inidb` tool and its use as a mirror for a backup based on a file system, see the *Database Administration Guide: SAP on IBM DB2 for Linux, UNIX, and Windows*.
- Standby database for backup purposes
  For more information about how to use the `db2inidb` tool to create a standby database for backup purposes, see the *Database Administration Guide: SAP on IBM DB2 for Linux, UNIX, and Windows*.

More Information

Database Administration Guide for SAP on IBM Db2 for Linux, UNIX, and Windows at https://help.sap.com/viewer/db6_admin on SAP Help Portal

10.1.7.2 Archiving Log Files To Tape

Context

You can archive log files to tape using the job *Archive Log Files to Tape* in the DBA Planning Calendar.

Procedure

1. Start the *Schedule an Action* wizard by clicking a calendar cell or by choosing the *Schedule an Action* link in your *Favorites*.
2. In the *Job Selection* step, choose *Archive Log Files to Tape* from the *Action* dropdown list.
3. In the *Parameter* step, specify the required parameters.

   > **i** Note
   >
   > The DB2 tape manager is used to archive log files to tape. Besides the standard parameters (for example, start time, date, number of log files to be saved, and tape label), you can also specify the option of the tape manager to use for archiving log files:
   >
   > - *Double Store*
   > - *Overwrite Expired Tapes*
   > - *Eject Tape at End of Operation*
   >
   > For more information, see the *Database Administration Guide – SAP on IBM DB2 for Linux, UNIX, and Windows*.

4. Schedule the job by choosing the *Execute* pushbutton on the *Summary* and exit the wizard.

Next Steps

For more information, see the *Database Administration Guide for SAP on IBM Db2 for Linux, UNIX, and Windows* at https://help.sap.com/viewer/db6_admin on SAP Help Portal.

10.1.7.3 Updating Statistics

Context

You can use the DBA Planning Calendar to schedule an update of the database statistics. In general, Db2 updates the database statistics automatically using the automatic RUNSTATS function.

If the automatically maintained statistics need to be updated or if a different type of statistics – other than the Db2 default – is required, you can schedule the job *RUNSTATS and REORGCHK for Set of Table* in the DBA Planning Calendar. This job performs a RUNSTATS for a single table or a set of tables that is specified by a name using wildcards.

> → Recommendation
>
> Since the RUNSTATS can affect system performance in case of large tables, we recommend that you schedule the job *RUNSTATS and REORGCHK for Single Table* to run outside normal working hours, for example, on Sundays.

Procedure

1. Start the *Schedule an Action* wizard by clicking a calendar cell or by choosing the *Schedule an Action* link in your *Favorites*.
2. In the *Job Selection* step, choose *RUNSTATS and REORGCHK for Set of Tables* from the *Action* dropdown list.
3. In the *Parameters* step, do the following:
   - Specify the tables where the statistics are to be updated. You can use wildcards for the table names.
   - Specify how you take statistic profiles into account by choosing one of the following options:
     - *Use Customized Settings or Existing Profiles*
       For tables that have statistic profiles, the RUNSTATS is performed using exactly the same settings as specified in the profile. For all other tables, the RUNSTATS is performed as specified in the wizard.
     - *Use Customized Settings Ignoring Existing Profiles*
       If you choose this option, the RUNSTATS is performed as specified in the wizard regardless whether or not statistic profiles exist.
   - Specify the following settings for the RUNSTATS:

Field	Description
Table Analysis Method	

Field	Description
Basic	Basic statistics for the table
Distributed Statistics	Distributed statistics for the table
None	No statistics for the table ⚠ Caution Selecting this option only freezes already existing old table statistics but it does neither delete nor invalidate them.
Sampling of [ ] %	Percentage of entries to be used for sampling
System (Page Sampling)	The data to be sampled is selected page by page.
BERNOULLI	The data to be sampled is selected row by row.
Analyze Key Columns only	Table statistics are gathered only for key columns of the table.
Index Analysis Method	
Basic	Basic statistics for the index are gathered.
Detailed Without Sampling	Detailed statistics for the index are gathered.
Detailed With Sampling of [ ] %	Detailed statistics for the index using sampling are gathered.
System (Page Sampling)	The index data to be sampled is selected page by page.
BERNOULLI	The index data to be sampled is selected row by row.
None	No statistics for the index are gathered. ⚠ Caution Selecting this option only freezes already existing old index statistics but it does neither delete nor invalidate them.

If you have specified more than one table, you can choose the degree of parallelism by setting the value for *Number of Parallel Jobs* higher than one. In this case, the RUNSTATS job can be parallelized if there are enough system resources available (for example, background processes and the appropriate number of processors). Doing so results in additional jobs that are scheduled by the regular DB13 job and which perform the RUNSTATS on tables in parallel. The SAP system makes sure that the number of parallel jobs does not exceed the number of available background processes.

> ⚠ Caution
>
> Handle the parameter *Number of Parallel Jobs* with care. Starting more jobs can have a high impact on the overall system performance even though the parameter *Number of Parallel Jobs* significantly reduces the amount of time for the job execution.

4. Schedule the job by choosing the *Execute* pushbutton on the *Summary* screen and exit the wizard.

10.1.7.4 Reorganizing Tables

Use

You can use the DBA Planning Calendar to schedule a reorganization of a set of tables. In general, Db2 reorganizes the tables using its automatic `REORG` function. If a reorganization is required that is not covered by automatic `REORG` (for example, due to table compression), you can schedule the job *REORG and RUNSTATS for Set of Tables* in the DBA Planning Calendar.

Procedure

1. Start the *Schedule an Action* wizard by choosing the *Schedule an Action* link in your *Favorites* or by clicking a calendar cell.
2. In the *Job Selection* step, choose *REORG and RUNSTATS for Set of Tables* from the *Action* dropdown list.
3. In the *Parameters* step, specify the following parameters:

Parameter	Description
Table Schema	Name of the schema to which the table belongs
Table	Name of the table

Parameter	Description
Offline	Schedules an offline reorganization Optionally, you can also specify the following parameters: • *Use Temporary Tablespace* If you select this option, a temporary tablespace is used for the reorganization. > **Note** > If no temporary tablespace is used for the `REORG`, it is performed in the tablespace where the table or index resides. You must make sure that there is enough free space in this tablespace (approximately the size of the table or index). If this tablespace already has a high I/O load, we recommend that you use a temporary tablespace for performance reasons. • *With Long Fields and LOB Data* If you select this option, long field and LOB data areas are also reorganized. • *Reset Dictionary* If you choose this option, a compression dictionary is rebuilt if the table is compressed or the *Switch Compression on* option is set. Otherwise, an existing compression dictionary is removed. • *Keep Dictionary* If you choose this option, an already existing compression dictionary is kept and not rebuilt. If the table is compressed or the *Switch Compression on* option is set and a dictionary does not yet exist, the compression dictionary is built.
Online	Schedules an online reorganization of the table
Index	Schedules only a reorganization of indexes If the database is range-partitioned, you can further select a single index. Otherwise, all indexes are reorganized.
Reclaim Space for Clustered Tables	Schedules a reorganization that reclaims space for MDC tables
Perform Reorganization on Specified Data Partition	Restricts reorganization to the selected data partition This function is only available if the table is range-partitioned.

Parameter	Description
Switch Compression on	Switches compression on Depending on the selected compression option, the table or its indexes are enabled for compression before the `REORG` operation is performed. Possible values are: • *Static* • *Adaptive* • *Indexes Only*

4. Schedule the job by choosing the *Execute* pushbutton on the *Summary* screen and exit the wizard.

10.1.7.5 Scheduling a REORGCHK for All Tables

> ⚠ Caution
>
> This section is only relevant if the data collection framework has not been set up and you have selected the Collection of Space and Performance History Data (Deprecated) [page 52] (not recommened).

You can use the DBA Planning Calendar to schedule an overall check of all tables using the job *REORGCHK for All Tables*. This job collects the following:

- Findings related to reorganization for tables and indexes that are provided by the `DB2 REORGCHK` command
- Information about current table and index sizes

> → Recommendation
>
> The job *REORGCHK for All Tables* must run once a week.

Procedure

1. Start the *Schedule an Action* wizard by clicking a calendar cell or by choosing the *Schedule an Action* link in your *Favorites*.
2. In the *Job Selection* step, choose *REORGCHK for All Tables* from the *Action* dropdown list.

3. In the *Parameters* step, specify the required parameters as follows:

Parameter	Description
All Tables	Specifies that the job is called for all tables By default, this parameter is selected.
Selected Tables	Specifies that this job is restricted to a set of tables If you choose this option, you also have to specify the *Table Schema* and *Table Name*. **i Note** **Only** choose this option if you require an up-to-date analysis for the selected tables.
With Compression Check	Analyzes the tables and checks for candidates to be compressed By default, you should not activate this option for the `REORGCHK` job that is scheduled weekly. For performance reasons, only perform compression checks in larger time intervals.
Minimum Table Size for Check	Defines the minimum size limit for checking how much space can be saved by compressing the table. **→ Recommendation** We recommend that you set this limit to prevent too small tables that do not benefit from row compression from being checked. **i Note** You can only specify this value if you have chosen *With Compression Check*.
Maximum Runtime	Restricts the runtime of this job

4. Schedule the job by choosing the *Execute* pushbutton on the *Summary* screen and exit the wizard.

10.1.7.6 Scheduling Scripts

Context

You can use scripts to schedule time-consuming and non standard database administration tasks using the job *CLP Script*.

Procedure

1. Start the *Schedule an Action* wizard by clicking a calendar cell or by choosing the *Schedule an Action* link in your *Favorites*.
2. In the *Job Selection* step, choose *CLP Script* from the *Action* dropdown list.
3. Specify SQL statements directly as job parameters.

   > **i** Note
   >
   > Alternatively, you can use scripts that have been created before. For more information, see Managing Your Own DB2 Scripts with SQL Script Maintenance [page 278].

4. Save your entries by choosing the *Execute* pushbutton on the *Summary* screen.
5. Exit the wizard

10.1.7.7 Running the NLS Cleanup Job

Context

> **i** Note
>
> The following section **only** applies if you are using a near-line storage (NLS) database for your local BW system.

When data is reloaded from the NLS database into the BW system, it continues to exist in the NLS database. It is marked, however, as invalidated. To remove this invalidated data and to reduce the space consumption in the NLS database, you can use the *NLS Cleanup* job in the DBA Planning Calendar.

Procedure

1. Start the *Schedule an Action* wizard by clicking a calendar cell or by choosing the *Schedule an Action* link in your *Favorites*.
2. In the *Job Selection* step, choose *NLS Cleanup* from the *Action* dropdown list.
3. In the *Parameter* step, specify the following:
   - In the *NLS Connection* field, choose an existing NLS connection.
   - In the *InfoProvider* field, choose the name of the InfoProvider.

   > ⚠ Caution
   >
   > If you do **not** specify an InfoProvider, the cleanup is performed for all InfoProviders. To run the job for many InfoProviders with similar names, you can use the wildcard character '*'.
4. Save your entries by choosing the *Execute* pushbutton on the *Summary* screen.
5. Exit the wizard.

Next Steps

For more information about reloading data from the NLS database into the BW system, see the document *Enabling SAP Business Warehouse Systems to Use IBM Db2 for Linux, UNIX, and Windows as Near-Line Storage (NLS)* at https://help.sap.com/viewer/db6_nls on SAP Help Portal.

10.1.7.8 Reducing High-Water Marks of Tablespaces on a Regular Basis

Context

You can use the DBA Planning Calendar to schedule a job that checks high-water marks of tablespaces and reduces these high-water marks according to user-defined conditions. You can schedule this job regularly to release unused space that is allocated to the tablespaces back to the file systems.

Procedure

1. Start the *Schedule an Action* wizard by clicking a calendar cell or by choosing the *Schedule an Action* link in your *Favorites*.
2. In the *Job Selection* step, choose *Reduce High-Water Mark for Tablespaces* from the *Action* dropdown list.

3. In the *Parameters* step, specify the required parameters as follows:

Parameter	Description
Reduce To Minimum	If you choose this option, the high-water mark for all selected tablespaces is reduced to the minimum, without any other conditions. This will always release the maximum possible amount of free space. If regularly scheduled, this might cause unnecessary cycles of releasing and reclaiming space if the tablespace size often changes.
Reduce by Specified Size Limit	If you choose this option the high-water mark for all selected tablespaces is reduced, but not more than the amount that is specified in the size limit. If a tablespace has less free space than specified in the size limit, the high-water mark is reduced as far as possible (to the minimum).
Reduce To Minimum If Size Limit Is Reached	If you choose this option, the high-water mark is reduced if the free space of the selected tablespaces has reached the specified size limit. The high-water mark is reduced to the minimum.
Reduce by Specified Size Limit If Reached	If you choose this option, the high-water mark is reduced if the free space of the selected tablespaces has reached the specified size limit. The high-water mark is reduced by exactly the amount that is specified in the size limit.
Size Limit of Free Space	You can specify the size limit of the free space and the unit of the size limit.
For All Tablespaces	If you choose this option, the high-water mark is reduced for **all** tablespaces that have been created with the option *Reclaimable Space*.
For Selected Tablespaces	If you choose this option, the high-water mark is only reduced for the tablespaces that you have selected in the tablespace list (and that have been created with the option *Reclaimable Space*).

4. Schedule the job by choosing the *Execute* button on the *Summary* screen and exit the wizard.

10.2 Displaying the DBA Log

Context

The DBA log provides information about protocol records that are written by all database-related monitoring programs and administration tools.

Procedure

1. Call the DBA Cockpit and choose *Jobs* > *DBA Log* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

   When you access the DBA log for the first time, the log information of all actions for the last 24 hours is displayed by default.
2. Specify a time frame for which you want to display data.
3. Choose a job action from the dropdown menu.
4. Apply your filter selection. The following information is displayed:

Column	Description
Partition	Name of the partition (This information is only displayed for DPF systems.)
Start Date	Start date of the action
Start Time	Start time of the action
End Date	End date of the action
End Time	End time of the action
Runtime	Runtime in HH:MM:SS
Action	Description of the action
Return Code	Return code of the action

5. For more information about an action, select an entry from the overview table. The *Log Details* area (Web browser) or a dialog window (SAP GUI) appears, displaying more details about the log file.

10.3 Controlling the Execution of Background Jobs

Context

To control the execution of background jobs, you can configure the back end of the DBA Planning Calendar. You can configure the back end for all systems, for selected database platforms, or for single systems only.

The respective system configuration is determined by applying the first available configuration in the following order:

1. The system-specific configuration
2. The configuration for the database platform
3. The configuration valid for all platforms
4. The standard configuration, that is, the current user, selection of background server by background dispatcher, and low job priority

Procedure

1. Call the DBA Cockpit and choose *Jobs* > *Back–End Configuration* on the *Database* tab page of the DBA Cockpit.
2. To switch to editing mode, choose the *Change* pushbutton.
3. In the *Selected Scope* area, choose the scope of the configuration entry.
4. In the *Background Dispatching* area, specify the following parameters:

Field	Description
Background Server	Server where scheduled jobs are executed If no server is specified, the background job dispatcher dynamically selects the server.
Job Priority	Priority of the job If no priority is specified, jobs run with default priority (low).
User	Name of the user to execute the job If no user is specified, the dialog user is used.

5. Save your changes.

10.4 Managing Your Own DB2 Scripts With SQL Script Maintenance

Use

You use the *SQL Script Maintenance* function to manage your own DB2 scripts.

Procedure

1. To access the *SQL Script Maintenance*, call the DBA Cockpit and choose *Jobs* > *SQL Script Maintenance* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).
   The *SQL Script Maintenance* screen appears displaying all the scripts located on your local monitoring system.
2. You can perform one of the following actions:

Action	Steps to Be Performed
Display an existing SQL script	Choose an entry from the list. Detailed information is displayed in the *SQL Script Details* detail area.
Edit an existing SQL script	1. Choose a script from the list and choose the *Change* button. 2. Enter and save your changes in the *Change Script* dialog box.
Delete an existing SQL script	1. Choose a script from the list and choose the *Delete* button. 2. Confirm the deletion.
Add a new SQL script	1. To add a new script, choose the *Add* button. 2. In the *Add Script* dialog box, enter a name in the *Script Name* field and start editing. 3. Save the new script.
Execute an existing SQL script	1. Choose a script from the list and choose the *Execute* button. 2. In the *Execute SQL Script* dialog box, enter the name of the SAP system where you want the script to be executed in the *Execute SQL Script on System* field. 3. To confirm your entry, choose the *Execute* button again. The *CLP Output Tray* appears showing an editor in the lower half of the screen where the result is displayed. **i Note** As an alternative to the *SQL Script Maintenance* function, you can also use the DBA Planning Calendar to execute a script by using the *CLP Script* job as described in Scheduling Scripts [page 274].

10.5 Monitoring Scheduled DB2 Tasks

Use

On the *Scheduled DB2 Tasks* screen, you can monitor the execution of tasks that have been scheduled from the DB2 Admin Task Scheduler and the status of the data collectors of the DCF.

Procedure

1. To access the *Scheduled DB2 Tasks* screen, call the DBA Cockpit and choose *Jobs* > *Scheduled DB2 Tasks*.
2. In the *Selection* area, you can specify the time frame to be displayed. By default, the jobs of the current day are displayed.
   The following information is displayed:

Column	Description
Name	Name of the task
Status	Status of the task The following values are possible: • *RUNNING* The task is currently running. • *COMPLETE* The task has finished running. • *NOTRUN* An error prevented the scheduler from calling the task's procedure. • *UNKNOWN* The task started running but an unexpected condition prevented the scheduler from recording the task outcome. This can occur if the system ends abnormally or a power failure happens while the task is running.
Invocation	Number of invocations so far
Time Started	Start time of the task If the status is *RUNNING*, *COMPLETE*, or *UNKNOWN*, this value indicates the time when the task started running. If the status *NOTRUN*, this value indicates the time when the task should have started.
Time Finished	Time when the task finished running This value is not set as long as the status is *RUNNING*. If the status is *UNKNOWN*, this value is the time when the task scheduler detected that the task was no longer running and updated the status table accordingly.
Agent TID	Agent thread ID of the application that executed the task

Column	Description
SQL Code	Value displayed depends on the status of this field. If the status is *COMPLETE*, this value indicates the SQL code that is returned by the CALL to the procedure. If the status is *NOTRUN*, this value indicates the SQL code of the error that prevented the task from running. **i Note** As long as the status is *RUNNING* or *UNKNOWN*, this value is not available.
SQL State	The SQL state that is related to the SQL code
Return Code	Return code of the procedure that was called As long as the status is not *COMPLETE*, this value is not available
Error Message	Full message text corresponding to the SQL Code

11 Alerts

Use

The DBA Cockpit is integrated with alerting infrastructures. The alerting infrastructures ensure that you are notified with alerts when critical incidents concerning databases arise. Alerts are raised, for example, when disk space becomes too small, when database performance is below a defined threshold, or when up-to-date database backups are missing.

Available Alerting Infrastructures

Depending on your use of the DBA Cockpit, alerts are generated using one of the following alerting infrastructures:

- **End-to-end monitoring and alerting infrastructure (MAI) in SAP Solution Manager**
  This alerting infrastructure is recommended when the DBA Cockpit runs on an SAP Solution Manager system and is used for central, remote monitoring of your system landscape. In this case, you can use the end-to-end monitoring and alerting infrastructure (MAI), which is available in SAP Solution Manager systems as of release 7.1.
- **Alert architecture of the Computing Center Management System (CCMS)**
  This alerting infrastructure can be used when the DBA Cockpit runs locally on an SAP system and is used for monitoring the database of the local system. In this case, the DBA Cockpit reuses data from the alert architecture of the Computing Center Management System (CCMS), which is available in every SAP NetWeaver-based system, to generate alerts.

We recommend that you use the MAI in SAP Solution Manager for remote monitoring. For compatibility reasons, however, it is also possible to establish alert monitoring of remote databases using the CCMS infrastructure instead of the MAI in SAP Solution Manager. This monitoring can be enabled for each system in the system configuration of the DBA Cockpit (see Configuring Systems for Remote Monitoring Manually [page 39]). Since remote monitoring using the CCMS has the same functions as local monitoring using the CCMS, it is not described in further detail here.

> **i** Note
>
> The availability of the MAI in SAP Solution Manager depends on your support license.

More Information

- Alerting in SAP Solution Manager [page 283]
- Alerting in a Local DBA Cockpit Using CCMS [page 286]

11.1 Alerting in SAP Solution Manager

Use

If the DBA Cockpit runs on an SAP Solution Manager system and is used for monitoring your system landscape centrally, you can use the end-to-end monitoring and alerting infrastructure (MAI). The MAI is available in SAP Solution Manager systems with release 7.1 or higher; the availability of the MAI also depends on your support license. If the MAI is available, you can use alert screens and functions of the MAI in SAP Solution Manager. The alert screens in the DBA Cockpit are deprecated and only available for customers who still want to use the CCMS.

The DBA Cockpit checks whether it runs on an SAP Solution Manager system. If so, it automatically switches from the alert infrastructure of the CCMS to the MAI provided by SAP Solution Manager.

> ⚠ Caution
>
> Avoid running the MAI in SAP Solution Manager **and** alerting in the CCMS infrastructure at the same time because this causes a higher system workload. To check whether the alerting in CCMS is deactivated for a system, go to the *System Configuration* screen. When you select a system configuration entry and choose *Change*, the *Collect Alert Data* checkbox must be deselected.

Features

Alerts in the DBA Cockpit Based on the MAI

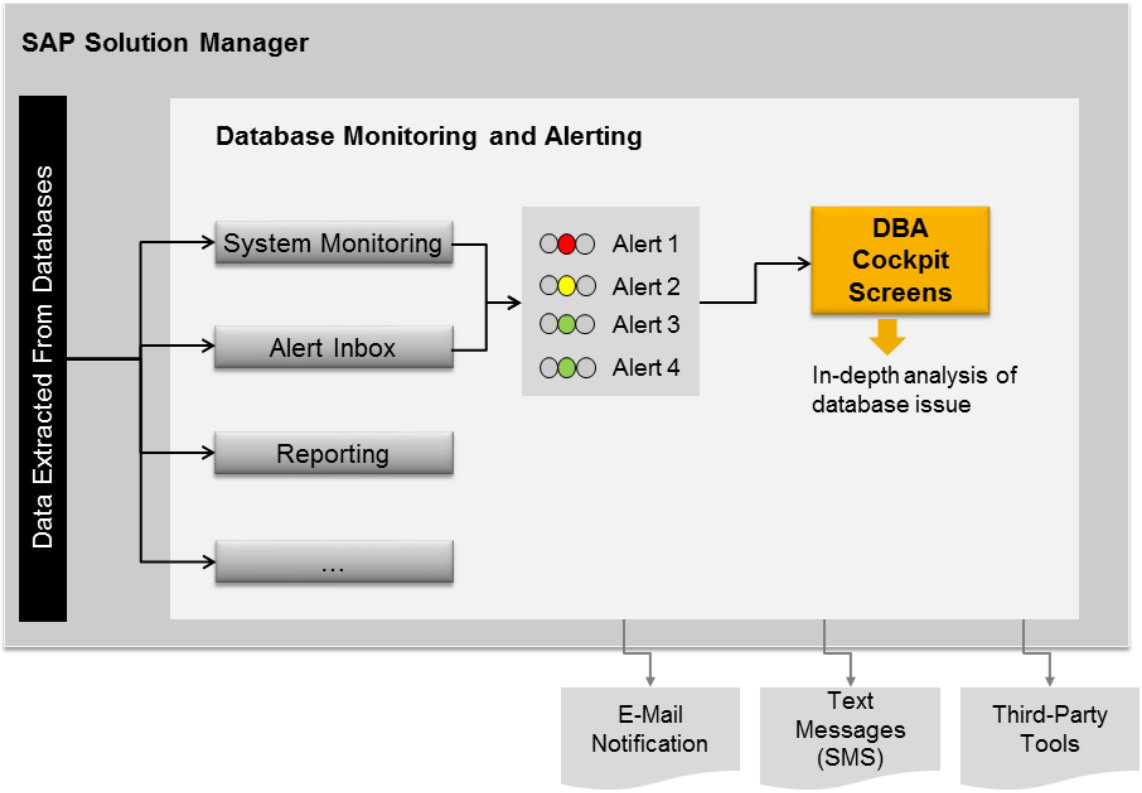
If the MAI is set up, the traffic lights in the system selector of the Web browser-based DBA Cockpit are based on alert results from the MAI, not from the CCMS. In the dashboard of the DBA Cockpit, some metrics are highlighted with traffic lights, which are also based on alert results from the MAI. If the alerting in the CCMS is not set up (as recommended), the alert screens in the DBA Cockpit are grayed out and not available. Use the alert screens in SAP Solution Manager instead.

Available Alert Functions in SAP Solution Manager

The functions for alert monitoring and handling in SAP Solution Manager include the following:

- The *System Monitoring* screen, which includes an overview of the system with a direct navigation to relevant alerts
- The alert inbox in technical monitoring

The monitoring and alerting tools in SAP Solution Manager also allow you to navigate from an alert to the relevant screen in the DBA Cockpit for an in-depth analysis of the underlying issue:



Database Monitoring and Alert Functions in SAP Solution Manager

The benefits of using the MAI in SAP Solution Manager include the following:

- Through its grouping and prioritization of metrics and alerts, the monitoring and alerting infrastructure prevents system and database administrators from being flooded with alerts.
- You can monitor alerts for a system landscape centrally from an SAP Solution Manager system.
- You can reuse available templates for monitoring and alerting for mass configuration of monitored systems.
- You can use the integrated reporting capabilities of SAP Solution Manager.
- You get a graphical overview of the landscape status.

More Information

Displaying and Managing Alerts in SAP Solution Manager [page 285]

11.1.1 Setting Up Alerting in SAP Solution Manager

To set up the end-to-end monitoring and alerting infrastructure for databases, use the standard setup for SAP Solution Manager. For more information, see Setting Up Database Monitoring and Alerting in SAP Solution

Manager [page 37] and Architecture Overview: End-to-End Monitoring and Alerting in SAP Solution Manager and DBA Cockpit [page 35].

11.1.2 Displaying and Managing Alerts in SAP Solution Manager

Use

If you use the end-to-end monitoring and alerting infrastructure in SAP Solution Manager, no dedicated screens for end-to-end alerting are available within the DBA Cockpit transaction itself. Instead, you use the alert functions and screens that are part of SAP Solution Manager.

> **i** Note
>
> The alert screens in the DBA Cockpit are deprecated and are only available for customers who still want to use the CCMS.

Procedure

Displaying Alerts in System Monitoring

1. To call up system monitoring in SAP Solution Manager, call up the work center using the `SOLMAN_WORKCENTER` transaction.
2. Choose the *Technical Monitoring* tab page and then *System Monitoring*.
   On the *System Monitoring* screen, you get an overview of the status of your systems from which you can navigate to details and the relevant alerts.

Accessing the Alert Inbox in Technical Monitoring

1. To access the alert inbox directly, call up the work center of the SAP Solution Manager system using the `SOLMAN_WORKCENTER` transaction.
2. Choose the *Technical Monitoring* tab page and then *Alert Inbox*.

More Information

For more information about how to manage alerts and how to use the alert inbox in SAP Solution Manager, see the SAP Library for SAP Solution Manager on SAP Help Portal at http://help.sap.com/solutionmanager *<Your SAP Solution Manager release>* > *Application Help* > *SAP Library (English)* > *Technical Operations* > *Technical Monitoring and Alerting*.

11.2 Alerting in a Local DBA Cockpit Using CCMS

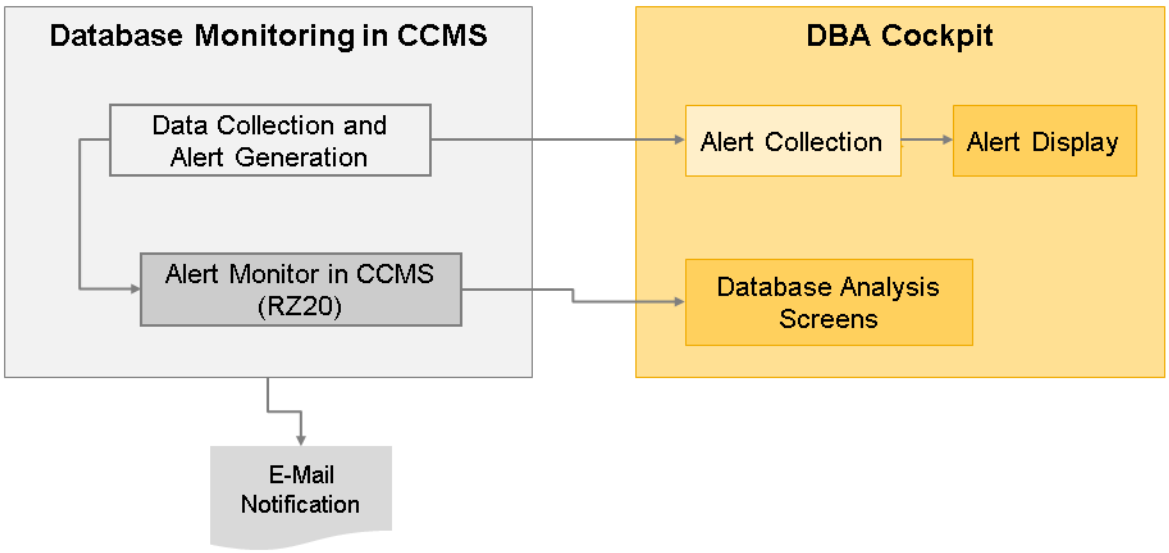
Use

The monitoring architecture of the Computing Center Management System (CCMS) centrally monitors any IT environments, including SAP systems and databases. The CCMS is part of every SAP NetWeaver system and also includes preconfigured database monitoring content as part of the CCMS alert monitor.

The database monitor in the CCMS checks snapshot data and the database history file. If a critical situation, such as a threshold violation, is detected, alerts are raised so that you can identify and solve problems at an early stage.

Integration

The database monitor and alerting infrastructure of the CCMS can be used together with a DBA Cockpit running on the same system.



Alerting in the CCMS and in the DBA Cockpit

The alerts raised by the alert monitor in the CCMS are automatically taken over by the DBA Cockpit and displayed on the alert screens of the DBA Cockpit. In the CCMS, you can also display monitoring and alerting details using the standard CCMS alert monitor. The alert monitor provides more detailed information about alerts than the DBA Cockpit. In the alert monitor in the CCMS, you can navigate to the relevant analysis tools for each alert type. Then CCMS navigates to the relevant database analysis screens of the DBA Cockpit, where you can continue with an in-depth analysis of the database issue.

For more information about the integration of database system monitoring in CCMS and in the DBA Cockpit, see Database Alerts in the CCMS and DBA Cockpit: Overall Architecture and Setup [page 288].

Features

Alert Monitoring in the CCMS

The Alert Monitor in the CCMS provides the following functions:

- **Alert tree structure**
  All errors generate alerts, which are displayed in a tree structure.
- **Alert status**
  The alerts contain a status indicator with a color and a numerical value. Yellow means a warning, red means a problem, and the numerical value shows the severity of the reported error. In the tree structure, the most severe alerts are passed upward in the display hierarchy. If a tree node does not display an alert, there is no error in the entire branch below it. For more information about database alerts, see Displaying Database Alerts in the Alert Monitor [page 290].
- **Assignment of analysis and auto-reaction methods**
  You can assign analysis and auto-reaction methods to the alerts, which contribute to faster processing of the error. If you double-click an alert, the monitoring architecture starts the assigned analysis method (such as the DBA Cockpit). As opposed to an analysis method, an auto-reaction method starts automatically as soon as the alert occurs. This includes, for example, sending an e-mail to the database administration.

Alert Monitoring in the DBA Cockpit

The alert monitor in the DBA Cockpit provides the following:

- An overview of alerts, which are displayed in a tree structure (see also Displaying Database Alerts in the Alert Monitor [page 290])
- A possibility to search for and display alerts for a defined time period in the alert message log (see also Displaying Alerts in the Alert Message Log [page 291])

Categories for Database Monitoring

The following categories of information are monitored:

- Disk space of the tablespaces and file systems that are required for the database system
- Performance indicators such as the following:
  - Access behavior of database buffers
  - Lock behavior of the application, monitoring of deadlock situations, and lock escalations
- Availability of backup and recovery mechanisms:
  - Last available backup
  - Availability of the log files that are necessary to restore the current state of the database from the last available backup
- Configuration parameters
- Availability data of tablespaces and containers

> → Recommendation
>
> We recommend that you check the alert monitor daily for information about database system monitoring.

More Information

For more information about the CCMS and the alert monitor in CCMS, see SAP Library for SAP NetWeaver on SAP Help Portal at http://help.sap.com/nw <Your SAP NetWeaver release>. Search for the keywords `Monitoring in the CCMS` and `Alert Monitor`.

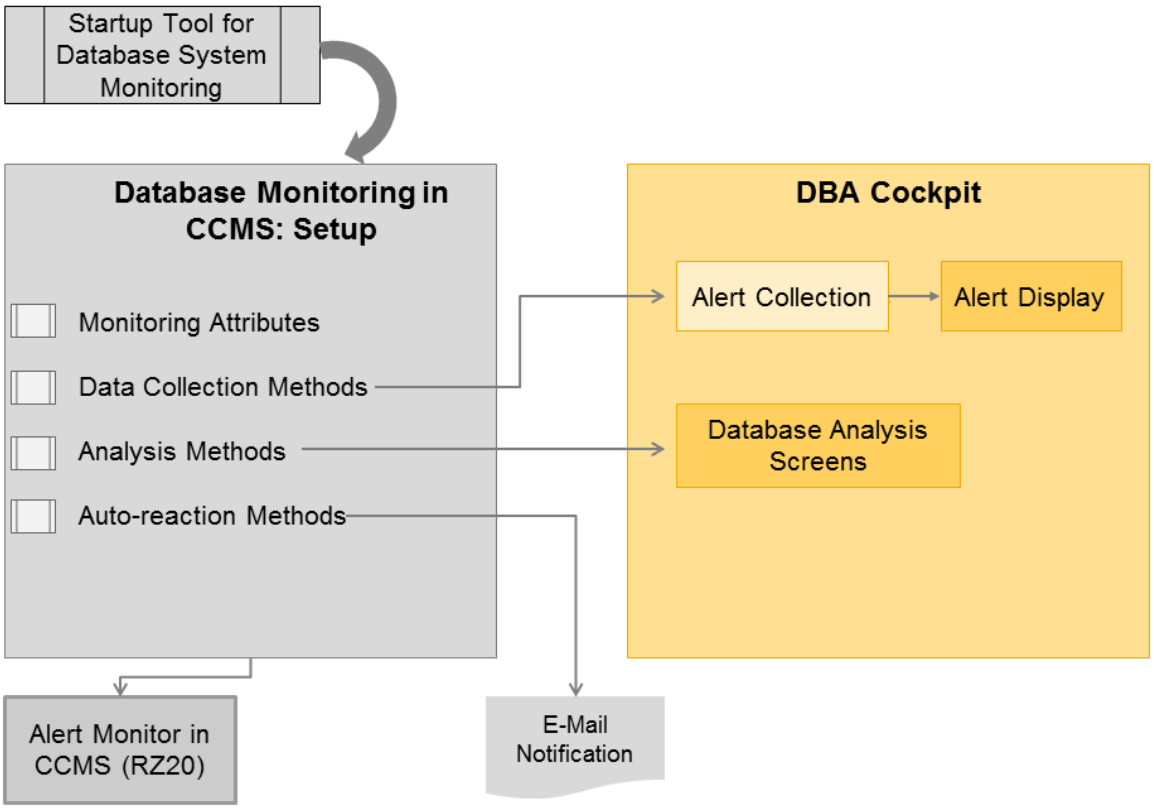
For more information about the CCMS in SAP NetWeaver 7.3 with Enhancement Package 1, for example, see SAP Library for SAP NetWeaver on SAP Help Portal at http://help.sap.com/nw731 *Application Help* > *SAP Library (English)* > *SAP NetWeaver Library: Function-Oriented View* > *Solution Life Cycle Management* > *Solution Monitoring* > *Monitoring in the CCMS* > *Alert Monitor*.

11.2.1 Database Alerts in the CCMS and DBA Cockpit: Overall Architecture and Setup

The database monitors in the CCMS are automatically generated by a startup tool during the startup of an SAP NetWeaver instance. The automatic setup includes the following:

- **Monitoring attributes**, such as the status, performance, or the availability (heartbeat) of the database
- **Data collection methods** that are assigned to the monitoring attributes, which define how information about the database is collected and then transferred to the monitoring infrastructure of the CCMS
- **Analysis methods**, which allow you to perform a detailed analysis of error situations in the appropriate transactions by double-clicking the alert in the alert tree
  In the case of a database, the monitoring architecture navigates to the relevant screen in the DBA Cockpit, where you can perform an in-depth analysis of the error that triggered the alert.
- **Auto-reaction methods**, which start automatically when an alert is triggered
  There are several predefined auto-reaction methods in the CCMS monitoring infrastructure that you can choose as an auto-reaction to an alert, such as e-mail notifications.

The startup tool considers all databases that have the *Collect Alert Data* option enabled in the system configuration of the DBA Cockpit. The startup tool automatically analyzes the database configuration and layout of the SAP system and adapts the database monitor accordingly.



Monitoring and Alerting in CCMS and in the DBA Cockpit: Architecture and Integration

For more information about displaying alerts in the DBA Cockpit and the CCMS, see Displaying Database Alerts in the Alert Monitor [page 290] and Displaying Alerts in the Alert Message Log [page 291].

11.2.2 Setting Up Database Alerts in the CCMS and in the DBA Cockpit

Use

Most of the alerting infrastructure in the CCMS is set up automatically for databases (see Database Alerts in the CCMS and DBA Cockpit: Overall Architecture and Setup [page 288]). There are two settings, however, that need to be performed manually:

- Configuration of auto-reaction methods such as the configuration of e-mail notifications for alerts
  The central, automated notification function informs you of an alert by e-mail. If you want to be notified as soon as an alert is raised, you have to define yourself as a recipient of mails generated by this function.
- Activation of background dispatching

> **i Note**
>
> If you do not activate background dispatching, the system is not monitored at all.

> ⚠ Caution
>
> Do not change the preconfigured check categories and parameters of the DB2 database system monitoring. Changing the preconfigured database monitoring is not recommended and should only be done in exceptional cases and after agreement with SAP support (see Displaying and Changing the Alert Configuration (Optional) [page 292]).

Procedure

Enabling Automatic E-Mail Notification

1. Call transaction RZ21.
   The *Monitoring: Settings and Tool Maintenance* screen appears.
2. Choose *Tool Definition* and then *Display Overview*.
3. Scroll through the list until you find *CCMS_OnAlert_Email*.
4. Select this entry and choose *List* > *Selected Entries* > *Edit* (or *Edit* in the application toolbar).
   The *Monitoring: Tool Administration* screen appears.
5. Choose *Parameter*.
6. Choose *Tool Definitions* > *Display Change* (or *Display Change* in the application tool bar).
7. In the *SENDER* line in the *Parameter value* column, enter a valid user for your SAP system.
8. In the *RECIPIENT* line in the *Parameter value* column, enter a valid user for your SAP system who will be notified in the event of an alert.
9. Save your changes.

Activating Background Dispatching

1. Call transaction RZ21.
2. Choose *Technical Infrastructure* > *Method Execution* > *Activate Background Dispatching*.

11.2.3 Displaying Database Alerts in the Alert Monitor

Procedure

To retrieve information about alerts, choose one of the following options:

- For an overview of alert situations, call the DBA Cockpit and choose *Alerts* > *Alert Monitor* on the *Database* tab page of the DBA Cockpit.
  The *Alert Monitor* screen appears, showing the alert tree with the currently reported messages. To display the full message log for an alert, click the *Details* button, which is displayed after the message column.
- To start a more detailed analysis, call transaction RZ20 and choose *SAP CCMS Monitor Templates* > *Database* > *DB2 Universal Database for NT/UNIX* in the tree structure *CCMS monitor sets*.

> **Note**
>
> If you want to use transaction RZ20 for remote database systems, choose *SAP CCMS Monitor Templates* > *Remote Database* > *DB2 for Linux, UNIX, Windows*.

Result

Data Displayed in the Alert Monitor Tree

Regardless of the view variant you choose, information about the following is displayed:

- *Connection <Database name>*
- *Space management*
- *Performance*
- *Backup/restore*
- *SAP consistency*
- *Health*

The checked parameters are displayed in the following colors depending on the type of message:

Message Type	Color
Information	Green
Warning	Yellow
Error	Red

> **Note**
>
> If a check resulted in a warning or an error, a short text is additionally displayed next to the parameter.

11.2.4 Displaying Alerts in the Alert Message Log

Procedure

1. To access an overview of the system monitoring results and the alerts, call the DBA Cockpit and choose *Alerts* > *Alert Message Log* on the *Database* tab page of the DBA cockpit.

   The *Alert Message Log* screen appears.

2. To limit the display to specific error levels or check categories, use the list boxes in the *Current Selection* area. In the *Object* and *Attribute* fields, you can enable restrictions to single attributes. Additionally, you can specify a time period for which the messages are displayed.

   By default, the messages of the last seven days are displayed. Only the most important data is displayed in the overview. The results are displayed as notes, warnings, or errors and are sorted by log date by default.
3. To display more detailed information, select an alert message.

   The *Alert Message Details* area appears displaying the following:

   - *Alert Message Details*
     Complete description of the attribute as displayed in the alert monitor tree
   - *Logged Data*
     Information about the alert message, for example, type of error, reported value, and date and time when it occurred
   - *Description*
     Description of the type of error and which value or parameter is being monitored

11.2.5 Deleting Messages in the Alert Message Log

1. Call the DBA Cockpit and choose *Alerts* > *Alert Message Log* on the *Database* tab page of the DBA cockpit.
2. To delete alert messages, select the entries in the list and choose the *Delete* button.

> ⚠ Caution
>
> To avoid the log table becoming too large, automatic clean-up programs run and delete entries that are older than 30 days.

11.2.6 Displaying and Changing the Alert Configuration (Optional)

Use

The DBA Cockpit allows you to display the entire database-specific alert configuration. In addition, you can also change some of the alert configuration settings. Note that the DB2 database system monitoring has preconfigured check categories and parameters.

> ⚠ Caution
>
> Only experienced users should make changes to the alert configuration, and only in exceptional cases after consulting SAP support. We recommend that you leave the predefined alert configuration unchanged, because it is optimized for database monitoring and alerting. If you want to change the alert configuration, contact SAP support first.

Procedure

Displaying Details of the Alert Configuration

You can access information about alert configuration by calling the DBA Cockpit and choosing *Alerts* > *Alert Configuration* on the *Database* tab page of the DBA Cockpit. The *Alert Configuration* screen appears.

On this screen, you can display alert configuration details and select configuration options using the list boxes in the *Selection* area. In addition, you can activate or deactivate an alert configuration entry by selecting or deselecting the checkbox in the *Active* column.

> ⚠ Caution
>
> If you deactivate an alert configuration entry, you will no longer be notified of the respective alerts.

Displaying Configuration Details of an Alert

For more information about an alert configuration entry, select one from the list. The *Alert Configuration Details* area appears displaying the following tab pages:

- *General (RZ21)*
- *Administration*
- *Thresholds*

General (RZ21)

This tab page displays the scheduling data from the basic alert monitor configuration for completeness reasons. You **cannot** change the values on this tab page. You can make changes using the general maintenance function (transaction RZ21) in the alert monitor.

Administration

This tab page displays the user, date, and time of the last changes.

Thresholds

This tab page displays the threshold statuses of an alert configuration as follows:

- *Error Condition*
  If the error is not corrected, normal operation is endangered.
- *Warning Condition*
  Limited operation, for example, with reduced performance
- *Normal State Condition*

Each condition is described by a relational operator, a comparison value, and the unit of measurement for this value.

> ⚠ Caution
>
> You can also change these thresholds. Changes should only be done by experienced users and after consulting SAP support.

Changing Alert Conditions on the Thresholds Tab Page

1. To change the alert conditions of the threshold statuses in the alert configuration, choose the *Thresholds* tab page.

2. Choose the *Edit* pushbutton.
   The *Alert Configuration Details* dialog box appears.
3. Maintain the configuration values according to which the system is monitored as follows:
   - Relational operators
     You can specify how the defined comparison value should be compared with the current value. In addition to the relational operators, you can enter whether a value should lie inside or outside of a range of values. A full colon *(:)* must separate the two values. You can also specify whether or not discrete values are within a set of explicit values. Semicolons *(;)* must separate such values.
   - Comparison value
     You can specify a value, a list of values, or a value range depending on the operator. This value will later be compared with the currently measured value.
   - Unit of measurement of the comparison value
     You can specify the unit of measurement of the comparison value. This is important for time values, which are normally calculated internally in seconds, to be correctly converted before comparison.

     > **i Note**
     >
     > You do not have to enter values for every operation status. However, you must make sure that the sum of comparison values covers every possible value. If this is not the case, a special alert is triggered with the following message:
     >
     > `There is no configuration entry for the logged value`
4. Save your entries.
   The changes take effect immediately.

12 Diagnostics

12.1 Diagnostics: Displaying the Audit Log

Use the audit log to track changes to the database made by the DBA Cockpit and changes to the DBA Cockpit system configuration.

Context

On the *Audit Log* screen, you can track changes to the database that were made from the DBA Cockpit, changes to the DBA Cockpit system configuration, and changes to the monitoring setup. Changes made from outside – for example, using native database commands – are not displayed here.

> **i** Note
>
> The audit log in the DBA Cockpit is **not** a certified tool for auditing.
>
> SQL commands issued from the SQL editor in the DBA Cockpit are also logged in the security audit log (transaction `SM20`). For more information, see SAP Note 3331129.

Procedure

1. To access the *Audit Log* screen, call the DBA Cockpit and choose *Diagnostics* > *Audit Log* on the *Database* tab page (Web browser) or in the navigation frame of the DBA Cockpit (SAP GUI).

   The *Audit Log* screen appears displaying the following information:

Field	Description
Date	Start date of the action
Time	Start time of the action
System	Target system on which the action was performed
Action	Type of action (name of the action in the DBA Cockpit)
Command	Type of command (for example, `ADD`, `DELETE`, or `EDIT`)

Field	Description
Object	Name of the modified object (for example, database or tablespace name)
User	Name of the SAP user who performed the action
From System	System from where the action was performed

2. In the *Selection* area, you can specify a time frame for which you want to display data.

   By default, the audit logs of the current week are displayed.

12.2 Diagnostics: Missing Tables and Indexes

> **i Note**
>
> This function is only available for systems where you called up the DBA Cockpit locally, or for ABAP systems for which an additional RFC connection has been assigned.

You can find out whether tables or indexes are missing from either the database or the ABAP Dictionary by calling the DBA Cockpit and choosing *Diagnostics* > *Missing Tables and Indexes* on the *Database* tab page of the DBA Cockpit.

The results of the last consistency check are displayed in a tree structure that is grouped into the following sections:

Section	Description
Objects missing from the database	Objects that are defined in the ABAP Dictionary, but not found in the database
Unknown objects in the ABAP Dictionary	Objects that are found in the database, but not defined in the ABAP Dictionary
Inconsistent objects	Results of the detailed comparison of the ABAP Dictionary and the database are displayed here

Section	Description
Other checks	Different checks are performed here: • It is checked whether the primary index of tables defined in the ABAP Dictionary was created uniquely on the database. • Objects in the SAP system tables are checked, which cannot be described at all or which cannot be completely described in the ABAP Dictionary for technical reasons. • If inconsistencies for these objects are detected, they are also displayed here. Additional information about the type of inconsistency is also provided.
Optional indexes	Mismatch between ABAP Dictionary and database regarding secondary indexes

The consistency check is automatically executed once a day as part of the standard background job `SAP_COLLECTOR_FOR_PERFMONITOR`. If you need exact, up-to-date check results, choose the *Refresh* pushbutton.

12.3 EXPLAIN Function

You can use the EXPLAIN function to review the access plans of `SELECT`, `INSERT`, `UPDATE`, or `DELETE` statements. It is available on the Web browser-based user interface and on the SAP GUI-based user interface of the DBA Cockpit:

EXPLAIN Function (SAP GUI-Based Version) [page 299]

EXPLAIN Function (Web Browser-Based Version) [page 297]

12.3.1 EXPLAIN Function (Web Browser-Based Version)

You can use the Web browser-based EXPLAIN to review the access plans of all `SELECT`, `INSERT`, `UPDATE`, or `DELETE` statements.

You can access the Web browser-based version of the EXPLAIN function as follows:

- In the navigation frame of the SAP GUI-based user interface of the DBA Cockpit, choose *Diagnostics* ▸ *EXPLAIN Access Plan*.
  A Web browser opens and displays the screen *EXPLAIN Access Plan*. Enter an SQL statement and choose the *EXPLAIN* pushbutton.
- In the *Favorites* list of the Web browser-based user interface, choose *Explain Access Plan*.
- In the database-specific area of the Web browser-based user interface, choose *Performance* ▸ *Applications*. Double-click an application and choose the *Statement Text* tab page. If a statement is displayed, choose the *EXPLAIN* pushbutton to display the access plan.

For more information, see Application Details: Statement Text [page 402].

- In the database-specific area of the Web browser-based user interface, choose *Performance* > *SQL Cache*. Select a statement from the list and choose the *EXPLAIN* pushbutton.
  For more information, see Performance: SQL Cache [page 103].

> **Note**
>
> The statements might contain optional comments such as `--OPTLEVEL( <optlevel> ) --QUERY_DEGREE(< query_degree> --LOCATION( <report> , <position> )`. If no comments are specified, the statements are explained using the default `<optlevel>` and the default `<query_degree>` for the work process.

If a statement was explained successfully, information about the SQL statement text is provided on the following tab pages:

Tab Page	Description
Original Statement	Displays the original SQL statement
Optimized Statement	Displays the SQL statement that was rewritten by the DB2 optimizer
Access Plan	Displays the access plan that was generated by the DB2 optimizer
EXPLAIN Snapshot	Displays the output of the EXPLAIN snapshot **Note** The *EXPLAIN Snapshot* tab page is only available if the monitored database is DB2 V9.5 for Linux, UNIX, and Windows or higher.

Using the Access Plan

The access plan shows all database operations that are performed when the statement is executed. It is displayed as a graphical tree, and each node in the tree represents an operator of the access plan.

You can do the following:

- Display or hide details of an operator by choosing the *Open Node* or *Close Node* icon on the respective node
- Expand or collapse subtrees by choosing the *Show Child Node* icon or the *Hide Child Node* icon respectively
- View operation details by double-clicking an operator in the graphical tree
  Global details about an operator are displayed on the following tab pages:
  - *General*
    Displays global details about the access plan
  - *Operator <Name of operator>*
    Displays details for the selected operator

- *Catalog Information (Optional)*
  Displays details for the respective catalog object of the selected operator
- *Predicates (Optional)*
  Displays filter predicates for the selected operator

- Search for operators in a complex statement by choosing the *Find* pushbutton
- Open an extra navigation window for complex access plans by choosing the *Navigate* pushbutton
- Print the graphic by choosing the *Print* pushbutton
- Configure the graphic before you print it by choosing the*Standard Graph* or *Detailed Graph* pushbutton
- Display or hide all operators by choosing *Collapse* or *Expand*
- Display global details about the access plan by choosing the*Toggle* pushbutton
- Display information about the JNet version used (can be required by SAP Support) by choosing the *About* pushbutton

> **i** Note
>
> For each index used in the access plan, the number of key columns that were really used within the access plan is displayed. In the appropriate tool tip, the used index field names are also displayed.
>
> Volatile tables and indexes of volatile tables are marked with an extra volatile label. To change and re-explain the SQL statement, choose *Edit Statement*.

Changing the DB2 Optimizer Configuration

If you want to change the DB2 optimizer parameters, choose the *Optimizer* pushbutton. With this function, you can temporarily change the `OPTIMIZER LEVEL`, the `QUERY DEGREE`, and the `VOLATILE` flag for all tables referred to in the query. In addition, DB2 experts are able to specify optimization guidelines.

User Interface Settings

If you want to change the user interface of the Web browser-based user interface, choose the *Settings* pushbutton.

12.3.2 EXPLAIN Function (SAP GUI-Based Version)

You can use this function to explain all `SELECT`, `INSERT`, `UPDATE`, or `DELETE` statements.

> **i** Note
>
> The statements might contain optional comments such as `--OPTLEVEL( <optlevel> ) --QUERY_DEGREE(< query_degree> --LOCATION( <report>, <position> )`. If no comments are specified, the statements are explained using the default `<optlevel>` and default `<query_degree>` for the work process.

Accessing the EXPLAIN

You can call the EXPLAIN function in the following ways:

- Call the DBA Cockpit and choose *Diagnostics* > *EXPLAIN* in the navigation frame of the DBA Cockpit. On the *Diagnostics: EXPLAIN* screen, enter an SQL statement manually and choose *Explain*.
- Call the DBA Cockpit and choose *Performance* > *Applications* in the navigation frame of the DBA Cockpit.
- Call the DBA Cockpit and choose *Performance* > *SQL Cache* in the navigation frame of the DBA Cockpit.
- Call the DBA Cockpit and choose *Performance* > *Top SQL Statements* in the navigation frame of the DBA Cockpit.
- Call the DBA Cockpit and choose *Workload Management* > *Threshold Violations* in the navigation frame of the DBA Cockpit.
- Call the DBA Cockpit and choose *Diagnostics* > *Cumulative SQL Trace* in the navigation frame of the DBA Cockpit.
- Call transaction `ST05` and choose *Enter SQL statement*. Enter an SQL statement manually and choose *Explain*.
  If a statement cannot be explained, the *ERROR: Check SQL Statement* screen appears providing a detailed error message and the possibility to modify the statement. To continue, choose *Explain Again*.
- Call transaction `ST05` and choose *Trace list*. Select a statement and choose *Explain*.

"EXPLAIN" and "EXPLAIN from Cache"

On several screens where an SQL statement is displayed, you can find two options to determine the access plan: *EXPLAIN* and *EXPLAIN from Cache*.

The *EXPLAIN* button triggers a fresh calculation of the acces plan using the Db2 command EXPLAIN. The *EXPLAIN from Cache* button triggers the lookup of an access plan that has already been prepared and stored in the SQL cache. Technically, this is realized by a call of the Db2 procedure `EXPLAIN_FROM_SECTION`.

Displaying the Access Plan of a Statement

If a statement was explained successfully, the *Display Execution Plan for SQL Statement* screen appears, providing information on the SQL statement text, the `OPTLEVEL` and `QUERY_DEGREE` that was used to explain this statement, and the access plan. If the SQL statement finds a match in the list of SAP optimizer profiles for this system, a popup appears that allows you to run the `EXPLAIN` function using the provided Db2 optimizer guideline. For more information about optimizer profiles, see SAP Note 1818503.

The access plan generated by the Db2 optimizer is displayed as a tree structure. It consists of all database operations that will be performed when the statement is executed.

The estimated execution time is displayed in timerons (arbitrary IBM time unit). All operators are numbered, starting with zero. Operators can have the following extensions:

Extension	Description
`[O]/[I]`	Shows whether the operator acts as an outer/inner input branch for a subsequent join operation
(`<Partition>`)	Shows on which partition this operation was performed This is only displayed if you are using a multi-partition database.

Non-volatile tables and indexes of non-volatile tables are displayed in blue. Volatile tables and indexes of volatile tables are displayed in orange.

For each index used in the access plan, the number of *key columns*, that is, index fields that were really used within the access plan, are displayed.

For further analysis of the displayed information, you can choose from various options in the application tool bar.

Related Links

EXPLAIN Options [page 302]

Explain facility in the IBM Db2 documentation

SAP blog post on SAP Community: Explain Everything in the DBA Cockpit

12.3.2.1 EXPLAIN Options

On the *Display Execution Plan for SQL Statement* screen, the following options are available:

Option	Description
Details	If no operator in the access plan is highlighted when choosing this option, a dialog box is displayed providing detailed information on the statement and each operator. This output is similar to the one of the DB2 command line tool `db2exfmt`. If operator no. 0 is highlighted, only the original statement and optimized statement are displayed in a separate dialog box. If any other operator is highlighted, the system displays detailed information on the selected operator only.
Optimizer	The access plan may vary depending on the optimizer parameters specified. When you choose this button, the *Change Query Optimization* dialog box appears where you can change the parameters *OPTIMIZER LEVEL*, *QUERY DEGREE*, and the flag *VOLATILE* for the tables used in the access plan. To explain the statement with new parameters, choose the *Explain Again* pushbutton. An existing DB2 optimizer guideline for the statement is displayed. You can choose whether to use the guideline when explaining the statement or not, and you can edit the guideline. For more information, see SAP Note 868888.

Option	Description
DB Catalog	With this option, you can display system catalog information on tables and indexes that are shown in the access plan. The following information is displayed: • For a table: Selected information from table SYSCAT.TABLES is displayed. Additionally, all indexes of the table are displayed with their index columns. • For an index: Selected information from table SYSCAT.INDEXES for this index is displayed. Additionally, selected information from table SYSCAT.COLUMNS is displayed for all index columns. Depending on whether you have selected a table or an index, the following pushbuttons are available: • *Table* Displays selected information from table SYSCAT.TABLES Additionally, all indexes of the table are displayed with their index columns. • *Columns* Displays selected information from table SYSCAT.COLUMNS for all table columns • *Indexes* Displays information from table SYSCAT.INDEXES for all indexes of the table as well as information from table SYSCAT.COLUMNS for all index columns • *Update Statistics* Updates the catalog statistics for the table If the catalog statistics were updated successfully, the field `<stats-time>` is displayed in green.
Dictionary	With this option, you can display the ABAP Dictionary structure (definition) of a table by selecting the table in the access plan. If you do not select a table in the access plan, the ABAP Dictionary structure (definition) of the first dictionary object of the SQL statement is displayed. With this option, you can display the structure of views, even though views never appear in the access plan. **i Note** This function is not available for systems that are monitored using a remote database connection.

Option	Description
Test Execution	This option is only available in the following cases: • A SELECT statement is explained using transaction ST05 *Trace list*, the parameter values for all parameter markers of the statement are provided, and the operation is other than PREPARE. • A SELECT statement without parameter markers is explained. When you use the EXPLAIN function, the entered SQL statement is only prepared and the access plan of the optimizer is chosen because of the system catalog statistics. Based on this information, the optimizer estimates the costs for the execution of this statement. However, the estimated costs may not correspond to the real execution time. Reasons for this might be bad statistics, a bad database layout, or problems of the optimizer itself. The *Test Execution* option measures the real execution time and provides other snapshot data, such as the number of buffer pool accesses and sorts for the selected statement. When the statement is executed, the parameter markers are replaced by the actual parameter values. A dialog box appears where you can change these values to investigate the dependence of the execution time from these values. The result of several test executions of the same statement can vary because, for example, the buffer pool may already contain data that is necessary for the execution. **i Note** This function is not available for systems that are monitored using a remote database connection.
Tree Info	The following additional information is displayed or hidden: • *num_rows* Estimated number of rows (result set) • *tot_cost* Estimated total cost for this statement • *i/o_cost* Estimated I/O cost of the statement This information is also included in the output information when you choose *Details*.

Option	Description
Edit	When choosing this option, the system switches to an editor window in which you can modify the selected SQL statement and explain it again.
Source	This option is only available if the statement contains a LOCATION comment, for example, when you call EXPLAIN using transaction ST05 *Trace list*. The location of the statement in the ABAP source code is displayed in a separate window. **i Note** This function is only available for systems where you called up the DBA Cockpit locally, or for ABAP systems for which an additional RFC connection has been assigned.
Collect	When choosing this option, a dialog window appears that allows you to collect and download `db2support` data for the statement.
Download Plan from db2exfmt	When choosing this option under *Goto* from the SAP GUI menu, you can download an explained SQL statement in Db2 EXPLAIN format.

12.4 Diagnostics: Self-Monitoring

Use

You can use the *Self-Monitoring* screen to identify and solve issues related to database configuration and monitoring setup that keep the DBA Cockpit from running properly.

More Information

Self-Monitoring of the DBA Cockpit [page 64]

Displaying Check Results of Self-Monitoring [page 65]

12.5 Diagnostics: Lock-Wait Events

Use

You can access information about lock-wait events by calling the DBA Cockpit and choosing *Diagnostics* > *Lock-Wait Events* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI) of the DBA Cockpit.

With the information provided on this screen, you can:

- Review lock wait events that occurred in the past and that were captured by the lock event monitor. By default, information about lock events is collected by the *Lock Event* data collector.
  For more information, see Data Collection Framework: Configuration [page 243].
- Review lock wait events that are currently occurring on the database server.

In the *Selection* area, you can specify the time frame for which you want lock-wait events to be displayed as well as the type of lock-wait event, such as, lock waits, lock time outs, and lock waits with a specific lock time out.

After having applied your selection, the total number of lock-wait events that occurred and that you selected in the *Selection* area is displayed in the *Summary* area.

For each lock-wait event, the following information is displayed:

Column	Description
Event Type	Type of lock event, which can be a lock wait, a lock timeout or a deadlock
Event Occurrence	Time when the lock event first occurred
Event Resolution	Indicates when the lock event was resolved or whether it still exists
No. of Involved Agents	Number of agents involved in the lock event
Max. Wait Time	Maximum lock-wait time of any of the involved agents
Lock Objects	Database objects involved in the lock conflict

Activities

To display details of a lock event, choose a row from the overview table in the *Lock-Wait Events* section. The *Lock Event* area appears below showing the details in a table sorted by the application handles of the waiting applications. To learn more about a waiting application, choose the relevant application handle. Another detail area below the current one opens. This *Details of Agent <number>* area provides general information about the application that the lock-waiting agent is working for as well as about the application the lock-holding agent is working for.

In addition, information about the activity that is currently being executed or that was executed last by each agent is provided, such as the SQL statement text and general information about the SQL statement (for example, the compilation environment).

You can display the current access plan of the statement by choosing the EXPLAIN pushbutton. For more information, see EXPLAIN Function [page 297]. To display the ABAP source code from where the SQL statement was executed, you can choose the *Show Source* pushbutton.

> **i Note**
>
> The pushbutton is only available if the following applies:
>
> - The SAP system on which the currently monitored database is running is an ABAP system.
> - The SAP system has the minimum Support Package level as described in **SAP Note** 1493490.

If the lock event monitor has captured past activities or nested activities for a certain agent, the *Activity History* table appears providing information about the agent ID, its nesting level, and the SQL statement executed by each agent. You can display detailed information for each activity by choosing a row in the *Activity History* table. The information is displayed in the *Details of Activity <number>* detail area.

12.6 Diagnostics: Lock Waits and Deadlocks

The system searches for lock waits indicating that at least one process is locked by the lock on another process. A request waits for a resource (a database table or a row of a table) that is locked exclusively by another user. The *Lock Waits and Deadlocks* screen gives you an overview of the lock waits and deadlocks in your system.

> **i Note**
>
> This screen is only available in the SAP GUI-based version of the DBA Cockpit. In the Web browser-based version of the DBA Cockpit, use the screen *Diagnostics* > *Lock-Wait Events* (see Diagnostics: Lock-Wait Events [page 306]).

You can access the *Lock Waits and Deadlocks* screen by calling the DBA Cockpit and choosing *Diagnostics* > *Lock Waits and Deadlocks* in the navigation frame of the SAP GUI-based version of the DBA Cockpit.

All recorded lock waits and deadlocks are displayed in a tree structure. For each lock wait or deadlock situation that has been detected, the *Lock Wait* or *Deadlock* node is displayed as well as the date and time when the lock wait or deadlock situation occurred. If you open the subnodes of a deadlock or lock wait node, a hierarchical structure appears displaying the following information:

- *<Lock wait or deadlock>*
  - *Agent <Agent ID> (<Application Name>) waiting for Agent <Agent ID>*
    - *Client Process ID: <Process ID>*
    - *Host: <Host>*
    - *Lock Agent is waiting for:*
      Table: <Schema>.<Table>
      Lock Object Type: <Lock Object Type>
      Current Lock Mode: <Lock Mode>
      Requested Lock Mode: <Lock Mode>

To display the last SQL statement that was executed by one of the agents, choose *Last SQL Statement*. The last SQL statement of the respective agent is displayed in the editor window at the bottom of the screen.

Tree Node	Description
<Agent ID>	Agent handle of the application waiting for the lock to be released
<Application Name>	Name of the application waiting for the lock to be released
<Client Process ID>	Process ID of the application requesting the lock
<Host>	Host name of the server requesting the lock
Requested Lock Mode	Lock modes that the waiting application would like to set The following lock modes are possible: • `IS`: intention share lock • `IX`: intention exclusive lock • `S`: share lock • `SIX`: share with intention exclusive lock • `X`: exclusive lock • `IN`: intent none • `Z`: super exclusive lock • `U`: update lock • `NS`: next key share lock • `NX`: next key exclusive lock • `W`: weak exclusive lock • `NW`: next key weak exclusive lock
Current Lock Mode	Lock mode held
Lock Object Type	Type of object to be locked
Table	Table on which/on whose record the lock is held

> ⚠ Caution
>
> Lock wait situations are recognized by DB2. Database parameter `LOCKTIMEOUT` specifies how many seconds the system must wait before automatically resolving a lock wait situation.
>
> If `LOCKTIMEOUT` is set to `-1`, lock wait situations are not resolved.

> ⚠ Caution
>
> DB2 recognizes deadlocks automatically using parameter `DLCHKTIME` that specifies the time period during which the system analyzes lock situations or deadlocks.

12.7 Diagnostics: SQL Command Line

This screen provides a virtual DB2 command line processor.

You can access the *SQL Command Line* screen by calling the DBA Cockpit and choosing *Diagnostics* > *SQL Command Line*.

If you enter an SQL command, the following output options are available:

- On the tab page *Text-Based Output*, the result is displayed in a text editor in the same way as it is provided by the DB2 command line processor.
- On the tab page *ALV-Based Output*, the results are displayed in an ALV list . If there is more than one result set, you can switch between the resulting ALV lists by using the buttons *Show Next Result Set* and *Show Previous Result Set*.

You can also execute CLP commands that are supported by the stored procedure `ADMIN_CMD`.

12.8 Diagnostics: CLP Commands

You can access detailed information about CLP commands by calling the DBA Cockpit and choosing *Diagnostics* > *CLP Commands* on the *Database* tab page of the DBA Cockpit.

By selecting a specific database area from the *CLP Commands* dropdown list (for example, *Database Configuration*) and choosing the *Apply Selection* pushbutton, the corresponding DB2 CLP native commands are performed and the native output is provided on the *CLP Commands* screen.

> **i Note**
>
> The information provided on the *CLP Commands* screen is available in a formatted way also on other screens, for example, the *Database* screen in the *Performance* task area. But in contrast to these screens that only provide data for current database releases, the native view can provide the complete data also for future database releases.

12.9 Index Advisor

Use

Using the index advisor, you can retrieve recommendations about useful indexes and, therefore, improve the performance of queries. In addition, you can design new virtual indexes that can be validated before they are actually created.

You can access the index advisor by choosing *Index Advisor* in the *Favorites* list on the *System Landscape* tab page of the DBA Cockpit. The index advisor opens in a separate browser window.

Activities

You use the index advisor to perform one of the following actions:

- To evaluate the SQL statements in the dynamic SQL cache by receiving recommendations for potential new indexes that might improve the overall system performance
  Based on the current content of the dynamic SQL cache, the DB2 Design Advisor determines and recommends new indexes that might improve the overall system performance.
  For more information, see Retrieving Index Recommendations for the Dynamic SQL Cache [page 310].
- To determine and recommend new virtual indexes for a single SQL statement
  On the basis of a given SQL statement, the DB2 Design Advisor determines and recommends new indexes that might improve the performance of the query.
  For more information, see Retrieving Index Recommendations for a Single SQL Statement [page 312].
- To create user-defined virtual indexes
  If you are not satisfied with the recommendations of the DB2 Design Advisor, you create a virtual index specifically tailored to your requirements. For more information, see Defining Virtual User-Defined Indexes [page 314].
- To include the indexes in the EXPLAIN function when explaining a query
  You can check, for example, if the virtually defined indexes would improve the performance of a query. For more information, see Validating Indexes Using the EXPLAIN Function [page 315].

12.9.1 Retrieving Index Recommendations for the Dynamic SQL Cache

Procedure

1. On the *Index Advisor* screen, choose the *SQL Cache* radio button in the *Advisor Mode* area.
2. Choose the *Recommend Indexes* pushbutton.

   A background job starts. The DB2 Design Advisor analyses the current content of the dynamic SQL cache and the background job returns the results as soon as the DB2 Design Advisor has finished its analysis.

   The user interface remains in read-only mode while the analysis is still running. To check for results, you can either choose the *Update* pushbutton or wait until it is automatically checked for results every 60 seconds.

   The results are displayed in the following table:

Column	Description
Index Name	Name of the index

Column	Description
Type	• *Existing (not-used)* Index exists in the system catalog, but for the investigated SQL query it is not identified as usable by the DB2 optimizer. • *Recommended* Index is recommended by the DB2 Design Advisor. Recommended indexes that do not exist are candidates for new indexes to be created. **i Note** Indexes are displayed with the following background colors: • Green Recommended index that already exists and that the DB2 optimizer would use • White Existing index that is, however, not appropriate for the respective SQL statement • Red Recommended index that does not yet exist
Exists	• *Yes* Index exists in the database. • *No* Index is a virtual index.
Table Name	Table on which the index is defined
Schema	Name of the index schema
Uniqueness	Specifies a unique rule: • *Primary Key* • *Unique* • *Duplicates Allowed*
NLEAF	Number of leaf pages
NLEVELS	Number of index levels
Rev. Scans	• *Yes* Index supports reverse scans. • *No* Index does not support reverse scans.
Columns	Number of columns in the key plus the number of included columns if there have been any defined
INCLUDEs	Number of included columns

Column	Description
Column Names	List of column names

3. To retrieve more information about which SQL statement would benefit from the recommended indexes, select an index from the list.

   The details are displayed in the following table in the content detail area:

Column	Description
SQL Statement	Name of the SQL statement that is in the dynamic package cache and that would benefit from the index
Frequency	Number of times the statement has been executed since it has entered the dynamic SQL cache
Cost Saving	Estimated cost savings in percent after the index was created
Cost Before	Estimated SQL cost in timerons before the index was created
Cost After	Estimated SQL cost in timerons after the index was created

You can find the complete output of the DB2 design on the *Advisor Output* tab page. The complete output also includes the estimated space requirements of each recommended index.

> **i** Note
>
> If you do not want to display unused indexes, you can set a filter on the table accordingly.

12.9.2 Retrieving Index Recommendations for a Single SQL Statement

1. On the *Index Advisor* screen, choose the *Single Statement* radio button in the *Advisor Mode* area.
2. In the *SQL Statement* field below, enter the SQL statement that you want to investigate.
3. Choose the *Recommend Indexes* pushbutton.
   The DB2 Design Advisor evaluates existing indexes on the affected tables. If the DB2 Design Advisor cannot find an appropriate index in the system catalog, the tool recommends one or more indexes that might improve the performance of the query.
   The results are displayed in the following table:

Column	Description
Index Name	Name of the index

Column	Description
Type	• *Existing (not-used)* Index exists in the system catalog, but for the investigated SQL query it is not identified as usable by the DB2 optimizer. • *User-Defined* Index has been virtually defined by the user to determine whether such an index could be used to improve the query performance. Those indexes do not exist in the system catalog. • *Recommended* Index is recommended by the DB2 Design Advisor. Recommended indexes that do not exist are candidates for new indexes to be created. **Note** Indexes are displayed with the following background colors: • Green Recommended index that already exists and that the DB2 optimizer would use • White Existing index that is, however, not appropriate for the respective SQL statement • Red Recommended index that does not yet exist
Exists	• *Yes* Index exists in the database. • *No* Index is a virtual index.
Table Name	Table on which the index is defined
Schema	Name of the index schema
Uniqueness	Specifies a unique rule: • *Primary Key* • *Unique* • *Duplicates Allowed*
NLEAF	Number of leaf pages
NLEVELS	Number of index levels
Rev. Scans	• *Yes* Index supports reverse scans. • *No* Index does not support reverse scans.

Column	Description
Columns	Number of columns in the key plus the number of included columns if there have been any defined
INCLUDEs	Number of included columns
Column Names	List of column names

> **i** Note
>
> If you do not want to display unused indexes, you can set a filter on the table accordingly.

12.9.3 Defining Virtual User-Defined Indexes

Use

If the index recommendations provided by the DB2 Design Advisor do not meet your requirements, you can also define virtual user-defined indexes. In addition, you can validate their use by calling the EXPLAIN function.

Prerequisites

On the *Index Advisor* screen, you have chosen the *Single Statement* radio button in the *Advisor Mode* area and already retrieved index recommendations for a single SQL statement [page 312].

Procedure

1. On the *Indexes* tab page on the *Index Advisor* screen, choose the *Add Virtual Index* pushbutton.
2. In the *Index Advisor* dialog box, enter the schema and the table on which you want to define the virtual index.
3. If you want the virtual index to be unique, choose the *Unique* checkbox.

   > **i** Note
   >
   > By default, all new virtual indexes are created with the *Allow Reverse Scans* option on database level. However, in the ABAP Dictionary, you **cannot** define this option for new virtual indexes.

4. Choose the *Index Columns* pushbutton.
   The column names of the appropriate table are displayed in the *Table Columns* field.
5. To define index columns, either choose the *Add Column to Index* or the *Remove Column from Index* pushbutton.

6. To continue, choose the *Add Virtual Index* pushbutton.
   The new user-defined virtual index is added to the list of indexes.

> **i** Note
>
> User-defined indexes are always displayed by a red background color because they do not really exist like the recommended indexes. If you want to remove all user-defined indexes, choose the *Remove User-Defined Indexes* pushbutton.

Result

You can now use the EXPLAIN function to validate the existing, recommended, and newly created user-defined indexes. For more information, see *Validating Indexes Using the EXPLAIN Function*.

12.9.4 Validating Indexes Using the EXPLAIN Function

Procedure

On the *Index Advisor* screen, choose *EXPLAIN* and one of the following options from the dropdown list:

- *Only existing indexes*
  This option corresponds to the normal EXPLAIN function that is known from the SQL cache. Only indexes that are known from the system catalog are used to build the access plan.
- *Existing and recommended indexes*
  Indexes that are known from the system catalog and indexes that are recommended by DB2 are used to build the access plan.
- *Existing, recommended, and user-defined indexes*
  Indexes that are known from the system catalog and all virtual indexes (recommended and user-defined) are used to build the access plan.

Result

A new dialog window or Web browser appears displaying the access plan that the DB2 optimizer considers to be the most efficient one.

12.9.5 Creating Indexes in the ABAP Dictionary

Use

> **i** Note
>
> This function is only available for systems where you called up the DBA Cockpit locally, or for ABAP systems for which an additional RFC connection has been assigned.

You use the following procedure to create an index in the ABAP Dictionary that has been virtually defined before but does not yet exist.

> → Recommendation
>
> Additional indexes require additional space and need to be maintained when data is updated or inserted in a table. We recommend that you only create additional indexes if they really can improve the performance of queries that put a heavy load on your database.

Procedure

1. On the *Index Advisor* screen, choose an index (user-defined or recommended) that does not yet exist.
2. Choose the *Create Index in ABAP Dictionary* pushbutton (that is located next to the *Index Name* column).
   The *Create Index in ABAP Dictionary* dialog box appears.
3. Enter a description for the index and choose *Create*.
   The index is created in the ABAP Dictionary.
   After the index has successfully been created, you can decide if you want to schedule a `RUNSTATS` for the affected table.

> **i** Note
>
> If the index to be created is extending an existing unique index (including primary keys) with one or more INCLUDE columns, you have to replace the existing index with a new index instead of creating a new one.
>
> In this case, the *Replace Existing Unique Index* dialog box appears instead of the *Create Index in ABAP Dictionary* dialog box. If you replace the existing index, the system replaces the index **only** on database level. This means that no changes are applied to the ABAP Dictionary. The replacement is automatically scheduled as an SQL script in the DBA Planning Calendar.

12.10 Diagnostics: Cumulative SQL Trace

You can access trace information about SQL statements by calling the DBA Cockpit and choosing *Diagnostics* > *Cumulative SQL Trace* on the *Database* tab page of the DBA Cockpit. The *Cumulative SQL Trace* screen appears.

If you want to retrieve new data, choose the *Refresh* pushbutton. On the *EXECUTE Time*, *PREPARE Time*, and *FETCH Time* tab pages, the following information is displayed for the SQL statements:

Column	Description
Total Time	Cumulative execution time of a statement
%	Proportional execution time of one statement with regard to all executed statements
Count	Number of executions
Time/Stmt	Average execution time of one statement
Table	Name of the table the SQL statement reads from If the statement reads from more than one table, only the name of the first table is displayed on this screen. The other names are displayed under *Statement Information* on the detail screen.
SQL Statement	Complete SQL statement

If you want to display more detailed information, choose an entry from the list. The *Cumulative SQL Trace - Details* area appears providing information about the optimization level, the query degree as well as the first and last statement usage.

In addition, the following tab pages are displayed:

- *Statement Information*
  Displays the complete SQL statement, the application server where the statement was executed, and a list of all ABAP reports in which the statement can be found
- *Time Histograms*
  Displays the distribution times of the selected SQL statement

If you want to display the access plan for the execution of a single statement, select the relevant entry and choose the *Explain* pushbutton. This function provides a detailed analysis of expensive SQL statements. For more information, see EXPLAIN Function [page 297].

To display the ABAP source program where the statement was defined, choose an entry in the list of ABAP reports. An editor screen appears, which contains the related source.

> ⚠ Caution
>
> Since all trace data remains permanently in the database, you should delete obsolete data before starting a new trace. To do so, choose the *Delete* pushbutton in the global toolbar.

For information about activating the cumulative SQL trace function, see **SAP Note** 139286.

12.11 Diagnostics: DBSL Trace Directory

Use

> **i** Note
>
> This function is **not** available for systems monitored using a remote database connection.

You can access information about the sequential DBSL trace and the DBSL deadlock trace by calling the DBA Cockpit and choosing *Diagnostics* > *DBSL Trace Directory* in the navigation frame (SAP GUI) or on the *Database* tab page (Web browser) of the DBA Cockpit.

By default, the trace files are stored in the following directories:

- UNIX: `/tmp/TraceFiles`
- Windows: `<DRIVE>:\usr\sap\TraceFiles`

To perform actions such as activating and deactivating the trace function or changing the trace level, see Diagnostics: Trace Status [page 318].

More Information

- SAP Note 31707 (for information about the sequential DBSL trace)
- SAP Note 175036 (for information about the DBSL deadlock trace)

12.12 Diagnostics: Trace Status

> **i** Note
>
> This function is only available for systems where you called up the DBA Cockpit locally, or for ABAP systems for which an additional RFC connection has been asssigned.

You can access information about the current trace status by calling the DBA Cockpit and choosing *Diagnostics* > *Trace Status* on the *Database* tab page of the DBA Cockpit. On the *Trace Status* screen, the following information is displayed:

DBSL Trace	Description
Trace Level	Specifies the amount of data to be traced The following trace levels are available: • *0* or *1*: Trace is off. • *2*: Only statements are traced. • *3*: Statements and results are traced.
Number of I/O Records to Be Traced	Number of result records to be traced for a statement This value is only displayed if trace level *3* is activated.
Display Length for String/Raw Data	Maximum output length
DBSL Trace Search String	If provided, only SQL statements containing this string are traced.
DBSL Trace Minimum Time Limit	If provided, only SQL statements with execution times higher than this time limit are traced.
Cumulative Trace	
First Trace Entry	Displays the start time of this trace if trace information already exists
Last Trace Entry	Displays the end time of this trace if trace information already exists
Number of Entries	Displays the number of entries in this trace if trace information already exists
Deadlock Trace	
Detection Interval	Only SQL statements running longer than this time are recorded for deadlock detection.

For each trace, a pushbutton shows whether the trace is active or switched off.

To change the information to be displayed on the *Trace Status* screen, choose the *Change* pushbutton. You can then perform one or more of the following actions:

- Choose if you want the trace status to be displayed on the connected application server or on all application servers by using the relevant radio button.
- Activate or deactivate the trace function by choosing the *Trace Off* or *Trace On* pushbutton.
- Change the trace level.
- Maintain the trace parameters in the *DBSL Trace Attributes* area.

12.13 Diagnostics: Message History

> **Note**
>
> This function is only available if the data collection framework is set up properly. In particular, you have to ensure that the `Diag Log` data collector is correctly configured because it keeps a local history of special `db2diag.log` and notification log entries.
>
> - For more information, see Data Collection Framework: Configuration [page 243].

On the *Message History* screen, information about relevant DB2 errors and informational messages are displayed that help you identify critical situations.

You can access information about the history of messages that are written to log files by calling the DBA Cockpit and choosing *Diagnostics* > *Message History* on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI) of the DBA Cockpit.

In the *Selection* area, you can do the following:

- Specify a time frame for which you want to display log file messages.
- Set additional filter criteria by using the keyword search in the *Message Code* and *Message Text* fields. You can use the asterisk (*) character in these fields.

After you have applied your selections, the following information is displayed:

Column	Description
Time of Collection	Time when the message was created
Time Stamp of First Message	Time when the message was created **Note** This column replaces the *Time of Collection* column if you have chosen one of the grouping options as display type. Only the first message is displayed in detail whereas the other messages are only added up and given as a total number.
Member	Identifier for the database member from which the data was retrieved **Note** This column is only visible in a DPF system.

Column	Description
Count	Total number of messages matching the *Display Type* criteria **i Note** This column is only visible if you have chosen one of the grouping options as display type.
Message Type	Message type that is related to the message identifier **Example** *ADM* is used for administration or notification log messages.
Message Number	Message number of the associated message **Example** *6102* is the numerical part of the *ADM6102I* message code.
Level	Indicates the severity level of the message The following values are possible: • *C* Indicates a critical situation • *E* Indicates that an error occurred • *I* Displayed message is only informational • *S* Indicates a severe situation, for example, deadlocks have been detected • *W* Indicates that the message contains a warning
Impact	Qualifies the impact of the message from a DBA's perspective The following values are possible: • *CRITICAL* • *IMMEDIATE* • *NONE* • *POTENTIAL* • *UNLIKELY*
Message Text	Short description of the message

12.14 Diagnostics: Diagnostic Logs

The *Diagnostics Logs* screen provides you with information about all relevant log files of DB2 including the following:

- Database diagnostic log (`db2diag.log`)

  > **i Note**
  >
  > Since DB2's automatic `RUNSTATS` have been introduced, the `db2diag.log` can rapidly grow in size up to several gigabytes – depending on the value of the `DIAGLEVEL` database configuration parameter. By default, the value of this parameter is *3*. We therefore recommend that you regularly switch the `db2diag.log` using the automation function as described in Configuring Monitoring Settings [page 231].

- Database notification log (`<instance_name>.nfy`)
- Statistics log

You can access the DB2 logs by calling the DBA Cockpit and choosing *Diagnostics* > *Diagnostic Logs* on the *Database* tab page of the DBA Cockpit.

To avoid unnecessary system workload, you can restrict the amount of data that is processed by specifying the following in the *Selection* area:

- A *Log Facility*
- A *Record Type*
- The minimum impact level (*Impact*)
- A time range (*Messages From / To*)

If you apply your selection, information according to the specified values is displayed including the appropriate DB2 component. To display additional details about a log entry, click the relevant entry in the list.

12.15 Diagnostics: Dump Directory

The DB2 dump directory contains the following files:

- DB2 diag log (`db2diag.log`)
- DB2 notification log (`<instance_name>.nfy`)
- DB2 dump files
- User exit log and error files
- Trace files

The system displays the content of the directory specified by the diagnostic data directory path (`diagpath`). This path is configured in the database manager configuration.

You can access the dump directory by calling the DBA Cockpit and choosing *Diagnostics* > *Dump Directory* on the *Database* tab page (Web browser) or in the navigation frame of the DBA Cockpit (SAP GUI).

If you want to display the content of an error log or a trace file, click the relevant file. The *File Viewer* area appears displaying the file contents.

> ⚠ Caution
>
> Avoid viewing large files because you can run out of memory in the ABAP system. To view the `db2diag.log` file, for example, use the *Diagnostics Logs* or *Message History* screens instead.

12.16 Diagnostics: Displaying Extended Latch Waits

Context

On the *Extended Latch Waits* screen, you can display a list of current and past extended latch waits, including:

- The total extended latch wait time for each latch
- The average total extended latch wait time for each latch
- The total number of extended latch waits for each latch

This screen helps you to identify and analyze potential performance bottlenecks.

Procedure

1. In the DBA Cockpit, on the *Database* tab page, choose *Diagnostics* > *Extended Latch Waits*.
2. In the selection area, enter a time frame.

   The DBA Cockpit then displays an overview of all extended latch waits with their aggregated figures for the selected time frame.
3. To display details of a single extended latch wait, select a line in the overview table.

   The DBA Cockpit then displays additional tab pages with a summary, history data, and a graphical view of the history data.
4. To isolate the time frame during which the extended latch wait occurred, analyze the graphic on the *History: Graphical View* tab page, or check the corresponding data in the table on the *History* tab page.

   > i Note
   >
   > On the *History* tab page and on the *History: Graphical View* tab page, the total extended latch wait times are calculated as the elapsed times for extended latch waits between the last measuring point (time of collection) and the current measuring point (time of collection). Average total extended latch wait times and total extended latch waits are calculated in a similar way.

13 BW Administration

Under *BW Administration* in the DBA Cockpit menu, you can find tools for administrating SAP Business Warehouse (SAP BW) systems and near-line storage databases for SAP BW.

> **i** Note
>
> All BW-related actions in the DBA Cockpit except BW data distribution use BW-specific APIs, which are not available using a remote connection. Therefore, the BW-related screens are only available if you use the DBA Cockpit in a local system. The only exception is the *BW Data Distribution* screen, which can also be used for remote systems.

Related Information

BW Data Distribution [page 324]
Improving MDC Settings Using the MDC Advisor [page 326]
Scheduling a Health Check for the SAP BW System [page 327]
Administration and Monitoring of the Near-Line Storage (NLS) Database [page 330]
BW Object Checks [page 325]

13.1 BW Data Distribution

You can use the BW data distribution to add new partitions to an existing database partition group and redistribute the database partition group.

> **⚠** Caution
>
> This tool is available for compatibility with older SAP BW releases. Use it only for tables with little or no data. We recommend that you use program `DB6CONV` for all other cases.
>
> For more information about your tool options for data distribution, see the documentation *SAP Business Warehouse on IBM Db2 for Linux, UNIX, and Windows: Administration Tasks* on SAP Help Portal at https://help.sap.com/viewer/db6_bw.

Prerequisites

This screen is **only** available in the SAP GUI-based version of the DBA Cockpit.

Context

During the SAP system installation you can add additional database partitions either using the SAP installer (software provisioning manager) or manually using `db2start`. Before the partitions that you have added can become active, you can use the BW data distribution to perform the following actions:

- Change the assignment of database partitions to database partition groups
- Define tablespace containers on the new database partition
- Determine if and when the affected tablespaces will be redistributed

Procedure

1. In the navigation frame of the DBA Cockpit, choose *BW Administration* > *BW Data Distribution*.
2. Follow the instructions of the BW data distribution wizard.

13.2 BW Object Checks

The SAP BW object checks analyze the table organization and metadata of InfoCubes, DataStore Objects, InfoObjects, and PSAs and check these BW objects for consistency.

Prerequisites

The BW object checks are only available in SAP Business Warehouse (SAP BW) systems.

You need at least the support packages listed in SAP Note 2731030.

Context

The BW object checks are useful in the following situations:

- You have performance issues, which might be due to missing indexes or an inconsistent table organization of a BW object.
- You want to create a system copy or you are planning a database upgrade. You want to ensure that the current BW objects are consistent.
- You are planning a conversion of many BW objects to BLU Acceleration.
- SAP support asks you to run the BW object checks to find the root cause of system issues.
- You want to run these checks as regular "housekeeping" jobs, for example, once a year.

Procedure

1. Call the DBA Cockpit (transaction `DBACOCKPIT`).
2. In the menu, choose *BW Administration* > *BW Object Checks*.

   When you call up the BW object checks for the first time, no object list is shown. If you have already used the BW object checks, the object list from the last check is shown.
3. In the selection area at the top of the screen, select the BW object type that you want to check.
4. In the *Filter* field, you can enter character strings, including the asterisk (*), to filter for specific BW object names.
5. Choose *List Objects*.

   The system returns a list of BW objects that match your selection criteria. Note that no checks have run at this moment.
6. To initiate the checks, choose *Check Listed Objects*.

   The checks now run in the background.
7. Choose the *Refresh* button to update the status shown for the objects.

Results

For each BW object, the number of errors and warnings are shown. Expand the grouping for each object to see detailed log messages and their long texts (if available).

13.3 Improving MDC Settings Using the MDC Advisor

The MDC advisor is a DB2 tool that proposes multidimensional clustering (MDC) settings for tables using queries executed on these tables.

Context

You use the MDC advisor to collect BW reporting queries for an InfoProvider, that is, for InfoCubes and DataStore objects. The MDC advisor analyzes the collected BW reporting queries and returns a proposal for MDC settings for the fact tables of an InfoCube and for the active table of a DataStore object. Based on this MDC proposal, you can change the clustering of the analyzed InfoProvider using transaction RSA1 in your SAP system.

Procedure

To access the MDC advisor, call the DBA Cockpit (transaction `DBACOCKPIT`) and choose *BW Administration* ▸ *MDC Advisor* in the SAP GUI navigation frame of the DBA Cockpit.

More Information

For more information about the MDC advisor, see the guide *SAP Business Warehouse on IBM Db2 for Linux, UNIX, and Windows: Administration Tasks* on SAP Help Portal at https://help.sap.com/viewer/db6_bw.

13.4 Scheduling a Health Check for the SAP BW System

Use

You use BW health checks in the DBA Cockpit to verify if the main DB2 features for SAP BW have been correctly implemented in your system. Using the data provided on the *BW Health Checks* screen, you can check the following:

- If there is a high disk space consumption because of multidimensional clustering (MDC) for InfoCubes and DataStore objects that was unsuitably defined
- If you have wrong distribution keys for InfoCubes, DataStore objects, and PSA tables
- If the fact tables of your InfoCubes are distributed over the same database partitions
- If the partitioning columns of your InfoCubes are filled correctly

Prerequisites

This screen is **only** available if you run the DBA Cockpit in a local SAP BW system and if your SAP system release is based on SAP NetWeaver 7.3 or higher.

Process

Displaying BW Health Checks

1. To access the *BW Health Checks* screen, call transaction `DBACOCKPIT` and choose *BW Administration* ▸ *BW Health Checks* in the SAP GUI navigation frame of the DBA Cockpit.
   The *BW Health Checks* screen appears displaying the date and time of the last MDC space check, distribution key check, and BW-specific layout check.

2. To display more information about each kind of check, choose the information pushbutton.

Scheduling BW Health Checks

1. On the *BW Health Checks* screen, choose the *Schedule BW Health Checks* pushbutton.
2. In the *Select BW Health Checks* dialog box, select one or more of the following options:
   - *MDC Space Check*:
     If MDC has been defined for an InfoCube or all standard and transactional DataStore objects, this check verifies if the disk space consumption exceeds a specific threshold.
   - *Distribution Key Check*
     This check verifies if the distribution keys of your InfoProviders and PSA tables have been correctly defined. The following objects are checked:
     - All InfoCubes and their aggregates
     - All standard, transactional, and write-optimized DataStore objects
     - All PSA tables

     > → Recommendation
     >
     > If you are using a single-partition database, InfoProviders and PSA tables are already created with a distribution key. We recommend that you check the distribution keys of your InfoProviders and PSA tables **before** you add new partitions to your database.
   - *BW-Specific Layout Check*
     This check verifies if both fact tables of an InfoCube are distributed over the same database partitions. In addition, it checks whether the column `SID_0CALMONTH` or `SID_0FISCPER` is filled correctly with data if MDC was defined for an InfoCube with one of these columns. You **cannot** use the *BW-Specific Layout Check* to check SAP BW compression.
3. Choose the *Execute* pushbutton.
   The *Schedule Job* dialog box appears.
4. Choose if you want the check to start immediately in the background or at a later date, and confirm your selection.

After the jobs have finished, the results are displayed in the *Results of Consistency Check* area as follows:

Type of Check	Result	Description

MDC Space Check	*MDC Disk Space Consumption*	• The *MDC Space Check* returns **no error** if one of the following applies:
		  • MDC is not defined for an InfoCube or DataStore object.
		  • MDC is defined for an InfoCube or DataStore object, and the additional disk space consumption with MDC is less or equal 33%.
		• The *MDC Space Check* returns a **warning** if MDC is defined for an InfoCube or a DataStore object, and the additional disk space consumption with MDC is more than 33% but less than 50%.
		• The *MDC Space Check* returns an **error** if MDC is defined for an InfoCube or DataStore object, and the additional disk space consumption with MDC is more than 50%.
		i Note
		If the *MDC Space Check* returns a warning or an error, you should change the defined MDC settings using the *Repair* pushbutton.

Distribution Key Check	*Distribution Key*	• The *Distribution Key Check* returns **no errors** if the distribution key of an InfoCube, DataStore object, or PSA table is correct. • The *Distribution Key Check* returns an **error** if one of the distribution keys of an InfoCube, DataStore object, or PSA table is missing or is not correct or is not defined. If the database is not partitioned, you can repair the distribution key using the *Repair* pushbutton. Otherwise, we recommend that you redistribute the table.

Type of Check	Result	Description
BW-Specific Layout Check	• *Data Distribution* • *PARTTIM Column*	• The *BW-Specific Layout Check* returns **no errors** if both fact tables of an InfoCube are distributed over the same database partitions or if the column `SID_0CALMONTH` or `SID_0FISCPER` is filled correctly with data. • The *BW-Specific Layout Check* returns **warnings** if both fact tables of an InfoCube are not distributed over the same database partitions or if the column `SID_0CALMONTH` or `SID_0FISCPER` is not correctly filled with data. If the distribution check of the fact tables returns warnings, we recommend that you redistribute the tables using DB6CONV. If the check of the column `SID_0CALMONTH` or `SID_0FISCPER` returns warnings, open a customer message on SAP Service Marketplace.

> **i** Note
>
> You can monitor the background job in the *DBA Planning Calendar*.

More Information

DBA Planning Calendar [page 255]

13.5 Administration and Monitoring of the Near-Line Storage (NLS) Database

The DBA Cockpit provides a few functions that help with the administration and monitoring of a near-line storage database.

Near-Line Storage

Near-line storage (NLS) is a category of data persistency that is similar to archiving. The overall goal is to take read-only data out of the BW database and to store it in an additional near-line storage DB2 database.

The database server and the storage devices of the near-line storage solution can be separated from the SAP system database hardware, but you can still access the separated data transparently from an SAP BW system.

Near-Line Storage Administration and Monitoring in the DBA Cockpit

Using the DBA Cockpit, you can perform the following administrative and monitoring tasks for NLS:

- Configure the NLS database as a monitored system in the DBA Cockpit
  For more information, see Setting Up the DBA Cockpit for Remote Monitoring [page 34].
- Configure the database connections that are required to access the NLS database from the local BW system using the *NLS Configuration* screen in the DBA Cockpit

  > **i Note**
  >
  > This task is an extension to the standard database connection maintenance.

- Monitor the BW objects of which data has been archived to the NLS database using the *NLS Overview* screen in the DBA Cockpit
  On this screen, you can monitor the space consumption of BW objects in both the SAP BW database and in the related NLS database.
- Clean up invalidated requests in the NLS database using the NLS Cleanup Job in the DBA Planning Calendar [page 274]

More Information

For more information, see the documentation *Enabling SAP Business Warehouse Systems to Use IBM Db2 for Linux, UNIX, and Windows as Near-Line Storage (NLS)* on SAP Help Portal at https://help.sap.com/viewer/db6_nls.

13.5.1 Configuring the Connection to the NLS Database

You use the *NLS Configuration* screen to configure the connection to the NLS database.

> **⚠ Caution**
>
> If you create the NLS connection using the *Database Connection* screen rather than the *NLS Configuration* screen, the configuration data for the NLS connection is **not** created. Make sure that you use the *NLS Configuration* screen.
>
> The connect user specified for the NLS database connection requires access to application data. Therefore, the NLS database connection should not be the same as the database connection that is used for the monitoring of the NLS database itself.

Prerequisites

This screen is only available if you run the DBA Cockpit in a local SAP BW system.

Procedure

1. To access the *BW Administration – NLS Configuration* screen, call the DBA Cockpit and choose *BW Administration* > *NLS Configuration*.
   The *NLS Configuration* screen shows all NLS databases that are connected to your SAP BW system and provides the following information:

Column	Description
Connection Name	Name of the database connection that is used to access the database
DB Name	Name of the assigned NLS database
DB Server	Name of the host where the NLS database is installed
Schema Name	Name of the NLS database schema
User Name	Name of the database user that is used to connect to the NLS database
Configuration Status	Indicates if the configuration information, which is stored as part of the NLS connection, matches the SAP BW system information If the *Configuration Status* is not correct, the SAP BW system cannot use an NLS connection. To update the configuration of the NLS connection, choose the *Update Configuration* pushbutton.
Connection Status	Indicates if the remote NLS database is available **i Note** By default, this field is empty. To check the connection status, choose the *Test Connection* pushbutton.

2. You can now add new NLS configuration entries, change NLS configuration entries, or delete them.
   To perform these tasks, the standard maintenance screen for the configuration of database connections is used. In all cases, a dialog box appears with a detailed maintenance screen for the database connection related to the NLS database.
3. To check the availability of the NLS connection, choose the *Test Connection* pushbutton.
   The result of the connection test appears in the action message window in the lower half of the screen.

More Information

Configuring Database Connections [page 45]

13.5.2 Overview of the NLS Database

Use

The *NLS Overview* screen provides you with an overview of InfoProviders that are connected to an NLS database. You can use the information on this screen to analyze the space consumption of an InfoProvider in the BW database in comparison to the remote NLS database. In addition, you can update the configuration of your BW queries so that they automatically read data from the NLS database.

Prerequisites

This screen is **only** available if you run the DBA Cockpit in a **local** SAP BW system.

Process

To access the *NLS Overview* screen, call the DBA Cockpit and choose *BW Administration* > *NLS Overview*. To specify which InfoProviders are displayed, enter the required data in the following input fields and choose the *Apply* pushbutton:

- *InfoProvider Name*
- *InfoProvider Size (KB)*
- *NLS Connection*
- *NLS Size*

The screen then displays the following information about all InfoProviders that meet the specified selection criteria:

Column	Description
InfoProvider Name	Name of the InfoProvider

Column	Description
InfoProvider Size (KB)	Total size of all database tables of the InfoProvider in KB **i Note** This size information is based on data collected by the DCF. For more information, see Enablement of Databases for the Data Collection Framework [page 48].
NLS Connection	Name of the NLS connection
NLS Name	Name of the NLS object
NLS Size	Total size of all database tables of the InfoProvider in the NLS database **i Note** This value is only available if the NLS database has been configured and properly set up for monitoring in the DBA Cockpit. For more information, see Setting Up the DBA Cockpit for Remote Monitoring [page 34] and Scheduling a REORGCHK for All Tables [page 272].
Query Enabled	Indicates whether the BW queries of the InfoProvider have been enabled to read data from the NLS database

Enabling BW Queries to Use the Near-Line Storage Database

By choosing the *Enable Queries* pushbutton, you enable all BW queries of a single InfoProvider to read data from the NLS database. After a data archiving process has been created for an InfoProvider, the InfoProvider can no longer access the NLS database to build the query results.

Displaying Details of InfoProviders

To display detailed information about an InfoProvider, double-click an InfoProvider, or select it and choose the *Details* pushbutton. The *NLS Details* screen appears displaying details of the InfoProviders in two tables:

In the left table, the schema name, table name, and size of the database tables that are associated with the InfoProvider in the BW database are displayed. The right table displays the same information for the database tables representing the InfoProvider in the NLS database.

Analyzing Single Tables of the BW and the NLS Database

To analyze a single table of an InfoProvider, double-click an InfoProvider in the left table on the *NLS Details* screen, or select the InfoProvider and choose the *Details* pushbutton. The *Table and Indexes Detail* screen appears providing detailed technical information, for example, about the structure of the selected table, its indexes, and the tablespaces used.

⚠ Caution

If you want to analyze a single NLS table, you **cannot** navigate to the *Single Table Analysis* screen in the *Space* task area by double-clicking a table cell in the right table of the *NLS Details* screen.

Instead, proceed as follows:

1. On the *System Landscape* tab page, select your NLS database system.
2. Go to the *Database* tab page and choose ▶ *Space* ▶ *Single Table Analysis* ▶.
3. On the *Single Table Analysis* screen in the *Space* task area, enter the schema and the table name of the NLS table in the appropriate fields.

More Information

Analyzing and Maintaining a Single Table [page 152]

13.5.3 Synchronization Check of the NLS Database

On the *NLS Synchronization Check* screen, you can check whether the NLS database is consistent with the attached SAP BW system. If, for example, you recovered either the BW database or the NLS database, you must check whether both systems are synchronized and consistent.

Prerequisites

This screen is **only** available if you run the DBA Cockpit in a **local** SAP BW system.

Procedure

1. To access the NLS synchronisation check, call the DBA Cockpit and choose ▶ *BW Administration* ▶ *NLS Synchronisation Check* ▶.
2. Choose the NLS connection that you want to check from the dropdown list.
   The consistency of the NLS database and the SAP BW system is checked. The following output is displayed:

Column	Description
Checked Object	Tree structure with the name of the InfoProvider and of inconsistent archiving requests
Required Action	Action that is required to repair the inconsistent request
Description	Additional information about the required action

Column	Description
Target Timestamp	Time stamp of the operation as contained in the SAP BW system

The consistency check returns one of the following results:

- `The NLS database is consistent`
  No further action is required.
- `The NLS database is not consistent and it can be resynchronized`
  If the NLS connection is not consistent but it can be resynchronized, the archiving requests with problems are displayed for each InfoProvider as well as the corrective action that you must take. To resynchronize the NLS database, choose the *Resynchronize* pushbutton.
- `The NLS database is not consistent and it must be restored`
  In this case, you have to restore the NLS database to the point in time that is displayed in column *Target Timestamp*.

3. For more information about the steps to take to achieve consistency, consult the detailed documentation *Enabling SAP Business Warehouse Systems to Use IBM Db2 for Linux, UNIX, and Windows as Near-Line Storage (NLS)* on SAP Help Portal at https://help.sap.com/viewer/db6_nls, especially the section *Achieving System Consistency After an SAP BW or NLS Database Recovery*.

14 Workload Management

Use

Workload management lets you distinguish and prioritize different types of work on the database. Workloads identify the submitters of work by connection properties and assign incoming work to service classes. You use service classes to monitor and control resource consumption of the different workloads.

In addition, the information provided on the screens in the *Workload Management* task area allows you to define thresholds on resources to capture information or even cancel database activities that violate a threshold. Thresholds are either checked proactively before the activity begins based on the projected impact, or reactively while the activity is running and consuming resources.

You can access data about workload management by calling the DBA Cockpit and choosing *Workload Management* > `<activity option>` on the *Database* tab page (Web browser) or in the navigation frame (SAP GUI).

> **i** Note
>
> Setting up workload management is optional. If workload management has not been configured, the database does not distinguish different types of work.

More Information

Introduction to DB2 workload management concepts in the IBM Db2 documentation

Control of work with thresholds in the IBM Db2 documentation

14.1 Workload Management: Threshold Violations

On the *Threshold Violations* screen, you can analyze threshold violations that occurred on the database. You can display threshold violations that were captured during a certain period of time by specifying a time frame in the *Selection* area and choosing the *Apply Selection* pushbutton.

The following information is displayed:

Column	Description
Violation Time	Time when the threshold violation occurred

Column	Description
Member	Database member where the violation occurred
Predicate	Predicate that was violated **i Note** Activities with the value `MANUAL` were captured manually on the *Applications* screen. For more information, see Performance: Applications (DB2 V9.7 or Higher) [page 93].
Violated Value	Value that was exceeded and that violated the threshold predicate
Service Superclass	Name of the service superclass in which the violation occurred
Service Subclass	Name of the service subclass in which the violation occurred

By default, the history of a threshold violation is kept for two weeks. To delete all recorded threshold violations, choose the *Reset Violation History* pushbutton.

For more information about how to change the size of the violation history, see Data Collection Framework: Configuration [page 243].

Displaying Details About Threshold Violations

To display details about a threshold violation, choose a line in the overview table.

If the selected threshold violation has caused the recording of an SQL statement, you can use the *EXPLAIN from Activity* button to display the access plan of this statement including section actuals. Section actuals are runtime statistics that are collected during the execution of an SQL statement. Section actuals contain, for example, the actual number of rows for each database object and each operation in the execution plan.

i Note

To record the SQL statement, its access plan, and section actuals with a threshold violation, you need to choose *Activity Data with Statements, Bind Values and Sections* for the *Data Collection upon Threshold Violation* field when defining the threshold.

All other details about the selected threshold violation are displayed in the content detail area on the following tab pages:

- *General*
  Displays details about the activity that violated the threshold as well as the execution statistics for that particular activity.
- *SQL Statement(s)*
  Displays details about the SQL statement that was executed as part of the activity.
  For nested statement calls - for example, during the execution of stored procedures - more than one SQL statement can be available.

To retrieve a newly generated access plan, pick an SQL statement from the list and choose the *EXPLAIN* pushbutton. The new access plan is displayed in a separate window.

14.2 Workload Management: Threshold Configuration

Use

On the *Threshold Configuration* screen, you can maintain thresholds on the database system, that is, create and drop thresholds, enable and disable thresholds, or change predicate values.

Features

On the *Threshold Configuration* screen, all defined thresholds for the database system are displayed. You can perform the following actions:

- Enable or disable a threshold by selecting a line from the list and choosing the *Enable / Disable Threshold* button.
- Create a threshold by choosing the *Create Threshold* button and entering the relevant settings in the dialog window that appears.

  > **i Note**
  >
  > The predicates that are available for threshold settings in the DBA Cockpit are limited to those that are considered to be a use case in an SAP environment. DB2, however, allows you to define thresholds on more predicates than are available in the DBA Cockpit.
  >
  > Be aware that inappropriate predicate values can cause a system slowdown due to increased I/O. Thresholds that are violated too often force the event monitor to collect and write details of each threshold violation to the respective event monitor tables. To get meaningful predicate values when defining thresholds, use the statistics that are available on the Performance: Workload Statistics [page 124] screen.
- Edit an existing threshold by selecting one of the displayed thresholds and choosing the *Edit Threshold* button.
- Drop a threshold by selecting one of the displayed thresholds and choosing the *Drop Threshold* button.

> **i Note**
>
> If the WLM event monitors have not yet been created on the database, you can do so on the *Data Collection Framework: Configuration* screen under *Configuration* in the DBA Cockpit.
>
> For more information, see Data Collection Framework: Configuration [page 243].

14.3 Workload Management: Workloads and Service Classes

The *Workloads and Service Classes* screen provides you with an overview of the workload management configuration of the database system. In a tree structure, all configured workloads, service superclasses, and subclasses are displayed including the workload evaluation order and whether they are enabled or not.

If you choose a service class from the tree structure, details about this service class are displayed in the *Class Definition*, *Data Collection*, and *Prioritization* sections in the content detail area.

If you choose a workload from the tree structure, the relevant details are displayed in the *Workload Definition* and *Data Collection* sections in the content detail area.

In addition, if you choose a workload from the overview, the connection attributes are displayed, that is, all the attributes that associate an incoming activity with the selected workload. For an activity to be associated with a certain workload, all the connection attributes of the incoming activity must match the definition of the workload (Boolean *AND*). If a single attribute type is specified more than once, this indicates that only one of them must match the activity (Boolean *OR*).

14.4 Workload Management: SAP WLM Setup Status

In addition to the default workloads and service classes that are created automatically during the setup of history data collectors in an SAP environment, you can set up additional workloads and service classes using the *SAP WLM Setup Status* screen. The workload management setup that is supported by the DBA Cockpit is separated into the following categories:

- SAP enhanced prioritization scheme
- SAPTOOLS workload and service class
- Priority aging service class

SAP Enhanced Prioritization Scheme

In addition to the workload management setup that is based on the work process types and that is set up by default for SAP systems (see also The Data Collection Framework [page 48]), you can create an additional workload and service class using the enhanced prioritization scheme. This additional workload and service class consists of one of the following groups:

- SAP users
- SAP transactions
- SAP application servers

You can monitor and prioritize this special group separately from the workloads that are based on work process types.

If you want to drop the enhanced prioritization scheme, choose the *Modify Enhanced Prioritization Scheme* button. In the wizard that appears, choose *No enhanced priority scheme*.

If you want to drop the enhanced prioritization scheme as well as the workloads and service classes that identify the SAP work process types, choose the *Drop SAP Workloads and Service Classes* button.

> → Recommendation
>
> We recommend that you always keep the work process type- based WLM setup to allow monitoring based on SAP work process types.

SAPTOOLS Workload and Service Class

For SAP systems, the separation of SAP monitoring tools is set up by default (see also The Data Collection Framework [page 48]). If you want to remove the SAPTOOLS workload and service class, you can do so by choosing the *Drop SAPTOOLS Workload and Service Class* button.

If the SAPTOOLS workload and the service class have not yet been created, you can do so by choosing the *Set Up SAPTOOLS Workload and Service Class* button.

Priority Aging Service Class

The priority aging feature lets you change the priority of in-progress activities over time. You use priority aging to decrease the priority of longer-running activities so that the throughput for shorter-running activities stays at a stable level. If the priority aging service class has not yet been created, you can create it by choosing the *Set Up Service Class for Priority Aging* button. To drop the priority aging service class, choose the *Drop Service Class for Priority Aging* button. To enable priority aging based on defined thresholds, you have to set up corresponding service class thresholds as described in Workload Management: Threshold Configuration [page 339].

For more information about the priority aging feature, see the IBM Db2 documentation .

14.5 Setup of Workloads and Service Classes in SAP Environments

If the database that is set up for monitoring belongs to an SAP system (ABAP stack) with SAP Basis Release 6.20 or higher, a set of SAP-specific workloads and service classes is created with the *Framework* data collector.

Dedicated workloads are created to identify each of the work process types in an SAP environment, that is, one workload per SAP work process type. Each of these workloads is then assigned to its own service class. This setup lets you prioritize and monitor work per SAP work process type. This means that the drill-down capabilities of monitoring data in an SAP environment are improved significantly.

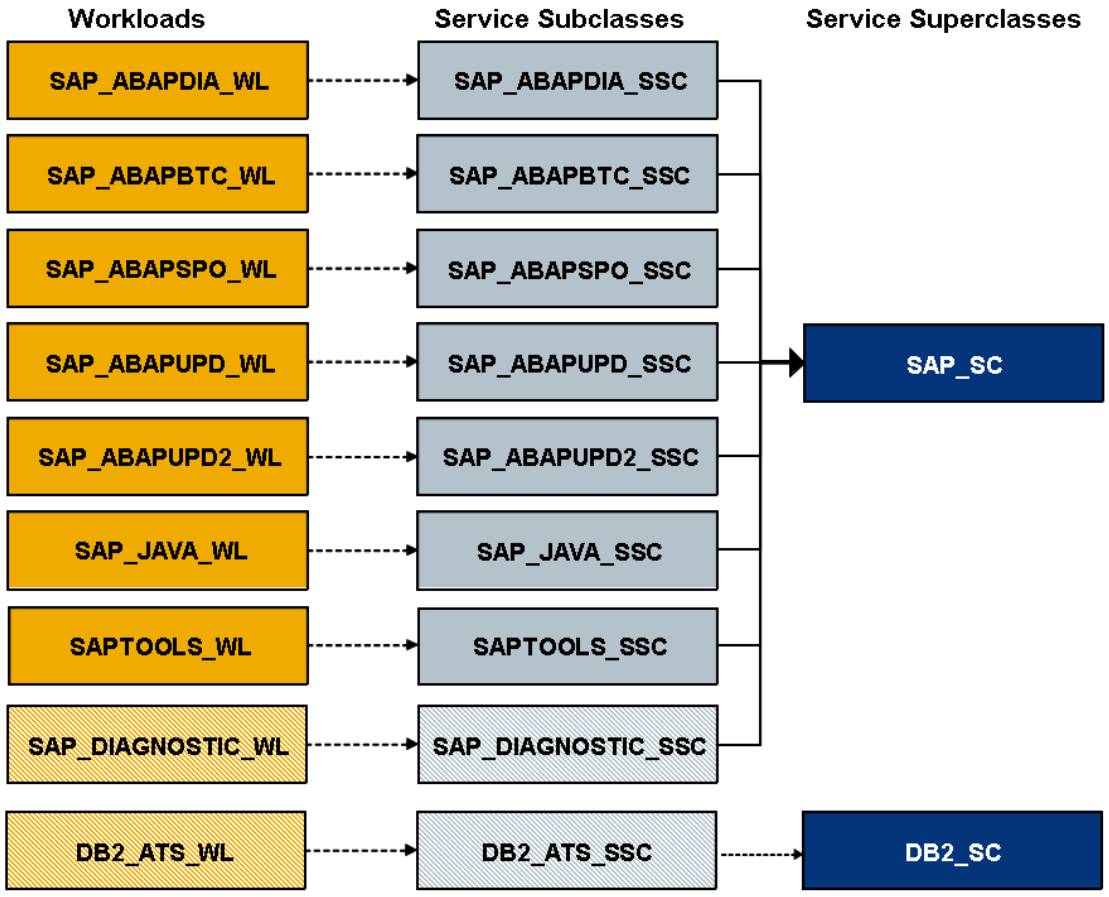
The following work process types are identified in an SAP environment:

- ABAP dialog
- ABAP batch
- ABAP spool
- ABAP updater (primary and secondary)
- Java (if available)

> **i** Note
>
> Since the SAP Enqueue work process is not connected to the database, **no** workload and service class are created for this type of work process.

All work coming from outside the SAP system still runs in the DB2 default workload and service classes.

The following figure provides an overview of the workloads and service classes that are created:



Overview of Workloads and Service Classes

- The `SAP_DIAGNOSTIC_WL` workload as well as the `SAP_DIAGNOSTIC_SSC` service class are created for SAP-internal use. Currently, no work is executed in this service class.
- The `SAP_SAPTOOLS_WL` workload as well as the `SAP_SAPTOOLS_SSC` service class are created to separate work that originates from SAP monitoring tools. You can monitor the `SAP_SAPTOOLS_SSC` service class to determine the amount of resources that are consumed by SAP monitoring tools such as the DBA Cockpit or the Performance Warehouse.

- The DB2_ATS_WL workload as well as the DB2_ATS_SSC service class are created to separate work that is executed by the DB2 task scheduler. The DCF uses the DB2 task scheduler to execute the data collectors. Since in typical SAP systems no other component uses this mechanism, the workload that is measured on this service class is caused by the DCF of the DBA Cockpit.
- The service superclass SAP_SC is only created for structural reasons. No work is executed in service superclasses.

In addition to the default workloads and service classes that are created in SAP environments, you can set up additional workloads and service classes as described in Workload Management: SAP WLM Setup Status [page 340].

15 Monitoring DB2 pureScale in the DBA Cockpit

You can use the DBA Cockpit to monitor DB2 pureScale members and CFs. The DBA Cockpit provides the following information about DB2 pureScale:

- An overview of the cluster topology and cluster alerts
- An overview of the DB2 member performance in the DB2 pureScale cluster
- Information about the group buffer pool and local buffer pool hit ratio
- The content of the (global) package cache
- Information about the memory configuration and consumption of the cluster caching facilities (CFs)
- Information about page contention in the DB2 pureScale cluster
- Information about connected clients and the client configuration (that is, the content of the`db2dsdriver.cfg` file)

You can find information about DB2 pureScale on the following screens in the *Performance* area:

Navigation Path	Available For	More Information
Database	SAP GUI and Web browser-based UI	Performance: Database [page 76]
Buffer Pools	SAP GUI and Web browser-based UI	Performance: Buffer Pools [page 80]
pureScale: Cluster Status	SAP GUI only	Performance - pureScale: Cluster Status [page 67]
pureScale: Page Reclaim	SAP GUI only	Performance - pureScale: Page Reclaim [page 68]
pureScale: Cluster Caching Facility	SAP GUI and Web browser-based UI	Performance: pureScale: Cluster Caching Facility [page 68]
SQL Cache	SAP GUI and Web browser-based UI	Performance: SQL Cache (DB2 V9.7 or Higher) [page 103]
Top SQL Statements	SAP GUI and Web browser-based UI	Performance: Top SQL Statement Analysis [page 99]

In addition, DB2 pureScale information is available on the *Client Configuration* screen in the *Configuration* area (SAP GUI only, see also Configuration: Client Configuration [page 220]).

16 References

Documentation by SAP

The following documentation is available on SAP Help Portal and SAP Support Portal:

> → Recommendation
>
> For central access to **all** our documentation, use our SAP on IBM Db2 overview page on SAP Help Portal.

Description	Links
Database Administration Guide for SAP on IBM Db2 for Linux, UNIX, and Windows	https://help.sap.com/viewer/db6_admin
Implementation documentation such as guides for installation, system copy, and SAP system upgrades	Installation guides for SAP Application Server systems based on SAP NetWeaver Software Update Manager (SUM) guides System copy guides
DBA Cockpit documentation and videos	Database Administration Using the DBA Cockpit: IBM Db2 for Linux, UNIX, and Windows Videos: DBA Cockpit Overview Network Statistics Monitoring in the DBA Cockpit
SAP Business Warehouse on IBM Db2 for Linux, UNIX, and Windows: Administration Tasks	Db2 10.5 and higher Db2 10.1 and lower (out of mainstream maintenance)
Database upgrade guides	Db2 11.5 Db2 11.1 Db2 10.5 Db2 10.1 (out of mainstream maintenance) Db2 9.7 (out of mainstream maintenance)

Description	Links
Running an SAP System on IBM Db2 with the Db2 pureScale Feature	Db2 11.5 Db2 11.1 Db2 10.5 Db2 10.1 (out of mainstream maintenance)
Enabling SAP Business Warehouse Systems to Use IBM Db2 for Linux, UNIX, and Windows as Near-Line Storage (NLS)	http://help.sap.com/viewer/db6_nls
IBM Db2 High Availability Solution: IBM Tivoli System Automation for Multiplatforms	https://help.sap.com/viewer/db6_samp
Central access to SAP Notes	Find SAP Notes on SAP Support Portal
Central access to all Db2 for LUW-related guides (including the above mentioned)	SAP on IBM Db2 overview page

SAP on Db2 for Linux, UNIX, and Windows Community

Check out and participate in our SAP community for IBM Db2. Here you'll find blogs, Q&As, whitepapers, videos, and guides. You can also post blogs and questions and search the community archive for more.

Documentation by IBM

For IBM product documentation on all Db2 versions, go to the IBM Db2 documentation .

17 Appendix

17.1 Database Details: Buffer Pool

The following information is displayed on the *Buffer Pool* tab page in the content detail area of the *Database* screen:

Field	Description
Buffer Pools	
Number	Number of buffer pools
Total Size	Total size in KB of all buffer pools
Buffer Quality	
Overall Buffer Quality	Percentage at which the data is read from the buffer pool, rather than directly from the hard disk This is calculated using the following formula: (logical reads - physical reads) / (logical reads) * 100
Data Hit Ratio	Percentage at which data (without index data) is read from the buffer pool, rather than directly from the hard disk
Index Hit Ratio	Frequency as a percentage at which index data is read from the buffer pool, rather than directly from the hard disk
Columnar Hit Ratio	Frequency as a percentage at which columnar data is read from the buffer pool, rather than directly from the hard disk
No Victim Buffers	Number of times an agent did not have a preselected victim buffer available
Average Time	
Physical Reads	Average time in milliseconds required to read data from disk into the buffer pool
Physical Writes	Average time in milliseconds required to write data from the buffer pool to disk
Data	
Logical Reads	Number of read accesses to data in the buffer pool

Field	Description
Physical Reads	Number of read accesses to data on disk (I/O) This value includes the number of physical reads that were performed synchronously (by the database manager agents) and asynchronously (by the I/O servers).
Physical Writes	Number of write accesses to data on disk (I/O) This value includes the number of physical writes that were performed synchronously (by the database manager agents) and asynchronously (by the I/O cleaners).
Synchronous Reads	Number of read accesses to data on disk (by agents)
Synchronous Writes	Number of write accesses to data on disk (by agents)
Temporary Logical Reads	Number of logical read requests that required I/O to get data pages into the temporary tablespace
Temporary Physical Reads	Number of physical read requests that required I/O to get data pages into the temporary tablespace
Temporary Physical Reads	Number of physical read requests that required I/O to get data pages into the temporary tablespace
Index	
Logical Reads	Number of read accesses to index data in the buffer pool
Physical Reads	Number of read accesses to index data on disk (I/O) Since index data is read only by the database manager agents, this value includes the number of synchronously read index pages.
Physical Writes	Number of write accesses to index data on disk (I/O) This value includes the number of physical writes that were performed synchronously (by the database manager agents) and asynchronously (by the I/O cleaners).
Synchronous Reads	Number of read accesses to index data on disk (by agents)
Synchronous Writes	Number of write accesses to index data on disk (by agents)
Temporary Logical Reads	Number of logical read requests that required I/O to get index pages into the temporary tablespace
Temporary Physical Reads	Number of physical read requests that required I/O to get index pages into the temporary tablespace

Field	Description
Columnar	
Logical Reads	Number of read accesses to columnar data in the buffer pool
Physical Reads	Number of read accesses to columnar data on disk (I/O) This value includes the number of physical reads that were performed synchronously (by the database manager agents) and asynchronously (by the I/O servers).
Physical Writes	Number of write accesses to columnar data on disk (I/O) This value includes the number of physical writes that were performed synchronously (by the database manager agents) and asynchronously (by the I/O cleaners).
Synchronous Reads	Number of read accesses to columnar data on disk (by agents)
Synchronous Writes	Number of write accesses to columnar data on disk (by agents)
Temporary Logical Reads	Number of logical read requests that required I/O to get columnar data pages into the temporary tablespace
Temporary Physical Reads	Number of physical read requests that required I/O to get columnar data pages into the temporary tablespace

> **i** Note
>
> Data is read or written in pages. A page can be 4 KB, 8 KB, 16 KB, or 32 KB in size.
>
> Unless otherwise specified, no distinction is made between synchronous and asynchronous accesses.

17.2 Group Buffer Pool (GBP) I/O

The following metrics are displayed in the DBA Cockpit if the Db2 database is installed as pureScale system with a set of members. The metrics are displayed on various screens in the snapshot area, for example, for databases, bufferpools, and applications.

Before you work with these metrics, make sure that you have understood the IBM concepts of group buffer pool (GBP) and local buffer pool (LBP), which are relevant in a Db2 pureScale environment. For more information, see Buffer pools in a Db2 pureScale environment in the IBM Db2 documentation.

Metric	Description
Buffer Quality	This group of counters summarizes the access to data, indexes, and pages containing XML data (XDA).
Local Buffer Pool Hit Ratio	Percentage at which pages are read from the local buffer pool rather than directly from the hard disk
Local Buffer Pool Hit Ratio (GBP-Independent Pages)	Local buffer pool hit ratio with respect to GBP-independent pages only. A GBP-independent page is a page that only one member accesses. It exists only in the local buffer pool (LBP) of a member and does not need coordination by the group buffer pool (GBP).
Local Buffer Pool Hit Ratio (GBP-Dependent Pages)	Local buffer pool hit ratio with respect to GBP-dependent pages only. A GBP-dependent page is a page that is accessed by different members in a Db2 pureScale environment, so copies of the page in different local buffer pools are coordinated by the group buffer pool (GBP).
Group Buffer Pool Hit Ratio	Percentage of pages that are read from the global buffer pool rather than directly from the hard disk
Percentage of Pages Found in GBP by Prefetcher	Percentage of pages that have been requested by the prefetcher, but that are already available in the GBP

The following information is related to specific pages. The information is available on a tab page with three sub tabs that group the relevant information by data pages, index pages, and pages containing XML data (XDA).

> **i** Note
>
> Self-explanatory metrics are not listed in this table.

Metric	Description
GBP-Independent Pages Found in LBP	The number of GBP-independent data pages found in a local buffer pool (LBP) by an agent
GBP Pages Invalid in LBP	The number of times that data pages were invalid in the local buffer pool and were read from the group buffer pool instead
GBP Logical Reads	The number of times GBP-dependent data pages were attempted to be read from the group buffer pool because the pages were either invalid or not found in a local buffer pool

Metric	Description
GBP Physical Reads	The number of times that GBP-dependent index pages were read into the local buffer pool from disk because they were not found in the GBP
Pages Found in LBP	The number of times that data pages were found in the local buffer pool (LBP)
GBP-Independent Pages Found in LBP	The number of GBP-independent data pages found in a local buffer pool by asynchronous EDUs
Pages Found in LBP (GBP-Dependent and Independent)	The number of times that data pages were found in the local buffer pool when a prefetcher attempted to access them
Pages Read from GBP (Invalid in LBP)	The number of times that a prefetcher attempted to read data pages from the group buffer pool because the pages were invalid in the local buffer pool
GBP-Dependent Pages Read from GBP	The number of times that a prefetcher attempted to read GBP-dependent data pages from the group buffer pool because the pages were either invalid or not found in the local buffer pool
GBP-Dependent Pages Read from Disk	The number of times that GBP-dependent data pages were read from disk and transferred into a local buffer pool by a prefetcher because they were not found in the GBP

More Information

For more technical details about the displayed metrics, see Monitor elements for viewing Db2 pureScale buffer pool activity in the IBM Db2 documentation.

For a detailed description of the formulas for buffer pool hit ratios, see Formulas for calculating buffer pool hit ratios in the IBM Db2 documentation.

17.3 Database Details: Cache

The following information is displayed on the *Cache* tab page in the content detail area of the *Database* screen:

Field	Description
Catalog Cache	

Field	Description
Size	Maximum allowed size in KB for the catalog cache The catalog cache is accessed each time a transaction accesses a table, view, or alias. The cache is allocated dynamically from the heap. The maximum allowed size is determined by database configuration parameter `CATALOGCACHE_SZ`.
Quality	Percentage at which the data is read from the catalog cache, rather than directly from the hard disk This is calculated using the following formula: (catalog cache lookups - catalog cache inserts) / catalog cache lookups * 100
Lookups	Number of times that the catalog cache was referenced to obtain table descriptor information
Inserts	Number of times that the system tried to insert table descriptor information into the catalog cache
Overflows	Number of times that an insert into the catalog cache failed due to the catalog cache being full
High-Water Mark	Largest size reached by package cache
Package Cache	
Size	Maximum allowed size in KB for the package cache The package cache contains access plans. The maximum allowed size is determined by database configuration parameter `PCKCACHESZ`.
Quality	Indicates percentage at which the data is read from the package cache, rather than directly from the hard disk This is calculated using the following formula: (package cache lookups - package cache inserts) / package cache lookups *100
Lookups	Number of times an application looked for a section in the package cache
Inserts	Total number of times that an access plan was not available for use and had to be loaded into the package cache
Overflows	Number of times that the package cache overflowed the bounds of its allocated memory

Field	Description
High-Water Mark	Largest size reached by the package cache

17.4 Database Details: Asynchronous I/O

The following information is displayed on the *Asynchronous I/O* tab page in the content detail area of the *Database* screen:

Field	Description
I/O Configuration	
Number of I/O Servers	Number of I/O servers that read data asynchronously from the hard disk into the buffer pool
Number of I/O Cleaners	Number of I/O cleaners that write data asynchronously from the buffer pool to the hard disk
Average Time	
Asynchronous Physical Reads	Average time in milliseconds required by the I/O servers to read a page from disk and write it to the buffer pool
Asynchronous Physical Writes	Average time in milliseconds required by the I/O cleaners to read a page from the buffer pool and write it to the hard disk
Data	
Asynchronous Physical Reads	Number of data pages that were read asynchronously from disk and written to the buffer pool by the I/O servers (pre-fetch)
Asynchronous Physical Writes	Number of data pages that were written asynchronously from buffer pool to disk (I/O cleaners)
Asynchronous Read Requests	Number of asynchronous data read requests
Index	
Asynchronous Physical Reads	Number of index pages that were read asynchronously from disk and written to the buffer pool by the I/O servers (pre-fetch)
Asynchronous Physical Writes	Number of index pages that were written asynchronously from buffer pool to disk (I/O cleaners)

Field	Description
Asynchronous Read Requests	Number of asynchronous index read requests

Non-Prefetch I/O

Field	Description
Non-Prefetch Requests	The number of requests for non-prefetch work successfully added to the prefetch queue
Failed Non-Prefetch Requests	The number of times an attempt to queue a non-prefetch request was made but failed. This element is for non-prefetch work done by prefetchers. A possible reason for the failed request is that the prefetch queue is full

Prefetcher I/O

Field	Description
Unread Prefetched Pages	Indicates the number of pages that the prefetcher read and that were never used
Prefetch Waits	The number of times an application spent waiting for an I/O server (prefetcher) to finish loading pages into the buffer pool
Prefetch Wait Time	The time an application spent waiting for an I/O server (prefetcher) to finish loading pages into the buffer pool. The value is given in milliseconds.
Data Pages Requested for Prefetch	The number of data pages successfully requested for prefetching
Data Prefetch Requests	The number of data prefetch requests successfully added to the prefetch queue
Index Pages Requested for Prefetch	The number of index pages successfully requested for prefetching
Index Prefetch Requests	The number of index prefetch requests that were successfully added to the prefetch queue
Columnar Pages Requested for Prefetch	The number of columnar data pages successfully requested for prefetching
Columnar Prefetch Requests	The number of columnar data prefetch requests successfully added to the prefetch queue

Field	Description
XDA Pages Requested for Prefetch	The number of XML storage object (XDA) data pages successfully requested for prefetching.
XDA Prefetch Requests	The number of XML storage object (XDA) data prefetch requests that were successfully added to the prefetch queue
Temp Data Pages Requested for Prefetch	The number of data pages for temporary tablespaces that were successfully requested for prefetching
Temp Data Prefetch Requests	The number of data prefetch requests for temporary tablespaces that were successfully added to the prefetch queue
Temp Index Pages Requested for Prefetch	The number of index pages for temporary tablespaces that were successfully requested for prefetching
Temp Index Prefetch Requests	The number of index prefetch requests for temporary tablespaces that were successfully added to the prefetch queue
Temp Columnar Pages Requested for Prefetch	The number of columnar pages for temporary tablespaces that were successfully requested for prefetching
Temp Columnar Prefetch Requests	The number of columnar prefetch requests for temporary tablespaces that were successfully added to the prefetch queue
Temp XDA Pages Requested for Prefetch	The number of XML storage object (XDA) data pages for temporary tablespaces that were successfully requested for prefetching
Temp XDA Prefetch Requests	The number of XML storage object (XDA) data prefetch requests for temporary tablespaces that were successfully added to the prefetch queue
Failed Data Prefetch Requests	The number of times that an attempt to queue a data prefetch request was made but failed. One possible reason is that the prefetch queue is full.
Failed Index Prefetch Requests	The number of times that an attempt to queue an index prefetch request was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed Columnar Prefetch Requests	The number of times that an attempt to queue a columnar prefetch request was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.

Field	Description
Failed XDA Prefetch Requests	The number of times that an attempt to queue an XML storage object (XDA) data prefetch request was made but failed. One possible reason is the prefetch queue is full and a request could not be obtained from the free list.
Failed Temp Data Prefetch Requests	The number of times that an attempt to queue a data prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed Temp Index Prefetch Requests	The number of times that an attempt to queue an index prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed Temp Columnar Prefetch Requests	The number of times that an attempt to queue a columnar prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed Temp XDA Prefetch Requests	The number of times that an attempt to queue an XML storage object (XDA) data prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.

Total No. of Skipped Prefetch Pages

Field	Description
Skipped Data Pages (Total)	The number of data pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool
Skipped Index Pages (Total)	The number of index pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool
Skipped Columnar Pages (Total)	The number of columnar pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool
Skipped XDA Pages (Total)	The number of XML storage object (XDA) data pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool

Field	Description
Skipped Temp Data Pages (Total)	The number of data pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool
Skipped Temp Index Pages (Total)	The number of index pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool
Skipped Temp Columnar Pages (Total)	The number of columnar pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool
Skipped Temp XDA Pages (Total)	The number of XML storage object (XDA) data pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool

Skipped Prefetch Pages Loaded by Agents in UOW

Field	Description
Skipped Data Pages (Loaded by Agent)	The number of data pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped Index Pages (Loades by Agent)	The number of index pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped Columnar Pages (Loaded by Agent)	The number of columnar pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped XDA Pages (Loaded by Agent)	The number of XML storage object (XDA) data pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped Temp Data Pages (Loaded by Agent)	The number of data pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped Temp Index Pages (Loaded by Agent)	The number of index pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work

Field	Description
Skipped Temp Columnar Pages (Loaded by Agent)	The number of columnar pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work by an agent in the same unit of work
Skipped Temp XDA Pages (Loaded by Agent)	The number of XML storage object (XDA) data pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work

Columnar

Field	Description
Asynchronous Physical Reads	Number of columnar data pages that were read asynchronously from disk and written to the buffer pool by the I/O servers (prefetch)
Asynchronous Physical Writes	Number of columnar data pages that were written asynchronously from buffer pool to disk (I/O cleaners)
Asynchronous Read Requests	Number of asynchronous columnar data read requests

> **i** Note
>
> Data is read or written in pages. A page can be 4 KB, 8 KB, 16 KB, or 32 KB in size.

17.5 Database Details: Direct I/O

The following information is displayed on the *Direct I/O* tab page in the content detail area of the *Database* screen:

Field	Description
Average Time	
Direct Reads	Average time in milliseconds required to read directly from disk
Direct Writes	Average time in milliseconds required to write directly to disk
I/O	

Field	Description
Direct Reads	Read accesses from disk that do not use the buffer pool (LONG VARCHAR fields, backup)
Direct Writes	Write accesses to disk that do not use the buffer pool (LONG VARCHAR fields, restore, load)
Average I/O per Request	
Direct Reads	Average number of requests to read directly from disk
Direct Writes	Average number of requests to write directly to disk

17.6 Database Details: Real-Time Statistics

The following information is displayed on the *Real-Time Statistics* tab page in the content detail area of the *Database* screen:

Field	Description
Size of Statistics Cache	Size of statistics cache in bytes
Number of Asynchronously Collected Statistics	Total number of successful asynchronous statistics collection activities
Number of Statistics Collections During Query Compilation	Total number of statistics collection activities for creating statistics by the system without table or index scan during query compilation
Time Spent During Query Compilation	Total time spent on creating statistics by system without table or index scan during query compilation in milliseconds
Number of Synchronously Collected Statistics	Total number of synchronous statistics collection activities during query compilation
Time Spent on Synchronous Statistics Collection Activities	Total time spent on synchronous statistics collection activities in milliseconds

17.7 Database Details: Locks and Deadlocks

The following information is displayed on the *Locks and Deadlocks* tab page in the content detail area of the *Database* screen:

Field	Description
Lock List	
Size	Maximum length of the list (database configuration parameter `LOCKLIST`) Database locks are managed in a list. The lock list is allocated dynamically.
In Use	Current size of the lock list
Lock Waits	
Total	Total number of times that applications or connections waited for locks
Time Waited	Total amount of elapsed time in milliseconds that applications waited for a lock to be granted
Average Time Waited	Average time in milliseconds waited for a lock
Escalations	
Lock Escalations	Number of times that locks have been escalated from several row locks to a table lock If the maximum allowed length of the lock list is reached, row locks are converted to table locks to save space in the lock list. This process is called lock escalations
Exclusive Lock Escalations	Number of times that locks have been escalated from several row locks to one exclusive table lock, or the number of times an exclusive lock on a row caused the table lock to become an exclusive lock Exclusive locks are important to track since they can impact the concurrency of your data because other applications cannot access data held by an exclusive lock.
Locks	
Locks Currently Held	Total number of locks currently held by the applications

Field	Description
Deadlocks Detected	Number of deadlocks that have occurred Deadlock situations are recognized and resolved automatically by the database. The database configuration parameter lock escalations determines when a lock wait situation is resolved.
Lock Timeouts	Number of times that a request to lock an object timed out instead of being granted The database configuration parameter lock escalations determines when a lock wait situation is resolved.

17.8 Database Details: Logging

The following information is displayed on the *Logging* tab page in the content detail area of the *Database* screen:

Field	Description
Log Files	
Primary	Number of primary log files The database configuration parameter `LOGPRIMARY` determines this value.
Secondary	Number of secondary log files The database configuration parameter `LOGSECOND` determines this value.
Size	Number of pages in each log file The database configuration parameter `LOGFILSIZ` determines this value. Each page has 4 KB.
Total Log	
Available to Database	Amount of primary log space in bytes in the database that is not being used by uncommitted transactions
Used by Database	Total amount of primary log space in bytes currently used in the database
Maximum Space Used	Maximum amount of primary log space used in bytes

Field	Description
Node with Least Available Space	Node with the least amount of available log space in bytes **i Note** This field is only displayed for global snapshots over all partitions.
Application with Oldest Transaction	Application ID (that corresponds to the `agent_id` value from the application snapshot) of the application that has the oldest transaction
Secondary Log	
Logs Currently Allocated	Total number of secondary log files that are currently being used for the database
Maximum Space Used	Maximum amount of secondary log space used in bytes
Log Pages	
Read	Number of log pages read from disk
Written	Number of log pages written to disk
Log Buffer Consumption	
LSN Gap	Percentage of log space held by dirty pages in relation to log space specified by parameter `SOFTMAX`
Restart Range	Percentage of log space held that will have to be redone for crash recovery pages in relation to log space specified by parameter `SOFTMAX`
Log Buffer Quality	
Log Buffer Hit Ratio	Ratio of log data read from the buffer in relation to log data read from disk
Log Buffer Overflows	Number of times that agents have to wait for log data to write to disk while copying log records into the log buffer This value is incremented per agent per incident. For example, if two agents attempt to copy log data while the buffer is full, then this value is incremented by two.
Log Buffer I/O	
Average Write Time/Page	Average time per page in microseconds required to write log data to disk

Field	Description
Average Write Time/IO	Average time per I/O in microseconds required to write log data to disk
Average Read Time/Page	Average time per page in microseconds required to read log data from disk
Average Read Time/IO	Average time per I/O in microseconds required to read log data from disk

17.9 Database Details: Calls

The following information is displayed on the *Calls* tab page in the content detail area of the *Database* screen:

Field	Description
Rows	
Read	Number of data records that were read
Selected	Number of data records that were selected
Deleted	Number of data records that were deleted
Inserted	Number of data records that were inserted
Updated	Number of data records that were updated
Rows Read / Rows Processed	Ratio of rows read from the base table compared to rows processed, which can be either rows returned to the application (`SELECT` statements) or rows written (`UPDATE`, `INSERT`, `DELETE` statements) A value of *1* indicates an optimal access to the requested data. High values indicate statements with inefficient access. **i Note** This metric is only available if your database is DB2 V9.5 for Linux, UNIX, and Windows Fix Pack 1 or higher.

Field	Description
BP Gets / Rows Processed	Average number of pages read from the buffer pool per rows processed, which can be either rows returned to the application (`SELECT` statements) or rows written (`UPDATE`, `INSERT`, `DELETE` statements) **i Note** This metric is only available if your database is DB2 V9.5 for Linux, UNIX, and Windows Fix Pack 1 or higher.
Statements Executed	
SELECT SQL	Number of `SELECT` statements that were executed
UPDATE/INSERT/DELETE	Number of `UPDATE`, `INSERT`, and `DELETE` statements that were executed
DDL	Number of Data Definition Language (DDL) statements that were executed, for example, `CREATE TABLE`, `CREATE VIEW`, `ALTER TABLE`, and `DROP INDEX`
Elapsed Time (sec)	Sum of the host execution times in seconds for all the statements that were executed
Elapsed Time (microsec)	Remaining part of the above elapsed time in microseconds
Statements Attempted	
COMMITs	Number of `COMMIT` statements that have been attempted
Rollbacks	Number of `ROLLBACK` statements that have been attempted Automatic rollbacks caused by error situations or deadlocks are not included.
Dynamic SQL	Number of dynamic SQL statements attempted
Static SQL	Number of static SQL statements attempted
Failed SQL	Number of attempted SQL statements that failed
Hash Joins	
Total	Total number of hash joins executed
Total Hash Loops	Total number of times that a single partition of a hash join was larger that the available sort heap space
Overflows	Number of times that hash join data exceeded the available sort heap space

Field	Description
Small Overflows	Number of times that hash join data exceeded the available sort heap space by less than 10%
Post Threshold	Total number of hash joins that were throttled back by the sort memory throttling algorithm A throttled hash join is a hash join that was granted less memory than requested by the sort memory manager. A hash join is throttled back when the memory allocation from the shared sort heap is close to the limit set by database configuration parameter `sheapthres_shr`. This throttling significantly reduces the number of overflows over the`sheapthres_shr` limit in a system that is not properly configured. The data reported in this element only reflects hash joins using memory allocated from the shared sort heap.

17.10 Database Details: Sorts

The following information is displayed on the *Sorts* tab page in the content detail area of the *Database* screen:

Field	Description
Sort Heap	
Total Size	Amount of memory in KB available for each sort as defined in the database configuration parameter SORTHEAP (in pages)
Allocated	Total number of allocated space of sort heap space for all sorts and at the time the snapshot was taken
Sort Time	
Total	Total time in milliseconds required for all sort processes
Average	Average sort time in milliseconds
Sorts	
Total Sorts	Total number of sorts that have been executed
Sort Overflows	If the storage area allocated for sorting is not large enough, a sort overflow occurs. The hard disk is then used temporarily.

Field	Description
Active Sorts	Number of sorts in the database that currently have a sort heap allocated
Post Threshold Sorts	Total number of sorts that were throttled back by the sort memory throttling algorithm. A throttled sort is a sort that was granted less memory than requested by the sort memory manager. A sort is throttled back when the memory allocation for sorts is close to the limit set by database configuration parameter `sheapthres_shr`. This throttling significantly reduces the number of overflows over `sheapthres_shr` limit in a system that is not properly configured. The data reported in this element only reflects sorts using memory allocated from the shared sort heap.

17.11 Database Details: XML Storage

The following information is displayed on the *XML Storage* tab page in the content detail area of the *Database* screen:

Field	Description
Pool Data	
Logical Reads	Number of data pages for XML storage objects (XDAs) that have been requested from the buffer pool (logical reads) for regular and large tablespaces
Physical Reads	Number of data pages for XML storage objects (XDAs) that have been read from the tablespace containers (physical reads) for regular and large tablespaces
Write Accesses	Number of times a buffer pool data page for an XML storage object (XDA) was physically written to disk
Temporary Data	
Logical Reads	Number of pages for XML storage objects (XDA) that have been requested from the buffer pool (logical reads) for temporary tablespaces.

Field	Description
Physical Reads	Number of pages for XML storage objects that have been (XDA) read from the tablespace containers (physical reads) for temporary tablespaces
Asynchronous I/O	
Physical Reads	Number of XML storage object (XDA) data pages that have been read in from the tablespace containers (physical reads) by asynchronous engine dispatchable units (EDUs) for all types of tablespaces.
Physical Writes	Number of times a buffer pool data page for an XML storage object (XDA) was physically written to disk by either an asynchronous page cleaner, or a prefetcher A prefetcher may have written dirty pages to disk to create space for the pages being prefetched.
Read Requests	Number of asynchronous read requests for XML storage object (XDA) data

17.12 Buffer Pool Details: Buffer Pool

The following information is displayed on the *Buffer Pool* tab page in the content detail area of the *Buffer Pool* screen:

Field	Description
Buffer Pool	
Name	Name of the buffer pool
Current Size	Size of the buffer pool in KB and in pages ⚠ Caution If *Automatic* is selected, the buffer pool is automatically tuned (if DB2's Self-Tuning Memory Management is activated).
New Size	Size of the buffer pool in pages after a database restart
Pages Left to Remove	Number of pages that are still to be removed

Field	Description
Tablespace Use Count	Number of tablespaces that belong to this buffer pool
Buffer Quality	
Overall Buffer Quality	Percentage at which the data is read from the buffer pool, rather than directly from the hard disk This is calculated using the following formula: (logical reads - physical reads) / (logical reads) * 100
Data Hit Ratio	Percentage at which only data (without index data) is read from the buffer pool, rather than directly from the hard disk
Index Hit Ratio	Percentage at which index data is read from the buffer pool, rather than directly from the hard disk
Columnar Hit Ratio	Percentage at which columnar data is read from the buffer pool, rather than directly from the hard disk
Average Time	
Physical Reads	Average time in milliseconds required to read data from disk into the buffer pool
Physical Writes	Average time in milliseconds required to write data from the buffer pool to disk
Data	
Logical Reads	Number of read accesses to data in the buffer pool
Physical Reads	Number of read accesses to data on disk (I/O) This value includes the number of physical reads that were performed synchronously (by the database manager agents) and asynchronously (by the I/O servers).
Physical Writes	Number of write accesses to data on disk (I/O) This value includes the number of physical writes that were performed synchronously (by the database manager agents) and asynchronously (by the I/O cleaners).
Synchronous Reads	Number of read accesses to data on disk (by agents)
Synchronous Writes	Number of write accesses to data on disk (by agents)
Temporary Logical Reads	Number of logical read requests that required I/O to get data pages into the temporary tablespace

Field	Description
Temporary Physical Reads	Number of physical read requests that required I/O to get data pages into the temporary tablespace
Index	
Logical Reads	Number of read accesses to index data in the buffer pool
Physical Reads	Number of read accesses to index data on disk (I/O) Since index data is read only by the database manager agents, this value includes the number of synchronously read index pages.
Physical Writes	Number of write accesses to index data on disk (I/O) This value includes the number of physical writes that were performed synchronously (by the database manager agents) and asynchronously (by the I/O cleaners).
Synchronous Reads	Number of read accesses to index data on disk (by agents)
Synchronous Writes	Number of write accesses to index data on disk (by agents)
Temporary Logical Reads	Number of logical read requests that required I/O to get index pages into the temporary tablespace
Temporary Physical Reads	Number of physical read requests that required I/O to get index pages into the temporary tablespace
Columnar	
Logical Reads	Number of read accesses to columnar data in the buffer pool
Physical Reads	Number of read accesses to columnar data on disk (I/O) This value includes the number of physical reads that were performed synchronously (by the database manager agents) and asynchronously (by the I/O servers).
Physical Writes	Number of write accesses to columnar data on disk (I/O) This value includes the number of physical writes that were performed synchronously (by the database manager agents) and asynchronously (by the I/O cleaners).
Synchronous Reads	Number of read accesses to columnar data on disk (by agents)
Synchronous Writes	Number of write accesses to columnar data on disk (by agents)

Field	Description
Temporary Logical Reads	Number of logical read requests that required I/O to get columnar data pages into the temporary tablespace
Temporary Physical Reads	Number of physical read requests that required I/O to get columnar data pages into the temporary tablespace

17.13 Buffer Pool Details: Asynchronous I/O

The following information is displayed on the *Asynchronous I/O* tab page in the content detail area of the *Buffer Pool* screen:

Field	Description
Average Time	
Asynchronous Physical Reads	Average time in milliseconds required by the I/O servers to read a page from disk and write it to the buffer pool
Asynchronous Physical Writes	Average time in milliseconds required by the I/O cleaners to read a page from the buffer pool and write it to the hard disk
Data	
Asynchronous Physical Reads	Number of data pages that were read asynchronously from disk and written to the buffer pool by the I/O servers (pre-fetch)
Asynchronous Physical Writes	Number of data pages that were written asynchronously from buffer pool to disk (I/O cleaners)
Asynchronous Read Requests	Number of asynchronous data read requests
Index	
Asynchronous Physical Reads	Number of index pages that were read asynchronously from disk and written to the buffer pool by the I/O servers (pre-fetch)
Asynchronous Physical Writes	Number of index pages that were written asynchronously from buffer pool to disk (I/O cleaners)
Asynchronous Read Requests	Number of asynchronous index read requests

The following metrics apply to DB2 Version 10 and higher only:

Non-Prefetch I/O

Field	Description
Non-Prefetch Requests	The number of requests for non-prefetch work successfully added to the prefetch queue
Failed Non-Prefetch Requests	The number of times an attempt to queue a non-prefetch request was made but failed. This element is for non-prefetch work done by prefetchers. A possible reason for the failed request is that the prefetch queue is full

Prefetcher I/O

Field	Description
Unread Prefetched Pages	Indicates the number of pages that the prefetcher read and that were never used
Prefetch Waits	The number of times an application spent waiting for an I/O server (prefetcher) to finish loading pages into the buffer pool
Prefetch Wait Time	The time an application spent waiting for an I/O server (prefetcher) to finish loading pages into the buffer pool. The value is given in milliseconds.
Data Pages Requested for Prefetch	The number of data pages successfully requested for prefetching
Data Prefetch Requests	The number of data prefetch requests successfully added to the prefetch queue
Index Pages Requested for Prefetch	The number of index pages successfully requested for prefetching
Index Prefetch Requests	The number of index prefetch requests that were successfully added to the prefetch queue
Columnar Pages Requested for Prefetch	The number of columnar data pages successfully requested for prefetching
Columnar Prefetch Requests	The number of columnar data prefetch requests successfully added to the prefetch queue
XDA Pages Requested for Prefetch	The number of XML storage object (XDA) data pages successfully requested for prefetching.

Field	Description
XDA Prefetch Requests	The number of XML storage object (XDA) data prefetch requests that were successfully added to the prefetch queue
Temp Data Pages Requested for Prefetch	The number of data pages for temporary tablespaces that were successfully requested for prefetching
Temp Data Prefetch Requests	The number of data prefetch requests for temporary tablespaces that were successfully added to the prefetch queue
Temp Index Pages Requested for Prefetch	The number of index pages for temporary tablespaces that were successfully requested for prefetching
Temp Index Prefetch Requests	The number of index prefetch requests for temporary tablespaces that were successfully added to the prefetch queue
Temp Columnar Pages Requested for Prefetch	The number of columnar pages for temporary tablespaces that were successfully requested for prefetching
Temp Columnar Prefetch Requests	The number of columnar prefetch requests for temporary tablespaces that were successfully added to the prefetch queue
Temp XDA Pages Requested for Prefetch	The number of XML storage object (XDA) data pages for temporary tablespaces that were successfully requested for prefetching
Temp XDA Prefetch Requests	The number of XML storage object (XDA) data prefetch requests for temporary tablespaces that were successfully added to the prefetch queue
Failed Data Prefetch Requests	The number of times that an attempt to queue a data prefetch request was made but failed. One possible reason is that the prefetch queue is full.
Failed Index Prefetch Requests	The number of times that an attempt to queue an index prefetch request was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed Columnar Prefetch Requests	The number of times that an attempt to queue a columnar prefetch request was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed XDA Prefetch Requests	The number of times that an attempt to queue an XML storage object (XDA) data prefetch request was made but failed. One possible reason is the prefetch queue is full and a request could not be obtained from the free list.

Field	Description
Failed Temp Data Prefetch Requests	The number of times that an attempt to queue a data prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed Temp Index Prefetch Requests	The number of times that an attempt to queue an index prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed Temp Columnar Prefetch Requests	The number of times that an attempt to queue a columnar prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed Temp XDA Prefetch Requests	The number of times that an attempt to queue an XML storage object (XDA) data prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.

Skipped Prefetch Pages Loaded by Agents in UOW

Field	Description
Skipped Data Pages (Loaded by Agent)	The number of data pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped Index Pages (Loades by Agent)	The number of index pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped Columnar Pages (Loaded by Agent)	The number of columnar pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped XDA Pages (Loaded by Agent)	The number of XML storage object (XDA) data pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped Temp Data Pages (Loaded by Agent)	The number of data pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work

Field	Description
Skipped Temp Index Pages (Loaded by Agent)	The number of index pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped Temp Columnar Pages (Loaded by Agent)	The number of columnar pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work by an agent in the same unit of work
Skipped Temp XDA Pages (Loaded by Agent)	The number of XML storage object (XDA) data pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work

> **i** Note
>
> Data is read or written in pages. A page can be 4 KB, 8 KB, 16 KB, or 32 KB in size.

17.14 Buffer Pool Details: Direct I/O

To display detailed information about direct I/O for your selected buffer pools, choose the *Direct I/O* tab page in the content detail area of the *Buffer Pool* screen.

The following information is displayed on the *Direct I/O* tab page in the content detail area of the *Buffer Pool* screen:

Field	Description
Average Time	
Direct Reads	Average time in milliseconds required to read directly from disk
Direct Writes	Average time in milliseconds required to write directly to disk
I/O	
Direct Reads	Read accesses from disk that do not use the buffer pool (LONG VARCHAR fields, backup)
Direct Writes	Write accesses to disk that do not use the buffer pool (LONG VARCHAR fields, restore, load)

Field	Description
Average I/O per Request	
Direct Reads	Average number of requests to read directly from disk
Direct Writes	Average number of requests to write directly to disk

17.15 Buffer Pool Details: XML Storage

The following information is displayed on the *XML Storage* tab page in the content detail area of the *Buffer Pool* screen:

Field	Description
Pool Data	
Logical Reads	Number of data pages for XML storage objects (XDAs) that have been requested from the buffer pool (logical reads) for regular and large tablespaces
Physical Reads	Number of data pages for XML storage objects (XDAs) that have been read from the tablespace containers (physical reads) for regular and large tablespaces
Write Accesses	Number of times a buffer pool data page for an XML storage object (XDA) was physically written to disk
Temporary Data	
Logical Reads	Number of pages for XML storage objects (XDA) that have been requested from the buffer pool (logical reads) for temporary tablespaces.
Physical Reads	Number of pages for XML storage objects that have been (XDA) read from the tablespace containers (physical reads) for temporary tablespaces
Asynchronous I/O	
Physical Reads	Number of XML storage object (XDA) data pages that have been read in from the tablespace containers (physical reads) by asynchronous engine dispatchable units (EDUs) for all types of tablespaces.

Field	Description
Physical Writes	Number of times a buffer pool data page for an XML storage object (XDA) was physically written to disk by either an asynchronous page cleaner, or a prefetcher A prefetcher may have written dirty pages to disk to create space for the pages being prefetched.
Read Requests	Number of asynchronous read requests for XML storage object (XDA) data

17.16 Tablespace Details: Buffer Pool

The following information is displayed on the *Buffer Pool* tab page in the content detail area of the *Tablespaces* screen:

Field	Description
Tablespace	
Name	Name of the tablespace
Buffer Quality	
Buffer Pool	Name of the buffer pool associated with the selected tablespace
Overall Buffer Quality	Percentage at which the data is read from the buffer pool, rather than directly from the hard disk This is calculated using the following formula: (logical reads - physical reads) / (logical reads) * 100
Data Hit Ratio	Indicates percentage at which data (without index data) is read from the buffer pool, rather than directly from the hard disk
Index Hit Ratio	Indicates percentage at which index data is read from the buffer pool, rather than directly from the hard disk
Columnar Hit Ratio	Indicates percentage at which columnar data is read from the buffer pool, rather than directly from the hard disk
No Victim Buffer	Number of times an agent did not have a preselected victim buffer available

Field	Description
Average Time	
Physical Reads	Average time in milliseconds required to read data from disk into the buffer pool
Physical Writes	Average time in milliseconds required to write data from the buffer pool to disk
Data	
Logical Reads	Number of read accesses to data in the buffer pool
Physical Reads	Number of read accesses to data on disk (I/O) The value includes the number of physical reads that were performed synchronously (by the database manager agents) and asynchronously (by the I/O servers).
Physical Writes	Number of write accesses to data on disk (I/O) The value includes the number of physical writes that were performed synchronously (by the database manager agents) and asynchronously (by the I/O cleaners).
Synchronous Reads	Number of read accesses to data on disk (by agents)
Synchronous Writes	Number of write accesses to data on disk (by agents)
Temporary Logical Reads	Number of logical read requests that required I/O to get data pages into the temporary tablespace
Temporary Physical Reads	Number of physical read requests that required I/O to get data pages into the temporary tablespace
Index	
Logical Reads	Number of read accesses to index data in the buffer pool
Physical Reads	Number of read accesses to index data on disk (I/O) Since index data is read only by the database manager agents, this value includes the number of synchronously read index pages.
Physical Writes	Number of write accesses to index data on disk (I/O) The value includes the number of physical writes that were performed synchronously (by the database manager agents) and asynchronously (by the I/O cleaners).
Synchronous Reads	Number of read accesses to index data on disk (by agents)

Field	Description
Synchronous Writes	Number of write accesses to index data on disk (by agents)
Temporary Logical Reads	Number of logical read requests that required I/O to get index pages into the temporary tablespace
Temporary Physical Reads	Number of physical read requests that required I/O to get index pages into the temporary tablespace
Columnar	
Logical Reads	Number of read accesses to columnar data in the buffer pool
Physical Reads	Number of read accesses to columnar data on disk (I/O) The value includes the number of physical reads that were performed synchronously (by the database manager agents) and asynchronously (by the I/O servers).
Physical Writes	Number of write accesses to columnar data on disk (I/O) The value includes the number of physical writes that were performed synchronously (by the database manager agents) and asynchronously (by the I/O cleaners).
Synchronous Reads	Number of read accesses to columnar data on disk (by agents)
Synchronous Writes	Number of write accesses to columnar data on disk (by agents)
Temporary Logical Reads	Number of logical read requests that required I/O to get columnar data pages into the temporary tablespace
Temporary Physical Reads	Number of physical read requests that required I/O to get columnar data pages into the temporary tablespace

17.17 Tablespace Details: Asynchronous I/O

The following information is displayed on the *Asynchronous I/O* tab page in the content detail area of the *Tablespaces* screen:

Field	Description
Average Time	

Field	Description
Asynchronous Physical Reads	Average time in milliseconds required by the I/O servers to read a page from disk and write it to the buffer pool
Asynchronous Physical Writes	Average time in milliseconds required by the I/O cleaners to read a page from the buffer pool and write it to the hard disk
Data	
Asynchronous Physical Reads	Number of data pages that were read asynchronously from disk and written to the buffer pool by the I/O servers (pre-fetch)
Asynchronous Physical Writes	Number of data pages that were written asynchronously from buffer pool to disk (I/O cleaners)
Asynchronous Read Requests	Number of asynchronous data read requests
Index	
Asynchronous Physical Reads	Number of index pages that were read asynchronously from disk and written to the buffer pool by the I/O servers (pre-fetch)
Asynchronous Physical Writes	Number of index pages that were written asynchronously buffer pool to disk (I/O cleaners)
Asynchronous Read Requests	Number of asynchronous index read requests

The following metrics apply to DB2 Version 10 and higher only:

Non-Prefetch I/O

Field	Description
Non-Prefetch Requests	The number of requests for non-prefetch work successfully added to the prefetch queue
Failed Non-Prefetch Requests	The number of times an attempt to queue a non-prefetch request was made but failed. This element is for non-prefetch work done by prefetchers. A possible reason for the failed request is that the prefetch queue is full

Prefetcher I/O

Field	Description
Unread Prefetched Pages	Indicates the number of pages that the prefetcher read and that were never used

Field	Description
Prefetch Waits	The number of times an application spent waiting for an I/O server (prefetcher) to finish loading pages into the buffer pool
Prefetch Wait Time	The time an application spent waiting for an I/O server (prefetcher) to finish loading pages into the buffer pool. The value is given in milliseconds.
Data Pages Requested for Prefetch	The number of data pages successfully requested for prefetching
Data Prefetch Requests	The number of data prefetch requests successfully added to the prefetch queue
Index Pages Requested for Prefetch	The number of index pages successfully requested for prefetching
Index Prefetch Requests	The number of index prefetch requests that were successfully added to the prefetch queue
Columnar Pages Requested for Prefetch	The number of columnar data pages successfully requested for prefetching
Columnar Prefetch Requests	The number of columnar data prefetch requests successfully added to the prefetch queue
XDA Pages Requested for Prefetch	The number of XML storage object (XDA) data pages successfully requested for prefetching.
XDA Prefetch Requests	The number of XML storage object (XDA) data prefetch requests that were successfully added to the prefetch queue
Temp Data Pages Requested for Prefetch	The number of data pages for temporary tablespaces that were successfully requested for prefetching
Temp Data Prefetch Requests	The number of data prefetch requests for temporary tablespaces that were successfully added to the prefetch queue
Temp Index Pages Requested for Prefetch	The number of index pages for temporary tablespaces that were successfully requested for prefetching
Temp Index Prefetch Requests	The number of index prefetch requests for temporary tablespaces that were successfully added to the prefetch queue
Temp Columnar Pages Requested for Prefetch	The number of columnar pages for temporary tablespaces that were successfully requested for prefetching

Field	Description
Temp Columnar Prefetch Requests	The number of columnar prefetch requests for temporary tablespaces that were successfully added to the prefetch queue
Temp XDA Pages Requested for Prefetch	The number of XML storage object (XDA) data pages for temporary tablespaces that were successfully requested for prefetching
Temp XDA Prefetch Requests	The number of XML storage object (XDA) data prefetch requests for temporary tablespaces that were successfully added to the prefetch queue
Failed Data Prefetch Requests	The number of times that an attempt to queue a data pre-fetch request was made but failed. One possible reason is that the prefetch queue is full.
Failed Index Prefetch Requests	The number of times that an attempt to queue an index prefetch request was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed Columnar Prefetch Requests	The number of times that an attempt to queue a columnar prefetch request was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed XDA Prefetch Requests	The number of times that an attempt to queue an XML storage object (XDA) data prefetch request was made but failed. One possible reason is the prefetch queue is full and a request could not be obtained from the free list.
Failed Temp Data Prefetch Requests	The number of times that an attempt to queue a data prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed Temp Index Prefetch Requests	The number of times that an attempt to queue an index prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed Temp Columnar Prefetch Requests	The number of times that an attempt to queue a columnar prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.

Field	Description
Failed Temp XDA Prefetch Requests	The number of times that an attempt to queue an XML storage object (XDA) data prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.

Skipped Prefetch Pages Loaded by Agents in UOW

Field	Description
Skipped Data Pages (Loaded by Agent)	The number of data pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped Index Pages (Loades by Agent)	The number of index pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped Columnar Pages (Loaded by Agent)	The number of columnar pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped XDA Pages (Loaded by Agent)	The number of XML storage object (XDA) data pages that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped Temp Data Pages (Loaded by Agent)	The number of data pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped Temp Index Pages (Loaded by Agent)	The number of index pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work
Skipped Temp Columnar Pages (Loaded by Agent)	The number of columnar pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work by an agent in the same unit of work
Skipped Temp XDA Pages (Loaded by Agent)	The number of XML storage object (XDA) data pages for temporary tablespaces that an I/O server (prefetcher) skipped because the pages were already loaded into the buffer pool by an agent in the same unit of work

17.18 Tablespace Details: Direct I/O

The following information is displayed on the *Direct I/O* tab page in the content detail area of the *Tablespaces* screen:

Field	Description
Average Time	
Direct Reads	Average time in milliseconds required to read directly from disk
Direct Writes	Average time in milliseconds required to write directly to disk
I/O	
Direct Reads	Read accesses from disk that do not use the buffer pool (LONG VARCHAR fields, backup)
Direct Writes	Write accesses to disk that do not use the buffer pool (LONG VARCHAR fields, restore, load)
Average I/O per Request	
Direct Reads	Average number of requests to read directly from disk
Direct Writes	Average number of requests to write directly to disk

17.19 Tablespace Details: XML Storage

The following information is displayed on the *XML Storage* tab page in the content detail area of the *Tablespace* screen:

Field	Description
Pool Data	
Logical Reads	Number of data pages for XML storage objects (XDAs) that have been requested from the buffer pool (logical reads) for regular and large tablespaces
Physical Reads	Number of data pages for XML storage objects (XDAs) that have been read from the tablespace containers (physical reads) for regular and large tablespaces

Field	Description
Write Accesses	Number of times a buffer pool data page for an XML storage object (XDA) was physically written to disk
Temporary Data	
Logical Reads	Number of pages for XML storage objects (XDA) that have been requested from the buffer pool (logical reads) for temporary tablespaces.
Physical Reads	Number of pages for XML storage objects that have been (XDA) read from the tablespace containers (physical reads) for temporary tablespaces
Asynchronous I/O	
Physical Reads	Number of XML storage object (XDA) data pages that have been read in from the tablespace containers (physical reads) by asynchronous engine dispatchable units (EDUs) for all types of tablespaces.
Physical Writes	Number of times a buffer pool data page for an XML storage object (XDA) was physically written to disk by either an asynchronous page cleaner, or a prefetcher A prefetcher may have written dirty pages to disk to create space for the pages being prefetched.
Read Requests	Number of asynchronous read requests for XML storage object (XDA) data

17.20 Application Details: Application

The following information is displayed on the *Application* tab page in the content detail area of the *Applications* screen:

Field	Description
Application	
PID	Process ID of the database process belonging to an SAP work process
Handle	In multi partition database systems, this ID is the same on every member where the application might span subagents.

Field	Description
Connect Start	Start time when the application connected to the database
Platform	Operating system on which the client application is running
Host	Host name of the application server where the application server is running
Name	Name of application running at the client as known to the database manager or DB2 connect

Field	Description
Status	Current status of the application. Possible values are: • `Database Connect Pending` The application has initiated a database connection but the request has not yet completed. • `Database Connect Completed` The application has initiated a database connection and the request has completed. • `Unit of Work Executing` The database manager is executing requests on behalf of the unit of work. • `Unit of Work Waiting` The database manager is waiting on behalf of the unit of work in the application. This status typically means that the system is executing in the application's code. • `Lock Wait` The unit of work is waiting for a lock. After the lock is granted, the status is restored to its previous value. • `Commit Active` The unit of work is committing its database changes. • `Rollback Active` The unit of work is rolling back its database changes. • `Recompiling` The database manager is compiling an SQL statement or precompiling a plan on behalf of the application. • `Request Interrupted` An interrupt of a request is in progress. • `Database Disconnect Pending` The application has initiated a database disconnect but the command has not yet completed executing. The application might not have explicitly executed the database disconnect command. The database manager disconnects from a database if the application ends without disconnecting. • `Transaction prepared` The unit of work is part of a global transaction that has entered the prepared phase of the two-phase commit protocol. • `Transaction Heuristically` The unit of work is part of a global transaction that has been heuristically committed. • `Transaction Heuristically Rolled Back` The unit of work is part of a global transaction that has been heuristically rolled-back. • `Transaction Ended`

Field	Description
	The unit of work is part of a global transaction that has ended but has not yet entered the prepared phase of the two-phase commit protocol. • `Creating Database` The agent has initiated a request to create a database and that request has not yet completed. • `Restarting Database` The application is restarting a database to perform crash recovery. • `Restoring Database` The application is restoring a backup image to the database. • `Backing Up Database` The application is performing a "fast load" of data into the database. • `Data Fast Load` The application is performing a "fast load" of data into the database. • `Data Fast Unload` The application is performing a "fast unload" of data from the database. • `Wait to Disable Tablespace` The application has detected an I/O error and is attempting to disable a particular tablespace. The application has to wait for all other active transactions on the tablespace to complete before it can disable the tablespace. • `Quiescing a Tablespace` The application is performing a QUIESCE TABLESPACE request. • `Wait for Remote Partition` The application is waiting for a response from a remote partition in a partitioned database instance.
Agent	
PID	Process ID of an SAP work process that made the connection to the database
Client Information	
User ID	Client user ID that is generated by the transaction manager and provided to the server, if the `sqleseti` API is used For ABAP systems: name of the SAP user
Workstation	Identifies the client's system or workstation (for example, CICS EITERMID) if the `sqleseti` API was used in this connection.

Field	Description
Application	Identifies the server transaction program performing the transaction if the `sqleseti` API was used in this connection. For ABAP systems: name of the SAP transaction
Accounting	The data passed to the target database for logging and diagnostic purposes, if the `sqleseti` API was used in this connection For ABAP systems: name of the program executing the SQL statement

17.21 Application Details: Agents

> **i** Note
>
> This tab page is **only** available if you are using the SAP GUI-based user interface of the DBA Cockpit.

The following information is displayed on the *Agents* tab page in the content detail area of the *Applications* screen:

Field	Description
Agents	
Coordinator Agent PID	Process ID (UNIX systems) or thread ID (OS/2 or Windows systems) of the coordinator agent for the application
Currently Associated	Number of agents currently associated with the application
Associated with This Appl.	Number of agents participating in this application (high-water mark)
Stolen from Application	Number of agents removed from this application and subsequently used by another application This only happens if the agent was not busy.
Times Used by Agent(s)	
User CPU Time	Total user CPU time in seconds consumed by agents
System CPU Time	Total system CPU time in seconds consumed by agents
Idle Time	Total idle time in seconds
Waited for Prefetch	Total time in milliseconds waited for prefetch

Field	Description
Memory Pools Allocated to Agent	Displays information about memory pools allocated to this application The table contains the following columns: • *Partition* Partition number • *PID* Process ID (UNIX) or thread ID (Windows) of the agent • *Pool ID* Type of memory pool • *Current Size (KB)* Current size of the memory pool • *High-Water Mark (KB)* Largest size of the memory pool since its creation • *Configured Size (KB)* Configured size of the memory pool

17.22 Application Details: Assigned Agents

> **i** Note
>
> This tab page is **only** available if you are using the Web browser-based user interface of the DBA Cockpit.

The following information is displayed on the *Assigned Agents* tab page in the content detail area of the *Applications* screen:

Field	Description
Agents	
Coordinator Agent PID	Process ID (UNIX systems) or thread ID (Windows) of the coordinator agent for the application
Currently Associated	Number of agents currently associated with the application
Associated with This Appl.	Number of agents participating in this application (high-water mark)
Stolen from Application	Number of agents removed from this application and subsequently used by another application This only happens if the agent was not busy.
Times Spent by Agent(s)	

Field	Description
User CPU Time	Total user CPU time in seconds consumed by agents
System CPU Time	Total system CPU time in seconds consumed by agents
Idle Time	Total idle time in seconds
Waited for Prefetch	Total time in milliseconds waited for prefetch
Assigned Agents	This table displays information about all agents that are the co-ordinator agent itself or agents working as a subagent for the coordinator agent: • *Partition* Partition number • *Agent Type* Type of the agent • *Nesting Level* Nesting level of the agent • *Entity* Entity of the agent • *State* Indicates whether an agent is associated or active • *TID* Thread ID of the agent • *Service Class* Service class the agent is assigned to • *Event Type* Type of event that was last processed by the agent • *Event Object* Object of the event that was last processed by the agent • *Event State* State of the event that was last processed by the agent

17.23 Application Details: Agents Memory

> **i** Note
>
> This tab page is **only** available if you are using the Web browser-based user interface of the DBA Cockpit.

The following information is displayed on the *Agents Memory* tab page in the content detail area of the *Applications* screen:

Column	Description
Partition	Partition number
PID	Process ID (UNIX) or thread ID (Windows) of the agent
Pool ID	Type of memory pool
Current Size (KB)	Current size of the memory pool
High-Water Mark (KB)	Largest size of the memory pool since its creation
Configured Size (KB)	Configured size of the memory pool

17.24 Application Details: Buffer Pool

The following information is displayed on the *Buffer Pool* tab page in the content detail area of the *Applications* screen:

Field	Description
Buffer Quality	
Overall Buffer Quality	Indicates percentage at which the data is read from the buffer pool, rather than directly from the hard disk This is calculated using the following formula: (logical reads - physical reads) / (logical reads) * 100
Data Hit Ratio	Indicates percentage at which data (without index data) is read from the buffer pool, rather than directly from the hard disk
Index Hit Ratio	Indicates percentage at which index data is read from the buffer pool, rather than directly from the hard disk
Average Time	
Physical Reads	Average time in milliseconds required to read data from disk into the buffer pool
Physical Writes	Average time in milliseconds required to write data from the buffer pool to disk
Data	

Field	Description
Logical Reads	Number of read accesses to data in the buffer pool
Physical Reads	Number of read accesses to data on disk (I/O) The value includes the number of physical reads that were performed synchronously (by the database manager agents) and asynchronously (by the I/O servers).
Physical Writes	Number of write accesses to data on disk (I/O) It includes the number of physical writes that were performed synchronously (by the database manager agents) and asynchronously (by the I/O cleaners).
Temporary Logical Reads	Number of logical read requests that required I/O to get data pages into the temporary tablespace
Temporary Physical Reads	Number of physical read requests that required I/O to get data pages into the temporary tablespace
Index	
Logical Reads	Number of read accesses to index data in the buffer pool
Physical Reads	Number of read accesses to index data on disk (I/O) Since index data is read only by the database manager agents, this value contains the number of synchronously read index pages.
Physical Writes	Number of write accesses to index data on disk (I/O) The value includes the number of physical writes that were performed synchronously (by the database manager agents) and asynchronously (by the I/O cleaners).
Temporary Logical Reads	Number of logical read requests that required I/O to get index pages into the temporary tablespace
Temporary Physical Reads	Number of physical read requests that required I/O to get index pages into the temporary tablespace

17.25 Application Details: Direct I/O

The following information is displayed on the *Direct I/O* tab page in the content detail area of the *Applications* screen:

Field	Description
Average Time	
Direct Reads	Average time in milliseconds required to read directly from disk
Direct Writes	Average time in milliseconds required to write directly to disk
I/O	
Direct Reads	Read accesses from disk that do not use the buffer pool (LONG VARCHAR fields, backup)
Direct Writes	Write accesses to disk that do not use the buffer pool (LONG VARCHAR fields, restore, load)
Average I/O per Request	
Direct Reads	Average number of requests to read directly from disk
Direct Writes	Average number of requests to write directly to disk

17.26 Application Details: XML Storage

The following information is displayed on the *XML Storage* tab page in the content detail area of the *Applications* screen:

Field	Description
Pool Data	
Logical Reads	Number of data pages for XML storage objects (XDAs) that have been requested from the buffer pool (logical reads) for regular and large tablespaces
Physical Reads	Number of data pages for XML storage objects (XDAs) that have been read from the tablespace containers (physical reads) for regular and large tablespaces

Field	Description
Write Accesses	Number of times a buffer pool data page for an XML storage object (XDA) was physically written to disk
Temporary Data	
Logical Reads	Number of pages for XML storage objects (XDA) that have been requested from the buffer pool (logical reads) for temporary tablespaces.
Physical Reads	Number of pages for XML storage objects that have been (XDA) read from the tablespace containers (physical reads) for temporary tablespaces
Asynchronous I/O	
Physical Reads	Number of XML storage object (XDA) data pages that have been read in from the tablespace containers (physical reads) by asynchronous engine dispatchable units (EDUs) for all types of tablespaces.
Physical Writes	Number of times a buffer pool data page for an XML storage object (XDA) was physically written to disk by either an asynchronous page cleaner, or a prefetcher A prefetcher may have written dirty pages to disk to create space for the pages being prefetched.
Read Requests	Number of asynchronous read requests for XML storage object (XDA) data

17.27 Application Details: Locks and Deadlocks

The following information is displayed on the *Locks and Deadlocks* tab page in the content detail area of the *Applications* screen:

Fields	Description
Lock Waits	
Total	Total number of times that this application requested a lock, but had to wait because another application was already holding a lock on the data

Fields	Description
Time Waited	Total amount of elapsed time in milliseconds that this application has waited for a lock to be granted
Average Time Waited	Average time in milliseconds waited for a lock
Escalations	
Lock Escalations	Number of times that locks have been escalated from several row locks to a table lock If the maximum allowed length of the lock list is reached, row locks are converted to table locks to save space in the lock list. This process is called "lock escalation".
Exclusive Lock Escalations	Number of times that locks have been escalated from several row locks to one exclusive table lock, or the number of times an exclusive lock on a row caused the table lock to become an exclusive lock Exclusive locks are important to track since they can impact the concurrency of your data because other applications cannot access data held by an exclusive lock.
Locks	
Locks Currently Held	Total number of locks currently held by the application
Deadlocks Detected	Number of deadlocks that have occurred Deadlock situations are recognized and resolved automatically by the database The database configuration parameter `DLCHKTIME` determines when a lock wait situation is resolved.
Lock Timeouts	Number of times that a request to lock an object timed out instead of being granted The database configuration parameter `LOCKTIMEOUT` determines when a lock wait situation is resolved.
Lock Timeout Value	Value of the database configuration parameter`LOCKTIMEOUT`
Deadlock Event Monitor	
Statement History List Size	When a detailed deadlock event monitor with history is running, this element reports the number of bytes being used from the database monitor heap (`HON_HEAP_S`) to keep track of the statement history list entries.

17.28 Application Details: Calls

The following information is displayed on the *Calls* tab page in the content detail area of the *Applications* screen:

Field	Description
Rows	
Deleted	Number of data records that were deleted
Inserted	Number of data records that were inserted
Selected	Number of data records that were selected
Updated	Number of data records that were updated
Statements Executed	
SELECT SQL	Number of SELECT statements that were executed
UPDATE/INSERT/DELETE	Number of UPDATE, INSERT, and DELETE statements that were executed
DDL	Number of Data Definition Language (DDL) statements that were executed, such as CREATE TABLE, CREATE VIEW, ALTER TABLE, and DROP INDEX.
Statements Attempted	
COMMITs	Number of COMMIT statements that have been attempted
Rollbacks	Number of ROLLBACK statements that have been attempted Automatic rollbacks caused by error situations or deadlocks are not included.
Dynamic SQL	Number of dynamic SQL statements attempted
Static SQL	Number of static SQL statements attempted
Failed SQL	Number of attempted SQL statements that failed
Hash Joins	
Total	Total number of hash joins executed
Total Hash Loops	Total number of times that a single partition of a hash join was larger than the available sort heap space

Field	Description
Overflows	Number of times that hash join data exceeded the available sort heap space
Small Overflows	Number of times that hash join data exceeded the available sort heap space by less than 10%

17.29 Application Details: Sorts

The following information is displayed on the *Sorts* tab page in the content detail area of the *Applications* screen:

Field	Description
Sort Time	
Total	Total time in milliseconds required for all sort processes
Average	Average sort time in milliseconds
Sorts	
Total Sorts	Total number of sorts that have been executed
Sort Overflows	If the storage area allocated for sorting is not large enough, a sort overflow occurs. The hard disk is then used temporarily.

17.30 Application Details: Cache

The following information is displayed on the :*Cache* tab page in the content detail area of the *Applications* screen:

Field	Description
Catalog Cache	

Field	Description
Quality	Indicates percentage at which the data is read from the catalog cache, rather than directly from the hard disk This is calculated using the following formula: (catalog cache lookups - catalog cache insert) / catalog cache lookups * 100
Lookups	Number of times that the catalog cache was referenced to obtain table descriptor information
Inserts	Number of times that the system tried to insert table descriptor information into the catalog cache
Overflows	Number of times that an insert into the catalog cache failed due to the catalog cache being full
Heap Full	Number of times that an insert into the catalog cache failed due to the database heap being full
Package Cache	
Quality	Indicates percentage at which the data is read from the package cache, rather than directly from the hard disk This is calculated using the following formula: (package cache lookups - package cache inserts) / package cache lookups * 100
Lookups	Number of times an application looked for a section in the package cache
Inserts	Total number of times that a request section was not available for use and had to be loaded into the package cache

17.31 Application Details: Unit of Work

On the *Unit of Work* tab page in the content detail area of the *Applications* screen, various metrics related to applications are displayed, including, for example, the start and stop time of the unit of work.

The following metrics apply to DB2 Version 10 and higher only:

Non-Prefetch I/O

Field	Description
Non-Prefetch Requests	The number of requests for non-prefetch work successfully added to the prefetch queue

Field	Description
Failed Non-Prefetch Requests	The number of times an attempt to queue a non-prefetch request was made but failed. This element is for non-prefetch work done by prefetchers. A possible reason for the failed request is that the prefetch queue is full

Prefetcher I/O

Field	Description
Unread Prefetched Pages	Indicates the number of pages that the prefetcher read and that were never used
Prefetch Waits	The number of times an application spent waiting for an I/O server (prefetcher) to finish loading pages into the buffer pool
Prefetch Wait Time	The time an application spent waiting for an I/O server (prefetcher) to finish loading pages into the buffer pool. The value is given in milliseconds.
Data Pages Requested for Prefetch	The number of data pages successfully requested for prefetching
Data Prefetch Requests	The number of data prefetch requests successfully added to the prefetch queue
Index Pages Requested for Prefetch	The number of index pages successfully requested for prefetching
Index Prefetch Requests	The number of index prefetch requests that were successfully added to the prefetch queue
Columnar Pages Requested for Prefetch	The number of columnar data pages successfully requested for prefetching
Columnar Prefetch Requests	The number of columnar data prefetch requests successfully added to the prefetch queue
XDA Pages Requested for Prefetch	The number of XML storage object (XDA) data pages successfully requested for prefetching.
XDA Prefetch Requests	The number of XML storage object (XDA) data prefetch requests that were successfully added to the prefetch queue
Temp Data Pages Requested for Prefetch	The number of data pages for temporary tablespaces that were successfully requested for prefetching

Field	Description
Temp Data Prefetch Requests	The number of data prefetch requests for temporary table-spaces that were successfully added to the prefetch queue
Temp Index Pages Requested for Prefetch	The number of index pages for temporary tablespaces that were successfully requested for prefetching
Temp Index Prefetch Requests	The number of index prefetch requests for temporary table-spaces that were successfully added to the prefetch queue
Temp Columnar Pages Requested for Prefetch	The number of columnar pages for temporary tablespaces that were successfully requested for prefetching
Temp Columnar Prefetch Requests	The number of columnar prefetch requests for temporary tablespaces that were successfully added to the prefetch queue
Temp XDA Pages Requested for Prefetch	The number of XML storage object (XDA) data pages for temporary tablespaces that were successfully requested for prefetching
Temp XDA Prefetch Requests	The number of XML storage object (XDA) data prefetch requests for temporary tablespaces that were successfully added to the prefetch queue
Failed Data Prefetch Requests	The number of times that an attempt to queue a data pre-fetch request was made but failed. One possible reason is that the prefetch queue is full.
Failed Index Prefetch Requests	The number of times that an attempt to queue an index prefetch request was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed Columnar Prefetch Requests	The number of times that an attempt to queue a columnar prefetch request was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed XDA Prefetch Requests	The number of times that an attempt to queue an XML storage object (XDA) data prefetch request was made but failed. One possible reason is the prefetch queue is full and a request could not be obtained from the free list.
Failed Temp Data Prefetch Requests	The number of times that an attempt to queue a data prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.

Field	Description
Failed Temp Index Prefetch Requests	The number of times that an attempt to queue an index prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed Temp Columnar Prefetch Requests	The number of times that an attempt to queue a columnar prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.
Failed Temp XDA Prefetch Requests	The number of times that an attempt to queue an XML storage object (XDA) data prefetch request for temporary tablespaces was made but failed. One possible reason is that the prefetch queue is full and a request could not be obtained from the free list.

17.32 Application Details: Statement

The following information is displayed on the *Statement* tab page in the content detail area of the *Applications* screen:

Field	Description
Statement	
Start Time	Date and time when unit of work starts, the statement starts, or a deadlock is detected
Stop Time	Date and time when the statement stopped executing
Elapsed Time(s)	Duration of unit of work in seconds
Elapsed Time (µsec)	Duration of unit of work in microseconds
Rows	
Rows Read	Number of rows read from the table
Rows Written	Number of rows changed (inserted, deleted, or updated) in the table
Sort	

Field	Description
Sort Overflow	If the storage area allocated for sorting is not large enough, a sort overflow occurs. The hard disk is then temporarily used.
Total Sort Time	Total time for all sort processes
Number of Statement Sorts	Total number of sorts that have been executed
Data	
Logical Reads	Number of read accesses to data in the buffer pool
Physical Reads	Number of read accesses to data on disk
Temporary Logical Reads	Number of logical read requests that required I/O to get data pages into the temporary tablespace
Temporary Physical Reads	Number of physical read requests that required I/O to get data pages into the temporary tablespace
Index	
Logical Reads	Number of read accesses to data in the buffer pool
Physical Reads	Number of read accesses to data on disk
Temporary Logical Reads	Number of logical read requests that required I/O to get index pages into the temporary tablespace
Temporary Physical Reads	Number of physical read requests that required I/O to get index pages into the temporary tablespace

17.33 Application Details: Statement Text

Use

The following information is displayed on the *Statement Text* tab page in the content detail area of the *Applications* screen:

Field	Description
Statement	

Field	Description
Type	Type of statement processed Possible types are: • Static SQL statement • Dynamic SQL statement • An operation other than an SQL statement, for example, a bind or precompile operation
Operation	Operation currently being processed or most recently processed (if none is currently running) Possible operations are: • `SELECT` • `PREPARE` • `EXECUTE` • `EXECUTE IMMEDIATE` • `OPEN` • `FETCH` • `CLOSE` • `DESCRIBE` • `STATIC COMMIT` • `STATIC ROLLBACK` • `FREE LOCATOR` • `PREP_COMMIT` • `CALL` • `PREP_OPEN` • `PREP_EXEC` • `COMPILE`
Blocking Cursor	Indicates if the statement that is executed is using a blocking cursor (*YES*) or not (*NO*) If data is transferred in blocks and not row by row, the performance of the corresponding query is improved.
Statement	Text of dynamic SQL statement that was being processed when the snapshot was taken It can also be the text of the statement that was most recently processed if no statement was being processed at the time when the snapshot was taken.

If a statement is displayed, you can choose *EXPLAIN* to list the access plan for the statement execution. This function provides a detailed analysis of expensive SQL statements.

To show the current access plan of the statement, you can choose the *EXPLAIN* pushbutton. To display the ABAP source code from where the SQL statement was executed, you can choose the *Show Source* pushbutton.

> **i Note**
>
> The *Show Source* pushbutton is only available if the following applies:
>
> - The SAP system on which the currently monitored database is running is an ABAP system.
> - The SAP system has the minimum Support Package level as described in **SAP Note** 1493490.

More Information

EXPLAIN Function [page 297]

17.34 Application Details: SQL Workspace

The following information is displayed on the *SQL Workspace* tab page in the content detail area of the *Applications* screen:

Field	Description
Private Workspace	
Lookups	Number of times an application looked for an SQL section in the agent's private workspace
Inserts	Inserts of SQL sections by an application into the private workspace
Overflows	Number of times that the private workspace overflowed the bounds of its allocated memory
High-Water Mark	Largest size reached by the private workspace
Shared Workspace	
Lookups	Number of times an application looked for an SQL section in the agent's shared workspace
Inserts	Inserts of SQL sections by an application into the shared workspace
Overflows	Number of times that the shared workspace overflowed the bounds of its allocated memory
High-Water Mark	Largest size reached by the shared workspace

18 Disclaimer

Links to IBM Documentation in This Guide

By following links to IBM Documentation you are leaving the SAP product documentation and entering a site that is not hosted by SAP. By using the link, YOU AGREE that unless expressly stated otherwise in your agreements with SAP you are about to access an external webpage which is not part of SAP's offering: (i) the content of the linked-to site and any further external site is not product documentation and you may not infer any product documentation claims against SAP based on this information; (ii) the fact that SAP provides links to external sites does not imply that SAP agrees or disagrees with the contents and information provided on such sites. SAP does not guarantee the correctness of the information provided. (III) SAP DOES NOT GIVE ANY REPRESENTATION REGARDING THE QUALITY, SAFETY, SUITABILITY, ACCURACY OR RELIABILITY OF ANY EXTERNAL WEBPAGE OR ANY OF INFORMATION, CONTENT AND MATERIALS PROVIDED THEREON; (IV) YOU VISIT THOSE EXTERNAL WEBPAGES ENTIRELY AT YOUR OWN RISK. SAP SHALL NOT BE DIRECTLY OR INDIRECTLY RESPONSIBLE OR LIABLE FOR ANY DAMAGE OR LOSS CAUSED OR ALLEGED TO BE CAUSED BY OR IN CONNECTION WITH YOUR USE OF OR RELIANCE ON ANY CONTENT, GOODS OR SERVICES AVAILABLE ON OR THROUGH ANY SUCH LINKED WEBPAGE.

Important Disclaimers and Legal Information

Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.
About the icons:

- Links with the icon : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon : You are leaving the documentation for that particular SAP product or service and are entering an SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.
The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

Bias-Free Language

SAP supports a culture of diversity and inclusion. Whenever possible, we use unbiased language in our documentation to refer to people of all cultures, ethnicities, genders, and abilities.

www.sap.com/contactsap

THE BEST RUN SAP