

Customizing System Setting (BC-CUS)



HELP.BCBCUSIMG

Release 4.6C



Copyright

© Copyright 2001 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

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

Microsoft[®], WINDOWS[®], NT[®], EXCEL[®], Word[®], PowerPoint[®] and SQL Server[®] are registered trademarks of Microsoft Corporation.

IBM[®], DB2[®], OS/2[®], DB2/6000[®], Parallel Sysplex[®], MVS/ESA[®], RS/6000[®], AIX[®], S/390[®], AS/400[®], OS/390[®], and OS/400[®] are registered trademarks of IBM Corporation.

ORACLE[®] is a registered trademark of ORACLE Corporation.

INFORMIX[®]-OnLine for SAP and Informix[®] Dynamic Server[™] are registered trademarks of Informix Software Incorporated.

UNIX[®], X/Open[®], OSF/1[®], and Motif[®] are registered trademarks of the Open Group.

HTML, DHTML, XML, XHTML are trademarks or registered trademarks of W3C[®], World Wide Web Consortium, Massachusetts Institute of Technology.

JAVA[®] is a registered trademark of Sun Microsystems, Inc.

JAVASCRIPT[®] is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, SAP Logo, R/2, RIVA, R/3, ABAP, SAP ArchiveLink, SAP Business Workflow, WebFlow, SAP EarlyWatch, BAPI, SAPPHIRE, Management Cockpit, mySAP.com Logo and mySAP.com are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other products mentioned are trademarks or registered trademarks of their respective companies.

Icons

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

Contents

Customizing System Setting (BC-CUS)	5
View-Led Customizing Transaction	6
Working with View-Led Customizing Transactions	7
View Cluster Transaction	8
Using View Cluster Transactions	9
Views and View Cluster Fields	10
Working with Navigation Boxes	11
Displaying Navigation Boxes in Detail Screens	12
Displaying Table Entries	13
Changing Table Entries	14
Selecting Table Entries	15
Deleting Table Entries	16
Creating New Table Entries	18
Going To Other Table Entries	20
Selecting Table Entries by Content	22
Open the Detail Screen	23
Deselecting Table Entries	24
Changing the Content of Fields	25
Printing Customizing Lists	26
Setting the REC/CLIENT Parameter in the System Profile	27
Defining Validity Periods of Customizing Tables	28
Service Functions in Customizing Transactions	29
Comparison of Customizing Objects	30
Logging Customizing Objects	31
Analyzing Logged Customizing Objects	33
Analyzing Change Logs	34
List the change logs	35
Displaying Logged Tables	36
Counting Change Documents	37
Archiving Change Logs	38
Deleting Change Logs	39
Logging Releases <4.0A	40
Table History Functionality	41
Comparing Current Table Content with Earlier Table Content	42
Checking Dictionary Settings	43
Listing Change Logs	44

Customizing System Setting (BC-CUS)

The sections that follow contain instructions for configuring your R/3 System using Customizing transactions. They describe, for example, how to work with views, view clusters, Customizing tables, table history, and the analysis, comparison, and adjustment options.

View-Led Customizing Transaction

View-Led Customizing Transaction

To be able to set business objects (for example, purchasing group, currencies) you must maintain fields in several tables or several individual fields in one table. View-led Customizing transactions conceal the physical table structures. In order to make your job simpler, all of the relevant fields for an object are grouped together in one **view**.

Views are logical tables that group together fields from physical tables to give them meaningful business context.



Views are Data Dictionary objects.

Advantages of Views

Views have the following advantages:

- To configure a particular object, you need only one call to maintain several tables.
- You only see those fields from the physical tables that are relevant to the business object.
- All view-led Customizing transactions have a standard dialog, standard functions and the same initial screen.

One-level and two-level procedures

View-led Customizing transactions are either one-level or two-level procedures:

One-level procedures

One-level procedures contain only an **initial screen**. You can change all of the fields that are displayed.

Two-level procedures

Two-level procedures contain an **initial screen** and a **detail screen**. The Customizing transaction will lead you to the detail screen. You can edit the fields displayed in the initial screen and in the detail screen.

See also:

[Using view-led Customizing transactions \[Page 7\]](#)

Working with View-Led Customizing Transactions

You can carry out the following activities in view-led Customizing transaction:

- [Create a new table entry \[Page 18\]](#)
- [Changing table entries \[Page 14\]](#)
- [Deleting table entries \[Page 16\]](#)
- [Displaying table entries \[Page 13\]](#)
- [Printing Customizing lists \[Page 26\]](#)
- [Put table entries in a task \[Ext.\]](#)
- [Analyzing tables \[Page 30\]](#) (Customizing object comparison)



The transport connection in the Customizing transactions is dealt with in [Setting up the production systems \[Ext.\]](#).

View Cluster Transaction

View Cluster Transaction

A view cluster is the set of objects, such as views, reports or tables which belong to one business function. The data can be arranged either hierarchically or non-hierarchically in view clusters.

A view-cluster transaction edits the complex cross-table data in a view cluster. The transaction navigation box shows the complex business object with its components or subobjects.

See also:

[Using View Cluster transactions \[Page 9\]](#)

Using View Cluster Transactions

You can perform the following activities in view-cluster transactions:

- [Using the navigation window \[Page 11\]](#)
- [Create a new table entry \[Page 18\]](#)
- [Changing table entries \[Page 14\]](#)
- [Deleting table entries \[Page 16\]](#)
- [Displaying table entries \[Page 13\]](#)
- [Printing Customizing lists \[Page 26\]](#)
- [Analyzing tables \[Page 30\]](#) (Customizing object comparison)

You can save your work either on each navigation level or on the highest navigation level (that is, you save all object components together when you finish working).



The transport connection in the Customizing transactions is dealt with in [Transport Customizing elements \[Ext.\]](#).

Views and View Cluster Fields

Views and View Cluster Fields

There are two types of fields in views and view clusters:

- **Key field**

A key consists of one or more key fields. A key field is accordingly a part of a key which explicitly specifies the entry in a table.

You **cannot** change existing key fields.

- **Function field**

The function fields contain the characteristic or the value of a key. You can put values in function fields.

Working with Navigation Boxes

In the **navigation box** you can see the complex business object with its "parts" or object components.

The object components are displayed in different dependency levels. The complex business object is referred to as a higher-level object; the object components are referred to as dependent objects.

To use the navigation box:

1. [Select \[Page 15\]](#) an object in the navigation box.
2. Position the cursor on an object component in the object.
3. Choose *Choose*.

The content of the chosen object component and the selected higher-level object is displayed in a screen under the navigation window.

4. To open the detail screen, choose *Goto → Details*.

Displaying Navigation Boxes in Detail Screens

Displaying Navigation Boxes in Detail Screens

Normally the navigation box is not displayed in detail screens.

If you want to see the navigation box, choose *Table view* → *Other view*.

Displaying Table Entries

Views and view clusters can both have:

- **Single-level views**

In single-level views (for example, the *Regions* view from *Insert regions* in the IMG), all table entries are displayed in the overview screen.

- **Two-level views**

In two-level views (for example the *Country Global Parameters* view for *Define countries* in the IMG), the keys and the most important fields are displayed in the overview screen.

To display the table entries in the detail screen:

1. In the overview screen, position the cursor on the table entry for which you want to see the further table entries.
2. Choose *Goto* → *Details*

The associated table entries are displayed.

Changing Table Entries

Changing Table Entries

If you are in a *Display* screen, choose *Table view* → *Display/Change* to maintain the table entries. The views and view clusters are set to *Change* as standard. If you have the required authorization, you can maintain the entries in the *Change* screen.

In the overview screen, you can change the function fields or create new keys and function fields.

Selecting Table Entries

Many options (for example, deleting table entries, changing field content, copying table entries to a task) require that you first select the table entries that you want to work on:

- **Selecting single table entries**

Click the required table entry.

- **Selecting a block of table entries**

1. Position the cursor on the table entry at the start of the block.
2. Choose *Edit* → *Select* → *Select block*
3. Position the cursor on the table entry at the end of the block.
4. Choose *Edit* → *Select* → *Select block*

The required block is selected.

- **Selecting all table entries**

Choose *Edit* → *Select* → *Select all*

All table entries are selected.

Deleting Table Entries

Deleting Table Entries

To delete a table entry or a block of entries, first select the entry (entries). After it has been deleted, a selected entry is assigned a deletion indicator in an internal buffer and no longer appears in the list of the table entries in the overview screen. As long as you have not yet saved your deletion, you can display and retrieve the entries which are buffered internally.

Procedure:

1. [Select \[Page 15\]](#) the table entry or the block you want to delete.
2. Choose *Edit* → *Delete*.
3. Check that you really want to delete the table entries concerned.
Choose *Selection* → *Display deleted entries*.

This takes you to the internal buffer (the *Overview of Deleted Entries* screen).

4. If you find table entries that you do not want to delete, select them in this internal buffer.
5. To retrieve the table entries that you selected in Step 4, choose *Retrieve*.
6. To definitively confirm your deletions, save the contents of the internal buffer in the *Overview of Deleted Entries* screen.

You can now no longer retrieve the deleted table entries.

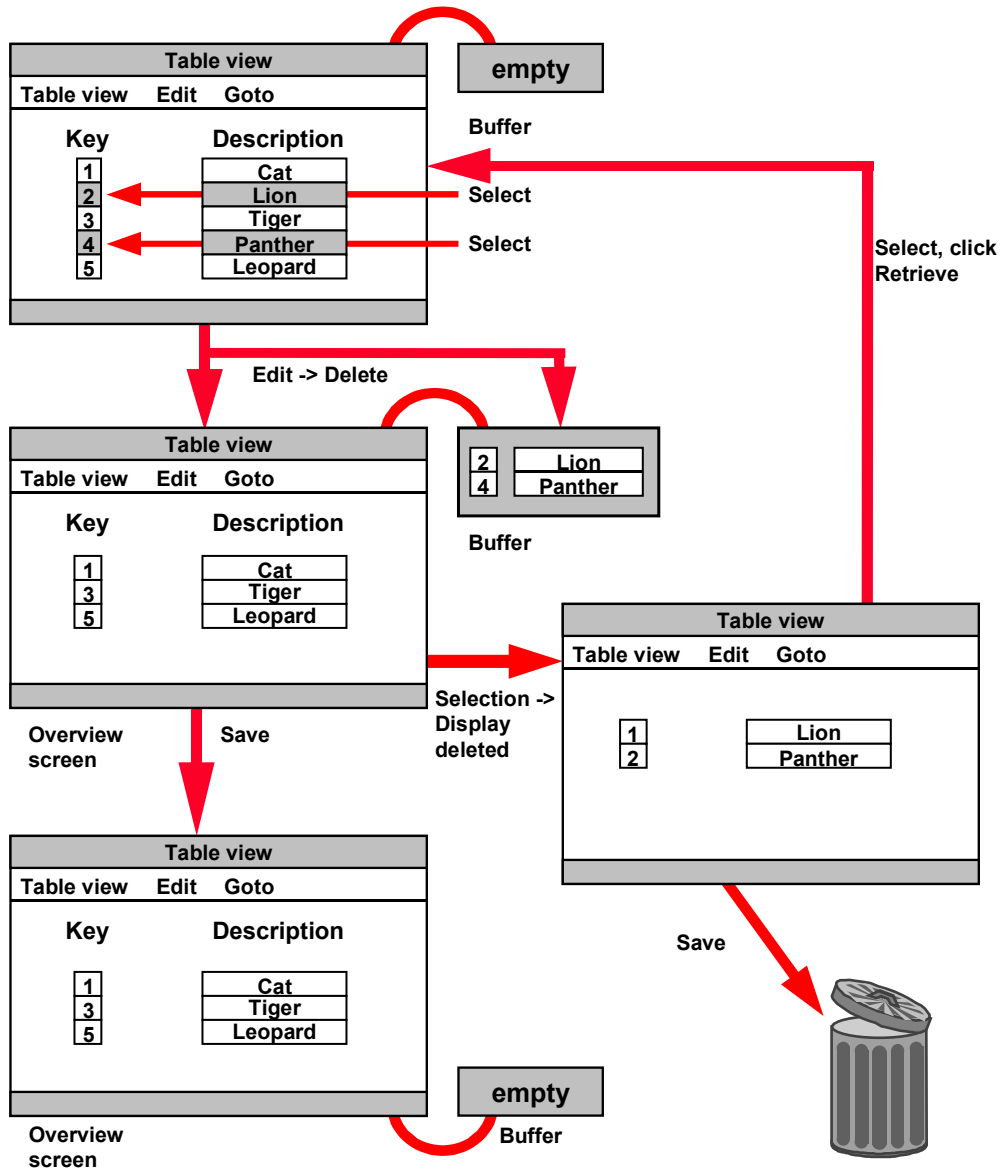


When you delete table entries in a view cluster, the R/3 System checks whether the table entry to be deleted contains dependent table entries. If it does, the dependent table entries are deleted too.

When you retrieve entries in a view cluster, the R/3 System also retrieves the dependent and higher-level table entries automatically.

Illustration: What Happens when You Delete Table Entries

Deleting Table Entries



Creating New Table Entries

Creating New Table Entries

You can create new entries by reference or without a reference.

Creating New Table Entries Without References

When you create new table entries without a reference, in single-level views the system presents an empty overview screen. In two-level views, the system presents a detail screen.

Procedure:

1. Choose *Edit* → *New entries*.
2. Enter values in the key fields and function fields.
3. Save your work.

If you enter a key that is the same as one that already exists, the system displays a message in the status bar to tell you so.

In this case you have to:

- a) Change the new key
- b) Choose *Continue*.
- c) Save your work.

The data is now copied into an internal buffer. You can now only maintain the function fields.

Creating New Table Entries by Reference

When you create new table entries by reference, you select an existing table entry as a reference, which you then change to meet your requirements.

1. [Select \[Page 15\]](#) the table entry that you want to use as a reference (reference entry).
2. Choose *Edit* → *Copy as*.

or

Choose *Copy as*.

3. Change the key fields and, if appropriate, the function fields. They should be different than the reference entry.
4. Choose *Continue*.

If you enter a key that is the same as one that already exists, the system displays a message in the status bar to tell you so.

In this case you have to:

- a) Change the new key
- b) Choose *Continue*.

The data is now copied into an internal buffer. You can now only maintain the function fields.

Creating New Table Entries

- c) Choose *Continue* again.
The overview screen is presented.
- d) Save your work.

Until you have copied all of the table entries you selected:

- Further overview screens are presented ([single-level views \[Page 13\]](#))
- Further detail screens are presented ([two-level views \[Page 13\]](#)).

When you have copied all of the reference entries, choosing *Transfer* will take you back to the original overview screen.

The status bar shows you how many table entries you have copied.



When you copy entries in a view cluster, the object components are not copied too.

Displaying Created Table Entries

When you create a new table entry, this table entry is first stored in an internal buffer. This table entry is only really created when you save it.

Before you save, you can display the table entries you have just created and that are buffered internally. Choose *Selection* → *All created entries*.

Going To Other Table Entries

Going To Other Table Entries

You can go to a another table entry in the following ways:

- **Call up the next table entry**

Choose *Goto* → *Next entry*.

The cursor is positioned on the next table entry down.

- **Using *Previous entry***

Choose *Goto* → *Previous entry*.

The cursor is positioned on the previous table entry (above).

- **Using *Other entry***

1. Choose *Goto* → *Other entry*.

A further screen appears in which you enter the key of the entry you want to go to.

2. Choose *Continue*.



If you do not know the precise key (for example, CL for Chile), you can enter the beginning of the key (in this case C for Chile). The R/3 System then goes automatically to the first table entry whose key matches this letter. In this case the cursor goes to CA for Canada.

- **Call up table entry using name**

1. Choose *Selection* → *By content*

2. Select *Name*, and then choose *Continue*.

3. Enter the part of name you know (for example Africa for South Africa).

4. Select **cs** as the operand (stands for "contains string").

5. Choose *Choose*.

The system displays a list with the table entries containing the name you entered. Since a name, unlike a key, need **not** be unique, this list can contain several table entries.

or

1. Choose *Selection* → *By content*.

2. Select **Name**, and then choose *Continue*.

3. Enter the letters of the name which you know (for example USA for USA or United States of America).

4. Select **CA** as the operand (stands for "contains at least one character").

5. Choose *Choose*.

Going To Other Table Entries

The system displays a list of all the names containing the letters you entered. This list can be very long because the letters are linked with **OR**. In the above example, all country names are displayed that contain the letter **U**, the letter **S**, or the letter **A**.

Selecting Table Entries by Content

Selecting Table Entries by Content

You can select table entries according to their field contents. This allows you to combine several fields using different operands (for example >, <, =) in order to obtain a special evaluation.

To do this:

1. Choose *Selection* → *By content*.
Each function field and key field that occurs in the View that you edited is displayed.
2. If necessary, scroll through the list and position the cursor on the required field.
3. Choose *Continue*.
A dialog box is displayed, in which you can enter an operand in the *Co* field and the comparison value for the selected field in the *Field content* field. The operand = (equal to symbol) is set by default.
4. Enter the operand.
5. Enter the comparison value.
6. Choose *Choose*.
A list of all table entries which correspond to the operands and comparison values is displayed.

You can now specify the selection by means of further fields.

1. Choose *Selection* → *By content*.
Your previous selection is displayed.
2. Choose *Append*.
All key fields and function fields are listed again.
3. Position the cursor on the field you want to use to as a further restriction criterion.
4. Choose *Continue*.
A further dialog box is displayed, in which you can choose the link category (for example, **AND**, **OR**), the comparison operator and the comparison value.
5. Choose *Choose*.
A list of all table entries that correspond to the comparison operands, the comparison values, and the linking operands, is displayed.

You can make more changes to your selection by:

- Appending fields using logical links AND and OR
- Delete fields from selection
- Changing the sequence of the selection criteria.

Open the Detail Screen

If you are working in a [two-level view \[Page 13\]](#) you can open the detail screen.

To do this:

1. Position the cursor on the table entry for which you want to open the detail screen.
2. Choose *Goto* → *Details*
3. Edit the table entries in the detail screen.

Deselecting Table Entries

Deselecting Table Entries

To deselect all of the table entries you selected, choose *Edit* → *Select* → *Deselect all*.

Changing the Content of Fields

If you have a great number of table entries that you want to change in a field, you can do this in one step. However, you can only edit function fields, not key fields.

To do this:

1. [Select \[Page 15\]](#) all of the table entries that you want to change.
2. To change the field contents, choose *Edit* → *Change field contents*.
The system displays a list of all of the function fields.
3. Position the cursor on the required function field and choose *Continue*.
A further screen is displayed where you have to enter the value of the field which should be used in all table entries.
4. Enter the required value.
5. Choose *Replace*.
All selected table entries of the chosen field are replaced by the entered value. At this stage, your changes are stored in an internal buffer.
6. Make sure you really want to change the field contents of all selected table entries.
To do this, Choose *Selection* → *All changed entries*.
7. In the list, select the table entries that you want to leave in their original state.
8. Choose *Edit* → *Undo change*.
9. Confirm that you want to reset to the original.
10. Save your entries.

Printing Customizing Lists

Printing Customizing Lists

You can print lists in Customizing transactions or view cluster transactions.

To do this:

1. Choose a table view.
2. Choose *Table view* → *Print*.
The table view is displayed
3. To print the list, choose *List* → *Print*.
4. Enter the number of copies in the output device dialog box.



The table view screen display provides new functions, e.g. export the table view to MS Excel or MS Word. These new functions are based on the use of the ABAP List Viewer for screen output.

See also:

[ABAP List Viewer \[Ext.\]](#)

Setting the REC/CLIENT Parameter in the System Profile

SAP recommends that automatic table change logging is authorized only in systems where this is worthwhile. In general this includes the following systems:

- The R/3 System where your Customizing work is done
- Your production system

Do not allow automatic logging in a test system under any circumstances.

Table change logging is controlled by parameter REC/CLIENT in the R/3 system profile. The parameter can be given the following values:

Parameter value	Means
rec/client = off	Table change logging is off
rec/client = all	Table change logging active in all clients (See Caution below)
rec/client = <client#>	Table change logging active in listed clients

[Automatic recording \[Page 43\]](#) works only when REC/CLIENT is set to:

- **REC/CLIENT = all**
- **REC/CLIENT = <client#>...**

Report **RSPARAM** displays all profiles in the system with their values.



Setting the REC/CLIENT parameter to *all* can seriously reduce system performance.

SAP recommends you do **not** use the *all* value. You should limit automatic logging to a particular client.

Defining Validity Periods of Customizing Tables

Defining Validity Periods of Customizing Tables

The Human Resources component (HR) contains time-dependent tables. The validity period is in the key of time-dependent tables. An example of a time-dependent table is the table with contributions for pension insurance. The records change with time and are therefore only valid up to a certain date.

To maintain values for any validity period, choose *Edit* → *Limit validity*.

In the view or in view cluster, only the last validity period is displayed with its value.

Choose *Edit* → *Expand/collapse* to:

- Display the previous ranges for the validity period, with values (*Expand*)
- "Hide" the displayed previous ranges and values for the validity period again, so that only the last valid duration is displayed (*Collapse*).



For views or view clusters without a validity period, these options are **not** available.

Service Functions in Customizing Transactions

You can transport the configurations which you maintained with the Customizing transactions into other clients or systems. See also [Transporting Customizing elements \[Ext.\]](#).

SAP provides service functions to facilitate these tasks in the Customizing transactions:

- You can record the system changes you make in Customizing transactions in a **Customizing request**.

[Create tasks and change requests \[Ext.\]](#)

- You can transfer table entries from client *000* or from another client directly into your Customizing client with the toolbar, without having to use the transport system.

[View Maintenance: Comparing/Adjusting Between Clients/Systems \[Ext.\]](#)



You can activate automatic recording for each client.

Comparison of Customizing Objects

Comparison of Customizing Objects

You can :

- **Compare** views or tables with each other
- **Adjust** views or tables in a logon (target) client to match a comparison client (copying entries from a comparison client into the logon client or deleting entries in the logon client).

Prerequisites

To compare or adjust Customizing objects, you need authorization object S_TABU_CLI.

You need protection level 0 or 1 (a setting in Table T000) in the comparison client.

The client in which you run the comparison tool is known as the **logon client**, and the system in which you are working is known as the **logon system**.

The client against which you run the comparison is known as the **comparison client**, and the system it is in is known as the **comparison system**.

The clients can be in the same R/3 System or in two different R/3 Systems.



You can compare between different clients in the same system or in different systems for **client-dependent** views or tables

For **cross-client** views or tables, only comparisons between different systems are appropriate.

To perform the comparison, see the [Customizing Cross-System Tools \[Ext.\]](#) documentation.

Logging Customizing Objects

Use

The analysis of logged Customizing objects allows you to answer the following questions about an Customizing setting changes:

- who made a change?
- what was changed?
- when was the change made?

Logs are analyzed by object, i.e. the changes can be shown in their Customizing context, not just technically at table level.

Change logs can be deleted or archived. . If you archive logs, you must ensure that the analysis includes the archived logs.

The analysis only applies to the logon client. If you want to see change logs for another client, you must logon in that client. Logs of cross-client objects are shown in every logon client.



Productive data changes should **not** be logged, for performance reasons.

Prerequisites

The analysis can only be used when:

- the *Log data changes* checkbox is set in the technical settings of the table, or the tables in the view, in the ABAP Dictionary



The Dictionary setting is predefined by SAP for the objects to be logged. You should **not** deactivate logging for these objects, as analysis inconsistencies can occur if customizing settings are sometimes maintained without logging.

To activate logging:

1. Call transaction SE13.
 2. Enter the table name.
 3. Choose *Change*.
 4. Set the checkbox *Log data changes*.
 5. Save your entries.
- the profile parameter *rec/client* is maintained.



The logging profile parameter should always be set in the customizing system and client. Logging in the productive system or client can impair performance.

Logging Customizing Objects

To log transports, the transport control profile must contain the entry `r3transoptions = recclient="xxx"`, where `xxx` has the same value as the `rec/client` profile parameter.

To set the profile parameter `rec/client` for logging:

1. Call transaction RZ11.
2. Enter `rec/client` as the profile parameter to be changed.
3. Set the profile parameter to the desired value.

If the profile parameter is `OFF`, there is no logging in the system. `ALL` activates logging in all clients in the system. A value `051`, `100` logs all client-specific tables in the specified clients (`051` and `100`). Cross-client tables are logged in all clients.

4. Save your entries.

Analysis

The analysis is made as follows.

- Go to the IMG and choose the activity whose change history you want to see.
- Enter the period you want to analyze and start the analysis.
- The changes are displayed as a list. The following information is provided for each change:
 1. Date and time
 2. Key field(s)
 3. Comparison of the old and new function field values, or new or deleted record notes.

You can call detail and technical information for individual entries.



You can call the analysis in the customizing menu via *Tools* → *Table history* instead of from the IMG. You specify here whether you want to see the sequence of changes to a table or Customizing object, or the differences between the current version and a previous version of a table.

If you choose the change sequence, you must enter the name of the object or table in the next screen, as well as the entries described above.

The analysis can also be called from view or view cluster maintenance via *Utilities* → *Change logs*.

The comparison is described in the table history documentation.

[List change logs \[Page 35\]](#)

[Table history functionality \[Page 41\]](#)

[Archiving table change logs \[Ext.\]](#)

Analyzing Logged Customizing Objects

Use



This screen only appears when you start the analysis from the Customizing menu via *Tools* → *Table history*.

You can see the changes made to a table/Customizing object or compare the current status of a table with a past status. Other functions:

- List logged tables
- Get number of logs for all or selected tables
- Archive logs
- Delete logs

Further Information

[Logging Customizing objects \[Page 31\]](#)

Analyzing Change Logs

Analyzing Change Logs

Use

You can see the changes made to a Customizing object or table during a specified period.

Functions

You can make the following entries:

- **Customizing object/table:** This option is only visible when you enter the analysis via the Customizing menu.
- **Group box Time restriction:** enter the date and time of the beginning and end of the period for which you want to see the changes made.
- **Analysis of:** specify whether you want to see changes made to a Customizing object or a table. This option is only visible when you enter the analysis via the Customizing menu.
- **Archived logs checkbox:** Select the field if the analysis period contains archived logs which are to be included in the analysis.
- Other functions:
- **Logging: Display status:** Tells you the clients in which logging is active and whether the tables in the objects are logged.
- **Execute:** Run the analysis. You can perform the analysis in the background or print the result directly in the *Program* menu.

Further Information

[Logging Customizing objects \[Page 31\]](#)

List the change logs

Use

The change log list has two sections. The first contains the following context information:

- **Analysis parameters:** System information, Release, client, date and time, user.
- **Selected period**
- **Selected Customizing objects and tables**

The second section lists the changes per object in chronological order with the following information.

- **Change time**
- **Key of record changed, inserted or deleted**
- **Old and new function field values.** If the entry is new, the note *New data record* appears. If the entry was deleted, the note *Data record deleted* appears.

Functions

Functions:

- *Display detail:* A complete list of all fields of the old and changed record. For new entries only the new record, for deleted entries only the old record, is displayed.
- *Techn. information:* Technical object information.

To get the information, position the cursor on the change log. Choose a function. The information is displayed in a new window. The following fields are highlighted:

- Key fields = green background
- Language-dependent fields = yellow background
- Changed fields = red background (only in *Display detail* function)
- Other functions:
 - **List → Print**
 - **List → Save in PC file**

Further Information

[Logging Customizing objects \[Page 31\]](#)

Displaying Logged Tables

Displaying Logged Tables

To log table changes automatically, you must have [allowed data change logging \[Page 41\]](#) in the Data Dictionary. You can list the tables for which automatic recording is set, in the table history.

Proceed as follows:

1. Choose *Tools* → *Table history* in the customizing menu.
2. Choose *List logged tables*.

A list of all table names is displayed. You can print this list.

Counting Change Documents

You can use table history to count the number of change documents. Procedure:

1. In the *Customizing* screen, choose *Tools* → *Table history*.
2. To find out how many change documents there are, choose *Administration* → *Total change docs (all)*

To find out how many change documents there are for selected tables, choose *Administration* → *Total change docs (selection)*. You can restrict the search to a stated period of time or particular tables .

Archiving Change Logs

Archiving Change Logs

1. Choose *Tools* → *Table history* in the customizing menu.
2. Choose *Edit* → *Logs* → *Archive*.

Further information

[Archiving table change logs \[Ext.\]](#)

The selected change logs are archived in a file and deleted from the log table.

Deleting Change Logs



Change logs are no longer available after deletion. They can also not be retrieved later. If you want to delete the change logs from the database but continue to use them for analyses, you must archive them.

To delete change logs, proceed as follows:

1. Choose *Tools* → *Table history* in the customizing menu.
2. Choose *Edit* → *Logs* → *Archive*.
3. Enter the end date and the table(s) whose change logs you want to delete.
All change documents as old as or older than the end date are deleted.
If you do not specify a table name, all change documents which satisfy the date criterion are deleted.
4. Choose *Execute*.
5. Confirm the deletion.

Logging Releases <4.0A**Logging Releases <4.0A**

The log table DBTABPRT was replaced by the table DBTABLOG in Release 4.0A. If logs which were written before Release 4.0A are to be used, they must be converted using the report RSDBLOGCONV40. The logs can then be analyzed and/or archived.

Table History Functionality

The **table history** compares the current status of a table with an earlier version.

To perform this analysis, you must first [activate the logging of table changes \[Page 43\]](#). Only in this way can the SAP System create a change entry for each change in the log table.

The change log contains the time of the change and the name of the changer.



Displaying the list and analyzing the table log database affect the performance of the SAP System and have long runtimes. Response times of several minutes are normal.

The table log database is accessed directly via the database interface. The table history provides only a technical user view of the SQL tables in the database. The transaction codes are the only information about where and in which customizing project changes were made.

Comparing Current Table Content with Earlier Table Content

Comparing Current Table Content with Earlier Table Content

So that you can use this function, the following prerequisites must be fulfilled:

- The [REC/CLIENT parameter \[Page 27\]](#) in the system profile is set to allow data logging.
- [Logging \[Page 43\]](#) is activated for the table concerned.

Use table history to compare the current content of a table with the content at an earlier time

Procedure:

1. In the Customizing screen, choose *Tools* → *Table history*.
2. Select *Comparison: History* ↔ *Current* and choose *Analyze change documents*
3. Enter the name of the table/view for which you want a comparison, and a date and time with which to compare.
4. Choose *Compare*.

The content of the table at the time you specified is compared and the current content are displayed. Differences in field content are marked.

The status codes on the left have the following meanings:

Statures	Means
(none)	Entry present in current and old version, and is identical.
C	Entry present in current version, absent in old version.
M	Entry present in current and old version, and is not identical.
S	Entry absent in current version, present in old version.
*	Entry flagged for deletion.



You cannot compare system tables using this function.

Checking Dictionary Settings

To check the Dictionary settings:

1. Choose *Tools* → *ABAP Workbench* → *Development* → *Dictionary* in the initial screen.
2. Enter the name of the table whose log settings you want to check.
3. Choose *Display*.

The table is displayed.

4. Choose *Goto* → *Technical settings*.
5. Check whether the checkbox *Log data changes* is marked.



All customizing tables are automatically recorded by default. Changing this default setting changes the Data Dictionary. You need a change licence, which you can get from SAP via OSS, in this case.

Listing Change Logs

Listing Change Logs

This function has the following prerequisites:

- The [Rec/client \[Page 37\]](#) system profile parameter must be set to allow data logging.
- [Logging \[Page 41\]](#) must be activated for the desired table.

To list change logs, proceed as follows:

1. Choose *Tools* → *Table history* in the customizing menu.
2. Choose the change log list selection criteria.
3. Choose *Compare* or *Display*, depending on whether you want to see the current contents or not.