Developer's Guide
SAP NetWeaver 2004s SPS 7

Business Planning and Analytical Services

Document Version 3.00 – März 2006

THE BEST-RUN BUSINESSES RUN SAP

SAP AG
Dietmar-Hopp-Allee 16
69190 Walldorf
Germany
T +49/18 05/34 34 24
F +49/18 05/34 34 20
www.sap.com

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes
only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

Disclaimer

Some components of this product are based on Java™. Any code change in these components may cause unpredictable and severe malfunctions and is therefore expressively prohibited, as is any decompilation of these components.

Any Java™ Source Code delivered with this product is only to be used by SAP's Support Services and may not be modified or altered in any way.

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, except if such damages were caused by SAP intentionally or grossly negligent.

Typographic Conventions

Type Style	Represents
Example Text	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Cross-references to other documentation.
Example text	Emphasized words or phrases in body text, graphic titles, and table titles.
EXAMPLE TEXT	Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.
`Example text`	Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
`Example text`	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
`<Example text>`	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
`EXAMPLE TEXT`	Keys on the keyboard, for example, F2 or ENTER.

Icons

Icon	Meaning
⚠	Caution
💬	Example
💡	Note
🧭	Recommendation
SYN	Syntax

Contents

1 Business Planning and Analytical Services

Purpose

Various BI interfaces and tools are available if you want to modify the *Business Planning and Analytical Services* scenario.

Advantages for Application Development

In this section, we distinguish between the two BI planning solutions, BI integrated planning and the BW-BPS, and analysis process design (for example, for data mining solutions).

BI Integrated Planning

- You can develop your own data models and planning-specific metadata objects for your business planning.
- You can use the BEx Query Designer to define input-ready queries for the manual entry of plan data.
- In the BEx Analyzer and Web Application Designer, you can develop planning applications that support both manual and automatic data entry and changes.
- With the SAP enhancement concept, you can make enhancements to the standard in the BI system. Within the BI system, you can use customer exits and BAdIs to make enhancements in the Query Designer and the Web Application Designer.

Business Planning and Simulation (BW-BPS)

- You can use the BW-BPS Web Interface Builder to create Web-enabled planning applications in the form of Business Server Page applications (BSP applications).
- Services that are based on the SAP NetWeaver Internet Communication Framework (ICF) are delivered with BI. The service for the Status and Tracking System (STS) is implemented as a Web service.

Analysis Process Design

- You use the analysis process designer to define analysis processes that explore and identify hidden or complex relationships between BI data.
- You use the data mining workbench to create models. This allows you to use the methods according to your requirements.

Prerequisites

Area	Prerequisites
BI integrated planning: modeling the data basis	-
BI integrated planning: modeling planning-specific metadata objects	-
BI integrated planning: definition of an input-ready query	-

BI integrated planning: creation of Web templates	Proficiency in standard markup languages
Enhancements using function exits and BAdIs	ABAP proficiency
BW-BPS: Web service for STS	-
BW-BPS: Web Interface Builder of BW-BPS	ABAP proficiency
Analysis Process Design	-
Data mining	-

2 Getting Involved

This section provides an overview of the concepts and the development environment.

2.1 Working with the Development Environment

Purpose

The BI system provides heterogeneous development environments for the *Business Planning and Analysis Services* scenario.

Area	**Development Environment**
BI integrated planning: modeling the data basis	Data Warehousing Workbench
BI integrated planning: modeling planning-specific metadata objects	Planning Modeler
BI integrated planning: definition of an input-ready query	BEx Query Designer
BI integrated planning: creation of Web templates	BEx Web Application Designer and BEx Analyzer
Enhancements using function exits and BAdIs	ABAP Workbench tools
BW-BPS: Web service for STS	Internet Communication Framework (ICF)
BW-BPS: Web Interface Builder of BW-BPS	Web Application Builder of the Application Server
Analysis Process Design	Analysis Process Designer
Data mining	Data Mining Workbench, Analysis Process Designer

3 Go and Create

We provide instructions for first development in the following areas:

BI Integrated Planning: Modeling Planning Scenarios

You use the planning modeler and the planning wizard to model, administer, and test all the metadata that belongs to a planning scenario.

For information about creating planning models, see Modeling Planning Scenarios [Page 21].

BI Integrated Planning: Implementing Our Own Planning Function Types

Planning function types are parameterizable processes to change transaction data within BI Integrated Planning.

For information about creating planning function types, see Implementing Planning Function Types [Page 71].

BW-BPS: Overview of Planning with BW-BPS

For an overview of the required and optional steps in planning with BW-BPS, see Overview of Planning with BW-BPS [Page 88].

BW-BPS: Web Interface Builder

You use the Web interface Builder to create Web interfaces. You generate BSP applications on the basis of the Web interface. The BSP application accesses the planning objects using a Web browser.

For information about creating a simple Web interface, see Creating and Editing Web Interfaces [Page 288].

Analysis Process Design

Analysis processes allow you to explore and identify complex relations between BI data in a simple way.

For information on creating a simple analysis process using the Analysis Process Designer, see Creating Analysis Processes [Page 389].

Data Mining

You create a model for a data mining method so that you can apply the method according to your business requirements.

For information on creating a model in the Data Mining Workbench, see Creating, Changing and Activating Models [Page 408].

3.1 Modeling Planning Scenarios

Purpose

To model your planning scenarios, BI Integrated Planning provides you with the *Planning Modeler* and the *Planning Wizard*.

Both tools are Web dynpro-based applications that have to be installed on the SAP J2EE Server. You can allow access to these applications using links or iViews in the portal. It is not necessary, therefore, to install the SAP front end locally.

Planning Modeler

You use the planning modeler to model, manage, and test all the metadata that belongs to a planning scenario.

Interface

The tab pages *InfoProvider*, *Aggregation Levels*, *Filters*, *Planning Functions* and *Planning Sequences* are structured in such a way that in the upper part of the screen you have the option to search using objects that can be selected in the system, and a table which displays the results of the search. If you select or create an entry, in the lower part of the screen the system displays the properties of the respective object and provides the user with options to edit the object.

You can modify the interface as required by hiding or showing the subareas.

To modify the table layout, you can:

- Choose *Filter On* and enter descriptions in the input-ready rows by which the table columns are filtered.
- Choose *Settings* and select table columns and define the sequence and the general settings for the table layout. When you upgrade, it cannot be guaranteed that the user-specific settings for the table views in the planning modeler will be retained, or that you will be able to reuse them if you have saved them locally.

Functions

The planning modeler provides the following functions:

- **InfoProvider selection, characteristic relationship and data slice assignments, selection, modification, and creation of InfoProvider of type aggregation level**

  You define the corresponding settings on the *InfoProvider* und *Aggregation Levels* tab pages in the planning modeler.

Tab Page	Related Information
InfoProvider	The InfoProvider defines the data basis for planning. This involves real-time InfoCubes and MultiProviders. See InfoProviders [Page 24]. For real-time InfoCubes you can define permitted combinations of characteristic values in the form of characteristic relationships and create data slices for data that you want to protect. For more information, see Characteristic Relationships [Page 26] and Data Slices [Page 30]. On the *Settings* tab page, you can set a *Key Date* as the default key date for planning. See Standard Key Date in Planning Functions [Page 62].

Aggregation Levels	An aggregation level is a virtual InfoProvider that has been especially designed to be able to plan data manually or change it using planning functions. An aggregation level represents a selection of characteristics and key figures for the underlying InfoProvider and determines as such the granularity of the planning. You can create several aggregation levels for an InfoProvider and, therefore, model various levels of planning and, for example, hierarchical structures. Note, however, that aggregation levels cannot be nested. You can change an aggregation level by selecting InfoObjects in the lower part of the screen that are to be used or not. For more information, see Aggregation Level [Page 31].

The following InfoProviders are can be used as the basis **for an input-ready query**:

- The InfoProvider is an aggregation level that is defined on a real-time-enabled InfoCube (simple aggregation level).
- The InfoProvider is an aggregation level that is defined on a MultiProvider (complex aggregation level). The following prerequisites must be fulfilled: The MultiProvider includes
  - at least one real-time InfoCube, and
  - no simple aggregation level.
- The InfoProvider is a MultiProvider that contains at least one simple aggregation level.

- **Creating and changing filters**

  With regards to the underlying InfoProvider, filter objects are global objects that restrict the dataset that is used in queries and planning functions. You require filters if you want to use a planning function in a planning sequence.

  You define the corresponding settings on the *Filter* tab page.

Tab Page	Related Information
Filter	You can restrict selected characteristics of the InfoProvider to single values, value ranges, hierarchy nodes, history, or favorites and determine whether they can be changed when you execute them. For more information, see Filter [Page 36].

- **Creating and changing planning functions and planning sequences**

  You define the corresponding settings on the *Planning Functions* and *Planning Sequences* tab pages.

Tab Page	Related Information

Planning functions	The system offers you standard planning functions. You can create the following types of planning functions: • Unit conversion • Generate combinations • Formula • Copy • Delete • Delete invalid combinations • Repost • Repost by characteristic relationships • Revaluate • Distribute by reference data • Distribute by key • Currency translation You can use FOX formulas for complex tasks or define customer-specific planning function types in ABAP using an exit. For more information, see Planning Functions [Page 39].
Planning sequences	You can determine steps for the input templates or planning functions by selecting the required aggregation level, filter, and planning function (if applicable). For more information, see Planning Sequences [Page 63].

- **Creating and changing variables**

  Variables can be used in queries and different areas of the planning model (see Variables [Page 64]). The system provides a variable wizard wherever you might want to use variables:

  - When defining characteristic relationships and data slices (*InfoProvider* tab page)
  - When defining filters (*Filter* tab page)
  - To parameterize planning functions (*Planning Functions* tab page)
  - To parameterize queries (in the BEx Query Designer)

Planning Wizard

To assist you in modeling planning for the first time, the planning wizard offers support in the form of an assistant that leads you through a simple scenario, starting with one InfoProvider.

You perform the following steps:

Step	**Related Information**

InfoProvider	You can select an InfoProvider. (You cannot, however, define characteristic relationships, data slices, and settings.)
Aggregation level	You create one or more aggregation levels.
Filter	You create one or more filters.
Planning function	You create one or more planning functions.
Test environment	The system integrates your planning model into a planning sequence. You can then execute this in the test environment.

Prerequisites

You require real-time-enabled InfoCubes as data stores. You have created these InfoCubes in the Data Warehousing Workbench. For more information, see Real-Time InfoCubes [Page 420].

Process Flow

1. You choose the appropriate InfoProvider.
2. You create one or more aggregation levels.
3. You create one or more filters.
4. You create one or more planning functions.
5. You create a planning sequence.
6. You test the planning model.

Result

You have created a planning model on the basis of which you can now run input-ready queries and automatic planning functions.

For more information, see Input-Ready Query [Page 69].

3.2 Overview of Planning with BW-BPS

Purpose

In this overview you learn how to proceed generally in order to execute planning with BW-BPS. This offers an initial overview of the required and optional steps and how these are related to each other. You will find more information on the individual steps in the corresponding sections of this documentation, which is referred to each time.

Process Flow

1. If an InfoCube with the required data is not already available in your BW system, create an in InfoCube [External] with the required characteristics and key figures. Normally you supply the InfoCube with data from the operative systems of your company. You can use this data as **actual data** as the basis of your planning. With BW-BPS you can also enter completely new **plan data**, without having to refer to existing actual data.

In both cases you need a transactional InfoCube for the plan data.

> You can find more information under Create InfoCube [External] and, in particular, under Transactional InfoCube [Page 420].

2. Create master data, master data texts, and hierarchies for the characteristics of the InfoCube.
3. Create a planning area [Page 93]. You assign the InfoCube to this planning area.

   Note that an InfoCube can be assigned to one planning area at most.

   If you specify an RFC destination in a planning area you can also access data from another BW system.
4. Create characteristic relationships [Page 110] to ensure the consistency of the plan data. This step is optional.
5. Create planning levels [Page 116] for the planning area. You include a selection of characteristics and key figures from the InfoCube in these planning levels. In this way you define on which aggregation level you are performing planning. Characteristics that you do not include in the planning level are handled by the system in the following way: When reading the data, the system aggregates using all existing values in the transaction data records. When the data is saved the values of these characteristics are replaced with the initial value.
6. Create planning packages [Page 119].

   A planning package represents the quantity of transaction data on which the planning functions and manual planning operate. In this way you distinguish the work lists of the different planners. When designing a planning application you have to consider how you want to separate work lists so that planners do not mutually overwrite plan data or mutually lock data. An alternative to working with planning packages is to use user-specific variables.

   Every planning level automatically contains a planning package; the ad hoc package. The ad hoc package can be used like a package that you have created. However, while the settings of packages created by you are saved permanently, the system resets all package settings for the ad hoc package when you exit the planning session.
7. Restrict the characteristics to your desired value ranges.

   For every characteristic, decide whether you want to carry out the restriction in the planning level or in the planning package. It is mostly advisable to restrict characteristics of general significance centrally in the planning level (for example fiscal year), while characteristics whose values describe certain subtasks, should be restricted in the package (for example planning for article 100 to 200, customer 1000, company code 2000 and 2100).

   Try to restrict the characteristic values in the planning level and package to as small an area as possible. This way, you reduce the data quantity represented by the planning package, and increase the execution speed of the planning functions.
8. For every planning level create the planning functions you require.

   Planning functions are created in the context of a planning level, and can access the characteristics and key figures that are contained in the planning level. For all planning functions, you require a parameter group (or several) in addition, which contains the concrete processing rules – for example for a function of the type revaluation, the percentage by which the values should be changed.

   You will normally always choose the function type manual planning [Page 127]. With the planning layouts [Page 128] that belong to it, you cannot only enter data manually, you can also look at existing data. Apart from manual planning, you can also create functions of various types, for example revaluation, copy, repost, and provide them with parameter groups.

9. If you want to offer employees an individual view of the planning object, then you create a planning profile [Page 123].

   You can assign planning profiles your employees. These ensure that, from the entire quantity of all planning objects (planning areas, levels, packages, functions), only those are visible that are relevant to the respective employee. This makes orientation within the planning environment [Page 90] easier, and makes sure that all employees only process the objects that are assigned to them.

   This step is optional.

10. Define planning folders or Web interfaces for manual planning. This step is optional.

11. Define planning sessions and subplans in the Status and Tracking System so that you are able to control planning processes.   This step is optional.

12. Execute planning.

    When executing planning in the planning workbench always select a planning package first (for example by double-clicking) to determine the dataspace in which the planning function should operate. Subsequently open a planning layout for data entry (in the case of manual planning), or execute a planning function by double-clicking on the desired parameter group. You can also execute the planning functions from an open planning layout. After executing the planning function, you then see the changed key figure values directly in the opened layout.

    The connection between planning packages, parameter groups, and layouts in the planning folders or Web interfaces are already fixed. End-users only have to select layouts or function keys to execute their planning tasks.

13. Save your data.

    During a planning session, you can navigate freely within the planning environment. Changes to data and planning objects are automatically stored temporarily, and must be explicitly saved when you end the session.

14. You can protect your data from unwanted changes.

    When you have finished a planning task, you can create data slices [Page 95] in the context of the planning area with which you can lock a greater area of complete data quantity of the planning area against changes.

    This step is optional.

Result

You have plan data at your disposal that you can compare with current actual data at any time, and that you can also use as the basis for decisions.

3.3 Creating and Editing Web Interfaces

Use

You create and edit Web interfaces in order to generate a BSP application that is based on the Web interface. The BSP application accesses the planning objects using a Web browser.

Prerequisites

To create a functional BSP application, you must already have created planning objects in the planning environment.

SAP delivers planning applications based on Web interfaces that have been created with the Web Interface Builder. To avoid conflicts with the delivered Web interfaces, create your Web interfaces in the **namespace** that is reserved for customers. This begins with **Y** or **Z**.

Note that the Web interface development objects delivered by SAP are part of the packages [External] delivered by SAP. We recommend that you do not change these objects as this represents a modification [External] and has organizational and technical consequences. However, if you do want to make changes, copy the delivered objects and then change the copies.

Procedure

Use the Wizard to Create a Web Interface

The Web Interface Builder has a wizard that supports you in creating a new Web interface:

1. On the *SAP Easy Access Menu* screen, choose *Business Planning and Simulation* → *Web Interface Builder* → *Customizing*. The initial screen of the Web Interface Builder appears.
2. Choose (*Web Interface* → *Create with Wizard)*. The start page of the wizard appears.
3. In the following steps of the wizard, you put together the elements that you require:
   a. *Start*
   b. *Specify Name for Web Interface*
   c. *Select Planning Areas*
   d. *Select Planning Levels*
   e. *Create Individual BSP Pages*
   f. *Complete*

   For each step, the wizard offers information about the possible settings. In the first step, you only have the option of switching to the next step. However, when you have carried out more than one step you can switch between the steps that you have already performed in any order. This allows you to change the settings you made for an earlier step later in the process.
4. When you have made all the settings and have arrived at the last step, choose *Complete*. The system creates a new Web interface in accordance with your settings.
5. To save the Web interface, choose (*Web Interface* → *Save)*.

   A Web interface that was created using the wizard is no different to a manually created Web interface. Therefore, the subsequent processing of a Web interface created using the wizard is the same as the subsequent processing of a manually created Web interface.

Create a Web Interface Manually

1. On the *SAP Easy Access Menu* screen, choose *Business Planning and Simulation* → *Web Interface Builder* → *Customizing*. The initial screen of the Web Interface Builder appears.
2. Choose 🗋 (*Web Interface* → *Create)*.

   The system displays a dialog box in which you can determine basic properties of the Web interface. Depending on what you enter here, the system automatically creates certain elements for you for the new Web interface.
3. In the *Number of Pages* field, enter whether the generated application should consist of one or more HTML pages.
4. Select which of the available *Standard Elements* you want the system to create on each page of the application.

   You can manually correct the selection that you make in the two last steps when you create a Web interface by adding further elements or removing elements that are not required.
5. Edit the new Web interface as described in the following *Edit a Web Interface* section.

Edit a Web Interface

1. On the *SAP Easy Access Menu* screen, choose *Business Planning and Simulation* → *Web Interface Builder* → *Customizing*. The initial screen of the Web Interface Builder appears.
2. Choose 📂 (*Web Interface* → *Open)*.
3. From the list of available Web interfaces, select the one you require. The system displays the hierarchical element structure of the Web interface.
4. Edit the components of the Web interface (see Components for Web Interfaces [Page 298]):
   - To add a new component to the Web interface, choose:
     - *Create Page* in the context menu of an element of type “application”. A dialog box appears where you can enter the name of the page.
     - *Create Subcomponents* in the context menu of an element of type “page” or “container”. A dialog box appears with an overview of the subcomponents available in the system. Select the required element type.
   - To edit the attributes of an existing component, double-click on its name in the element structure or choose *Change Attributes* in the context menu of the component.

   The system displays the attributes of the component in the attribute editor.

   The different attributes of the components are documented in the system. For more detailed information about a specific attribute, choose the appropriate attribute in the attribute editor and then use F1.
5. When you have created all required elements and have set their attributes, save the Web interface.

6. Choose (*Edit → Check Consistency)*. If applicable, the system highlights problems such as, for example, incorrect references between elements of the Web interface.
7. Choose (*Edit → Generate)*. The system generates the objects specified in the *Result* section below.
8. Choose *Edit → Display Preview* or *Display in External Browser* to test the application.

   If you are in the test phase when you create or change a Web interface and you want to check your results as quickly as possible, you can use the quick preview in the Web interface instead of the last steps two that are mentioned. To do this, choose *Goto → Settings*. Under *Preview* in the *Settings* dialog box, choose the *Quick Preview w/o Generating* option. This means that you can display the Web interface in the preview window without having to generate the BSP application first (which can be time-consuming).

Result

As a result of the subsequent generation in the Web Interface Builder, the system has generated the following objects:

- For each element of type “page” that you have created in the Web interface, the system generates a BSP page which contains the elements required for the Web interface. This page is stored in the system. At the runtime of the application, the SAP Web Application Server generates a HTML-format page from every requested BSP page. This is displayed in the browser.
- In addition to that, additional pages are generated for special purposes (for example, a page which is displayed when exiting the Web application).

When using the SEM-BPS design (see Design Templates for Web Interfaces [Page 299]): If you have specified in the properties of the Web interface that you want to use a customer-designed class, the system generates this class. The generated BSP pages then use the processing logic implemented in this class (and not the standard class of the Web Interface Builder).

See also:

SAP Web AS Architecture [External]

Creation of Web Applications with Business Server Pages [External]

3.4 Creating an Analysis Process

Procedure

You are in the SAP Easy Access SAP Business Information Warehouse. In the SAP menu, choose *Special Analysis Processes → Analysis Process Designer*. In order to create and execute a simple analysis process with transformation, proceed as follows:

1. Choose *Create.*

2. Select an application from the dropdown menu and select ✓ *Okay.* Your analysis process will be assigned to the appropriate folder on the left side of the screen.
3. Specify a description.
4. Drag a data source into the work area and make the following detailed settings in the dialog box that appears.
5. Drag a transformation into the work area. By double clicking on the transformation node, you can make the settings.
6. Drag a data target into the work area and make the following settings in the dialog box that appears.
7. Connect the nodes with the mouse.
8. To make an explicit field assignment, double click on the data flow arrow that connects the nodes.
9. Save your analysis process. Specify a technical name.
10. Before you execute your analysis process, you have the option of checking the data and of calculating intermediate results for performance optimization. See also Checking Data [Page 389].
11. Choose *Check.*
12. Choose *Activate.*
13. Execute the analysis process. The data is written to the data target and the log is displayed.

3.5 Creating, Changing, and Activating a Model

Use

You create a model for a data mining method so that you can apply the method according to your business requirements. You use model fields in a model to specify what is to be predicted and which data should form the basis of the prediction.

You can create a data mining model using the *Data Mining Workbench* or the *Analysis Process Designer* (APD). Once you have created and saved the model to meet your requirements, you can activate it.

Prerequisites

You must have been assigned to the role *Customer Behavior Analysis* (SAP_BW_CUSTOMER_BEHAVIOR) and you must have chosen *Customer Behavior Analysis* → *Customer Behavior Modeling* in the user menu.

Creating a Model in the Data Mining Workbench

1. Position the cursor on a data mining method and use the right-hand mouse to choose *Create* in the context menu.
2. In the step *Create Model*, enter a name and a description for the model. The method name for which you are creating a model is displayed. You have three options for model field selection:

- To create the model fields manually, select the *Manual* option.
- If you want to create a model that is similar to an existing model created previously, you can copy it choosing the *Use Model as Template* option. You can make minor changes to the copied version manually to suit your requirements.
- To create a model from a query, choose *Model Field Selection* and select the query which you want use as a source for model fields.

  Selecting a query at this point will assist you in creating the model. It is therefore **recommended** to enter the same query that you would like to use subsequently, while training the model. However, this is **not essential**. You can also use any other query as a template for your model.

The InfoObjects contained in the selected query are available in the next step as model fields.

3. In the step *Select InfoObjects*, select from the query those InfoObjects that you would like to use as model fields.

   If you would like to use other fields from the query as calculated or restricted key figures in your model, you need to include an SAP BW dummy InfoObject as a model field for each one. You can then assign the corresponding field to this model field in the *Change* mode.

4. In the step *Edit Model Fields*, specify the attributes for each field.

   - The description you give the model field does not necessarily have to be identical with that of the InfoObject.
   - The system automatically copies the attributes *Data Type* and *Length* from InfoObject (these cannot be modified).
   - The *value types* valid for a model field are dependent on the method that you are creating the model for and on the data type of the model field.

     The value type specified for a model field determines which entries can be made as *Field Parameters* and *Field Values*.

     The attributes for a model field that are listed below do not apply in the data mining method *Association Analysis*. No prediction is involved with this method. Instead, the association rules are determined by training and form the result. Consequently, the settings for the field parameters and field values do not apply.
   - Set the *Prediction Variable* indicator for the model field for which the subsequent prediction is to be made. Select as a prediction variable that model field for which you wish to gain more information (via the model).

     With the data mining method *Clustering*, the cluster is always the prediction variable. Consequently, you cannot specify a prediction variable for this method.
   - The *field parameters* are dependent on the value type of the model field and on the data mining method.

You cannot select any parameters for model fields where the value type KEY has been set.

- Under *Field* Values, you can specify how the system should interpret specific values that can be taken by a model field but have no bearing on the result.

5. In the *Model Parameters* step, enter the parameters that are valid for the entire model. The model parameters are dependent on the data mining method.
6. *Save* the model.

Result

You have performed all necessary steps for the creation of a model. The created model appears in the tree beneath the relevant method.

Changing the Model

You can make changes to the model that you have created.

1. Position the cursor on a model that you wish to change and use the right-hand mouse to choose *Change* in the context menu.
2. In the *Model Fields* tab page, make your changes to the model fields. You can change the attributes for the model fields or add more model fields.
3. In the *Model Parameters* tab page, make your changes to the model parameters.
4. Save your changes.

Activating the Model

Once a model meets your requirements, you can activate it. The active model is then used for creating other versions. This means that, when you change a model that has been activated, the active version remains unchanged and the changes are saved in a *Revised* version. The active version is only overwritten when you activate the modified version.

You can only train or valuate a model or use it for the prediction if the model has been activated.

If a model has a modified version, the model name in the tree is marked in blue.

To activate a model, proceed as follows:

1. Position the cursor on a model that you wish to activate and use the right-hand mouse to choose *Activate* in the context menu.

   The version displayed under *Model Information* is changed to *Active*.

2. Make any necessary changes to the model and save your changes.

   The version displayed under *Model Information* is changed to *Revised*.

3. To navigate between the active and modified versions, place the cursor in the model and choose ⇄.

4 Core Development Tasks

This section forms the core of the Developer's Guides and describes the central areas of the development phase.

4.1 Developing User Interfaces

Purpose

BW-BPS: Web Interface Builder of BW-BPS

You use the Web interface builder to create Web-enabled planning applications in the form of business server page applications (BSP applications).

For more information, see Web Interface Builder [Page 285].

4.2 Developing Business Logic

Purpose

BI Integrated Planning: Modeling Planning Scenarios

You can define your own planning-specific metadata objects with the planning modeler and the planning wizard. Both tools are Web dynpro-based applications that have to be installed on the SAP J2EE Server.

For more information, see Modeling Planning Scenarios [Page 21].

BI Integrated Planning: Planning Functions

Planning functions allow system-based processing or generation of data. A planning function describes the ways in which the transaction data for an aggregation level can be changed.

For more information see Planning Functions [Page 39].

You can implement your own planning function types in order to implement specific processes and then to apply them to transaction data.

For more information, see Implementing Planning Function Types [Page 71].

BI Integrated Planning: Input-Ready Query

You can use input-ready queries to create applications for manual planning, that extend from simple data recording to complex planning applications.

For more information, see Input-Ready Queries [Page 69]

Planning with BW-BPS

You can develop your own planning application with BW-BPS. The areas of application range from simple data entry to more complex planning scenarios with data extraction, automatic planning preparation, manual data entry, controlling planning process, and retracting plan data.

For more information on the individual steps, see Overview of Planning with BW-BPS [Page 88].

Analysis Process Design

You can define analysis processes that explore and identify hidden or complex relationships between BI data. You can create data mining models, use existing data mining methods, or perform various other transformations of data for this purpose.

For more information, see Analysis Process Designer [Page 362].

4.2.1 Business Planning

Purpose

Business planning with SAP NetWeaver Business Intelligence allows business experts to accelerate the decision-making process, predict future trends on the basis of historic analyses, and provide all decision makers with a central point of access to data and information.

To create and use planning scenarios or other applications, business planning with SAP NetWeaver Business Intelligence offers the following planning tools:

- **BI Integrated Planning**, a solution that is completely integrated into the BI system
- **BW-BPS** (Business Planning and Simulation)

  We recommend that you use the new BI Integrated Planning functionality when you implement new scenarios.

Business planning is part of the IT scenario *Business Planning and Analytical Services*. For more information, see Business Planning and Analytical Services [External].

Integration

Note the following points regarding the two planning tools, BI Integrated Planning and BW-BPS:

- Both planning tools use the same data basis and can be operated in parallel in one system. It is not necessary to migrate existing planning applications.
- Some functions (such as, lock procedure and formulas) are used by both BW-BPS and BI Integrated Planning.
- In BI Integrated Planning, most of the BEx and OLAP analysis functions are available for planning applications. In comparison to BW-BPS, you require fewer objects (for example, you use the same variables in analysis and planning) and tools (Query Designer and Web Application Designer).
- Since a large number of BW-BPS concepts are also used in BI Integrated Planning (for example, planning levels, planning functions, planning sequences, characteristic relationships, data slices), switching to BI Integrated Planning from BW-BPS requires a minimal effort.

Features

Business planning with SAP NetWeaver Business Intelligence covers processes that collect data from InfoProviders, queries or other BI objects, convert them using various methods, and write back new information to BI objects (such as InfoObjects or DataStore objects).

Using the Business Explorer for BI Integrated Planning allows you to build integrated analytical applications that encompass planning **and** analysis functionality.

For more information about planning tools, see:

Business Planning and Simulation (BW-BPS) [Page 87]

BI Integrated Planning [Page 18]

4.2.1.1 BI Integrated Planning

Purpose

BI Integrated Planning provides business experts with an infrastructure for realizing and operating planning scenarios or other applications. Planning covers a wide range of topics, ranging from the simple input of data to complex planning scenarios. In contrast to BW-BPS (Business Planning and Simulation), this solution is fully integrated into the BI system.

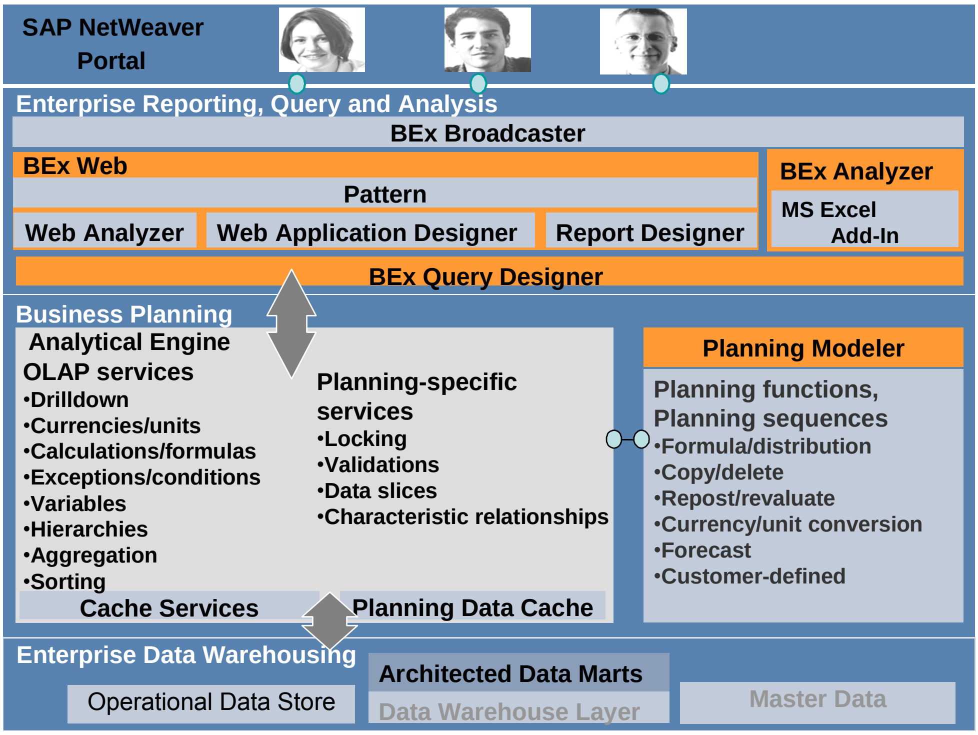
Integration

The following tools are available for modeling planning scenarios:

- To create the data basis, use the *Data Warehousing Workbench*.
- To model all planning-specific metadata objects, use the *Planning Modeler*. The planning modeler is a Web-based application that is installed on the J2EE Engine.
- To define an input-ready query for manually entering plan data, use the *BEx Query Designer*.
- To configure Web templates, use the *BEx Web Application Designer*; to configure Excel applications, use the *BEx Analyzer*.

You can use the Data Warehousing Workbench and the various Business Explorer tools to analyze, plan and enter data.

The following graphic provides an overview of the architecture:



Features

The planning model incorporates:

- Data (stored in InfoCubes)
- (Structuring) views of data (aggregation levels, MultiProvider, characteristic relationships, if required)
- Methods to change data (planning functions, planning sequences, manual planning in the form of input-ready queries, in addition to process chains)
- Utilities (filters that can be used in queries and planning functions; variables used to parameterize objects that can usually be used where selections are used, for example, in data slices)
- Concepts for (where applicable, time-restricted) central protection of data (data slices)

For more information about transporting planning model objects, see Transport of Planning Objects [Page 86].

The most important concepts and terminology for the BI Integrated Planning planning model are discussed in the next section.

Data Basis and Lock Concept

Real-time InfoCubes are used to store data.

To ensure that one user only is able to change data, “their” data is locked and cannot be changed by other users. Depending on the expected load (determined by the number of users working in parallel and the complexity of the selection), you can specify one of several lock processes as the default. The lock algorithm is used by BW-BPS and BI Integrated Planning.

Modeling in the Planning Modeler

In the planning modeler, you edit the following objects of the planning model:

- **Aggregation levels**

  To determine the level on which data can be entered or changed (manually through user input or automatically by a planning function), an InfoProvider of type aggregation level has to be defined. An aggregation level consists of a subset of the characteristics and key figures of a MultiProvider or real-time InfoCube. Real-time InfoCubes are used to store data.

- **Characteristic relationships**

  You can use characteristic relationships to model semantic relationships between characteristics (such as product group and product). In this way you check, for example, whether a particular combination of characteristics can be generated (if this combination is permitted) or whether a cell is input ready. Characteristic relationships are created for an InfoCube.

- **Data slices**

  You use data slices to protect whole areas of data globally against changes (for example, current values or historic values).

- **Planning functions**

  Planning functions allow system-based processing or generation of data. The BW-BPS function types are provided as standard. Functions can be executed immediately (using the pushbutton) or in the background as a planning sequence. You can also define your own function types.

- **Planning sequences**

  A planning sequence is a sequence of planning functions and manual input templates that are executed sequentially. You can also schedule planning sequences to be processed in the background as a step in a process chain.

- **Filter**

  A filter describes a section of a dataset which is processed, for example, in a query or a planning function. (For example, calendar year 2004 – 2005, customer group XY).

- **Variables**

  Variables can be used in various places; in the filter for selections of characteristic values that can be parameterized, to parameterize planning functions or planning sequences.

Input-Ready Query

A query that is defined for an InfoProvider of type aggregation level. It is input ready and can be used for manual planning. Whether a particular cell is input ready depends on the drilldown, specifically whether characteristic relationships and data slices are permitted for the cell.

Complex Planning Applications

In the BEx Analyzer and Web Application Designer you can build planning applications that support both manual and automatic data entry and changes.

See also:

Authorizations for BI Integrated Planning [External]

4.2.1.1.1 Modeling Planning Scenarios

Purpose

To model your planning scenarios, BI Integrated Planning provides you with the *Planning Modeler* and the *Planning Wizard*.

Both tools are Web dynpro-based applications that have to be installed on the SAP J2EE Server. You can allow access to these applications using links or iViews in the portal. It is not necessary, therefore, to install the SAP front end locally.

Planning Modeler

You use the planning modeler to model, manage, and test all the metadata that belongs to a planning scenario.

Interface

The tab pages *InfoProvider*, *Aggregation Levels*, *Filters*, *Planning Functions* and *Planning Sequences* are structured in such a way that in the upper part of the screen you have the option to search using objects that can be selected in the system, and a table which displays the results of the search. If you select or create an entry, in the lower part of the screen the system displays the properties of the respective object and provides the user with options to edit the object.

You can modify the interface as required by hiding or showing the subareas.

To modify the table layout, you can:

- Choose *Filter On* and enter descriptions in the input-ready rows by which the table columns are filtered.
- Choose *Settings* and select table columns and define the sequence and the general settings for the table layout. When you upgrade, it cannot be guaranteed that the user-specific settings for the table views in the planning modeler will be retained, or that you will be able to reuse them if you have saved them locally.

Functions

The planning modeler provides the following functions:

- **InfoProvider selection, characteristic relationship and data slice assignments, selection, modification, and creation of InfoProvider of type aggregation level**

  You define the corresponding settings on the *InfoProvider* und *Aggregation Levels* tab pages in the planning modeler.

Tab Page	Related Information

InfoProvider	The InfoProvider defines the data basis for planning. This involves real-time InfoCubes and MultiProviders. See InfoProviders [Page 24]. For real-time InfoCubes you can define permitted combinations of characteristic values in the form of characteristic relationships and create data slices for data that you want to protect. For more information, see Characteristic Relationships [Page 26] and Data Slices [Page 30]. On the *Settings* tab page, you can set a *Key Date* as the default key date for planning. See Standard Key Date in Planning Functions [Page 62].
Aggregation Levels	An aggregation level is a virtual InfoProvider that has been especially designed to be able to plan data manually or change it using planning functions. An aggregation level represents a selection of characteristics and key figures for the underlying InfoProvider and determines as such the granularity of the planning. You can create several aggregation levels for an InfoProvider and, therefore, model various levels of planning and, for example, hierarchical structures. Note, however, that aggregation levels cannot be nested. You can change an aggregation level by selecting InfoObjects in the lower part of the screen that are to be used or not. For more information, see Aggregation Level [Page 31].

The following InfoProviders are can be used as the basis **for an input-ready query**:

- The InfoProvider is an aggregation level that is defined on a real-time-enabled InfoCube (simple aggregation level).
- The InfoProvider is an aggregation level that is defined on a MultiProvider (complex aggregation level). The following prerequisites must be fulfilled: The MultiProvider includes
  - at least one real-time InfoCube, and
  - no simple aggregation level.
- The InfoProvider is a MultiProvider that contains at least one simple aggregation level.

- **Creating and changing filters**

  With regards to the underlying InfoProvider, filter objects are global objects that restrict the dataset that is used in queries and planning functions. You require filters if you want to use a planning function in a planning sequence.

  You define the corresponding settings on the *Filter* tab page.

Tab Page	**Related Information**
Filter	You can restrict selected characteristics of the InfoProvider to single values, value ranges, hierarchy nodes, history, or favorites and determine whether they can be changed when you execute them. For more information, see Filter [Page 36].

- **Creating and changing planning functions and planning sequences**

You define the corresponding settings on the *Planning Functions* and *Planning Sequences* tab pages.

Tab Page	Related Information
Planning functions	The system offers you standard planning functions. You can create the following types of planning functions: • Unit conversion • Generate combinations • Formula • Copy • Delete • Delete invalid combinations • Repost • Repost by characteristic relationships • Revaluate • Distribute by reference data • Distribute by key • Currency translation You can use FOX formulas for complex tasks or define customer-specific planning function types in ABAP using an exit. For more information, see Planning Functions [Page 39].
Planning sequences	You can determine steps for the input templates or planning functions by selecting the required aggregation level, filter, and planning function (if applicable). For more information, see Planning Sequences [Page 63].

- **Creating and changing variables**

  Variables can be used in queries and different areas of the planning model (see Variables [Page 64]). The system provides a variable wizard wherever you might want to use variables:

  - When defining characteristic relationships and data slices (*InfoProvider* tab page)
  - When defining filters (*Filter* tab page)
  - To parameterize planning functions (*Planning Functions* tab page)
  - To parameterize queries (in the BEx Query Designer)

Planning Wizard

To assist you in modeling planning for the first time, the planning wizard offers support in the form of an assistant that leads you through a simple scenario, starting with one InfoProvider.

You perform the following steps:

Step	Related Information
InfoProvider	You can select an InfoProvider. (You cannot, however, define characteristic relationships, data slices, and settings.)
Aggregation level	You create one or more aggregation levels.
Filter	You create one or more filters.
Planning function	You create one or more planning functions.
Test environment	The system integrates your planning model into a planning sequence. You can then execute this in the test environment.

Prerequisites

You require real-time-enabled InfoCubes as data stores. You have created these InfoCubes in the Data Warehousing Workbench. For more information, see Real-Time InfoCubes [Page 420].

Process Flow

1. You choose the appropriate InfoProvider.
2. You create one or more aggregation levels.
3. You create one or more filters.
4. You create one or more planning functions.
5. You create a planning sequence.
6. You test the planning model.

Result

You have created a planning model on the basis of which you can now run input-ready queries and automatic planning functions.

For more information, see Input-Ready Query [Page 69].

4.2.1.1.1.1 InfoProvider

Use

InfoProviders that contain real-time InfoCubes provide the data basis for BI Integrated Planning. Aggregation levels are a type of virtual InfoProvider and are created on the basis of a real-time InfoCube, or a MultiProvider that contains InfoCubes of this type. Aggregation levels are specifically designed so that you can plan data manually or change it using planning functions.

For more information about the types of InfoProvider, see:

- Real-Time InfoCube [Page 420]
- MultiProvider [External]

Integration

In the *Modeling* functional area of the *Data Warehousing Workbench*, you create InfoProviders as the data basis for BI Integrated Planning.

For more information, see InfoProviders [External], Creating InfoCubes [External], and Creating MultiProviders [External].

In the *Planning Modeler*, you select the InfoProvider that you want to use as the data basis for BI Integrated Planning. On the *Aggregation Levels* tab page, you create one or more aggregation levels for this InfoProvider.

For more information, see Aggregation Levels [Page 31].

Prerequisites

You have created a suitable InfoProvider as the data basis for BI Integrated Planning and filled it with data.

Features

InfoProvider Selection

You can restrict the number of InfoProviders displayed by specifying the technical name or description or by making an entry for last changed by.

You can change, check and save the selected InfoProviders.

InfoObjects

On the *InfoObjects* tab page, the system displays the InfoObjects that belong to the InfoProvider (see InfoObject [External]). They are listed in the following tables:

- *Dimensions*, with the characteristics assigned to them
- *Navigation Attributes* for the characteristics contained in the InfoProvider
- *Key figures*

Under *Settings*, you can choose to display additional columns.

Characteristic Relationships and Data Slices

In **change mode** you can define the permitted combinations of characteristic values in the form of characteristic relationships and create data slices for the data that you want to protect for real-time-enabled InfoCubes.

For more information, see Characteristic Relationships [Page 26] and Data Slices [Page 30].

Default Key Date for Planning

On the *Settings* tab page in **change mode**, you can set a *Key Date* as the default key date for planning. If time-dependent objects, such as attributes or hierarchies, are used in objects of the planning model, you can always refer to the default key date for planning. In this way, you can ensure that a uniform key date is used in the planning model. The objects in the planning model that are relevant for this are characteristic relationships, data slices and parameters of planning functions.

For more information see Standard Key Date in Planning Functions [Page 62].

4.2.1.1.1.1.1 Characteristic Relationships

Use

Characteristic relationships are used to relate characteristics that correspond to each other with regard to content. You can use characteristic relationships to define rules in order to check permitted combinations of characteristic values for each real-time enabled InfoCube. You can also define rules that the system uses to derive values from characteristics for other characteristics. This is useful, for example, when the derivable characteristics are to be available for further analysis.

You can define characteristic relationships for the master data of a characteristic (type **attribute**), a hierarchy (type **hierarchy**), a DataStore object (type **DataStore**) or an exit class (type **Exit**).

Integration

If characteristic relationships are defined in relation to attributes and hierarchies, the system offers those attributes and hierarchies that were created in the BI system for a characteristic (see Tab Page: Attributes [External] and Tab Page: Hierarchy [External] and Hierarchy [External]).

Characteristic relationships are created on a real-time enabled InfoCube. They then affect all InfoProvider relevant for planning that reference this InfoCube.

Each input-ready query and each planning function then automatically takes the characteristic relationships into account:

- This means that cells for invalid characteristic combinations are not input ready in an input-ready query and new data records with invalid characteristic combinations cannot be created.
- Planning functions constantly check whether new characteristic combinations are valid according to the characteristic relationships. In case of invalid combinations, the system informs you with an error message.

The possible characteristic derivations take place when the delta records are determined in the delta buffer. The possible source characteristics are the characteristics of the real-time enabled InfoCube that are filled by characteristics from the participating aggregation levels. If characteristic relationships are changed, the data records in the InfoCube have to adapted to the new structure. The planning function *Reposting Characteristic Relationships* is used for this purpose.

Prerequisites

The following prerequisites must be fulfilled in order to define characteristic relationships:

- The InfoProvider must be a real-time enabled InfoCube. The characteristic relationships defined for a real-time enabled InfoCube are also effective in the MultiProviders that contain a real-time enabled InfoCube. See InfoProviders [Page 24].
- In characteristic relationships of the type attribute, the target characteristic must be defined as an attribute of the basic characteristic and must itself be contained in the InfoCube.

- In characteristic relationships of the type **hierarchy**, the target characteristic must be contained in a hierarchy and in the InfoCube. The hierarchy is mainly intended for modeling a derivation relationship; thus the hierarchy cannot contain a leaf or an inner node more than once. Link nodes are also not permitted.
- With characteristic relationships of type **DataStore**, only standard DataStore objects are permitted. Thus you can use all methods for managing and monitoring available in the Data Warehousing Workbench.

Features

Definition of a Characteristic Relationship

Characteristic relationships are created on a real-time enabled InfoCube. A characteristic relationship comprises a set of steps that link characteristics and are numbered sequentially. Each of these relations links a set of characteristics. These relations represent the smallest units of a characteristic relationship.

Behavior of Combination Checks with and Without Derivation

Relations can only be used to check characteristic combinations or can be used for a characteristic derivation. You set this behavior in the definition of a relation. You can link several relations of type *Derivation* if the targets of one relation are the sources of another relation. Redundancy should be avoided here so that the relations actually represent the smallest unit of the characteristic relationships.

At runtime, the system determines which relationships in the InfoProviders that are relevant for planning are used.

- **Combination check**: A relation is only used in an aggregation level when every characteristic of the relation occurs in the aggregation level. With derivations, these are the source and target characteristics. In this case, nothing is derived and only a combination check is executed.
- **Characteristic derivation**: Derivation does not take place within one aggregation level. Derivations are only performed for records of the real-time enabled InfoCubes. First the system determines the set S of characteristics that are filled by the aggregation level involved. If all the source characteristics are included in the set S, the system applies the derivation relations in the next step. The target characteristics of these derivations can then serve as sources in the steps that follow. Thus the system performs the maximum possible derivation in the InfoCube. If characteristic values that were already derived are changed again in subsequent steps, the derivation is incorrect. The system produces an error message.

Types of Characteristic Relationships

The following types of characteristic relationships exist:

Type	More Information
Attribute	You can select an attribute of the basic characteristic as the target characteristic (for example, the characteristic *currency* is an attribute of the characteristic *controlling area*). The existing combinations of characteristics and attribute values are always permitted combinations.

Hierarchy	All characteristics are available as source or target characteristics if they have been set as *External Characteristics in Hierarchy* in InfoObject maintenance. In addition to the hierarchy basic characteristic, the hierarchy must include at least one other characteristic. Only one characteristic is permitted as a source and/or target characteristic (here the superordinate characteristics are not counted in compounded characteristics). The permitted combinations are taken from the hierarchy structure. A hierarchy can be used in multiple relations: in one step, you derive a characteristic that is on the next higher level from the hierarchy basic characteristic; in the second step you take the derived characteristic and derive the characteristic on the next level. Depending on the property of the hierarchy, the hierarchies used can be parameterized with the appropriate variables.
DataStore	The data records located in the DataStore define the valid characteristic combinations and are used for characteristic derivation. **Only Combination Check**: All InfoObjects from the DataStore object (except for key figures) can be selected. **With Derivation**: The keys of the DataStore object have to be selected as source characteristics. Target characteristics can be InfoObjects from the data part of the DataStore object (except for key figures). The keys for the DataStore objects can be restricted in any case; the restricted part is then used for the combination check or derivation. The restrictions can be parameterized with variables that must be replaceable without the dialog. We recommend that you use only small DataStore objects (with few characteristics, few records).
Exit	The valid characteristic combinations and derivable characteristic values result from the customer-specific implementation of the specified exit class. The exit class must implement the interface 'IF_RSPLS_CR_EXIT'. Only these types of classes are offered for editing in maintenance. We recommend that you derive your own class from the example class 'CL_RSPLS_CR_EXIT_BASE'. You then only have to implement the methods ‘CHECK’, ‘DERIVE’ and ‘CREATE’. The class 'CL_RSPLS_CR_EXIT_BASE' itself can be used directly, but it does not execute an action. Also note the commented example source text in the template class; an infrastructure for buffering is provided there, for example.

In addition to the characteristic relationships above, which you can edit, there are other predefined characteristic relationships for the BI time characteristics in the system that are always active: when there are redundant time characteristics in the real-time enabled InfoCube, for example (such as 0FISCPER and the pair 0FISCPER3 and 0FISCYEAR), the system also applies these relations between time characteristics to the combination check or derivation as well, as it does other relations.

Note that the initial characteristic value (# *not assigned*) plays a special role within a relation. This has to do with the fact that characteristics that do **not** occur in an aggregation level (and are **not** derivable) are updated with the initial value.

- Combination check without derivation:

Let us assume that there is a relation between the characteristics *Product* and *Assortment*; usually there is no derivation relationship between the two. In aggregation levels with *Product* and without *Assortment*, *Assortment* is updated with the initial value. Thus these types of combinations are always valid; they cannot be forbidden. The same applies to combinations with the initial value for *Product*.

- Combination check with derivation

Let us assume there is a relation between the characteristics *Cost Center* and *Profit Center*; here *Profit Center* can be derived from *Cost Center.* In an aggregation level with *Profit Center* and without *Cost Center*, however, *Cost Center* is also updated with the initial value. These combinations cannot be forbidden. The reverse is not valid due to the possible derivation of *Profit Center* from *Cost Center.*

Activities

1. You are in the *InfoProvider* tab page of the *Planning Modeler.* Choose the required InfoProvider of type real-time enabled InfoCube. In the lower screen area, the system displays the tab pages *Characteristic Relationships, Data Slices* and *Settings*.
2. To create or change characteristic relationships for the selected InfoProvider, choose *Change*.
3. You set whether the system is to perform only combination checks and proposals, or whether a derivation of the target characteristic from the basic characteristic is also to be performed. For each step, choose *with* or *without derivation*.
4. Select the **type of characteristic relationship**.
5. Specify the **basis** of the relationship. This is different for each type of relationship:
   - **Attribute**: a master-data bearing characteristic
   - **Hierarchy**: the hierarchy basic characteristic
   - **DataStore**: a DataStore object
   - **Exit**: an exit class.
6. The system shows further settings options depending on the relationship type selected. Depending on the type of step, you select the check characteristics or source and target characteristics.
   - In the case of attribute and hierarchy relationships such as DataStore objects, you can select check characteristics.
   - In the case of an exit class, you can select source and target characteristics.
7. The system completes your information automatically on the *Characteristic Relationships* tab page. Before you select another InfoProvider or leave the tab page, save the defined characteristic relationship.

4.2.1.1.1.1.2 Data Slices

Use

Data slices are a concept for protecting the main data of a real-time enabled InfoCube against changes. This protection affects input-ready queries and all planning functions that use this InfoCube.

If you want to ensure that certain plan versions can no longer be changed after a certain point in time, for example, and current data is not overwritten, you can use a data slice that contains these plan versions.

Prerequisites

Data slices are created on a real-time enabled InfoCube. They then affect all InfoProviders from planning that include this InfoCube. See InfoProviders [Page 24].

Features

There are two types of data slice:

- The data slice is based on a **selection**. Here you determine the restrictions for the characteristics that you wish to protect against changes.
- The data slice is based on an **exit class**. In the exit class, you can implement a customer-specific logic to protect data records.

In general, the following rules apply to data slices:

- If no data slice is defined for a real-time enabled data slice, any valid characteristic combination can be posted in this InfoCube (also see Characteristic Relationships [Page 26].
- Every data record that is part of the selection of a data slice is protected against changes. The associated cells in input-ready queries are **not** changeable. This type of record **cannot** be changed and saved using planning functions. The data slices cumulate in effect.
- If a real-time enabled InfoProvider contains a data slice that includes no characteristic value restrictions at all, the data slice acts as a lock for postings of all types in the entire real-time enabled InfoProvider.
- After you have created a data slice it is activated automatically. The settings made in the definition of the data slice have an immediate effect on the ability to update data. You can deactivate an existing data slice at any time (status *inactive*). Then this data slice is no longer taken into account.

Activities

1. You are on the *InfoProvider* tab page of the *Planning Modeler.* Choose the required InfoProvider of type real-time enabled InfoCube. In the lower screen area, the system displays the tab pages *Characteristic Relationships, Data Slices* and *Settings*.
2. To change data slices, choose *Change*.
3. Change to the *Data Slices* tab page as needed.
4. To create a data slice, choose *Create*. The system marks the rows for the data slice that is to be created.
5. In the *Data Slice Description* enter a text for the data slice to be created.

6. Determine whether the data slice is based on a selection or on an exit class.
7. If the data slice is to be based on a **selection**, set the required characteristic values in the *Change Characteristic Selections* screen area.
   - Select the characteristic to be restricted.
   - Choose the symbol in the last column of the selected row. The dialog box for determining characteristic restriction appears.
   - Select one or more values from the value list. The selection can also contain variables as long as they do not send variables at runtime.
   - Choose *Add* and save the affected selection with *OK*.
8. If the data slice is to be based on an **exit class**, enter the name of the exit class.
   - Under *Restricted*, choose the characteristics that you need in the exit. You will only get the current values for these characteristics in the exit. If you are also interested in the initial values in the exit, set the indicator *also #*. This is the default setting. If this indicator is **not** set for a characteristic, the exit is not called for those aggregation levels that do **not** contain the characteristic, for example, because the affected characteristic value would be initial in this case.
   - The exit class must implement the **interface** 'IF_RSPLS_DS_EXIT'. Only these types of classes are offered for editing in maintenance. We recommend that the customer class inherit from the **template class** 'CL_RSPLS_DS_EXIT_BASE'. The template class itself is can be run directly, but it does not execute an action. Re-implement the **method** ‘IS_PROTECTED’. Also note the commented example source text in the template class; an infrastructure for buffering is provided there, for example.
9. If the data slice should **not** be active at first, set the associated indicator in the field *inactive*.

4.2.1.1.1.2 Aggregation Level

Use

Aggregation levels are used as InfoProviders for planning: with an aggregation level, you model levels whose data can be changed manually using input-ready queries or automatically using planning functions.

An aggregation level is set using a set of characteristics and key figures from the underlying InfoProvider. The key figures included in the aggregation level are aggregated using the characteristics that are not included in the aggregation level.

In the simplest case, an aggregation level is located on a real-time enabled InfoCube. For more information on the functioning principle of aggregation and saving the changed data records for an aggregation level by means of a simple example, see Simple Aggregation Level [Page 34].

Aggregation levels can also be created on MultiProviders.

Integration

You can create multiple aggregation levels for an InfoProvider. Use the *Planning Modeler* or the *Planning Wizard* for this.

In the *Modeling* functional area of the Data Warehousing Workbench, the system also displays the aggregation levels (symbol) and the underlying InfoProviders in the InfoProvider overview. When you double-click on the aggregation level, you can branch to the Planning Modeler and edit the selected aggregation level.

Prerequisites

In the *Planning Modeler* or *Planning Wizard* you have selected (and if necessary edited) an InfoProvider to act as the basis of the aggregation level. This InfoProvider includes at least one real-time-enabled InfoCube. For more information about the corresponding processing step, see InfoProvider [Page 24].

Features

Simple Aggregation Level

A real-time enabled InfoCube is the basis of a simple aggregation level. You can find a simple example under Simple Aggregation Level [Page 34].

Complex Aggregation Level

A MultiProvider that includes at least one real-time enabled InfoCube, but no simple aggregation level, is the basis of a complex aggregation level.

> You want to copy current data from an actual InfoCube to a plan InfoCube with a planning function of type *Copy*. To do this, you use an aggregation level based on a MultiProvider that includes the plan and actual InfoCubes.

Aggregation levels, like MultiProviders, cannot be nested.

With a complex aggregation level, note how data records from the InfoProviders included in the MultiProviders are embedded in the MultiProviders (and thus also the aggregation levels) and how the system writes changes to data records of the aggregation level back to the InfoProviders included in the MultiProviders. For more information on these MultiProvider-specific features - with simple examples - see Complex Aggregation Levels [Page 34].

The following conditions apply to both types of aggregation level:

- At least one key figure and one characteristic have to be included in the aggregation level.
- The key figures used have to have the database aggregations SUM, MIN or MAX. With MIN or MAX, key figure values can only be displayed. They cannot be changed using manual planning or planning functions.
- For key figures of type date or time, only the data type ‘DEC’ is supported.
- Referencing key figures (and thus also non-cumulative key figures or elimination of internal business volume) are **not** supported in aggregation layers.
- If a characteristic is compounded and used in an aggregation level, the aggregation level must also contain all compounding "parent" characteristics.

- If a key figure is used in an aggregation level and does not have a fixed unit of measure or currency, the aggregation level must contain the associated characteristic for the unit.
- If a key figure with exception aggregation is used in an aggregation level, the aggregation level must also contain the characteristic for exception aggregation if it occurs in the underlying InfoProvider.
- The aggregation level inherits a navigation attribute from the underlying InfoProvider if it includes the basic characteristic of the navigation attribute. Note that the navigation attribute for an aggregation level is not visible in the Planning Modeler. It is only visible in the Query Designer.
- An aggregation level cannot be created on MultiProviders if a characteristic of an InfoProvider contained in the MultiProvider supplies two different characteristics in the MultiProvider.
- If a characteristic on the InfoProvider that serves as the basis for an aggregation level is constant, this characteristic has to be included in the aggregation level.

Activities

You are in the *Aggregation Level*s tab page of the *Planning Modeler.* In the *Aggregation Level Selection* screen area, you can create, copy, delete, change, check, save and activate aggregation levels.

Creating Aggregation Levels

1. To create an aggregation level, choose *Create*. The *Create Aggregation Level* dialog box appears.
2. Enter a technical name and a description.
3. Choose the appropriate InfoProvider. If you do not enter a search term and choose Start, the system shows all the InfoProviders available in your system.
4. Choose *Transfer.* In the lower screen area of the *Planning Modeler*, the system displays an overview of all InfoObjects of the InfoProvider.
5. Choose the InfoObjects that are to be included in the aggregation level. Note the conditions listed above.
6. To save the definition of the aggregation level, choose *Save.*
7. To check the definition of the aggregation level in view of consistency, choose *Check*.

   When you choose *Check*, the system tries to complete necessary objects, such as superordinate characteristics from compounded characteristics.
8. If the definition is consistent, choose *Activate*. Once it has been activated, the aggregation level is ready for use.

Changing Aggregation Levels

1. To change an aggregation level, choose *Change*. In the lower screen area of the *Planning Modeler*, the system displays an overview of all InfoObjects of the InfoProvider used in the aggregation level. The *InfoObjects* selection list allows you to display all InfoObjects for the InfoProvider, only those used in the aggregation level, or those not used in the aggregation level.

2. Change the definition as required.
3. Save, check and activate the changed definition.

4.2.1.1.1.2.1 Simple Aggregation Level

The following example demonstrates how the system works when a key figure value is changed (manually or automatically).

Assuming there is an InfoCube IC with the characteristics product, product group, version and year, along with the key figure revenue. The aggregation level ALVL includes the same objects with the exception of the characteristic, product.

Fact Table of the InfoCube IC

Product	Product Group	Version	Year	Revenue
P1	PG1	V1	2005	10
P2	PG1	V1	2005	20
P3	PG2	V1	2005	42

The key figure revenue includes the database aggregation SUM. Accordingly, we get the following result when the transaction data for the aggregation level ALVL is read from the database without restriction:

Aggregation Level ALVL (Key Figure Aggregated on the Database Level)

Product Group	Version	Year	Revenue
PG1	V1	2005	30
PG2	V1	2005	42

If you have changed the revenue from 30 to 40 and is saved as a new value, the system writes a new record with the difference of the key figure value to the fact table of the InfoCube IC:

Delta Record in the Fact Table of the InfoCube IC

Product	Product Group	Version	Year	Revenue
#	PG1	V1	2005	10

In this type of delta records, all characteristics of the InfoCube that are not included in the aggregation level get the initial value (not assigned: #). (Here we are assuming that no derivations were used. For more information on this concept, see Characteristic Relationships [Page 26].)

4.2.1.1.1.2.2 Complex Aggregation Level

The following examples show:

- how the system embeds data records from the InfoProviders contained in a MultiProvider in the MultiProvider.
- how the system writes new or changed data records of the MultiProvider to those included in this InfoProvider

Example: Characteristic Product in MultiProvider MP

Assuming there is a MultiProvider MP, that includes the actual-InfoCube IC_A and the plan InfoCube IC_P. The actual InfoCube IC_A includes the characteristics product, product group, version and year, as well as the key figure profit. The plan InfoCube IC_P includes the same objects with the exception of the characteristic, product. An aggregation level ALVL_MP is defined on the MultiProvider MP, which includes all characteristics of the MultiProvider.

The following two data records for the underlying InfoProvider yields the following data records in the MultiProvider:

Data Record in Actual InfoCube IC_A

Product	Product Group	Version	Year	Profit
P1	PG1	V1	2005	10

Data Record in Plan InfoCube IC_P

Product Group	Version	Year	Profit
PG1	V1	2005	30

Data Records in MultiProvider MP (or ALVL_MP)

InfoProviders	Product	Product Group	Version	Year	Profit
IC_A	P1	PG1	V1	2005	10
IC_P	#	PG1	V1	2005	30

The data records in the MultiProvider MP are - from a technical viewpoint - generated using a UNION operation from the records of the underlying InfoProvider. The InfoProvider is always included so that the "origin" of the respective data record is clear on the level of a data record.

If new data records are generated during manual planning or using the planning functions, the system ensure that every record of the MultiProvider can be assigned back to the InfoProviders contained in the MultiProvider uniquely and without loss of information.

The following table shows an example of a data record that could not be assigned.

Example of a Record in the MultiProvider MP that could not be assigned

InfoProviders	Product	Product Group	Version	Year	Profit
IC_P	P1	PG1	V1	2005	43

The data record is part of the InfoProvider IC_P. However, this InfoProvider does not provide a product in the MultiProvider. This means that P1 is not permitted.

In manual planning, data cells that could lead to this type of record are not input ready. In planning functions, the system ensure that these types of data records cannot be saved.

The same situation can occur for key figures in complex aggregation levels. If K is a key figure in MultiProvider MP that is supplied from the actual InfoCube IC_A, but not by the plan InfoCube IC_P, this key figure is always initial in a data record in the MultiProvider MP with the InfoProvider IC_P. This value also cannot be changed.

4.2.1.1.1.3 Filter

Use

A filter is an object that describes a multidimensional segment of data from a data set. Filters are used in reporting, analysis and planning, for example, to restrict data to a certain business area, certain product groups or certain time periods. This segmenting of the data can be achieved so that users or user groups only have access to the data that is relevant to them or that only certain data areas are available within an application scenario.

Within BI Integrated Planning, filters set the selection for the data upon which a planning function operations. A planning sequence comprises a set of planning functions. A filter is assigned to each of these functions.

> You want to revaluate your transaction data in your InfoProviders by a factor of 10%. However, you want to only perform the revaluation for certain groups of customers. To do this, you create a filter that contains the group of customers that you wish to revaluate.

Filters can be reused in planning functions and in queries.

Integration

You can create multiple filters for an InfoProvider. Here you can use the *Planning Modeler* or *Planning Wizard* and the *Query Designer.* You can only define filters on aggregation levels in the *Planning Modeler* or *Planning Wizard*.

For more information on filters in the query, see the documentation on the Query Designer under Filter [External]

Prerequisites

In order to create a filter for use in BI Integrated Planning, you need an aggregation level. For more information, see Aggregation Levels [Page 31].

Features

You choose the characteristics that you want to restrict from the characteristics of an aggregation level and insert them into the filter.

A filter has the following components:

Filter Components

Element	**Description**
Characteristic Restrictions	On the restriction dialog, you can further restrict the characteristic using single values, value ranges, hierarchy nodes and variables. These characteristic restrictions determine the selection of data for a filter.
Default Values	Default values are only relevant in queries. They can be defined in the same way as characteristic restrictions and set the initial filter status of the query upon execution.

To determine time-dependent selections, such as for determining a time-dependent hierarchy for time-dependent hierarchy node selections, a *Filter Key Date* can be specified.

> The delivered variable 0PLANDATA can be used on the characteristic 0CALDAY for synchronizing key dates in queries, filters, characteristic relationships, data slices and planning functions. In this way, you can ensure that the same key date is used in these objects.

The function of a filter depends on its use, either in a planning function or in a query.

Filters in Planning Functions

In connection with planning functions, a filter in the characteristic restrictions describes the data that is used for executing a planning function.

Selections in the default values are **not** used for executing planning functions.

You can also use a key date for the filter to determine time-dependent selections.

Filters in a Query

The values defined in the characteristic restrictions restrict the data that is available for further filtering at runtime of a query. Filtering a characteristic value outside of these value sets is then no longer possible.

The default values determine the initial filter status of the query.

The settings *Changeable upon Execution* and *Only Single Value* generally refer to the use of filters with a query.

Changeable upon Execution sets whether the value selection made in the characteristic restrictions can be changed during execution of the query. This setting is a prerequisite for the definition of default values for a characteristic.

When the option *Changeable upon Execution* was selected, you can use the *Only Single Value* option to determine whether only a single value may be used for filtering the query.

For more information on filters in the query, see the documentation on the Query Designer under Filter [External].

Activities

You are in the *InfoProvider* tab page of the *Planning Modeler.* In the *Filter Selection* screen area, you can create, copy, delete, change, check, save and activate filters.

Creating Filters

1. To create a filter, choose *Create*.
2. In the *Create Filter* screen area, enter a technical name and a description for the filter to be created.
3. In the ***Aggregation Level Selection*** screen area, choose the required aggregation level. If you do not enter a search term and choose *Start*, the system shows all the aggregation levels available in your system.

   Choose *Transfer.* In the lower part of the *Planning Modeler* screen area, the system displays the *Filter* and *Settings* tab pages.
4. On the ***Filter*** **tab page**, choose the characteristics to be restricted.

> You can adapt the display of the characteristics according to your needs (display with key, text, key/text or text/key).

Add the characteristics that you wish to restrict to the list. You can add individual or all characteristics of the aggregation level (using *Add* or *Add All*) to the list.

5. Select the characteristic to be restricted and choose the symbol for input help in the column after *Characteristic Restrictions*. The dialog box for determining characteristic restriction appears.

   > You can choose single values, value ranges and hierarchy nodes or variables as needed. You can also transfer values from the history or from favorites.
   >
   > You can choose one of the following views to select single values, value ranges and hierarchy nodes or variables:
   >
   > - *All Values* to display all characteristic values
   > - *Search* to search for a specific characteristic value or hierarchy node
   > - *Value Range* to define value ranges (such as intervals)
   > - *Variables* to select or create a variable
   > - *All Nodes* to display and select hierarchy nodes

6. In the value list, select one or more values, value ranges or hierarchy nodes and choose *Insert* and saved the affected selection with *OK*. The system transfers the affected settings to the list of restricted characteristics.
7. With *Show Enhanced Settings*, you can make further restrictions:

   - *Changeable upon Execution* (determines whether the characteristic restrictions can be changed at execution)

   When you select the *Changeable upon Execution* option, you can make further settings:

   - *Only Single Value* (determines whether only a single value may be used for filtering at runtime)
   - *Default Value.* Choose the symbol in the column after *Default Value*. The dialog box for determining the default value appears. Proceed as you did when restricting the characteristic values.

8. On the ***Settings* tab page** you can set the key date.
9. To save the definition of the filter, choose *Save.*
10. To check the definition of the filter in view of consistency, choose *Check*.

    > Even when the check for a filter fails, the filter can be saved in the Planning Modeler or Planning Wizard (like in the Query Designer). The procedure allows you to save filters without characteristic restrictions and then create them in the system later, for example. A consistency check is done, however, during execution of the filter, before it is used.

4.2.1.1.1.4 Planning Functions

Use

Planning functions are used for system-supported editing and generation of data within BI Integrated Planning.

A planning function specifies the ways in which the transaction data for an aggregation level can be changed. The following are determined for this purpose:

- The name of the aggregation level
- The type of planning function
- How characteristics are used
- The parameter values

The planning function type determines the way in which data is changed by a planning function. The BI system offers you a number of standard planning function types:

- Unit conversion
- Generate combinations
- Formula
- Copy
- Delete
- Delete invalid combinations
- Forecast
- Repost
- Repost by characteristic relationships
- Revaluation
- Distribute by reference data
- Distribute by key
- Currency translation

You can also implement customer-specific planning function types. For more information, see Implementing Planning Function Types [Page 71].

Integration

In the *Planning Modeler* and *Planning Wizard*, you create your planning functions (as well as the prerequisite objects of the planning model).

In the Data Warehousing Workbench, the planning function objects are displayed in the functional areas *Business Content* and *Transport Connection* in the *Planning* folder.

Prerequisites

You have created the following objects in the *Planning Modeler* or *Planning Wizard*:

Aggregation levels on which planning functions are created (see Aggregation Levels [Page 31]). Planning functions can be created and executed on each active aggregation level.

- **Filters** that are required when the planning function is executed. Filters determine the data for which the planning function is executed. The planning function locks the data defined in the filter in the real-time enabled InfoCubes that are part of the aggregation level. The filter has to be defined on the same aggregation level as the planning function (see Filter [Page 36]).

Features

You determine the planning function type and the aggregation level on which the planning function is to work. In addition, you can change the characteristic usage, the conditions and the parameter sets.

The features will now be explained, with an example of how to create a planning function of type *Repost*.

The following table shows the data for the InfoProvider before the planning function is executed:

Before executing the planning function

Product	Product Group	Version	Year	Sales
P1	PG1	V1	2007	10
P2	PG1	V1	2007	20

You want to repost all records in version “V1” to version “V2”. You do this by **reposting** all key figures. The following table shows the status after the planning function has been executed:

After executing the planning function

Product	Product Group	Version	Year	Sales
P1	PG1	V1	2007	0
P2	PG1	V1	2007	0
P1	PG1	V2	2007	10
P2	PG1	V2	2007	20

After reposting, the records that contain zeros only (empty records) remain in “V1”; the required records appear in "V2".

Characteristic Usage

The planning function type defines the options available for using characteristics and the parameters of the planning function. Together, the parameters of the planning function type define the parameter set.

With **characteristic usage**, the characteristics of the aggregation level are divided into *Characteristics to Be Changed* and *Block Characteristics* (characteristics that are not used). In this way, you specify the characteristic values that are changed when the planning function processes a data record. Block characteristics remain constant.

When you create a planning function of type *Repost* for the case described above, you first check which characteristic values should be changed and set the *to be changed* indicator accordingly. Since you want to repost the data from version “V1“ to version “V2“, you set the indicator for the *Version* characteristic as *To Be Changed* (which in this case means to be reposted).

Furthermore, you can select block characteristics as condition characteristics.

Parameter Sets and Conditions

The next step is to specify the detailed information. With most planning functions, all transaction data is processed with the same set of parameters. In this case, a block characteristic was not selected as a condition characteristic; only one parameter set has to be entered.

> The parameter set for the planning function type *Repost* includes a table for selecting the key figures that are to be reposted and a table in which you can enter the *From-To Value Pairs* for the characteristics to be reposted. In key figure selection, you set the indicator for *Select All Key Figures*. In the *From- and To Values for Reposting* table, you choose *Create Row* and enter “V1” under *From* and “V2” under *To.* The planning function is ready.

If you want to use different parameter sets to execute different transaction data records, you have to work with **conditions**. You have to select at least **one** block characteristic as a condition characteristic.

> If you want to increase the planned production for products in product group PG1 by 5% and the products of product group PG2 by 10%, choose the product group as the condition characteristic.

In the parameters, you can create multiple pairs of conditions and parameter sets. For each pair, use a filter to select condition characteristics. You can change the associated parameter set for each pair.

> From a technical viewpoint, the method that the planning function actually executes is called more than once. The data that was selected with the filter is divided into blocks. Each combination of characteristic values in the block characteristic forms a separate block (thus the name *Block Characteristics*). Planning function types that work with reference data can also have additional blocks (such as *Copy*). The method is then called once for each block with a table of records. The table includes those data records that correspond to the characteristic combination for the block in the block characteristics.
>
> For each block, the system checks whether there is a condition and parameter set pair so that the block fits the condition. The block is tested against the conditions along the sequence of pairs. The first pair in which the block matches the condition is used, that is, the method of the planning function type is executed with the block and the set of parameters that fits the condition. The remaining pairs are not taken into account. Therefore the method is only executed once for each block.

Variables in Planning Functions

The usual BI variable types are available in many planning function types (see Variables [Page 64]).

Working with Empty Records

Almost all planning function types do **not** read empty records and do **not** write empty delta records to the buffer. Exceptions to this are *Copy* and *Generate Combinations:* these two function types read empty records and write empty delta records.

Activities

You are on the *Planning Functions* tab page of the *Planning Modeler.* In the *Planning Function Selection* screen area, you can display, create, copy, delete, change, check, and save *Planning Functions*.

Create a Planning Function

1. To create a planning function, choose *Create*. The *Create Planning Function* dialog box appears.
2. Choose the planning function type.
3. Enter a technical name and a description for the planning function.
4. Select the aggregation level on which the planning function is to work.
5. Choose *For Characteristic Usage* and determine which characteristics are to be changed and used in conditions, as required.
6. Choose *For Parameters*. In the *Conditions with Parameters* screen area, you can create, delete and copy conditions. On the *Selected Conditions* tab page, you can use input help to select the condition values to which the condition is to be applied. On the *Associated Parameter Set* tab page, you maintain the parameter sets.

Check

During the check, you have to specify values for existing mandatory entry variables if values have not been selected for these variables in the current session. If this is the case, the initial screen appears.

Save

Planning functions can also be saved when they are **not** consistent.

Execute

Planning functions can be executed directly from a Web application or a BEx workbook. However, you have to enable this in the relevant design tool first.

In order to execute a planning function in the *Planning Modeler*, you must first include it in a planning sequence (see Planning Sequences [Page 63]).

In the *Planning Wizard*, you create a temporary planning sequence in the last step. This contains exactly one planning function.

Before the planning function is executed, the system checks whether the planning function is consistent.

Example

Process Flow of Planning Function: Distributing by Key [Page 43]

4.2.1.1.1.4.1 Process Flow of Planning Function: Distributing by Key

For the year “2007“ and the version "V1", the planned quantity per product is to be distributed to the available factories "W1" and "W2". The total quantity of each product stays the same. However, it is to be distributed evenly between the factories.

The following table shows the data for the InfoProvider before the planning function is executed:

Before executing the planning function

Year	Version	Product	Factory	Quantity
2007	V1	P1	W1	10
2007	V1	P1	W2	20
2007	V1	P2	W1	60
2007	V1	P2	W2	40

The following table shows the result executing the planning function:

After executing the planning function

Year	Version	Product	Factory	Quantity
2007	V1	P1	W1	15
2007	V1	P1	W2	15
2007	V1	P2	W1	50
2007	V1	P2	W2	50

In reality, the planning function only writes delta records to the InfoCube. The following table shows the delta records that were actually written:

Delta records in the InfoCube

Year	Version	Product	Factory	Quantity
2007	V1	P1	W1	5
2007	V1	P1	W2	-5
2007	V1	P2	W1	-10
2007	V1	P2	W2	10

Creating the Planning Function

When you create the planning function, you must first determine the characteristics in which the values change. With regard to the **characteristics *Year, Version* and *Product***, the total of values should not change; these characteristics remain constant and become **block characteristics.** The key figure values should be distributed according to the characteristic value *Factory.* Thus in characteristic usage, the ***Factory* characteristic** must be selected as **to be changed**.

The parameters for the distribution function are set as follows (see Standard Planning Function Types [Page 46]).

- Use the table for key figure selection to determine that the only key figure to be distributed is the ***Quantity* key figure**.
- Because within a block (characteristic combination (year, version, product)) the overall total always has to be distributed each time, the ***Top Down Distribution*** distribution type is applied with the ***Distribute All*** setting.
- The factories **"W1"** and **"W2"** are entered as to values and receive **identical keys** (such as 1).

Process Flow of the Planning Function

You want to execute the planning function with a filter that is restricted to year "2007" and version "V1". The planning function executes the following steps:

1. First the system loads the filter and the planning function.
2. It replaces the variables.
3. It checks the filter and the planning function for consistency.
4. Using the filter, the system requests the selected data and loads it into the buffer. In this example, we assume that the records displayed above in the "before executing the planning function" table are selected by the filter and transferred to the planning function. Whether the system reads or takes the existing empty records into account or not depends on the type of planning function (see Standard Planning Function Types [Page 46]). With the distribution types, the system does not read empty records.
5. In accordance with the characteristic usage, the system divides the transaction data into blocks. The tables below the process flow illustrate this. Two blocks result. They are defined by the characteristic combinations shown here.

Block formation in accordance with characteristic usage

Block No.		**Year**	**Version**	**Product**
1		2007	V1	P1
2		2007	V1	P2

6. For each block, the system uses the characteristic combinations to look for the correct parameter set from the condition-parameter set pairs. Since no conditions were created in this example, both blocks are executed with the only available parameter set.
7. For each block, the system runs the actual process, in this case, distribution. The tables after the process flow show the before-after values and the resulting delta records for each block.
8. Depending on the type of planning function, the system either processes any empty records that it finds or does not take them into account. In this section there are no empty records.
9. The system checks
    - Whether the resulting records are consistent with regard to master data and characteristic relationships.
    - Whether they are protected by data slices.
    - Whether they are located within the transferred filter.
10. Once the system has successfully processed all blocks, it writes the delta records it has collected back to the buffer. As required, derivation is performed in the buffer (see Characteristic Relationships [Page 26] and Aggregation Levels [Page 31]).

If one of the generated records has a consistency error, the entire planning function writes nothing to the buffer.

Overview: Block 1

Block 1: { 2007, V1, P1} Before

Year	Version	Product	Factory	Quantity
2007	V1	P1	W1	10
2007	V1	P1	W2	20

Block 1: { 2007, V1, P1} After

Year	Version	Product	Factory	Quantity
2007	V1	P1	W1	15
2007	V1	P1	W2	15

Block 1: { 2007, V1, P1} Delta

Year	Version	Product	Factory	Quantity
2007	V1	P1	W1	5
2007	V1	P1	W2	-5

Overview: Block 2

Block 2: { 2007, V1, P2} Before

Year	**Version**	**Product**	**Factory**	**Quantity**
2007	V1	P2	W1	60
2007	V1	P2	W2	40

Block 2: { 2007, V1, P2} After

Year	**Version**	**Product**	**Factory**	**Quantity**
2007	V1	P2	W1	50
2007	V1	P2	W2	50

Block 2: { 2007, V1, P2} Delta

Year	**Version**	**Product**	**Factory**	**Quantity**
2007	V1	P2	W1	-10
2007	V1	P2	W2	10

4.2.1.1.1.4.2 Standard Planning Function Types

Definition

The standard planning function types are part of the BI system. When you use a standard planning function type with a particular aggregation level, you define a planning function of this type.

Use

The following standard planning function types are delivered as part of the BI system.

Unit Conversion

You use the *Unit Conversion* function type to convert units of key figures into other key figures.

In the *Source / Target Key Figure Conversion Type* table, you can specify multiple conversions. For each conversion, you have to select the unit or quantity conversion type. The value in the target key figure is overwritten. This also applies when the source key figure is empty.

Since no characteristic values are changed during unit conversion between key figures, this option is not available in characteristic usage.

Generate Combinations

You use the *Generate Combinations* function type to generate empty records in an aggregation level for all permitted combinations according to the master data and characteristic combinations. These are exactly those combinations that are also valid during the check for this aggregation level (see Characteristic Relationships [Page 26]).

This function type does not permit any other settings: Because new data records are continuously generated for the whole aggregation level, all characteristics must be *To Be Changed.* The function type works without block characteristics.

The function type writes empty records (like with *Copy*).

Formula

You use the *Formula* function type to incorporate extended mathematical functions in the calculation of plan data. The same formulas are valid in BI Integrated Planning as were valid in BW-BPS. For more information, see the documentation on BW-BPS under Formula [Page 251].

Only those blocks that already contain data are processed.

Copy

You use the *Copy* function type to copy the key figure values from existing characteristic combinations to other characteristic combinations.

For example, if you want to copy the values from the year 2005 to 2007 without changing other characteristics, set the indicator *Will Be Changed* for the characteristic *Year*.

Using the table for key figure selection, you can set which key figures are to be copied.

Using the *From and To Values for Copying* table, you can either create a simple copying process or multiple copying processes within a planning function.

The function type allows complicated copying processes as well: both the from and to values can be characteristic restrictions on the characteristics to be changed. The system continually totals the values of the From key figures using all the records in the block that correspond to the characteristic restriction; it continually writes the To totals for each individual characteristic combination.

You can also copy values from **one** From value to **multiple** To values.

> You copy data from the year “2005“ in version “ACTUAL” to the combinations {“2006“, “PLAN01“} and {2007, „PLAN02“}.

The function type reads and writes empty records.

The following rules apply:

- The From values are read as reference data and do **not** need to be part of the filter that is transferred to the planning function.
- The To values are changed; they have to be included in the transferred filter.
- The key figure values for the To values are always overwritten during copying. This is also valid when the From values are empty.
- If there are no values to form a block for a characteristic restriction on the From side, the sub-process is not executed.
- Combinations are generated on the To side for the specified characteristic restrictions. This is consistent with the master data and the characteristic relationships defined for the InfoProviders. If the system cannot find a target for a block and a sub-process, the system terminates the function and produces an error message.
- If a target has been specified in one or more sub-processes, the function is executed and as a result, the target contains the appropriate totals.

The copying function also forms blocks of reference data.

Delete

You use the *Delete* function type to delete the key figure values for the selected data records.

No characteristic values are changed. In characteristic value usage, you can only select characteristics as condition characteristics.

You can select the key figures to be deleted in a table.

Delete Invalid Combinations

You use the *Delete Invalid Combinations* function type to delete all key figure values for all records where the combination of characteristic values does not correspond to the characteristic combinations that are defined in the underlying real-time enabled InfoCube.

> Note the following condition: A planning function of this type can only be created on an aggregation level that itself has been created directly on a real-time enabled InfoCube (a simple aggregation level, see Aggregation Level

[Page 31]). The aggregation level must include all of the InfoObjects the InfoCube that are valid in aggregation levels.

Forecasting

You use a forecast procedure to predict the future development of key figure values. The forecast function make various statistical forecast procedures available so that you can calculate forecast values from your historic data.

For more information, see Forecasting [Page 51].

Repost

You use the *Repost* function type, like *Copy*, to post the key figure values for existing characteristic combinations to other combinations. In contrast to copying, the key figure values of the From values are deleted. You can find an introductory example under Planning Functions [Page 39].

Using the table for key figure selection, you determine which key figures you want to repost.

Using the *From and To Values for Reposting*, you can either create a simple reposting process or multiple reposting processes within a planning function.

The following rules apply:

- Both the From and To values for the reposting process have to be single values.
- Both the From and the To values are changed and have to be included in the filter that is transferred to the planning function.
- When you perform repost, the key figure values are always added to the To values.

Repost by Characteristic Relationships

You use the *Repost by Characteristic Relationships* function to repost transaction data so that it is consistent with the characteristic relationships. The original records are deleted and reposted to the correct characteristic combinations.

In characteristic usage, you can determine the characteristics for which values are to be corrected. This only makes sense for characteristics that can be derived in accordance with the characteristic relationships.

Note the following condition: A planning function of this type can only be created on an aggregation level that itself has been created directly on a real-time enabled InfoCube (a simple aggregation level, see Aggregation Level [Page 31]). The aggregation level must include all of the InfoObjects the InfoCube that are valid in aggregation levels.

Revaluation

You use the *Revaluation* function type to increase or decrease key figures by one percentage.

No characteristic values are changed. In characteristic value usage, you can only select characteristics as condition characteristics.

You can choose whether you want to enter a common percentage for all key figures or want to revaluate key figures with individual percentages.

Distribution

For an introduction to how a planning function of type *Distribute* works, see Process Flow of Planning Function: Distributing by Key [Page 43].

You use the table for key figure selection to select the key figures that you want to distribute.

The distribution process can be controlled in different ways:

- With a top-down variant
    - *Distribute All*: The entire block is totaled.
    - *Only Distribute Not Assigned:* Values are only distributed if they have the characteristic value *Not Assigned [#]* in the current block for all of the characteristics to be changed.
- With the table *From and To Values for Distribution*: You can create one or more distribution processes by manually entering the characteristic restrictions onto the characteristics to be changed.

  The system continually totals the values of the key figures on the From side over all the records in the block that correspond to the characteristic restriction, and distributes this total to the To values. The distribution keys and the characteristic combinations to which they are distributed are different for the different available processes (*Distribution by Keys* or *by Reference Data)*.

  The system performs the following steps to execute a planning function of type *Distribution.*

    i. The total amount to be distributed is determined.

    ii. For the data records that are affected by the From values, the key figure values are deleted.

    iii. In accordance with the distribution keys, the totals are added together with the key figure values of the data records selected by the To values.

The following rules apply:

- Both the From and To values are changed and have to be included in the filter that is transferred to the planning function.

  If the From value is created so that it can select data records that are not in the filter, the system does not produce an error message. The data records are not part of the distribution.

  If the To value is created so that it can select data records that are not in the filter, the system terminates the entire planning function and produces an error message.

  If a target has been specified in one or more sub-processes, the function is executed and as a result, the target contains the appropriate totals.
- Rounding differences are distributed evenly to those target data records that are not equal to zero.

Distribution by keys

You use the *Distribution by Keys* function type to generate the characteristic combinations to which data is distributed in accordance with the master data and characteristic relationships. The key figure values are distributed according to the expressly specified distribution keys. These are distribution functions that determine the weighting of the distribution.

With the *From and To Values for Distribution* table, you can either enter a specific value in the *Factor* column or select a formula variable from the input help for the To values. The system displays them in the *Factor Variable Text* column.

If the characteristic restriction of a To value is **not** a single value, this key is valid for every characteristic combination that corresponds to the characteristic restriction.

The keys are normalized during processing; they are converted into percentages. The system first totals the distribution factors that are assigned to the individual characteristic values and then distributes the values relatively. This results in percentages that always add up to 100 %.

The same key applies to all the key figures to be distributed for each planning function.

Combinations are generated on the To side for the specified characteristic restrictions. This is consistent with the master data and the characteristic relationships defined for the InfoProviders. If the system cannot find a target for a block and a sub-process, the system terminates the function and produces an error message.

Distribution by reference data

You use the *Distribution by Reference Data* function type to generate all the characteristic combinations to which you want to distribute data in accordance with the reference data. The key figure values are distributed by percentage in accordance with the reference data.

The system provides two parameters for the *Selection of Reference Data.*

Parameter	**Description**
Reference Key Figure (optional)	You determine a key figure that is to be used in the reference data to calculate the distribution key. • When you have chosen this key figure, the system distributes all of the key figures to be distributed according to this key, that is, according to this reference key figure. • If **no** reference key figure is selected, the system calculates a key individually for each of the key figures to be distributed. The same key figure is used in the reference data that is distributed.
Reference Values on Any Block Characteristics	You set a reference value for one or more block characteristics. While reading the reference data for every block, the system overwrites the characteristic value of the block in the characteristic with this reference value. An InfoProvider provides data for the years 2005 and 2006. You want to change transaction data for the year 2006. Thus the data from the year 2006 is selected in the filter that is transferred to the planning function. Therefore all the blocks have the year 2006. However, if the reference value is 2005, you can ensure that the keys are calculated according to the data from 2005.

The system distributes data to precisely those characteristic combinations that exist in the reference data and fit the characteristic restrictions in the To values.

The percentage distribution corresponds to the percentage distribution in the reference data.

In top-down distribution, the system distributes to all characteristic combinations that exist in the reference data. Data records are excluded that have the value *Not Assigned [#]* for all characteristic values.

If there is no reference data for a sub-process, the key figure values for this sub-process are **not** distributed; the system produces a warning.

Currency Translation

You use the *Currency Translation* function type to convert currencies of key figures into other key figures.

In the *Key Figures and Currency Translation Types* table, you can specify multiple translations. For each translation, you have to select a currency translation type. The value in the target key figure is overwritten. This also applies when the source key figure is empty.

Since no characteristic values are changed during currency translation between key figures, this option is not available in characteristic usage.

4.2.1.1.1.4.2.1 Forecasting

Use

You use a forecast procedure to predict the future development of key figure values. The forecast function make various statistical forecast procedures available so that you can calculate forecast values from your historic data.

Integration

These forecast functions are available as planning function types within BI Integrated Planning (see Planning Functions [Page 39]). They are based on the same statistical procedures that are used in demand planning.

> For more information about forecasting in the context of demand planning, see http://help.sap.com/ → *Documentation* → *mySAP Business Suite* → *mySAP Supply Chain Management* → *SAP Supply Chain Management* → *SAP Advanced Planning and Optimization (SAP APO)* → *Demand Planning* → *Demand Planning Process* → *Definition/Redefinition of Forecast Models* → *Creating a Master Forecast Profile* → *Univariate Forecasting*.

Prerequisites

- Historic data is available for the forecast calculation.
- The aggregation levels on which you are creating a forecast function have to contain at least one time characteristic (for example, *Fiscal Year/Periods*).

Features

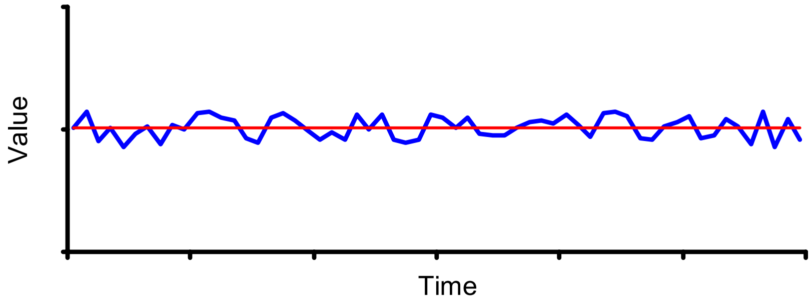
The forecast planning function type covers various univariate forecast procedures. In a forecast procedure, only the time series of the selected forecast key figure is taken into account; no additional data is input into the forecast calculation to interpret the development of the key figure.

Time Series Patterns

You can create forecasts for the following time series patterns:

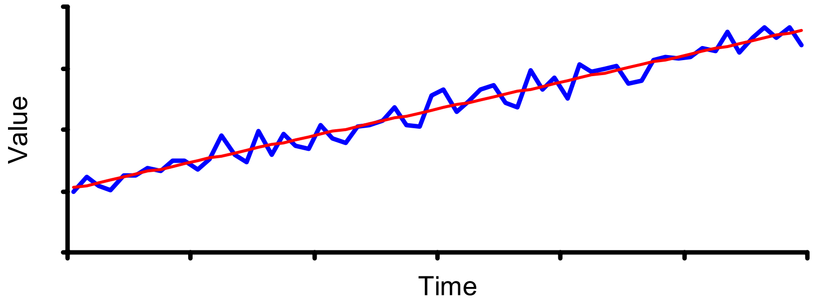
Constant

The historic data is essentially constant and varies very little from a stable mean value. In the following figure, this base value is represented by a red line:



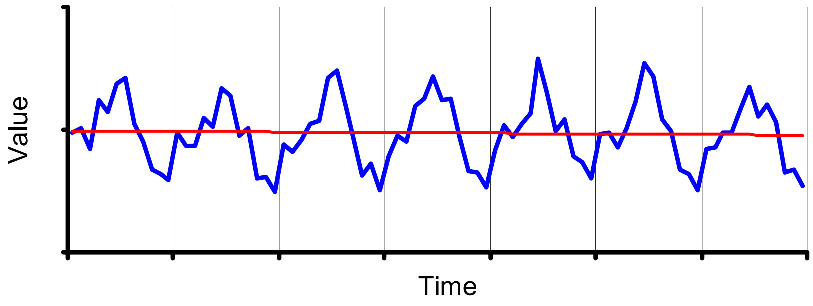
Trend

The time series pattern rises or falls continuously. In the following figure, this trend is represented by a red line:



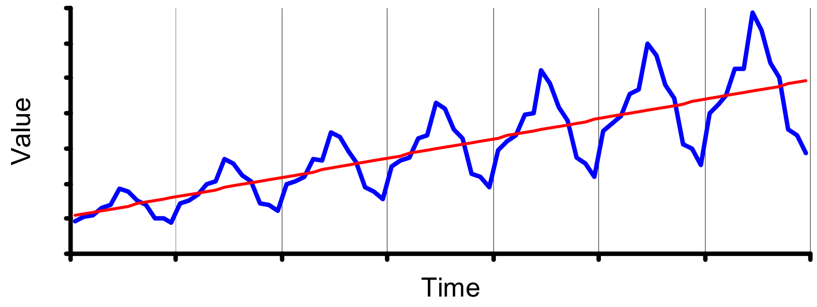
Seasonal

The values show periodically recurring peaks and troughs (on an annual basis). There is a stable mean value. In the following figure, this base value is represented by a red line:



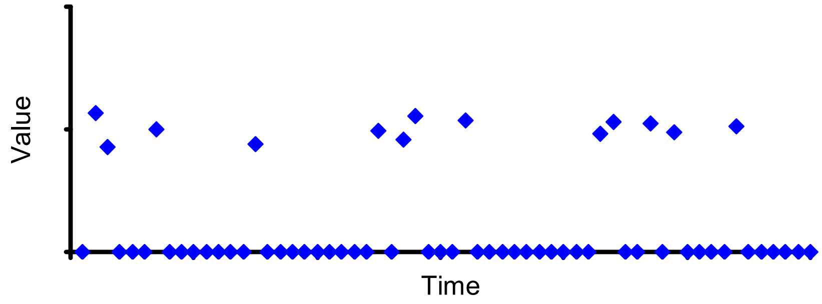
Seasonal Trend

This time series pattern is a combination of the trend and seasonal patterns. There are periodically recurring peaks and troughs, but with a continual increase or decrease in the mean value.



Intermittent

The value is zero at most points in the time series pattern. The values that are not zero fluctuate around a mean value.



Forecast Strategies

The forecast strategy determines which forecast procedure is used. In order to choose a suitable forecast strategy, base your decision on the time series pattern. The different forecast procedures are based on the different forecast models (time series models). They produce different results.

The following forecast strategies are available:

- *Average*
- *Moving average*
- *Weighted moving average*
- *Linear regression*
- *Seasonal linear regression*
- *Simple exponential smoothing (constant model)*
- *Simple exponential smoothing with alpha optimization (constant model)*
- *Linear exponential smoothing (trend model)*
- *Seasonal exponential smoothing (seasonal model)*
- *Seasonal trend exponential smoothing (seasonal trend model)*
- *Croston model*
- *Automatic model selection*

The *automatic model selection* forecast strategy allows you to let the system select the forecast model that best fits the trend of the historic data (see Automatic Model Selection [Page 58]).

If you already know that a particular forecast model is well matched to the time series trend, or if you explicitly want to use a particular forecast model for other reasons, you can select a forecast model (see Forecast Strategies [Page 55]).

Optional Functions for Forecast Strategies

The forecast strategies offer the following additional functions and possibilities:

- Outlier correction [Page 60]
- Logging statistical key figures [Page 61]
- Ignoring initial zeros [Page 62]

For exponential smoothing:

- Optimization of smoothing factors for exponential smoothing

For forecast models with trend components:

- Trend dampening [Page 61]

Activities

To create a planning function of type *Forecast*, the following steps are required:

1. Select a **time characteristic for the forecast**

   Choose *For Characteristic Usage*. You choose the time characteristic that represents the time dimension of the forecast.

You **cannot** include the selected time characteristic in the set of characteristics for conditions. For more information about using characteristics and condition characteristics, see Planning Functions [Page 39].

2. Select the **data for the forecast**

   Choose *For Parameters* and perform the following steps:

   a. Select the *Forecast Key Figures*

      You specify the key figures for which you want to calculate the forecast.

   b. Specify the *Forecast Time Frame*

      You specify the time frame for the forecast by restricting the time characteristic for the forecast. This is usually a time interval that represents the length of time for the required forecast.

      If the time characteristic for the forecast is *Fiscal Year/Periods* (0FISCPER), the system suggests the higher-level characteristic *Fiscal Year Variant*. You only have to restrict this characteristic if you are using variables with processing type *Customer and SAP Exit* in the restrictions for *Fiscal Year/Periods* (for example, *Current Periods*).

      The system ignores exceptional periods of the time characteristic *Fiscal Year/Periods* (0FISCPER) when it performs the calculations; values for periods of this type are not generated or changed.

3. Specify the **historic data**

   You specify the *Historic Time Frame* in the same way as the forecast time frame. The longer the time frame, the better the quality of the forecast results.

   You use the *Filter for Historic Data* if your historic data differs from the forecast data for particular characteristics. You have to specify a single value for each of these characteristics.

   This may be the case, for example, with the *Version* characteristic if the forecast data is in a plan version and the historic data is based on an actual version.

4. Select the **forecast procedure** and enter additional parameters

   You choose the required value for the *Forecast Strategy* parameter. Depending on the selected forecast strategy, the system proposes additional parameters. In certain cases, entering a value for individual parameters is mandatory.

   The system proposes *Automatic Model Selection* as the default forecast strategy. In comparison to other forecast strategies, this forecast strategy provides the largest number of parameters. Note that forecasting is accordingly time-consuming.

5. Save the planning function.

4.2.1.1.1.4.2.1.1 Forecast Strategies

Use

The forecast strategy determines how forecast values are calculated.

All forecast strategies are based on statistical forecast procedures and forecast models that represent the time series mathematically.

The exponential smoothing methods are currently the most widely used time series patterns (see Exponential Smoothing [External]).

If you expect historic values to continue to develop as they have in the past, choose a forecast model that fits the previous trend well.

The *Automatic Model Selection* strategy allows you to let the system select the forecast model that best fits the trend of historic data (see Automatic Model Selection [Page 58]).

Features

The following forecast strategies are available:

Average

The forecast value is calculated from the means of the historic values.

Optional forecast parameters: outlier correction [Page 60], logging statistical key figures [Page 61], ignoring initial zeros [Page 62].

Moving Average

The forecast value is calculated according to the order.

- Mandatory forecast parameter: *Order of Moving Average*.

  The order of the moving average is a number N that determines the length of the time interval for calculating the average. This is the number of chronologically sequential historic values. The forecast value is calculated as the average of the last N historic values.

  Do not enter a negative number for the order.

Optional forecast parameters: outlier correction [Page 60], logging statistical key figures [Page 61], ignoring initial zeros [Page 62].

Weighted Moving Average

When the system calculates the moving average, each historic value is given a particular weight.

- Mandatory forecast parameter: *Order of Moving Average*.

  The order of the moving average is a number N that determines the length of the time interval for calculating the average. This is the number of chronologically sequential historic values.

  Do not enter a negative number for the order.

- Mandatory forecast parameter: *Weighting Factors*.

  The weighting factors specify the relationship between the individual historic values and the average calculation. The sequence is important: Weighting factor 1 refers to the previous periods; weighting factor 2 refers to the periods before that, and so on.

  You want to create a forecast based on monthly values and choose a weighted moving average with an order that has the value 6. In this case, you want to place more weight on the most recent monthly values than on the less recent monthly values. The historic data is taken from months 5 to 10. The 6 weighting factors and the relevant months are as follows:

No.	Weighting Factor	Month
1	3,00	10
2	2,00	9
3	2,00	8
4	1,00	7
5	1,00	6
6	1,00	5

Optional forecast parameters: outlier correction [Page 60], logging statistical key figures [Page 61], ignoring initial zeros [Page 62].

Linear Regression

Simple linear regression (ordinary least squares).

Optional forecast parameters: trend dampening [Page 61], outlier correction [Page 60], logging statistical key figures [Page 61], ignoring initial zeros [Page 62].

Seasonal Linear Regression

Seasonal linear regression is based on the same statistical procedures as used in demand planning.

> For more information, see http://help.sap.com/ → *Documentation* → *mySAP Business Suite* → *mySAP Supply Chain Management* → *SAP Supply Chain Management* → *SAP Advanced Planning and Optimization (SAP APO)* → *Demand Planning* → *Demand Planning Process* → *Definition/Redefinition of Forecast Models* → *Creating a Master Forecast Profile* → *Forecast Strategies* → *Seasonal Linear Regression*.

- Mandatory forecast parameter: *Periods per Season*

Optional forecast parameters: trend dampening [Page 61], outlier correction [Page 60], logging statistical key figures [Page 61], ignoring initial zeros [Page 62].

Simple Exponential Smoothing (Constant Model)

Simple exponential smoothing is suitable if the historic data shows a constant trend.

- Smoothing factor settings: *Alpha* (base value [External])

Optional forecast parameters: outlier correction [Page 60], logging statistical key figures [Page 61], ignoring initial zeros [Page 62].

Simple Exponential Smoothing with Alpha Optimization (Constant Model)

This procedure corresponds to the “simple exponential smoothing“ above, with one modification; in addition, the system calculates the *Alpha* smoothing factor. The *Alpha* value is varigated in the interval using the defined step size and a forecast calculation (for the historic time frame) is performed in each case. The optimum value for *Alpha* is the value that produces the least number or errors in the forecast results.

- Smoothing factor settings: *Optimization Variable*, *Alpha From*, *Alpha To*, *Alpha Step Size*.

Linear Exponential Smoothing (Trend Model)

The forecast is calculated according to Holt’s method and is suitable if historic values display a rising or declining trend.

- Smoothing factor settings: *Alpha* (Base Value [External]), *Beta* (Trend Value [External]).

Optional forecast parameters: trend dampening [Page 61], outlier correction [Page 60], logging statistical key figures [Page 61], ignoring initial zeros [Page 62].

Seasonal Exponential Smoothing (Seasonal Model)

Choose this strategy if your historic values show seasonal fluctuations (for example, annual fluctuations) from a constant base value.

- Mandatory forecast parameter: *Periods per Season*

Smoothing factor settings: *Alpha* (Base Value [External]), *Gamma* (seasonal components).

Optional forecast parameters: outlier correction [Page 60], logging statistical key figures [Page 61], ignoring initial zeros [Page 62].

Seasonal Trend Exponential Smoothing

The forecast is calculated according to Winter/Holt's multiplicative method and is suitable if historic values display seasonal fluctuations from a rising or declining trend. Here the extent of the fluctuation depends on the strength of the trend.

> Ice cream sales in summer: It is taken for granted that ice cream sales rise by a trend of 10% annually. A seasonal increase of 30% each summer then leads to ever greater absolute fluctuations.

- Mandatory forecast parameter: *Periods per Season*

Smoothing factor settings: *Alpha* (Base Value [External]), *Beta* (Trend Value [External]), *Gamma* (seasonal components).

Optional forecast parameters: trend dampening [Page 61], outlier correction [Page 60], logging statistical key figures [Page 61], ignoring initial zeros [Page 62].

Croston Method

The Croston method was developed specifically for sporadic trends. This procedure uses exponential smoothing to calculate a mean time interval between the values in the time series that are not equal to zero.

> For more information, see http://help.sap.com/ → *Documentation* → *mySAP Business Suite* → *mySAP Supply Chain Management* → *SAP Supply Chain Management* → *SAP Advanced Planning and Optimization (SAP APO)* → *Demand Planning* → *Demand Planning Process* → *Definition/Redefinition of Forecast Models* → *Creating a Master Forecast Profile* → *Forecast Strategies* → *Croston Method*.

Check whether you want to aggregate the data in order to remove the gaps in the time series so that you can use procedures that consider trend or seasonal time series patterns. You can aggregate data in this way by choosing an imprecise time characteristic (month instead of day) or by forecasting values for product groups instead of individual products.

4.2.1.1.1.4.2.1.2 Automatic Model Selection

Use

Automatic model selection allows you to let the system determine which forecast model best fits your historic data.

> We recommend that you use automatic model selection if you do not know the trend of your historic data, if you cannot estimate how your data will develop, or if you do not want to specify a model.

Features

The system performs a number of tests and uses the results to determine the model to be used (see Forecast Strategies [Page 55]). If the model chosen is exponential smoothing, the system optimizes the relevant smoothing factors (*Alpha*, *Beta*, *Gamma*).

Note that automatic model selection requires a high calculation effort. This is particularly true of the seasonal trend model. The calculation effort also depends on the scope of the search space and the precision of the step sizes that you have set.

Activities

1. First the system tests for sporadic historic data by determining the number of periods that do not contain any data for the history key figure. If this number accounts for more than 66% of the total number of periods, the system automatically terminates model selection and uses the Croston method.
2. Then the system tests for white noise [External]. If there is white noise, the system automatically uses the constant method.
3. If both tests are negative, the system tests for seasonal and trend effects.
   a. First the system deletes existing trends. To test for seasonal effects, the system determines the auto-correlation coefficients. If the auto-correlation coefficient is greater than 0.3, the test is positive.
   b. To test for trend effects, the system determines the trend significance parameters. If the seasonal test is positive, the system removes possible seasonal effects. If no seasonal tests are determined, the system runs the test using the number of past periods minus 2. If seasonal effects are determined, it runs the test using the number of periods in a season plus 1.

   Since the results of these tests determine the model that the system checks in the next step, the *Periods per Season* parameter is highly significant. If your historic data contains, for example, a season with seven periods and you enter the value “3“ for *Periods per Season*, the seasonal test will probably be negative. In this case, the system does not check the seasonal model but checks the trend and constant models only.
4. The system uses the selected model (see the table below) and performs the forecast. It calculates all error measures. In models that use forecast parameters (*Alpha*, *Beta*, *Gamma*), these parameters vary to reflect the areas and step sizes specified in the forecast profile.

Test results and model selection

	Test for White Noise	Test for Sporadic Data	Seasonal Test	Trend Test
Croston model		X		
Trend model				X
Seasonal model			X	
Trend-seasonal			A	A

Linear regression			o	X
Seasonal linear regression			A	A

Legend for the characters in the table:

X – model is used if the test is positive.

A – model is used if all tests are positive.

o – model is used if the test is negative.

The constant model always runs, unless the test for sporadic data is positive. In this case, the Croston model is used exclusively (as a special variant of the constant model).

The system chooses the model with the parameters that produce the lowest error measure. The error measure is specified by the selection made in the *Error Measure* field in the forecast profile.

4.2.1.1.1.4.2.1.3 Outlier Correction

Use

Outliers are atypical values that cannot be explained by the forecast model. The results of the forecast can be heavily influenced by outliers.

The system is able to identify and replace outliers in the historic data. To do this, the forecast procedure calculates forecast values in the past period and compares them to the observed values. If the difference (the residual) exceeds a specific value, the observed value is replaced by the ex-post-forecast value for the corresponding time. After this correction, the forecast calculation is performed again with the amended historic data.

You determine a Sigma factor in order to define the threshold value.

Integration

Outlier detection depends on the individual forecast model because the forecast value is calculated using a model and its related algorithm.

Features

In the context of the forecast function, outliers in the historic values are defined by the Sigma factor. The greater the Sigma factor the more tolerant the system of atypical values, and the fewer outliers are determined.

The Sigma factor *fac* controls outlier detection in the following way: An observed value *y* is declared to be an outlier if the difference to the forecast value *e* is greater than *fac* *s. *s* denotes the standard deviation of the residuals.

Activities

Activate outlier correction if you think that outliers are having an unfavorable effect on the forecast result in accordance with the forecast function definition.

Do not select outlier correction if atypical values are always to be taken into consideration by the forecast.

4.2.1.1.1.4.2.1.4 Trend Dampening

Use

For forecast models with a trend component (*linear regression*, the trend models for *exponential smoothing with or without seasonality,* and, in some cases, *automatic model selection*), you can dampen the trend for future forecast values by specifying a trend dampening factor.

Use the trend dampening factor if you expect that the past growth rate will either slow down or intensify in the future.

Features

The trend dampening factor is a number that is multiplied by the trend value (growth rate) for the calculation of each forecast value at the respective time. In this way, growth in a long-term trend is further slowed or intensified:

- With a trend dampening factor that is less than 1, a type of saturation effect is produced. The trend value will exponentially converge to 0.

  A trend dampening factor of 0.9 decreases the growth rate for each period by 10% recursively.

  If, for example, the number of cars sold is currently growing by 1000 per period, the growth in the number of cars sold in the next period would be 0.9 * 1000 = 900, and would be 0.9 * 900 = 810 in the period after that, and so on.

- To achieve the reverse effect, you can enter a trend dampening factor that is greater than one.

Activities

You use this option to perform trend dampening. Specify a trend dampening factor that corresponds to your expectations.

4.2.1.1.1.4.2.1.5 Logging Statistical Key Figures

Use

This function logs statistical information about the forecast calculation.

After the forecast function has been executed, the system displays the corresponding statistical information and the error messages for the forecast calculation in the log.

Features

The following statistical information is logged:

- Error totals
- Mean absolute deviation (MAD)
- Mean absolute percent error (MAPE)
- Mean percentage error (MPE)

- Mean square error (MSE)
- Root of the mean square error (RMSE)
- Number of outliers (where outlier correction is active)
- Selected forecast model (with automatic model selection)
- Smoothing factors (with optimization of smoothing factors)

Activities

You choose this option in order to log statistical information about the forecast result.

4.2.1.1.1.4.2.1.6 Ignoring Initial Zeros

Use

You use this function to determine whether a string of zeros at the beginning of a selected period in the past are to be taken into consideration in the forecast.

> You want to calculate forecast values for your product. The past period covers the last 24 months. However, the time series of products that have only existed for a few months begins with a lengthy sequence of zero values. You do not want the forecast values to be influenced by these zeros so you select the option *Ignore Initial Zeros*.

Activities

Choose this option if initial leading zeros are to be excluded from the forecast calculation.

4.2.1.1.1.4.3 Standard Key Date in Planning Functions

Definition

The standard key date is a key date that is used in the entire planning model. It can be set for every real-time enabled InfoCube in the characteristic relationships.

Use

The following options are available to set a key date.

- *Unspecified* (default value is the system date if all of the participating InfoCubes have the key date = *unspecified*).
- *Fixed Date*
- *From Variable*

All objects that depend on the key date, such as the filter or hierarchies that are used in the filter or in the from and to values of the copy and distribution functions can be set to the standard key date.

If a planning function is executed, the system calculates the standard key date as follows.

- If the aggregation level was created directly on a real-time enabled InfoCube, the standard key date is the same key date that is set in the characteristic relationships on this InfoCube.
- If the aggregation level was created on a MultiProvider, all participating real-time enabled InfoCubes below the MultiProvider are checked. If all of these have the setting unspecified, the standard key date is the current system date. If one of the real-time enabled InfoCubes has another setting for its key date, the first one that returns a certain value wins. Variables are analyzed upon return.

All time-dependent objects continue to provide the option of using your own key date in your concrete application.

This can be useful in the following case: you want to use a specific hierarchy that is to be read with another key date than the master data attributes.

Integration

You make the key date setting for planning in the *Planning Modeler* in the *InfoProvider* area on the *Settings* tab page.

4.2.1.1.1.5 Planning Sequences

Use

Planning sequences are used for grouping planning functions within BI Integrated Planning. They allow you to save groups of planning functions in a sorted sequence, as well as sorted execution of the whole group.

Integration

Planning sequences can be edited, saved and tested in the Planning Modeler.

They can be included in a process chain as a step. They can also be linked with a variant of variable values there.

Features

A planning sequence can be made up of one or more steps. The following types of steps:

Step types of a planning sequence

Type	Description
Planning function	Based on an aggregation level, the system saves a planning function and a filter with which the planning function is executed.

Input template	You can embed input templates into the sequence especially for testing planning functions. Input templates are defined with an aggregation level and a filter. If a planning sequence is executed as a whole, the embedded input templates are not taken into account.

Activities

Testing a Planning Sequence in the Planning Modeler

In the Planning Modeler, you can test planning sequences as a whole or step by step using *Execute*. In the latter case, the input templates appear at the lower end of the application.

During execution of the steps, the system keeps the data in the buffer. Once you choose *Save*, the transaction data from the buffer is written to the database.

If the planning sequence includes input-ready variables, you can set them manually before execution of the steps and save the value combinations of the variables as a variant.

Execution of a Planning Sequence within a Process Chain

A planning sequence can be embedded in a process chain as the *Execute Planning Sequence* process type. The planning sequence can be linked with a stored variant for variable values.

4.2.1.1.1.6 Variables

Use

Variables are used to parameterize a query, a planning function, a filter, a characteristic relationship or a data slice. When a query, planning function or Web application is executed, they are filled with values.

Variables act as placeholders for characteristic values, hierarchies, hierarchy nodes, texts, and formula elements, and can be processed in many different ways.

- Depending on the objects for which you want to define variables, there are different **variable types**. For more information, see the documentation on the Query Designer under Variable Types [External].
- The **processing type** determines how a variable is filled with a value for the runtime of the query, planning function or Web application. For more information, see the documentation on the Query Designer under Processing Types for Variables [External].

> When you use variables, you can use **one** planning function definition, for example, as the basis of several different planning functions: you want to create a planning function of type *Copy* that copies your current data from the current version to another version. You use a variable for the *Version* characteristic in the To parameters of the planning function. Before you execute the planning function, you decide which version the current data is to be copied to.

The *Replacement Path* processing type is **not** available for formula variables that are used in planning functions, for example for conversion functions. Only the *Number* dimension is supported here.

You **cannot** create text variables in the planning modeler. In input-ready queries, however, you can use text variables without any restrictions.

Integration

Variables as Reusable Objects

Variables do **not** depend on an InfoProvider, but on the respective InfoObject only. A variable that you define for an InfoObject is available in all InfoProviders that use this InfoObject.

Variables can be defined in the Query Designer or in the planning modeler or planning wizard.

For example, when you define a variable for a planning function in the planning modeler, it is available for reuse for all queries or planning functions.

Features

Variables help you to flexibly set or parameterize your objects. The following objects support the use of variables:

Using variables

Object	**Using Variables**
Queries (especially input-ready queries)	• For example, to parameterize characteristic restrictions in the query • In formulas, conditions, exceptions and as a placeholder for text
Filter	To parameterize characteristic restrictions that describe the filter.
Planning functions	Depending on the respective planning function type, to parameterize conditions and parameters, for example, to parameterize the conversion factor in planning functions of type conversion.
Characteristic relationships	• To parameterize the hierarchy used • To parameterize selection from a DataStore object
Data slices	To parameterize characteristic restrictions that describe the data slice.
Additional objects	To parameterize the presentation hierarchy in the query.

Variables that are used in characteristic relationships and data slices cannot call a dialog for the manual entry of values. These variables must have a value at the time of execution.

In the planning modeler or planning wizard, as well as in the Query Designer, the following tools are available for creating and changing variable definitions **according to context**.

Tools for creating, changing and displaying variables

Tool	Description
Variable wizard	The wizard takes you through the process of **creating** a variable step-by-step. Each individual step is context-sensitive and is modified according to the combination of variable and processing types used. This means that the variable wizard only offers the selection options that are permitted for that combination of variable and processing types. For more information, see the documentation on the Query Designer under Defining Variables [External]. Note the following when you use the variable wizard in the context of the planning modeler or planning wizard: • The first step is *General Information*. • The system shows further dialog steps according to context: ○ Details [External] ○ Default values [External] ○ Replacement path [External] ○ Currencies and units [External] (There is no *Characteristic* dialog step).
Variable editor	The variable editor dialog box offers all the options for **changing** an existing variable. The individual tab pages of the dialog box show the previous settings for the variable. Not all settings for variables can be changed in the variable editor. For example, the variable type and processing type cannot be changed once the variable is created. Note the following when you use the variable editor in the context of the planning modeler or planning wizard: in addition to the change mode, the variable editor also has a **display mode**.

Activities

The tools for creating and changing variables are available wherever you enter constant values and can use variables.

In the planning modeler or planning wizard, you can also display and delete variables. In the restriction dialog for a characteristic in the **filter**, you see the list of the variables available for the selected characteristic in the *Variables* view for single values. The functions *Create, Edit, Display* and *Delete* are available here.

- If you choose *Create*, the variable wizard opens.

The variable wizard opens with the settings made for the current context: For example, if the variable wizard opens in the dialog for characteristic restriction, it has the settings that are appropriate for the type of variable (characteristic variable), the appropriate characteristic and the basic characteristic, as required.

- If you choose *Edit*, the variable editor opens and you can change certain properties.
- If you choose *Display*, the variable editor opens in display mode.
- If you choose *Delete*, the system deletes the selected variables.

For more information about using these two tools in the Query Designer, see Variables [External].

4.2.1.1.1.6.1 Changing Variable Values in the Planning Modeler

Use

In the planning modeler, you want to check a **planning function** that can be parameterized using variables or execute a **planning sequence** that contains objects that can be parameterized using variables. In both cases, you want to change the values of the variables in the planning modeler:

- In order to check a **planning function**, you have to assign variable values to all the variables that are set as mandatory; a dialog box appears in which you enter values for the variable.
- When you execute a **planning sequence**, you can change the values of the mandatory and optional variables in the appropriate input area at any time; the planning sequence is processed using the values that you enter.

You can also personalize individual values by assigning a particular value to it. This value is valid until you reset the personalized setting. Personalization means that you do not have to manually configure your personal variable settings each time. In this instance, BI Integrated Planning uses BEx personalization for variable values (see Personalization in BEx [External]).

In the planning modeler, the variable values entered in the current session are noted and set automatically, if they are required. This ensures that you do not have to manually maintain the same variables several times.

Features

In the planning modeler, you can change variable values to check a planning function or execute a planning sequence, if the variables are being used in the following objects:

- Variables in filters (see Filter [Page 36])
- Variables in planning functions (see Planning Functions [Page 39])

Depending on how the variable is defined and used, the variable values can come from different sources. Generally the following options are available:

- Default value for the variable
- Personalization for the variable
- Variants saved previously
- Variable values entered previously (automatic)
- Values maintained manually

Activities

You are on the *Planning Functions* tab page and have chosen the check function for a planning function with required entry variables, or you are on the *Planning Sequences* tab page in display mode for a planning sequence that contains variables. In the first case, the system shows a dialog box for entering variables; in the second, an input area. It lists all the variables that are available in the current application context. The following activities are possible:

Manual Maintenance of Variable Values

If you have chosen one of the layouts that contains the *Key*, you can enter variable values manually in the lead column of the variable value.

To display the lead column of the variable value, select the appropriate layout or change the table settings accordingly.

In addition, input help is available for selecting values for some variable types.

Personalizing Variables

1. To personalize values for variables or reset personalized values, choose *Personalized Variables*. A dialog box appears in which you can choose personalized variables.
2. To personalize variables, select one or more variables from the table of available variables and insert these values. To reset personalized values, choose one or more variables from the table of personalized variables and choose *Remove*.
3. You can also change the values in the personalized variables table at a later time.
4. Personalized variables are not displayed in the dialog box or input area by default. If you want to display personalized values, choose the *Display Personalized Variables* option.
5. Choose *OK*. The dialog box for entering variables or the corresponding input area appears.

In the table settings, you can display a column for personalized values and a column for resetting the personalized values. The column for personalized values indicates a personalized variable; the column for resetting the personalized value allows you to reset the current variable value to the personalized value.

Selection and Maintenance of Variants

In the *Available Variants* selection box, you can select an existing variant. The values determined for the variables in the variants are automatically set as the current values.

- To save the current values as a new variant, choose *Save*. The *Save Variant* dialog box appears. Enter a description for the variant and select *OK*.
- To change an existing variant, select an existing variant from the *Available Variants* selection box. As a result, the current variables contain the variable values from the variant. Change the values manually and choose *Save*.
- To save an existing variant as a new variant, select an existing variant from the *Available Variants* selection box. Choose *Save As*. The *Save Variant As* dialog box appears. Change the description of the variant and select *OK*.
- To delete an existing variant, select an existing variant from the *Available Variants* selection box. Choose *Delete*.
- To delete the properties of an existing variant, select an existing variant from the *Available Variants* selection box. Choose *Properties*. The *Change Variant Properties* dialog box appears. Change the properties of the variant and choose *OK*.

> An existing variant can be marked as an initial variant. This variant is set automatically each time variable input is called. To do this, you select an existing variant from the *Available Variants* selection box. Choose *Properties*. The *Change Variant Properties* dialog box appears. Choose *Use as Initial Variant* and choose *OK*.

Check Current Variable Values

To check the current values, choose *Check*.

4.2.1.1.1.7 Input-Ready Query

Use

You use input-ready queries to create applications for manual planning. These can range from simple data entry scenarios to complex planning applications.

Integration

You define a query that you want to use for manual planning in the BEx Query Designer (see Defining New Queries [External]).

In the Web Application Designer or the BEx Analyzer, you can combine the input-ready queries with other queries and planning functions to create complex planning applications.

Prerequisites

You can define an input-ready query on any of the following InfoProviders:

- Aggregation levels (see Aggregation Level [Page 31])

- MultiProviders that include at least one simple aggregation level

The aggregation levels are created in the planning modeler; MultiProviders are defined in the modeling functional area of the Data Warehousing Workbench.

Features

Definition of an Input-Ready Query

Once you have defined a query on an InfoProvider, you see the *Planning* tab page under the *Properties* of structural components (for example, in key figures or restricted key figures). The options provided there allow you to determine which structural components of an input-ready query are to be input ready at runtime and which are not. With structural components that are **not** input ready, you can also determine whether these components are viewed as reference data or are just protected against manual entry.

For the structural components, you also have the following options:

Input readiness of structural components of a query

Option	Description
Not input ready (reference data)	If they are being used as reference data, the structural components are **not** protected by data locks to ensure exclusive access for one user because this data serves as a reference for many users. This is the default setting.
Not input ready (no reference data)	If you want to protect structural components against manual entries but allow changes by planning functions, you can use locks to protect this data for one particular user. In this way you can ensure that the planning function works with the displayed data only and not with data that has been changed by other users.
Input ready	

You can also determine whether an input ready query is to be started in **change mode** or in **display mode**. You find this property in the *Query Properties* on the *Planning* tab page. If there is at least one input-ready query component, the query (as long as it has not been determined otherwise) is started in display mode.

In BI applications that use input ready queries as data providers, you can enter data manually at runtime. For more information, see Performing Manual Planning [Page 80] and Creation of Planning Applications [Page 74].

Example

You want to create an input-ready query for manual planning for a plan-actual comparison of revenues for a set of products. You want the plan data in a real-time-enabled InfoCube and the actual data in a standard InfoCube.

1. Create a MultiProvider that includes the InfoCubes for the plan and actual data.

2. Define an aggregation level on the MultiProvider which contains the characteristic *Product* and the key figure *Revenue*.
3. On the aggregation level, create two restricted key figures *Plan Revenue* and *Actual Revenue*. For restriction, choose the characteristic 0INFOPROV and restrict it to the plan or actual InfoCube.
4. Add the restricted key figures to the key figure structure. Insert *Product* into the rows. For *Plan Revenue*, choose *Input Ready* for the input-readiness option. For *Actual Revenue*, choose the option *Not Input Ready (Reference Data)*.
5. In the query properties, set the indicator that determines whether the queries are started in display or change mode as required.

Example of an input-ready query

Product	***Plan Revenue***	***Actual Revenue***
P01		20
P02		30

If you want to keep actual and plan data in a real-time enabled InfoCube, you do **not** require a MultiProvider for the task described above. Create an aggregation level on the InfoCube and define the input-ready query for the aggregation level. In the example above, a version characteristic acts as the InfoProvider. Create restricted key figures with the plan or actual version and proceed as in the previous example.

4.2.1.1.1.8 Implementing Planning Function Types

Use

Planning function types are parameterizable processes to change transaction data within BI Integrated Planning. The system offers you a number of default planning function types (such as copy, delete, reposting, revaluation, distribution by reference data or by key, unit and currency translation or FOX formula.

You can also implement customer-specific planning function types in order to realize specific processes and apply them to transaction data. Each planning function type comprises

- a definition part (metadata) that is created and changed in a transaction (RSPLF1),
- a ABAP-OO class in which the actual process is programmed. The class name is an element of the definition part.

Integration

In reference to transport, Business Content and activation, planning function types behave like other metadata objects of the BI system.

The active planning function types are visible in the Planning Modeler and can be used to create and execute planning functions.

Features

In the maintenance transaction for planning function types, you can display, change or create customer-specific planning function types.

Activities

Creating Planning Function Types

1. To get to the maintenance transaction for planning function types, on the *BI Integrated Planning* screen in the *Administration and Development* screen area, choose *Maintain Function Types*. The *Edit Planning Function Type* screen appears.
2. Enter a technical name for the planning function type.
3. Choose *Create.* The screen for creating a planning function type appears.
4. Specify a description for the planning function type and make the required settings on the *Properties* and *Parameters* tab pages.

Tab Page	Description
Properties	Screen areas: • *General Data* • *Implementation:* Enter the name of the ABAP class that implements the process. The ABAP class has to implement one of the two interfaces: ○ IF_RSPLFA_SRVTYPE_IMP_EXEC ○ IF_RSPLFA_SRVTYPE_IMP_EXEC_REF The latter interface is relevant when you need reference data for your process. The implementation of methods for the interface listed is optional with the exception of the method EXECUTE. In addition, the class can implement check methods that are executed at runtime. The interface IF_RSPLFA_SRVTYPE_IMP_CHECK serves this purpose. For more information, see the F1 Help for the input field for the ABAP class name. • *Interface Characteristic Usage* • *Interface Parameter*

Parameters	Create the required parameter using *Create Parameters* in the context menu of an object in the hierarchy parameter. The following parameter types are available: • **Elementary**: The value of an elementary parameter is the characteristic value of a specific InfoObject. This means that every elementary parameter is based on an InfoObject and thus inherits its technical properties. If the InfoObject is a characteristic, the system automatically checks the permissions of a value entered by the user against the master data. • **InfoObject of the InfoProvider**: This parameter can include the name of an InfoObject from the current InfoProvider (aggregation level). The InfoObjects permitted are determined with the restriction of the InfoObject selection. • **Data Selection**: Data selection parameters can include the selection criteria for multiple characteristics as they are needed for defining the filters. This is a specific selection table. The permitted characteristics are defined by the restriction of the characteristic selection. • **Structure**: Parameters can be combined into one structure. These parameters then comprise the components of the structure parameter. If you define the structure parameters for the table, the structure forms the row structure of the resulting table. • **Key Figure Selection**: This parameter type is used to select the processed key figures. This means it is a special case of type “InfoObject”. For more information, see the F1 help in the *Function Type* input field. With ▲ and ▼, you can change the sequence of the parameters. The sequence of the parameters is taken into account, when you create a planning function for this planning function type.

Example

The planning functions delivered by SAP are based on the same technical concept as the customer's own planning functions and can thus be viewed in the maintenance of the planning function types.

The function type *Delete* (0RSPL_DELETE) is a simple example. There is only one parameter (KYFSEL) for selection of the key figures to be deleted. In the associated ABAP class, the interfaces IF_RSPLFA_SRVTYPE_IMP_CHECK and IF_RSPLFA_SRVTYPE_IMP_EXEC are implemented.

4.2.1.1.2 Creation of Planning Applications

Purpose

Planning applications are BI applications that are based on a planning model. Power users combine the objects of the planning model into an interactive planning application which allows data to be entered and changed automatically or manually by users.

Planning model objects include:

- InfoProviders that contain data (see InfoProviders [Page 24])
- Aggregation levels as InfoProviders which provide a set of data with a particular level of granularity for data entry and change (see Aggregation Levels [Page 31])
- Input-ready queries which allow you to make manual entries for the aggregation level (see Input-Ready Queries [Page 69])
- Planning functions which allow automated changes to be made to data in the aggregation level and therefore model a part of the data flow (see Planning Functions [Page 39]).

In addition, planning sequences can belong to the planning model (see Planning Sequences [Page 63]).

Tools are available for creating planning scenarios. These tools are also used in reporting scenarios.

- For Excel-based planning applications: *BEx Analyzer*
- For Web-based planning applications: *BEx Web Application Designer*

Prerequisites

You have created the necessary planning model objects.

You have installed the frontend.

Example

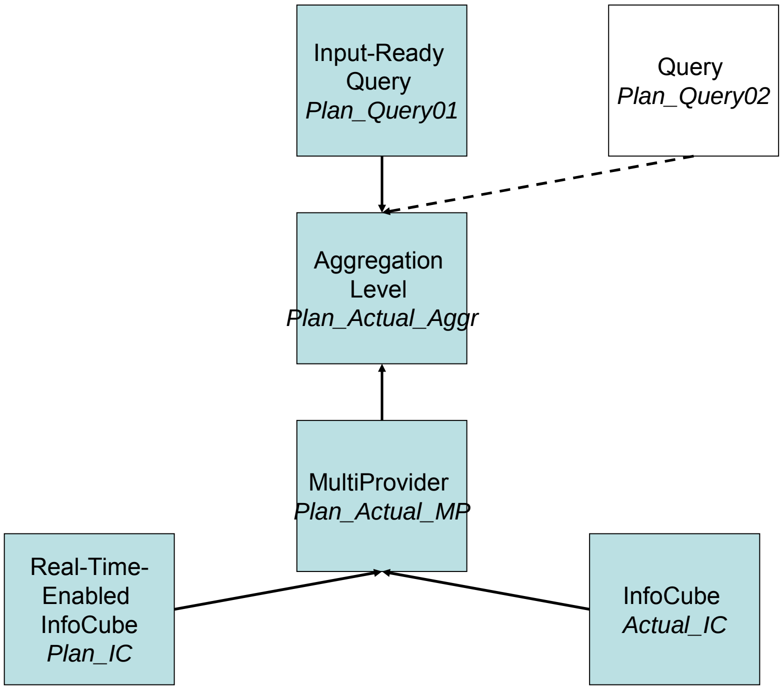
In order to illustrate the basic procedure, the following example will show you how to create an Excel-based planning application and a Web-based planning application on the basis of a simple planning model.

The underlying planning model consists of the following objects:

- A real-time-enabled InfoCube *Plan_IC* containing the planned sales data for next year
- A standard InfoCube *Actual_IC* containing the sales data for the previous year
- A MultiProvider *Plan_Actual_MP* containing the two InfoCubes
- An aggregation level *Plan_Actual_Aggr* for MultiProvider *Plan_Actual_MP*
- An input-ready query *Plan_Query01* which displays the actual and plan data and allows plan data to be entered manually.

The two InfoCubes contain the same characteristics and have at least one common key figure; the only difference is the key figure *Year*. One of the characteristics is *Country*. This has to be included in the query.

The following graphic shows how the objects in the planning model are related:



The planning application contains at least one of the following elements:

- A planning function for copying the data from the InfoCube containing actual data into the InfoCube containing plan data *PF_Copy*
- A selection list (dropdown box) that allows you to navigate in the query
- A planning function for revaluating plan data *PF_Revaluate01* (revaluate by %) or *PF_Revaluate02* (with a fixed percentage), where the selections in the selection list determine which data is to be revaluated
- A planning function for saving plan data
-

See also:

Creating Planning Applications in BEx Analyzer [Page 75]

Creating Planning Applications in the BEx Web Application Designer [Page 78]

4.2.1.1.2.1 Creating Planning Applications in the BEx Analyzer

Procedure

The following example shows how you create a workbook with a title, table, selection list and special pushbuttons (for functions such as copy, revaluate by %, delete and save).

1. Open the BEx Analyzer.

2. Check that you have the necessary security settings. Choose *Tools* → *Macro* → *Security*. Switch to the *Trusted Publishers* tab page and set the indicator for *Trust access to Visual Basic Project*.

   You only have to set this indicator when you are creating the planning function. You can deselect the *Trust access to Visual Basic Project* indicator afterwards.

3. Choose *New* to create a new workbook.

4. To enter the **title**, navigate to the appropriate cell in the workbook, enter the text and assign a suitable font size.

   In this example, the text *Sales Planning* is entered in cell B2 and given font size 18.

5. To start designing the workbook, choose to switch to **design mode**. The *SAP Logon* dialog box appears.

6. Log on to the BI server that you want to use.

7. To display the results of the query with the actual and plan data and enter plan data manually, navigate to the appropriate cell in the workbook and choose to insert an **analysis grid**.

In this example, we navigate to cell B6 and insert the analysis grid here. For more information about this design item, see Analysis Grid [External].

8. In the context menu of the analysis grid, choose *Properties*. The *Analysis Grid Properties* dialog box appears.

9. On the *General* tab page, create a new data provider. The *Create Data Provider* dialog box appears. In the *Data Provider* field, the system displays the name that is currently assigned to the data provider.

In this example, we create data provider *DP_01*. For more information, see Configuring Data Provider [External].

10. To determine the start view of the data provider, choose *Query/Query View*. The *Open* dialog box appears.

11. Select the required query or query view and choose *Open*. The system inserts the name of the InfoProvider on which the query or query view is based into the *InfoCube* field.

    In this example, we assign query *Plan_Query01* to data provider *DP_01*. The underlying aggregation level is *Plan_Actual_Aggr*.

12. As you want to save the navigational state after navigation steps have been performed in BEx Analyzer and plan data has been entered, deselect the *Reference the View* indicator.

13. When you have configured your data provider in this way, choose *OK*. The *Analysis Grid Properties* dialog box appears again.

14. Make sure that the *Apply Formatting* and *Allow Navigation* indicators are set and choose *OK*.

15. To select the values of a dimension in the query as a filter, navigate to the appropriate cell in the workbook and choose to insert a **dropdown box**.

In this example, we navigate to cell B4 and insert the dropdown box here. For more information about this design item, see Dropdown Box [External].

16. In the context menu of the dropdown box, choose *Properties*. The *Dropdown Box Properties* dialog box appears.

17. On the *General* tab page, select the configured data provider (*DP_01* in this example), and set the *Display Label* indicator.

18. On the *Dimensions* tab page, select the dimensions for which you want to be able to select values in the dropdown box. Make sure that the *Posted Values (Q)* entry is selected in the *Read Mode* field so that only posted values are displayed.

    In this example, we select the dimension *Country*.

19. For each special function that you want to include, choose to insert a design item of type **Button**.

In this example, we add pushbuttons in cells B18, B20; B22 and D22. For more information about this design item, see Button [External].

20. In the context menu of the button, choose *Properties*. The *Button Properties* dialog box appears. Enter a text for the pushbutton and static parameters in accordance with the following example:

Special Function Buttons

Button Text/ Command Range	**Static Parameters: Name**	**Static Parameters: Value**
Copy	CMD	EXECUTE_PLANNING_FUNCTION
	PLANNING_FUNCTION_NAME	*PF_Copy*
	DATA_PROVIDER_FILTER	*DP_01*
Revaluate by % Command range: $A$30:$C$30	VAR_NAME	OF_REVALUATION_FACTOR
	DATA_PROVIDER_FILTER	*DP_01*
	PLANNING_FUNCTION_NAME	*PF_Revaluate01*
	CMD	EXECUTE_PLANNING_FUNCTION
Delete	PLANNING_FUNCTION_NAME	*PF_Delete*
	DATA_PROVIDER_FILTER	*DP_01*
	CMD	EXECUTE_PLANNING_FUNCTION
Save	CMD	SAVE_AREA

21. As the workbook contains all the necessary elements, you can exit design mode. You do this by choosing .

22. Create the command range for the function *Revaluate by %*.

    In this example, we navigate to cell A30 and insert the text VAR_VALUE. Then we navigate to cell B30 and enter "0". In cell C30 we enter "=C20".

23. If you want to see a plain **background** instead of the table layout, select all rows and columns and choose *Fill Color*.

    In this example, we choose *White*.

    If you only want to hide the gridlines, you can change this setting. Choose *Tools* → *Options* → *View*; under the *Window options* group header, deselect the *Gridlines* indicator.

24. Choose 💾 to save your workbook. For more information, see Save [External].

Result

You have an input-ready query in the analysis grid in which you can enter plan data manually. You can use the planning functions you have created to calculate plan data. The data set is determined by the navigational state of data provider DP_01. Restrictions on structure elements (restricted key figures, for example) are not taken into consideration.

Test your workbook by entering, for example, values in cell C20 as revaluation factors.

For more information about the context menu of the cells in the analysis grid, see Manual Planning Functions [Page 83].

4.2.1.1.2.2 Creating Planning Applications in the BEx Web Application Designer

Use

The following simple example shows how you create a Web application with a selection list (button group), table and special pushbuttons for functions such as copy, revaluate (with a fixed percentage) and save.

Prerequisites

You have a planning model that contains aggregation level *Plan_Actual_Aggr* (defined on the basis of MultiProvider *Plan_Actual_MP*), query *Plan_Query02* (which is not input-ready), a filter and planning functions for copying *PF_Copy* and revaluating *PF_Revaluate02*.

You are familiar with the functions of the BEx Web Application Designer (see Creating a Web Application [External] and Creating Web Applications with the BEx Web Application Designer [External]).

Procedure

1. Open the BEx Web Application Designer.
2. Choose *Create New Web Template*.

   For more information about the Web template concept, see Web Template [External].

3. Double-click on *New Data Provider* to create a new **data provider** of type *Query View Data Provider*. The *Maintain Data Provider* dialog box appears.

   For more information about the data provider concept, see Data Providers in BI Applications [External].

4. In the *Name* field, the system displays the name it generated for the data provider. This is currently assigned to the data provider. You can retain this name.
5. In the *Define Data Provider Type* area, choose the *Query* option. Choose *Select Query*. The *Open* dialog box appears.
6. Select the query you want and choose *Open*. In the *Query* field, the system inserts the name of the query.
7. When you have defined your data provider in this way, choose *OK*.
8. Drag and drop the required **Web items** onto the *Layout* tab page of your Web template.

   In this example, we insert the following Web items:

   - Dropdown box
   - Analysis
   - Button group

   For more information, see Web Items [External].

9. If you click on a Web item or choose *Properties* in the context menu of a Web item, the system display the *Properties* screen area.
10. On the *Web Item Parameters* tab page, enter the required data in the highlighted fields.

    For the Web items in this example, the following data is required:

    For the **dropdown box** (see Dropdown Box [External]) under parameter *Data Binding → Data Binding Type → Selection of Characteristic*:

    - *Data provider*: *DP_01*
    - *Characteristic: Country*

    Set the indicator for the *Label Visible* parameter for the dropdown box so that the label is displayed for *Country*.

    The system uses the data provider you create first for all additional Web items. If you have performed the activities in the order described here, the system inserts data provider *DP_01* under the *Data Binding* parameter for the **Analysis** Web item (see Analysis [External]).

For the **button group** (see Button Group [External]), under parameter *Internal Display → List of Buttons → Caption* on the *Text Editing* dialog box (with option *Language-Independent Text*) we enter a text for the required pushbuttons. We choose the *Command* field to assign a suitable command to each button (see Command Wizard [External]). The *Edit Command* dialog box appears. On the *All Commands* tab page, select *Commands for Planning Applications* and choose the required functions in accordance with the following examples:

Special Function Buttons

Button Text	Command	Parameter
Copy	Execute planning function (simple) (EXEC_PLANNING_FUNCTION_SIMPLE)	*Data Binding → Reference to Data Provider of Type Filter*: *DP_01*
		Command-Specific Parameters → Planning Function: Select your copy planning function (*PF_Copy* in our example)
Revaluate	Execute planning function (simple) (EXEC_PLANNING_FUNCTION_SIMPLE)	*Data Binding → Reference to Data Provider of Type Filter*: *DP_01*
		Command-Specific Parameters → Planning Function: Select your revaluate planning function (*PF_Revaluate02* in our example)
Save	[SAVE_DATA]	*Data Binding*: No entry necessary

For more information, see Commands for Planning Applications [External].

11. Choose 💾 to save your Web template (menu path *Web Template* → 💾 *Save*).

12. Choose to execute your Web template.

Result

You can use the planning functions you have created to copy and calculate plan data. The data set is determined by the navigational state of data provider *DP_01*. You can save the entire Web application by choosing the *Save* pushbutton.

4.2.1.1.3 Performing Manual Planning

Use

In BI applications that use data providers that are assigned an input-ready query, the system allows the manual entry of data either in input-ready cells or input-ready new rows.

In input-ready cells, you can change individual key figures for a posted data record; in input-ready new rows, you can either change posted data records or enter new data records for characteristic combinations that do not yet exist.

Integration

You can use the BEx Analyzer to create and execute BI applications that provide this function; alternatively, you can create them in the Web Application Designer and execute them on the Web. In the BEx Analyzer, you use the *Analysis Grid* design item; on the Web, you use the *Analysis* Web item.

For more information about manually entering data in a BI application created using the BEx Analyzer, see Manual Planning Functions [Page 83].

Prerequisites

The following prerequisites must be met before you can change existing data:

- At least one input-ready key figure exists.
- No active conditions exist.

The following prerequisites must be met before you can enter data:

- No active universal display hierarchies exist.

Features

Input-Ready Queries at Execution Time

Whether the cells of an input-ready query can be changed upon execution depends on the query view and possibly other settings (such as settings for data slices and characteristic relationships).

With regard to whether the cells of a query view are input ready, note the following rules:

1. In a query that is used for manual planning, a cell is only input ready if each characteristic value of all the characteristics included in the aggregation level is unique. For this reason, none of the aggregated values on the aggregation level are input ready; totals, subtotals and inner hierarchy nodes are not input ready.
2. If a query that is used for manual planning includes a navigation attribute that is restricted using a fixed or dynamic filter or a restricted key figure, the system treats the navigation attribute as a normal characteristic. The rule listed under point 1 applies. The system only responds as if the navigation attribute is **not** part of the query if the navigation attribute is **not** restricted.
3. In a query that is defined on a MultiProvider or a complex aggregation level and is to be used for manual planning, cells are not input ready if the InfoProvider they refer back to is not a real-time enabled InfoCube or is a real-time enabled InfoCube that has been switched to load mode.
4. If an input-ready query is executed in change mode but the requested data is locked by another user, the query is started in display mode.

Input-Ready Objects (Cells and Rows)

Depending on the tool you are using, note the following:

BEx Analyzer	Web Application Designer

Format of input-ready cells All input-ready objects are displayed in format *SAPBEXinputData*.	
Format of cells that are not input ready Cells that are not input ready are not highlighted in any color. If you try to enter values in a cell that is not input ready, the system produces an error message.	*Format of cells that are not input ready* Columns that are not input ready are highlighted.

Input-Ready New Rows

Depending on the tool you are using, note the following:

BEx Analyzer	Web Application Designer
Posted data You can enter plan data irrespective of whether data has been posted. If no data is available in the analysis grid, you use input help to restrict the free characteristics to a single value and/or drag and drop the free characteristics into the rows. You cannot use characteristics in the columns and can use one structure only.	*Posted data* You can enter plan data if data has been posted.
Rows and columns You can use one structure only in the columns or rows. If data is not available in the analysis grid, you can use one structure only in the columns, but you cannot use a structure in the rows.	*Rows and columns* You can use one structure only in the columns or rows.
Format of input-ready rows Input-ready new cells are always displayed under the grid. You can enter data in all input-ready rows. The system determines the number of rows in which data has been entered by checking the cells under the grid in the first column. If one of these cells contains data, the system assumes that data has been entered for the corresponding row.	*Format of input-ready rows* You determine the number and position of the input-ready new rows in the parameters of the relevant Web item (under *Internal Display*).
Input help You access input help for a cell by double clicking on it or choosing the F4 button.	*Input help* Input help is not available.
Input You can enter a key or text. If you are working with text, the system has to be able to determine a key from the text in the logon language. For this reason we recommend that you work with keys.	*Input* You have to use the key.

Activities

Input-Ready Cells

In input-ready cells, you can change the individual key figure values for a posted data record.

Input-Ready Rows

You can enter new rows if you are using an appropriate analysis grid. In the simplest case, you use the key figure structure in the data columns and use input help to restrict the free characteristics to a single value or drag and drop a free characteristic into the rows or columns.

The system only display an input-ready new row if at least one of the cells in the row is defined uniquely with regard to the characteristics in the underlying aggregation level. When using navigation attributes and time characteristics, avoid redundancy by making sure that you do **not** have derivable characteristics in the drilldown at the same time (for example, month and quarter).

In the BEx Analyzer, note that the *Analysis Grid* design item has to be displayed in full size. You make this setting in design mode on the *Clipping* tab page in the *Analysis Grid Properties* dialog box. Choose *Full Size* as the horizontal and vertical setting.

If you want to create a Web application, first specify how many input-ready new cells you want and where they are to be displayed. You do this in the BEx Web Application Designer in the Web item parameters.

Depending on the tool you are using, proceed as follows:

BEx Analyzer	**Web Application**
You are in change mode. In the input-ready new rows, enter the required key figure and characteristic values.	Execute the Web template and in the input-ready new rows, enter the required key figure and characteristic values.
Choose *Transfer Values* to pass changed data records or new data records to the server to be checked.	Choose *Refresh* to pass changed data records or new data records to the server to be checked.
Choose *Save Values* to store the data records persistently on the database.	Choose *Save* to store the data records persistently on the database.

💡

The data is checked automatically during navigation (filter changes) and navigation is not permitted if incorrect entries exist.

4.2.1.1.3.1 Manual Planning Functions

Use

In the BEx Analyzer you can use input-ready queries to create applications that allow manual data entry. These can be simple applications in which you only use input-ready queries, or they can be complex applications for manual planning in which you combine numerous queries that are input-ready or can be defined for a particular analysis alongside planning functions that change data automatically.

The functions of the BEx Analyzer that support you in entering data are described below.

For an example of an application that is created using the BEx Analyzer, see Creation of Planning Applications [Page 74].

Prerequisites

You have defined one or more input-ready queries in order to assign data providers to them. For more information about defining an input-ready query, see Input-Ready Queries [Page 69].

Features

Manual Entry of Data in an Analysis Grid

In the BEx Analyzer, you use the *Analysis Grid* design item for manual planning. This uses the data provider you have selected to reference an input-ready query (or query view). See Analysis Grid [External]. In the analysis grid, a structural element of the query is input-ready if you set the corresponding indicator for the element in the query definition.

Note that a cell can only be input-ready if each characteristic for this cell is defined uniquely in the underlying aggregation level. The aggregation level models the levels on which key figure values can be changed. If you have free characteristics, you have to drag and drop these free characteristics into the rows or columns before cells can be input-ready.

Other settings in BI Integrated Planning, such as data slices or characteristic relationships, also affect whether data cells are input-ready.

In the simplest case, you want to manually enter or change values in **input-ready cells**.

The system uses a predefined MS Excel format template (*SAPBEXinputData*) to support you in visualizing the input-ready cells. You can format this according to your requirements so that you can distinguish input-ready cells from cells that are not input-ready.

By default, all the cells in a workbook can be changed. If you activate cell protection in the workbook, you can only change the input-ready cells. To do this, choose *Properties* in the context menu of the analysis grid and on the *General* tab page in the *Analysis Grid Properties* dialog box, set the *Enable Cell Protection* indicator.

Depending on the analysis grid, you can also enter **new rows**. In the simplest case, you use the key figure structure in the data columns and use input help to restrict the free characteristics to a single value or drag and drop a free characteristic into the rows or columns. If no data is available in the analysis grid, you use input help to restrict the free characteristics to a single value and/or drag and drop the free characteristics into the rows.

If new rows can be entered, the system displays a row under the analysis grid. This row is formatted in the same way as the input-ready cells. For more information, see Performing Manual Planning [Page 80].

Transfer Values

Changed cells and new rows are checked on the BI server. Choose ***Transfer Values*** from the context menu of a cell that has been changed. The system reads the values from the workbook, transfers them to the BI server and checks them for consistency against the planning model. The system only accepts the data if all entries are ‘correct’. As soon as the changed data is available on the BI server, it is automatically visible in all other components of the workbook for manual planning purposes. Other analysis grids that are input-ready or determined for a particular analysis are refreshed accordingly with the changed data.

💡

The values are also checked and transferred (if applicable) implicitly with each navigation.

⚠

In the workbook settings on the *General* tab page, you can change the setting that controls when plan data is transferred. We recommend that you use the default *Before Navigation* option. If you choose the *With Confirmation Prompt* option, the system displays a confirmation prompt every time before it transfers the values. If you choose the *No Transfer* option, the entered plan data is not transferred.

Save Values

Data that has been changed while the BEx Analyzer application is open is retained on the BI server (in the planning buffer) if it is consistent. In the context menu, choose ***Save Values*** to save this data on the BI server in the InfoProvider on the database.

💡

You can create functions for transferring (CHECK_VALUES) and saving (SAVE_AREA) changed data using buttons. For an example of how to create a save function, see Creating Planning Applications in the BEx Analyzer [Page 75].

Using a Formula to Change Values

In the context menu of an input-ready cell, you can choose *Change Values Using Formula* to access a tool that is similar to a calculator and supports you in entering data.

If you want to change individual cells or entire areas, highlight the area, enter a revaluation factor, for example, and choose enter or CTRL + enter. The selected area is revaluated accordingly.

💡

You can get information about the syntax that is valid here from the Office Assistant. Choose *Help* → *Show Office Assistant*.

4.2.1.1.4 Business Planning Portal Role

Use

In the portal, the *Business Planning* portal role provides users with a central point of access to Business Intelligence content, which contains the various tools of Business Planning.

Technical name of the portal role: `com.sap.ip.bi.business_planning_showcase`

Prerequisites

The system administrator for the portal has assigned the roles to the users.

Features

The *Business Planning* role in the portal contains the following subregisters:

Overview

This initial page provides an overview of the content of this portal role. In addition, it enables a **quick start** to call the Planning Model with one or more specific objects. Enter the search term (InfoProvider, aggregation level, filter, planning function, planning sequence) and choose *Start*. The system calls the Planning Modeler with the associated search results in a new window in the portal.

Planning Modeler

The Planning Modeler is the main application for access, modeling, testing and administration of all objects for which building a planning model is required in BI Integrated Planning.

Planning Wizard

The Planning Wizard helps you to quickly build a planning scenario. It leads you through all the necessary steps and includes a test environment for you newly created scenario. All objects created here can be changed and enhanced in the Planning Modeler.

BEx Web Analyzer

With the BEx Web Analyzer you can navigate in queries and analyze data. For more information, see BEx Web Analyzer [External].

With the *BEx Web Analyzer* iView, a *BEx Web Application iView* is called with the Web template 0ANALYZER. The *BEx Web Application Query String* property has value *bi_template =0ANALYSIS_PATTERN*. For more information, see BEx Web Application or Query as iView in the Portal [External]

4.2.1.1.5 Transport of Planning Objects

You use the BI transport connection and BEx transport requests to transport planning-model objects into another system (see Modeling Planning Scenarios [Page 21]).

For more information, see BI Transport Connection: Transporting Objects [External] and Transporting BEx Objects [External].

Variables and filters that you have created in the planning modeler behave like the corresponding query elements. You have transport object ELEM.

Unlike BEx objects, however, planning objects have the following distinctive feature: even if the standard transport system is switched on, they are written to the BEx transport request or requests.

This applies to the following object types from the planning area:

- ALVL aggregation level
- PLCR characteristic relationship
- PLDS data slice
- PLSE planning function
- PLSQ planning sequence
- PLST planning function type

For more information , see Transport-Relevant Metadata Objects [External].

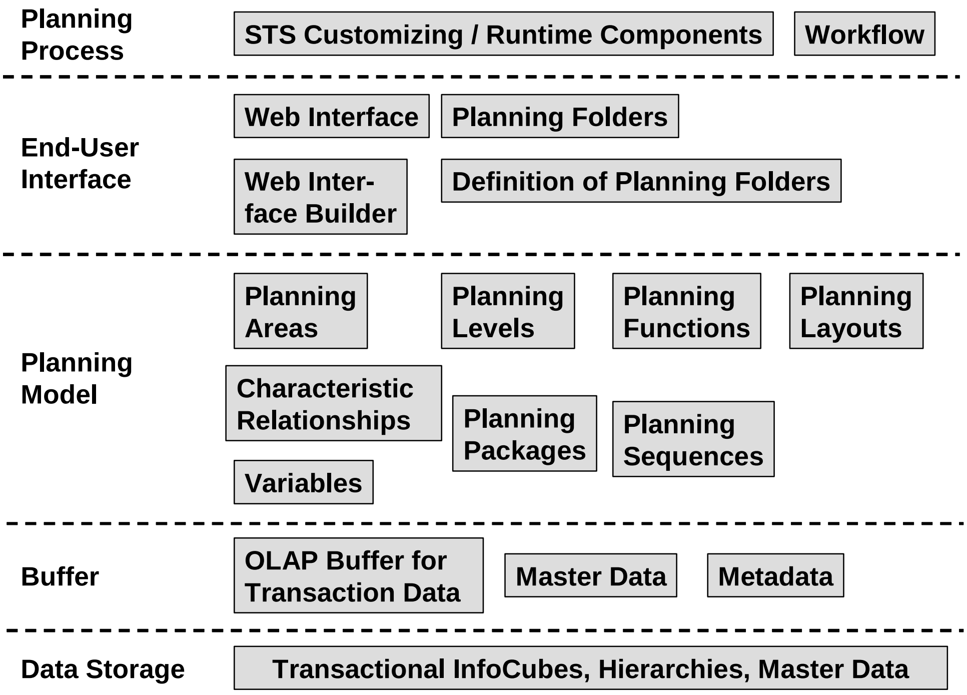
4.2.1.2 usiness Planning and Simulation (BW-BPS)

Purpose

Business Planning (BW-BPS) enables you to produce planning applications. You can either develop your own planning application or use the Business Content delivered by SAP. The area of application stretches from simple data input through to complex scenarios with data extraction, automatic planning preparation, manual data input, controlling the planning process, and retracting plan data. The architecture of BW-BPS is structured so that you are able to implement simple scenarios without great effort but are equally able to build sophisticated business-wide planning processes. Planning applications are built from the following components:

- A data basis for storing transaction and master data
- Layouts for manual data input in SAP GUI and on the Web
- Planning functions for editing transaction data
- Characteristic relationships with which you define consistency conditions for your planning application
- Planning packages and variables with which you define the work list for a planner
- A Status and Tracking System and a connection to the SAP Workflow for controlling the planning processes
- Retractors and extractors for exchanging plan data with the ERP system. These components are delivered with the Business Content.

The following graphic offers an overview of the BW-BPS architecture:



Planning Process	STS Customizing / Runtime Components Workflow
End-User Interface	Web Interface Planning Folders Web Interface Builder Definition of Planning Folders
Planning Model	Planning Areas Planning Levels Planning Functions Planning Layouts Characteristic Relationships Planning Packages Planning Sequences Variables
Buffer	OLAP Buffer for Transaction Data Master Data Metadata
Data Storage	Transactional InfoCubes, Hierarchies, Master Data

An introduction to planning with BW-BPS is offered in the section Overview of Planning with BW-BPS [Page 88].

4.2.1.2.1 Overview of Planning with BW-BPS

Purpose

In this overview you learn how to proceed generally in order to execute planning with BW-BPS. This offers an initial overview of the required and optional steps and how these are related to each other. You will find more information on the individual steps in the corresponding sections of this documentation, which is referred to each time.

Process Flow

1. If an InfoCube with the required data is not already available in your BW system, create an in InfoCube [External] with the required characteristics and key figures. Normally you supply the InfoCube with data from the operative systems of your company. You can use this data as **actual data** as the basis of your planning. With BW-BPS you can also enter completely new **plan data**, without having to refer to existing actual data.

   In both cases you need a transactional InfoCube for the plan data.

   > You can find more information under Create InfoCube [External] and, in particular, under Transactional InfoCube [Page 420].

2. Create master data, master data texts, and hierarchies for the characteristics of the InfoCube.
3. Create a planning area [Page 93]. You assign the InfoCube to this planning area.

Note that an InfoCube can be assigned to one planning area at most.

If you specify an RFC destination in a planning area you can also access data from another BW system.

4. Create characteristic relationships [Page 110] to ensure the consistency of the plan data. This step is optional.

5. Create planning levels [Page 116] for the planning area. You include a selection of characteristics and key figures from the InfoCube in these planning levels. In this way you define on which aggregation level you are performing planning. Characteristics that you do not include in the planning level are handled by the system in the following way: When reading the data, the system aggregates using all existing values in the transaction data records. When the data is saved the values of these characteristics are replaced with the initial value.

6. Create planning packages [Page 119].

   A planning package represents the quantity of transaction data on which the planning functions and manual planning operate. In this way you distinguish the work lists of the different planners. When designing a planning application you have to consider how you want to separate work lists so that planners do not mutually overwrite plan data or mutually lock data. An alternative to working with planning packages is to use user-specific variables.

   Every planning level automatically contains a planning package; the ad hoc package. The ad hoc package can be used like a package that you have created. However, while the settings of packages created by you are saved permanently, the system resets all package settings for the ad hoc package when you exit the planning session.

7. Restrict the characteristics to your desired value ranges.

   For every characteristic, decide whether you want to carry out the restriction in the planning level or in the planning package. It is mostly advisable to restrict characteristics of general significance centrally in the planning level (for example fiscal year), while characteristics whose values describe certain subtasks, should be restricted in the package (for example planning for article 100 to 200, customer 1000, company code 2000 and 2100).

   Try to restrict the characteristic values in the planning level and package to as small an area as possible. This way, you reduce the data quantity represented by the planning package, and increase the execution speed of the planning functions.

8. For every planning level create the planning functions you require.

   Planning functions are created in the context of a planning level, and can access the characteristics and key figures that are contained in the planning level. For all planning functions, you require a parameter group (or several) in addition, which contains the concrete processing rules – for example for a function of the type revaluation, the percentage by which the values should be changed.

   You will normally always choose the function type manual planning [Page 127]. With the planning layouts [Page 128] that belong to it, you cannot only enter data manually, you can also look at existing data. Apart from manual planning, you can also create functions of various types, for example revaluation, copy, repost, and provide them with parameter groups.

9. If you want to offer employees an individual view of the planning object, then you create a planning profile [Page 123].

   You can assign planning profiles your employees. These ensure that, from the entire quantity of all planning objects (planning areas, levels, packages, functions), only those are visible that are relevant to the respective employee. This makes orientation within the planning environment [Page 90] easier, and makes sure that all employees only process the objects that are assigned to them.

This step is optional.

10. Define planning folders or Web interfaces for manual planning. This step is optional.
11. Define planning sessions and subplans in the Status and Tracking System so that you are able to control planning processes. This step is optional.
12. Execute planning.

    When executing planning in the planning workbench always select a planning package first (for example by double-clicking) to determine the dataspace in which the planning function should operate. Subsequently open a planning layout for data entry (in the case of manual planning), or execute a planning function by double-clicking on the desired parameter group. You can also execute the planning functions from an open planning layout. After executing the planning function, you then see the changed key figure values directly in the opened layout.

    The connection between planning packages, parameter groups, and layouts in the planning folders or Web interfaces are already fixed. End-users only have to select layouts or function keys to execute their planning tasks.
13. Save your data.

    During a planning session, you can navigate freely within the planning environment. Changes to data and planning objects are automatically stored temporarily, and must be explicitly saved when you end the session.
14. You can protect your data from unwanted changes.

    When you have finished a planning task, you can create data slices [Page 95] in the context of the planning area with which you can lock a greater area of complete data quantity of the planning area against changes.

    This step is optional.

Result

You have plan data at your disposal that you can compare with current actual data at any time, and that you can also use as the basis for decisions.

4.2.1.2.2 Planning Environment

Definition

The planning environment is the central working environment for planning with BW-BPS. In this transaction you will find all the Customizing functions to model the planning architecture. You can also carry out planning from this screen.

The planning services represent one enhancement to the planning environment in BW-BPS. They form the technical infrastructure of the planning applications that are combined as CRM Analytics. CRM Analytics have been developed as application-specific in the interests of sales planning. Therefore, the following restrictions have to be noted:

- You cannot include CRM Analytics in Web interfaces.
- CRM Analytics is not included in the Status and Tracking System.

For more information see the Business Content documentation under *BI Content → Planning Content → CRM Analytics → Planning Services*.

Structure

BW-BPS planning environment objects are discussed in the following topics:

- Planning Area [Page 93]
- Planning Level [Page 116]
- Planning Package [Page 119]

Integration

The central buffer concept in BW-BPS is discussed in the following topic:

- Buffer Concept (BW-BPS) [Page 91]

4.2.1.2.2.1 Buffer Concept (BW-BPS)

Use

Planning methods work with data that is stored in a buffer. This is called the *planning buffer*. Data can be divided into:

- Transaction data (plan data)
- Master data, metadata (Customizing objects)

All changes to the data are written to the planning buffer. Therefore you do not have to save continually. Furthermore, performance is improved.

However, what data is written to the buffer is dependent on the data selections made in the planning package and in the detail applications (for example, the planning functions).

Detail applications are all elements that can be executed and are displayed in the right-hand screen in the planning environment; these include, for example, the planning definition screens or a planning function.

Each user writes data to a personal planning buffer. A user's data in their (personal) planning buffer is locked against changes by other users.

Activities

Every time a detail application is executed, the system first checks whether the requested data already exists in the planning buffer. If it does, the system accesses the version of data available in the planning buffer. If the data is not already in the buffer, the system reads it either from:

- The database table (Customizing objects), or

- The InfoCube (plan data)

and writes it to the planning buffer.

Commit Event

When you exit a detail application, the system writes the corresponding data to the planning buffer.

You get to a manual planning layout and execute a planning function. In the system, the following happens:

- ...

  i. The system writes the modified transaction data to the planning buffer.

  ii. The planning function refers to the planning buffer data; it modifies the data and writes it back to the planning buffer.

  iii. Manual planning reads the modified data from the planning buffer and displays it on the screen.

You can only exit a detail application when all of the data you have entered is consistent; that is, when your entries enable the detail application to be executed correctly. If necessary, you have to terminate the detail application. A termination only ends the current detail application and only causes the data from this detail application to be deleted in the buffer. All data for previous detail applications is still buffered.

Saving

You can save at any point. The data is then written to the InfoCube or the database table and also remains in the buffer. The buffered data is only deleted when you exit the planning application.

As the data is still in the planning buffer, locks that are set automatically remain set.

Therefore, BW-BPS never changes the data in a transactional InfoCube directly, but instead returns a delta. Delta records are transformed when they are transferred to the planning buffer. The planning buffer retains delta records as it does all other data records.

You can save data in two ways:

- *Save All*

  Saves master data, metadata and transaction data. Data (or delta) is written from the planning buffer back to the database tables and the InfoCube.

- *Save Model*

  Only saves master data and metadata. Data is written from the planning buffer back to the database tables.

4.2.1.2.2.2 Planning Area

Definition

A planning area contains two components: a data basis in which the plan data, including the master data, is provided and the settings in the planning environment that enable plan data to be edited either manually or automatically.

You can form multi-planning areas by combining several planning areas.

Structure

Planning areas are specified by their technical and descriptive names, their BW System, and the InfoCube that is to be used from that system. Once an InfoCube has been assigned to a planning area, it can no longer be assigned to other planning areas.

Integration

When creating a planning area you can branch straight to the Administrator Workbench [External]. Here you can create a new InfoCube or edit an existing InfoCube to assign to the planning area.

The data that you store in the InfoCubes and access using planning areas normally originates from the applications in an operative SAP system. These applications have data extractors that supply the Business Information Warehouse with data.

> If you are running BW-BPS in a distributed system landscape and create planning areas that access InfoCubes in different BW Systems, the settings that apply for all quantities and currencies are those basic settings (for example, units, decimal places) that you made in the system where BW-BPS is running. This applies particularly to exchange rates for currency translation. Recourse to the settings stored centrally in one system is necessary to maintain the consistency of the data.

See also:

Transactional InfoCubes [Page 420]

Editing Planning Areas [Page 93]

Multi-Planning Areas [Page 97]

Planning Environment [Page 90]

Planning Level [Page 116]

Data Slices [Page 95]

4.2.1.2.2.2.1 Editing a Planning Area

Use

When you edit a planning area you make central settings that are valid for all levels, packages and planning functions assigned to this planning area.

In addition, you can branch directly to the SAP BW system that contains the InfoCube to which the planning area is assigned.

Integration

Some of the processing functions enable direct access to the SAP BW InfoCube assigned to the planning area, and to the InfoObjects contained in this.

Features

The processing functions for planning areas are divided over the following tab pages:

Tab page *Attributes*

Here you can check which InfoCube is assigned to the planning area. If you are accessing a separate BW System (*Remote*), you can also check the RFC destination that produces the connection to the BW System here. You can switch straight to InfoCube maintenance using *Display InfoCube*. Here you can view the underlying InfoCube and, where necessary, edit it.

You normally only make settings for the InfoCube and RFC destination when creating a new planning area. This is because all additional objects that you create within the planning area depend on the key figures and characteristics available in the underlying InfoCube.

Therefore, it is only possible to make changes to the settings for the InfoCube and RFC destination, without necessitating any extra reworking, if the newly assigned InfoCube has the same key figures and characteristics as the previously assigned one. It may make sense to make supplementary changes to these settings in the following cases:

- You are using an existing InfoCube as the reference for a new InfoCube. You will add additional characteristics and key figures to the new InfoCube. You copy the planning area that is connected to the reference InfoCube and assign the new InfoCube to the new planning area. By doing this you can use all the objects that were available in the original planning area in the new planning area, without more Customizing work. You are also able to use the additional characteristics and key figures.

- You already have a BW system that is used company-wide and you want to retain the InfoCubes that you access with BW-BPS in that BW system too. Therefore, you transfer the InfoCubes that already exist in the planning system into your central BW system and add a corresponding RFC destination to the settings of the planning areas that previously accessed these InfoCubes within the planning system.

- You are operating several BW systems and have transferred an InfoCube to a system that is based on high-performance hardware. In this case, change the setting of the RFC destination in the planning area and you can profit from improved performance, without making further changes.

You can only change the settings for the InfoCube and RFC destination if you have not yet made changes to the planning area or its subordinate objects in the current planning session. If required, terminate the planning session and start again in order to execute the desired changes.

Tab page *Data slices*

Here you define value ranges for the characteristics in the planning area that you want to lock against further postings. For more information, see Data Slices [Page 95].

Tab page *Variables*

- Here you define variables for characteristic values that can be used in all planning levels, packages, and functions of the planning area, instead of fixed values. In this way you have the option to maintain central specifications, for example fiscal year, in one place, but to make them automatically effective in all the areas of the planning area that are affected. For more information, see Variables [Page 102].

Tab page *Master data*

Here you receive an overview of all the characteristics contained in the planning area and, where applicable, all the BW hierarchies belonging to them. Using the context menu of a characteristic, you can switch to the master data maintenance of the characteristic and enter new entries or change existing ones.

Note that master data for planning is read in the M version and not the A version. Time-dependent texts and attributes for master data can be created in BW. You can specify a key date for time-dependent master data. The master data is accessed with this key date. The system delivers those attribute values and texts that are valid on the specified key date. If you do not specify a key date, the master data is read with the current date. In multi-planning areas, the master data is accessed with the key date from the basic planning areas.

The system accesses master data with regard to time in the following cases:

- when reading regular-language texts for characteristic values
- when reading attributes for display in the planning layout or for characteristic relationships
- with the ATRV function in planning functions of type “formula”.

Tab page *Characteristic relationships*

You correlate formally independent characteristics in a planning area using characteristic relationships. You are able to define which combinations of values are valid, which values are derived from other values, and which values are suggested in manual planning. For more information, see Characteristic Relationships [Page 110].

4.2.1.2.2.2.2 Data Slices

Use

You use data slices when you want to explicitly lock certain subsets of a planning area dataset so that the data cannot be updated. Data slices serve a similar purpose as authorizations. Instead of working with data slices you could theoretically also retract the authorization for planners to change plan data during the course of a planning session. However this would prove costly in terms of administration.

When you create a planning area, there are no default settings that restrict the data you can update. In practice, however, it may be appropriate to only permit data updates for specific characteristic value combinations:

The InfoCube, on which your planning area is based, contains data from different companies in your enterprise. Normally, the companies themselves are responsible for importing the necessary data into the InfoCube in the SAP BW. However, you now want to enter further plan data for one of these companies. At the same time, you want to ensure that the data from the other companies is not changed in any way. To achieve this, proceed as follows:

Create a data slice for the planning area. Restrict the characteristic *company* to those companies for which no more data is to be entered. After you have saved the data slice, data can only be updated for that company (or companies) that is **not** included in the selection. You can continue to read data from the other companies as before.

Features

When you use data slices the following rules apply:

- If no data slice is defined for a planning area, you can update data for any characteristic value combination.
- A characteristic value restriction in a data slice acts as a lock that prevents changes. This means that you cannot update data for those characteristic value combinations that are included in the restrictions. You can change all the other data.
- You can create multiple data slices for a planning area. The change locks, set up using characteristic value restrictions, have a cumulative effect. This means that, as you add more data slices, you further restrict the data area where changes can be made.
- Whenever more than one data slice exists for a planning area these act as one. It is not possible with multiple data slices to have only some of them effective.
- If a planning area contains a data slice that includes no characteristic value restrictions at all then the data slice acts as a lock for postings of all types in the entire planning area.
- A data slice acts on all planning levels and packages that belong to the same planning area as the data slice.

Status of a data slice

After you have created a data slice it is activated automatically. The settings made in the definition of the data slice have an immediate effect on the ability to update data.

You can also deactivate data slices. This can, for example, be useful in the following situations:

- You set up the Customizing for planning in your company. You already know in advance that update locks will be necessary at a later point in time. You want to have these locks ready without them being effective immediately as you first want to fully complete Customizing. Activating the locks is to take place later.
- You have used a data slice to lock updates of new data. You later realize that subsequent updates are required because of late deliveries or payments.

To switch the status of a data slice, choose *Activate* or *Deactivate* from the context menu of the data slice. In the list of data slices, the system displays the present status of the data slice with a traffic light symbol (🟢 = active, 🟥 = inactive).

See also:

Planning Area [Page 93]

Planning Environment [Page 90]

4.2.1.2.2.2.3 Multi-Planning Area

Definition

A planning area that combines the characteristics and keys figures of several basic planning areas. A multi-planning area also contains the characteristic *planning area*. This is created automatically and can be used to identify and select all the basic planning areas combined in the multi-planning area. The technical names of the basic planning area are valid values for this characteristic. When the system reads the transaction data it fills this field automatically with the technical name of the planning area from which the data record was read. Consequently, you are also able to use non-transactional InfoCubes.

Characteristics that do **not** exist in a basic planning area but exist in the level of a multi-planning area are handled in the following way: When reading, the system checks whether the initial value for this characteristic is permitted in the selection condition of the planning level or planning package.

- If the initial values are contained in the selection condition, the data records are read from the basic planning area and transformed into records for the multi-planning area. The initial value is assigned to characteristics and key figures that are **not** available.
- If the initial values are not contained in the selection condition, no basic planning area data records are read.

If a planning function is to generate values that are not the same as the initial value for characteristics that are **not** available in the basic planning area, an error message is produced.

A multi-planning area does not contain any data of its own.

Use

Accessing Data in More Than One InfoCube

You use multi-planning areas in order to access data from two or more InfoCubes simultaneously. It is not a problem if the InfoCubes have different data models (also see Data Modeling: Key Figure Model and Account Model [Page 346]). The multi-planning areas are especially useful if you want to read data from one planning area and write it to another.

You want to plan resources for the sales department in your enterprise. The BW system that you are using as a data basis for BW-BPS contains one InfoCube with personnel data and one with sales data. To use the personnel data and the sales data at the same time, even though it is split into separate InfoCubes, create a multi-planning area. You then assign two basic planning areas to the multi-planning area. The two InfoCubes are already assigned to these basic planning areas.

Accessing Data in One InfoCube from More Than One Multi-Planning Area

You can also use multi-planning areas to get around a restriction to the basic planning areas: In the BW-BPS planning environment you are not permitted to assign the same InfoCube to more than one planning area. However, if you require a multiple assignment of this type for a particular reason, first create a basic planning area and assign the relevant InfoCube to it. To access the InfoCube, you can create as many multi-planning areas as required and assign the basic planning area you have created to them.

Reporting from Transactional InfoCubes

You have various options for displaying the data that you have planned in transactional InfoCubes.

- You can read the data straight from the transactional InfoCube.
- You can transfer the data from the transactional InfoCube into a basis InfoCube and display it there.

As the part of the data in the transactional InfoCube is located in an open request (which is therefore “yellow”), the query does not read the most up-to-date data by default. So that all the data is read you have to set a variable for the *Request ID* characteristic to the value *Most Current Data* in the query.

The following table offers an overview of the advantages and disadvantages of this type of reporting:

Reporting from Transactional InfoCubes

PRO	CONTRA
• Data is always up to date. • Data is always held in one place so there is no redundant data.	• Aggregates are not always up to date as data is often changed during planning. • If you have an *Oracle* database, you are not able to use the usual BITMAP indexes for the F table. Concurrent locks may be set if several users update data at the same time. Therefore the system uses the BTREE indexes which are somewhat slower.

Reporting from Basis InfoCubes

PRO	CONTRA
• You are always able to access up to date aggregates. • If you have an *Oracle* database, you are able to use the usual, quicker BITMAP indexes for the F table.	• Data is not always up to date. • Data is always held in two places so there may be redundant data.

Maintaining Multi-Planning Areas

You edit multi-planning areas just like normal (basic) planning areas: You create planning levels, define planning packages and planning functions, and execute these functions. Changes that you make to any data for a basic planning area, which is referenced by a multi-planning area, are automatically visible in the multi-planning area too, and vice versa. You can work with a multi-planning area and its underlying basic planning area simultaneously.

As a multi-planning area contains all the characteristics and key figures of the basic planning area, as well as the characteristic *planning area*, more main memory space and more CPU time is used than when working with basic planning areas. You can improve performance by reducing the size of the planning buffer. In this regard:

- only use as many basic planning areas in a multi-planning area as are strictly necessary
- execute planning functions that only affect one basic planning area directly in the basic planning area

Structure

A multi-planning area contains all the characteristics and key figures of those planning areas that are combined in the multi-planning area, as well as the *planning area* characteristic, which is generated by the system automatically. This characteristic enables you to assign key figures to a particular basic planning area. BW-BPS ensures that this characteristic is always selected. You can also use the *planning area* characteristic to read data from one planning area and write it to another. To do this, you create a planning function (for example, of type *copy*) and include *planning area* in the list of fields to be changed. Then, in a parameter group for the function, you determine which planning area the data is to be written to.

The procedure described above for cross-planning-area data processing is the only way to get around the fact that planning functions are restricted to the data in their own planning areas.

As well as using the copy function in a multi-planning area, you can transfer data from one InfoCube to another using a data mart or data staging. The following table offers a concise overview of the advantages and disadvantages of the two techniques:

Data Staging

PRO	CONTRA
• The server requires less memory capacity as there is no buffer. • Instead of having to recreate the database indexes of the database table after each data record transfer, you can save time by: ■ ... ■ ... b. deleting the database indexes c. transferring data d. recreating the database indexes • The data staging process is quicker than executing a planning function.	• As you are using a transactional InfoCube as the target, you have to switch this from *Planning enabled* to *Load enabled* before you start the data staging process and then reverse it again after.

Copying to the Multi-Planning Area

PRO	CONTRA

• The copy function is very easy to set up. If data has to be changed during the transfer process you can use the FOX formulas. • If data cannot be saved in the target InfoCube, you receive a detailed error message. • You do not have to switch between *Planning enabled* and *Load enabled* in the transactional InfoCube and can therefore continue planning while copying takes place. • You can use the additional features in the planning environment, for example, the flexible selection of data to be transferred using variables or derivations.	• The server requires a lot of memory capacity as all of the data to be copied is located in the planning buffer. When saving, you have to refresh the indexes of the database table after each data record.

Integration

In order to work with multi-planning areas you must first have created basic planning areas. You cannot delete a planning area that is referenced by a multi-planning area.

💡

> You can only change the composition of the basic planning areas that make up a multi-planning area until you save the multi-planning area. This is similar to the restriction with basic planning areas. You cannot subsequently change their assignment to an InfoCube.

Make sure that the technical settings for the characteristics and key figures in the various InfoCubes are the same, if:

1. you are using the same characteristics or key figures in different planning areas and
2. these planning areas are in different systems.

See also:

Multi-Planning Area: Application Examples [Page 100]

Planning Area [Page 93]

Planning Sequences [Page 267]

4.2.1.2.2.2.3.1 Multi-Planning Area: Application Examples

Here you will find some typical application examples for using multi-planning areas. The steps and objects that are relevant for the respective task are named for each object.

Copy

You want to copy the transaction data from planning area G1 into area G2. Both areas are structured the same, therefore they have the same characteristics and key figures.

Function type: Copy
Fields to be changed: Planning area
Parameter group: from G1 to G2

Copying with Automatic Summarization

You want to copy the transaction data from planning area G1 into area G2. Both areas contain the same key figures. Whereas the independent characteristics *article* and *article group* are in G1, G2 only contains the characteristic *article group*. For this reason, a breakdown according to individual articles will be lost for the data to be copied to G2: The data is automatically aggregated. To do this, create the copy function in the context of a planning level that only contains the characteristic *article group*. You require the following objects:

Planning level: Contains all desired key figures, but only the characteristic *article group* from G2
Function type: Copy
Fields to be changed: Planning area
Parameter group: from G1 to G2

Grouping Together Key Figures

The key figures *sales* and *discount* are managed in planning area G1, the key figure *net revenue* in planning area G2. You want to determine the net revenue using both key figures from the other planning area.

Function type: Formula
Fields to be changed: Planning area, key figure
Parameter group: **`{G2, net revenue} = {G1, sales} – {G2, discount}`**

Assignment of Key Figures to Accounts

Planning area G1 contains various financial key figures, but no account characteristic. However, there is an account characteristic in area G2, as well as a key figure value. You want to assign the amounts, which were posted in G1 for the key figures, to the appropriate accounts.

Function type: Formula
Fields to be changed: Planning area, account, key figure
Parameter group: **`{G2, 800000, value} = {G1, #, sales}`**

For syntactic reasons, the transfer of a value for the account characteristic is demanded in formula operands on the right of the assignment operator. As no such characteristic exists in planning area G1, the initial value # is transferred instead of an account.

Cost Center Assessment

Planning area G1 has the characteristics *cost center* and *account*, as well as the key figure *value*, which contains the amount per account and cost center. The characteristic *article group* and the key figure *administration costs* are managed in planning area G2. You want to apportion the amount posted to account 400000 of cost center 5000 to a quarter as administration costs to the article group AG1.

Function type: Formula
Fields to be changed: Planning area, article group, cost center, account, key figure
Parameter group: **`{G2, AG1, #, #, administration} = {G1, #, 5000, 400000, value}`**

4.2.1.2.2.2.4 Variables

Use

You can use variables to reduce the amount of recurrent Customizing work in the planning environment. You achieve this by restricting characteristics to one desired value or value range using variables in a central place within a planning area. You can also determine these values by selecting hierarchy nodes or attributes. The setting of the characteristic value which you make in this way then becomes automatically effective in all places within the planning area where the selection for the corresponding characteristic is referred to the variable you have defined. You can also create variables for numeric values which have no reference to a characteristic.

The use of variables offers the following advantages compared to a direct entry of selection conditions for a characteristic:

- Reduced Customizing effort: You only have to make settings that should be valid in the entire planning area as standard (the same as variable definition) in one place, and transfer these settings per variable reference to all planning levels and functions of the area.
- Flexibility: You can set the valid selection for a characteristic individually for every user.
- Dynamic, automated adjustment of Customizing: You can have the selection determined by a function module, where you have complete freedom in the definition of selection values. In this way it would be conceivable that certain characteristic value combinations can only be planned within a certain time period, and the assigned function module automatically sets another combination at the end of this period.
- Simplified structure of planning layouts: You can mark up characteristic values that are transferred per variable with an offset [External]. This leads, for example, to a drastic simplification of the maintenance of planning layouts for rolling planning over several periods, because with a period change you only ever have to update the value of the outgoing period and adjust all other periods automatically using graded offsets.

Integration

Variables that you define in the context of a planning area can be referenced anywhere within the planning area where a selection condition can be entered for this characteristic:

- In the planning level [Page 116]
- In the parameter groups of the user-defined planning functions [Page 196] for which the characteristics can be included in the list of fields to be changed (for example copying, reposting)
- In the formula editor of a formula function [Page 251] using the ABAP keyword `VARV` (variable value) and `VARI`

- In manual planning [Page 127] with column definition

  The option of restricting characteristic values through variable replacement is not offered in planning packages.

Features

Variable types

You can create the following types of variables:

- **Characteristic** You select one or several characteristics from the characteristic pool of the planning area and enter with which values of these characteristics the variable should be replaced at the time of execution.
- **Attribute** First you select a basic characteristic from the characteristic pool of the planning area and then enter one or several attributes of this characteristic. You define the desired attribute values for each of these attributes. The variable then makes those values available for the basic characteristic that corresponds to the selected attribute values (for example all articles that belong to a selected article group as an attribute value).
- **Hierarchy** First you select a characteristic from the characteristic pool of the planning area and then one of the BW hierarchies defined for this characteristic. Finally, you put together a selection of those values of the hierarchy-bearing characteristic that are contained in the hierarchy.
- **Number** You determine that the variable should be replaced at the time of execution by the number you defined. You use this type of variable where nondimensional sizes are used in BW-BPS (for example as a revaluation factor in functions of type revaluation [Page 215] or in formulas [Page 251] with the formula function `VARV` and `VARI`).

You can define any number of variables for every characteristic of the planning area. In addition, the possibility exists to assign several characteristics to one individual variable at the same time. If you assign a user to a variable defined in this way this results in the worklist of this user from a business view.

> You want to define a variable that describes the worklist of an employee who is responsible for the sales planning of antidiabetics in two sales regions. To do this, assign the characteristics product group and region to a variable `VPGRG` with the replacement type “User-specific values”, and define the *product group* to “Antidiabetics” as well as *region* to the values “North” and “South”. Specify the names of the variable (`VPGRG`) as selection for both characteristics respectively in a planning level. When the employee executes planning functions within this planning level, these automatically work on the values of the *product group* and *region* characteristics defined by the variable.

> The following special features apply for variables where several characteristics are assigned:
>
> - You can only assign a quantity of single values to the variable for every characteristic, but no value ranges.

  - You can only use the variable in planning levels, which contain all the characteristics which are also contained in the variable definition.
  - The indicator *Restriction of selection required by user* is automatically set by the system and cannot be reset by the user.

In addition to the reference characteristic, determine the replacement type and the required type of text replacement for every variable (except numeric variables).

Replacement type

The replacement type allows you to control the way in which the variable is replaced by a value or value range at runtime:

- **Fixed value**
  With this replacement type you enter a selection of values or value ranges for the variable. These are used standardized in the entire planning area for the replacement of variables.

- **User-specific values**
  With this replacement type you proceed as with replacement by a fixed value. You also enter the user for which the variable values should be valid. In this way you can make sure that the person responsible for sales planning sees the plan figures for all regions when he/she navigates within the planning area, while the processors responsible for individual regions are only offered the plan figures that fall within their area of responsibility. Users are distinguished by the user names with which they log onto the system.

  If a characteristic is to be replaced with the user-specific values of a variable but no values are defined for the current values, the planning functions affected by this cannot be executed.

  You have defined a variable **`VCOMPANY`** for the characteristic *company*, with replacement type "user-specific values", and entered different selections for the users `SMITH` and `JONES.` You have set a replacement with the variable `VCOMPANY` for this characteristic in the planning level. User `BROWN` tries to execute a planning function in which *company* is contained in the list of characteristics to be changed. The system issues an error message because it cannot determine the companies for which functions are to be executed with user `BROWN`.

- **User exit**
  With this replacement type you enter one of the function modules you created to determine the characteristic value to be used dynamically when the characteristic is accessed while executing a planning function. On this basis of this the system can compile the content of the planning package and transfer it to the planning function for processing.

- **Authorization**
  With this replacement type the system automatically determines those characteristic values for which the user logged onto the system has an authorization, and allows the user to make a selection from those values. The authorizations used here are the reporting authorizations from the BW system. This has the advantage that authorizations in reporting are automatically consistent with authorizations in planning, without any extra Customizing effort.

You can find authorization management in the SAP menu at *Business Information Warehouse → Business Explorer → Authorizations → Reporting – Authorization Objects*.
For more information, see Maintaining Authorizations [External].

Text replacement

Here you enter which of the various text types (short, medium, long) should be used to display the characteristic values in the column headers of the planning layouts of Manual Planning [Page 127]. If no texts are maintained in the logon language, the characteristic value itself is displayed instead. Here you can make the display of characteristic values mandatory.

See also:

Defining Variables [Page 105]

Using Variables [Page 108]

4.2.1.2.2.2.4.1 Defining Variables

Use

You define variables for the restriction of characteristic values in order to maintain selections that should be valid for the entire planning area in only one place and thereby reduce the amount of Customizing. In addition, you can define variables without characteristic references to be replaced by numeric values. You can use numeric variables of this type in planning functions of the type formula and revaluation.

Procedure

1. In the context menu of the desired planning area, choose *Change area*.
2. Switch to the tab page *Variables*.
3. Decide which action you want to execute:
    - To create a new variable, choose *Create*. Enter the technical name, a description and the type of variable.
    - To change the description of a variable, choose *Change description*.
    - To edit the properties of an existing variable, double-click on the variable name.

    The system displays the properties of the variable.
4. Make the desired settings for the variable. The possible settings comply with the type of variables:
    a. **Characteristic variables**

    Characteristics: Choose *Add Characteristic* to assign a characteristic to the variable for whose values you want to define a selection using variables.

The system displays a dialog box with the list of all characteristics of the planning area. Choose the characteristic that you want to assign to the variable from this list.

You can repeat this activity to assign several characteristics to the variable. If several characteristics are assigned to one variable, the selection can only contain single values for every characteristic, but no value ranges.

Choose *Delete characteristic* to remove the assignment of a characteristic to a variable. Note that you can only save a variable when at least one characteristic is assigned.

b. **Hierarchy variables**

Characteristic: Choose *characteristic*, to assign a characteristic to the variable.

Hierarchy: Choose one of the available hierarchies for the characteristic.

c. **Attribute variables**

Characteristic: Choose a *basic characteristic*, whose attributes you want to use.

Attributes: Choose one or more of the attributes defined for the characteristic.

d. **Numeric variables**

Numeric value: Enter a numeric value with which the variable should be replaced at the time of execution.

Value range for numeric values: Numbers up to 22 digits long are permitted with up to 9 decimal places.

The following fields are available with various variable types:

Text replacement (characteristic and attribute variables): Decide which of the various text types (short, medium, long) should be used, to portray the characteristic values in the planning layout of Manual Planning [Page 127].

Replacement type (characteristic, attribute and hierarchy variables): Decide in which way the values of variables should be defined. There are the following replacement types:

Fixed value: Enter the desired values in the table *Selection conditions*. Everywhere where you later want to refer to the variable, the variable is replaced by a value specified by you.

User-specific values: You define values for the variables similar to the replacement type *Fixed value*. However, in the list of *users*, you must first enter the user for which the values you subsequently enter are to be valid. When executing planning, the variable is then replaced by those values which were defined for the user who is logged onto the system.

If when executing planning, variables are used with the replacement type *user-specific values*, but no values are defined for the user logged onto the system, then this user cannot execute planning. However, you can avoid this execution lock by setting the indicator *Input allowed by user* for the variable.

User exit: You enter a function module, which dynamically determines the characteristic value to be used at runtime.

If you change the characteristic assignment or the replacement type of a variable, all previously entered characteristic value restrictions are deleted. In the case of a variable with the replacement type *User-specific values*, the list of assigned users is also deleted.

Restriction of values required by user: If this indicator is selected, then the system requests that the user restricts the possible selection values of the variable when executing planning. In this way you can force a two-step restriction of valid characteristic values:

- Global restriction by simulation of selection values in the variable definition
- Additional user-specific restriction of globally predefined values at runtime

For variables where more than one characteristic is assigned the system automatically sets this indicator and removes the ready for input status so that the setting cannot be changed.

Input allowed by user (only for variables with the replacement type *User-specific values*): If this indicator is set, the user has the option of restricting or also extending the globally predefined characteristic values further at the point of execution.

Selection conditions: Here you enter the characteristic values that should be made available by the variable at runtime. Depending on whether one or several characteristics are assigned to a variable, the type of characteristic restriction differs:

- One characteristic assigned: You can define single values and value ranges for the characteristic.
- Several characteristic assigned: You can only define single values for the characteristic.

5. In the *Selection conditions* table, enter the values that should be made available by the variable at the time of execution.

With variables of the type hierarchy, the cells of the *Selection conditions* table are locked against manual entry to avoid values being entered that are defined on the hierarchy-bearing characteristic but do not occur in the hierarchy. In order to define characteristic values for variables of the type hierarchy, use the F4-Help.

Result

After you have defined the variable you can use it in the planning levels and functions of the planning area to restrict characteristic values.

See also:

Using Variables [Page 108]

4.2.1.2.2.2.4.2 Using Variables

Use

You use variables for the characteristic value restriction in order to transfer, per reference to the variables, their centrally defined value for the current planning object.

Procedure

The way you use variables depends on the type of object that you want to use together with the variables.

Variables in the planning level

1. Select the planning level in which you want to use variables for characteristic value restriction and choose *Planning → Planning level → Change.*
2. Switch to the tab page *Selection.*
3. Position the cursor in the *From* column of the characteristic whose characteristic value you want to restrict using a variable.
4. Choose *Switch variable on/off* to activate variable replacement for the characteristic.

   When the variable replacement is activated the system automatically hides the *To* column. If you require a value range as a selection, you can achieve this by defining the variable correspondingly.
5. In the *From* column of the characteristic, enter the technical name of the desired variable. With the input help pushbutton you can also request a list of all variables that are defined for the characteristic and transfer the desired ones.
6. For time characteristics there is a dialog box in which you can also enter an offset. This makes it easy to create a copy function, for example, copy from the previous period.

Variables in Planning Functions (Except Formulas)

1. Select a function that contains a characteristic in at least one of the lists for fields to be changed or for conditions that you want to restrict with a variable.
2. Choose a parameter group for this function.
3. Position the cursor in the *From* column of the characteristic whose characteristic value you want to restrict using a variable.

   This step is valid for both characteristic to be changed and for characteristics from the list for condition fields.
4. Choose *Switch variable on/off*, to activate variable replacement for the characteristic.

   When the variable replacement is activated the system automatically hides the *To* column. If you require a value range as a selection, you can achieve this by defining the variable correspondingly.
5. In the *From* column of the characteristic, enter the technical name of the desired variable. With the input help pushbutton you can also request a list of all variables that are defined for the characteristic and transfer the desired ones.

Variables in Formulas

Proceed as follows when you want to access a variable defined in the planning area from a formula:

1. In the formula editor, create a local variable, whose type refers to the characteristic to which the (global) variable belongs (for example **`DATA VCOMP TYPE 0COMPANY`**).

   With global variables of the type number, you choose the predefined type F in the formula editor for the local variable.

2. Determine the value of the (global) variable using the language element **`VARV`** and allocate it to the local variable (for example **`VCOMP = VARV('VCOMPANY')`**).

Variables in Manual Planning

Whether or not you are able to use variables in manual planning depends on the settings you have made in the planning level.

1. Make sure that the characteristic value restriction by variables in the planning level was made so that it is suited to your planning layout. There are the following dependencies:

   The system evaluates the settings in the planning level and automatically ensures that each characteristic can only be used in suitable areas of the planning layout.

   The following steps only refer to characteristics with time base that are to be used in the data columns.

2. Position the cursor in the *From* column of the characteristic whose characteristic value you want to restrict using a variable.

3. Choose *Switch variable on/off*, to edit the variable replacement for the characteristic.

   The system displays a dialog box. Here, you can check the name of the variable that is used in the planning level for characteristic value restriction, enter an offset for the characteristic, or return to the entry of a fixed value.

4. Enter an offset with which the characteristic value, which results from variable replacement, should be marked up.

   Since in the definition of an individual data column with characteristic reference you can also only assign one individual characteristic value to the column, the following special feature applies in manual planning: With a characteristic, whose value is restricted by variable replacement and should be used in the data column, only the smallest value of the selection is used as the characteristic value; all other values that are possibly contained in the selection are ignored.

   You want to include the characteristic **`0FISCPER`** in a data column with the key figure values being displayed for the prior, current, and subsequent period. Period is restricted in the planning level by a variable **`VPERI`**, which contains the value range **`003...012`** as selections.

For the first data column choose *Switch variable on/off* to enter the offset **-1** for the past period in the dialog box (since as a result of this you fall below the smallest allowed value of the selection, the system automatically converts the data columns into a comparison column [External]). Repeat this step for the next column. Here you leave the field for the offset empty in order to transfer the current period. Finally for the third column you enter the offset **1** for the following period.

At the end of the period you can modify the selection of variable **VPERI** to the value range **004...012**. The data columns of the planning layout then automatically adjust to the changed setting, without you having to edit the layout manually.

You can also use the characteristic value determined by the variable replacement as a pure character string in column headings. To do this simply add the variable name (if required under the entry of the desired offset) in the text of the column heading, for example “sales in period **VPERI+2**”. The system recognizes at runtime that it concerns a variable and replaces the variable names with the current value.

4.2.1.2.2.2.5 Characteristic Relationships

Use

It is often the case that characteristics used within a planning task do not coexist in isolation. A relationship often exists between the different characteristics in terms of content, although this may not be immediately obvious from the technical modeling of the characteristics. **Characteristic compounding** indicates that a particular (higher-level) characteristic always has to be lead to a compound characteristic (see Tab Page: Compounding [External]). However, characteristic compounding does not make clear whether the particular values of one characteristic correspond to the particular values of the other. The content of, for example, the characteristics *Article* and *Article Group* correspond in this way.

Characteristic relationships are used to relate those characteristics that correspond to each other in terms of content. Using characteristic relationships you can set up procedures to check valid combinations of characteristic values and in addition, check that the system enters valid combinations of this sort within manual planning in the lead column or in the header area of a planning layout (default values). Furthermore, you can define rules according to which the system can derive values for further characteristics from the characteristics that you use for a planning task. These values are not contained in the planning level but should nevertheless be updated.

Data entry in manual planning has the following advantages:

- In a planning layout for which the “all possible characteristic combinations” option has been selected, those combinations are generated that have been defined as useful in the characteristic relationship. For example, in a lead column that contains the characteristics *cost centre* and *currency*, only the currency valid for this purpose is provided for data entry.
- By deriving characteristic values, the system is able to determine values of different characteristics from the entry of a single characteristic value. For example, the system

can determine the corresponding value of the characteristic *article group* from a value of the characteristic *article*.

- With characteristics between which there is a dependency, the system can check whether a combination of characteristic values that has been entered is valid. This prevents transaction data from being entered for invalid characteristic value combinations.

Integration

Characteristic relationships can be defined with reference to attributes and hierarchies on characteristics, among other things. Here, those attributes and hierarchies are used that were created in the BW system for a characteristic (see Tab Page: Attributes [External], Tab Page: Hierarchy [External] and Hierarchy [External]).

Characteristic relationships affect all function types that change data. Therefore, planning functions cannot update transaction data for value combinations that are not the result of a defined derivation or that are recognized as invalid during a check for valid combinations. Forming combination proposals only affects manual planning layouts for which the “All Possible Characteristic Combinations” option has been selected.

Prerequisites

The following prerequisites must be fulfilled in order to define characteristic relationships:

- The planning area must be a basic planning area

  The characteristic relationships defined in a basic planning area are also effective in multi-planning areas in which a basic planning area of this type is contained

- In characteristic relationships of the type attribute, the target characteristic must be defined as an attribute of the basic characteristic and must itself be contained in the planning area

- In characteristic relationships of the type hierarchy, the target characteristic must be contained in a hierarchy on the basic characteristic.

Features

For every characteristic relationship that you define it is possible to perform a combination check and to generate combination proposals in relation to the values of the characteristics involved. In addition, you can specify that the system should also derive the values of the target characteristic from the values of the basic characteristic.

> A characteristic relationship that supports the derivation of values of the target characteristic can only achieve the defined derivation when the basis characteristic is, but the target characteristic is not, contained in the planning level. Otherwise the effect the characteristic relationship has on the combination check and proposal is limited.

You can create up to 99 characteristic relationships in a planning area. It is possible to establish implicit “relationships between relationships” by using the target characteristic of a relationship A as the source characteristic of another relationship B. In this case, the only characteristic values from the source characteristic that enter relationship B are those that result from relationship A.

The following types of characteristic relationships exist:

- **Attribute**: The target characteristic is defined as the attribute of the basic characteristic (for example, the characteristic *currency* is an attribute of the characteristic *controlling area*).
- **Exit**: The values of the characteristics involved result from the implementation of the specified user exit.
- **Reference data**: Valid combinations are transferred from reference data that you determine using a suitable selection.
- **Hierarchy**: The characteristics involved are part of a BW characteristic hierarchy on the basic characteristic.

For each of these relationship types you decide, when defining a relationship, whether only combination checks and proposals should be generated or whether a derivation of the target characteristic from the basic characteristic should also be performed. In the case of derivation, the basic characteristic takes on the role of a source characteristic.

All relationship types offer the possibility to define relationships between a basic characteristic and several target characteristics. With the relationship types exit and reference data you can specify several basic characteristics.

Since characteristic relationships are defined in the context of a planning area they affect how all transaction data within the planning area is processed.

Adjusting Transaction Data after Changes to Characteristic Relationships

Transaction data stored for characteristics that are linked to another characteristic by a derivation relationship can also contain values for the derived characteristic. For example, a transaction data record for the characteristic *material* normally also contains the corresponding value of the attribute characteristic *material group*. If master data is revised and former assignments between characteristic and attribute are changed (for example, by arranging a material in a different material group), then inconsistent transaction data records occur.

You can also use characteristic relationships to consistently match the key values of existing transaction data records to the current valid attribute assignments defined in the master data. This allows you to transfer from the old to the new structure. To do this, create a characteristic relationship as a derivation and assign the characteristics affected. Create a planning function of the type repost (characteristic relationships) and assign the derived characteristic to it as the characteristic to be changed. You also require a parameter group for the function, in which no further settings are necessary. When you execute the function with the parameter group the transaction data is reassigned according to the current assignments between the values of the source characteristic and the derived characteristic.

The *Delete* function (*invalid combination*) deletes all plan data in which errors were found during the combination check.

⚠

The *Repost (characteristic relationships)* and *Delete (invalid combination)* functions can only be used in planning levels that contain all characteristics. This is the only way in which incorrect records can be cleaned up.

See also:

Reposting (Characteristic Relationships) [Page 212]

Deleting (Invalid Combination) [Page 214]

4.2.1.2.2.2.5.1 Defining Characteristic Relationships

Use

You define characteristic relationships in order to be able to use validations, default values, and derivations when generating transaction data for characteristic value combinations. For more information, see Characteristic Relationships [Page 110].

Procedure

1. In the planning environment, select a planning area by double-clicking on it and choose *Planning → Planning Area → Change*. (Alternatively, choose *Change Area* from the context menu for the desired planning area.)
2. Choose the tab page *Characteristic Relationships.*

   Each numbered row of the table corresponds to a characteristic relationship. You can create up to 99 relationships in a planning area and a characteristic can be used in several relationships.
3. To create a new relationship, select the input help in the *Type* column in a free row in the table and determine the type of relationship. Available options are *Attribute*, *Exit*, *Reference Data*, or *Hierarchy*.
4. In the *Detail* column, choose *Detail View*.

   The system shows the detailed settings for the relationship. The settings that are possible here comply with the type of relationship you selected.
5. Decide whether the system should just perform a *Combination Check and Proposal* on the relationship or whether a *Derivation* of characteristic values should also take place.
6. Specify the first characteristic of the relationship (with attributes and hierarchy relationships this is the basic characteristic).
7. If you are defining a relationship of the type hierarchy, specify a *hierarchy* which is defined on the basic characteristic.
8. Specify a *target characteristic* (or several target characteristics) for the relationship.

   Depending on the type of relationship, you can choose the following objects as target characteristics:

   - **Attribute**: One of the characteristics that are assigned to the basic characteristic as an attribute. This attribute characteristic must be contained in the planning area.
   - **Exit**: Any characteristic in the planning area.
   - **Hierarchy**: One of the characteristics that are contained in the hierarchy you selected on the basic characteristic.
   - **Reference data**: Using a selection, choose data from the planning area that you want to use as a reference (for example, you could specify a version with actual data)

     Only those characteristics that are not involved in the characteristic relationship are available for the selection of reference data. For technical reasons you must also specify a key figure for the selection of reference data. Transaction data must be contained in the planning area for this key figure.

9. Choose *Char. Relationships* to exit the detail view of relationships.

Result

The characteristic relationship is effective in the entire basic planning area and in the multi-planning areas that contain this basic planning area. When you write newly entered or changed data records in which the basic characteristic of the relationship occurs the system performs combination checks and derivations corresponding to the relationship you defined.

If you create a planning layout of manual planning in which basic and target characteristics are contained in the lead column or in the header area and select the option “All Possible Characteristic Combinations”, the layout is opened with those characteristic value combinations that result from the characteristic relationship.

Additional functions on the characteristic relationships tab page

To delete the characteristic relationship, select the corresponding row and choose *Delete Relationship*.

You can display the where-used list for the characteristic relationship using *Step* (*Using a Step*) and *All Steps* (*Using All Steps*). For more information, see Where-Used List for Characteristic Relationships [Page 114].

4.2.1.2.2.2.5.2 Where-Used List for Characteristic Relationships

Definition

The list of planning levels in which a characteristic relationship is being used is available in both the definition of the characteristic relationship for a planning area and the definition of the planning level.

Use

In the definition of characteristic relationships for a planning area

The following types of where-used list are available on the *Characteristic Relationships* tab page:

Where-used list in the definition of a planning area

Function key	Meaning
Step (Using a Step)	Provides an overview of the planning levels in which the selected characteristic relationship is used.
All Steps (Using All Steps)	Provides the corresponding overview for all characteristic relationships.

Click on *Close* to return to the previous screen.

You will find more information on the *Note* column in the following section:

In the definition of the planning level

On tab page *Characteristic Relationships*, in the *Note* column of table *Charcteristic Relationships Used*, those planning levels are displayed in which the respective step can be used.   These specifications form the basis of the selections in the planning levels. However, they only really represent one opportunity. The application of a step is dependent on additional factors:

- the actual data in the InfoCube
- the variables and selections in the packages

If, for example, *Package* is specified in the *Note* column for a particular planning level but the selection of a character in a lower-level package is restricted to '#' in the level, the step is **not** used in this planning package.

If it is possible on the basis of the variables and selections in the packages to further restrict the planning level selection so that this affects the possibility of executing the step, appropriate notes are visible in these columns:

Notes for using a step in the definition of a planning level

Note for using a step	**Meaning**
Empty	Step can be applied because of selection in planning level. This is not dependent on selections or variables in a planning package.
Variable	Step can be applied if variables have '*'. This is dependent on the characteristic values of the variables. There is one exception where this note is **not** displayed: The characteristic for the planning area is selected in a multi-planning area with a variable.
Var. PG	Step is only applied if a multi-planning area is affected. A variable is used in the generated *planning area* characteristic for the multi-planning area. This overwrites all other notes. If, however, the other notes are of interest you can display these by replacing the variable with an appropriate normal selection.
Package	The selection is currently **not** determined in the planning level. A selection is possible that causes the step to be applied.
Package PG	Step is only applied if a multi-planning area is affected. The selection of the generated *planning area* characteristic is set to "package selection". This overwrites all other notes.
Var./Package	Whether or not you can apply the step is dependent on both the selections and the characteristic values of variables in a planning package.
Multiple	The source characteristic of the step can be derived in more than one way. The system does not specify in any more detail whether dependencies on variables or selections in a planning package exist.

4.2.1.2.2.3 Planning Level

Definition

A planning level contains a quantity of characteristics and key figures from a planning area. It determines the level of detail of the planning. Planning area characteristics that do **not** exist in the planning level are used for aggregation. Selections can be specified for the characteristics in the planning level.

Use

Multi-dimensional integrated planning mostly takes place on different planning levels. The content from these planning levels is connected using bottom-up aggregation and top-down distribution. You establish this connection by creating planning levels whose characteristics refer to each other as regards content (for example, *region – country*, *product group – product*), and technically too, with compound characteristics. You create several planning levels for the same general planning task. The planning levels differ in the level of detail of the planned objects (for example *product line*, *product group*, *product*). Key figures, which you plan on the detail level (for example products), are automatically aggregated to higher levels (for example *product group*).

When addressing a specific business issue you will not need to access the entire amount of characteristics and key figures available in the InfoCube specified in the planning area. You create planning levels in order to limit the characteristics and key figures to those that are relevant to the business issue. You can also make characteristic value restrictions in the planning level in order to further restrict the area that the planning functions refer to.

> You can create planning levels for different planning tasks by combining appropriate characteristics and key figures for a specific purpose in one planning level. You could, for example, have a planning level for production planning where you plan the material requirements based on different plants. You could also have a planning level for sales planning where you plan the material requirements based on your customers or on specific sales channels. You can make appropriate selections to split the characteristic *customer* over several planning packages. Planning can then be carried out by those employees who are responsible for the customer groups combined in the respective packages.

> Note the following features:
>
> - Key figures of type date or time can only be used when the numeric type “DEC – counter or amount field with comma and sign” is set as the data type.
>
>   Key figures of the type date or time with the data type “DATS” or “TIMS” are **not** supported as these data types are saved internally as character strings and therefore cannot be used for calculations.
>
> - In planning, all key figures are handled as if they have the standard and exception aggregation behavior *Summation (SUM)*. This applies even when other settings have been made in key figure maintenance. For this reason, you should only use key figures that have the standard aggregation behavior *Summation*. As long as the exception aggregation characteristic is not used in any planning level, it does not matter which aggregation behavior is set for exception aggregation.

In some cases data inconsistencies can occur when aggregating or compressing data in the InfoProvider. This also applies to plan key figures that are seen as non-summative from a business perspective.

Interaction of Different Planning Levels

A planning area contains the characteristics *product* and *product group* (and possibly other characteristics too), with the key figure *sales quantity*.

Planning level 1 contains the characteristics *product* and *product group* and the key figure *sales quantity*. Planning level 2 contains the characteristic *product group* and the key figure *sales quantity*.

The following records are available to planning level 1:

Planning Level 1

Product	Product Group	Sales Quantity
P1	PG1	10
P2	PG1	20
P3	PG2	30
P4	PG2	40

In planning level 2 you see the following data. This is aggregated using the characteristic *product*:

Planning Level 2

	Product Group	Sales Quantity
	PG1	30
	PG2	70

The interaction between the planning levels becomes evident when changes are made: The sales quantity value for product group PG1 is increased from 30 to 40 in the first record and saved. The BW system saves the changed record. If you only select data in planning level 1, that is on the level *product – product group*, the results will be as follows:

Planning Level 1

Product	Product Group	Sales Quantity
	PG1	10
P1	PG1	10
P2	PG1	20
P3	PG2	30
P4	PG2	40

As planning is performed at product group level, the change to the sales quantity value is written to a “non-assigned product”.

Generally an initial characteristic value can be viewed as a “non-assigned object”. An initial characteristic value is marked with a # sign so that it can be distinguished in the settings from an entry where the user has not entered a value. The internal format is always the type-related initial value. For a date field this is, for example, eight zeros. Take this into account when you are programming exits.

You can change system behavior by creating characteristic relationships in which you activate derivation of characteristics. During derivation, the initial values are replaced with valid characteristic values. It is also possible to use a third planning level that only contains products. The product groups are then filled correctly during derivation.

Planning Level 3

Product		Sales Quantity
		10
P1		10
P2		20
P3		30
P4		40

If the value from P4 is increased with the value from 10 on this level, you will see the following records on planning level 2 (after both changes have been made):

Planning Level 2

	Product Group	Sales Quantity
	PG1	40
	PG2	80

Restricting Characteristics in the Planning Level

The values of a particular characteristic can either be restricted in the planning level or in the planning packages that are assigned to the level. A selection of values for the same characteristic in the planning level and also in the planning package is **not** possible.

If you restrict characteristics in the planning level to certain values, this restriction applies for all planning packages that are assigned to this planning level.

When you use some planning functions, the system will determine that characteristic value restrictions must be made at specific levels of the planning architecture.

When deciding which characteristics to restrict in the planning level and which to restrict in the planning package, SAP recommends the following procedure:

- Characteristics whose values should be kept constant for a particular planning task should be restricted in the planning level. This has the advantage that you make the setting centrally in one place and can change it with little effort if required. Typical candidates for a selection in the planning level are, for example, fiscal year or currency.
- Characteristics whose values are appropriate for delimiting planning subtasks should be restricted in the planning package. In this way you can split the whole task into subtasks that do not overlap. You can then assign them to a particular person responsible. Typical candidates to be selected in the planning package are, for example, article or customer.

Creating Planning Functions for a Planning Level

You can create planning functions for a planning level. In order to carry out these planning functions, however, you have to specify a planning package in the planning level that you can use to delimit the data area in which the planning functions will operate. For example, you can create a planning package for each cost center so that the planning functions that you execute reference just that one cost center (and not all cost centers simultaneously).

Copying Planning Levels

The system supports the creation of several planning levels with the same or similar structure through the *Copy Planning Level* function. This function enables you to copy a planning level (under a new name) into the same planning area as the original planning level.

Structure

You define a planning level by selecting the characteristics and key figures from the InfoCube that is assigned to the higher-level planning area. In the planning level you determine, for each characteristic, whether the characteristic is to be restricted to certain characteristic values in the planning level or in the lower-level planning packages. For documentation purposes, you can store a textual description for the planning level.

See also:

Planning Environment [Page 90]

Planning Package [Page 119]

Where-Used List for Characteristic Relationships [Page 114]

4.2.1.2.2.4 Planning Package

Definition

The planning package takes the characteristics and key figures selected in the planning level [Page 116] and serves to further restrict characteristics whose values have not already been restricted in the planning level. At the same time the planning package describes the data area on which the planning functions are to operate.

Use

You create planning packages to define subtasks within the overall planning process and to keep these subtasks separate. As you define the content of the various planning topics by including a selection of key figures and characteristics in planning levels, you can allocate the planning subtasks you have defined as planning packages to the relevant employees and have these processed independently. In this way you can break down the sales planning according to customers by including the characteristic *customer* in the planning level. You can restrict the characteristic in different planning packages in such a way that each package can be processed by the employee who is responsible for the relevant customer.

Restricting characteristics in the planning package

In the planning package you can make further restrictions that have not already been made in the superordinate planning level. In this way, you define the data area on which the planning functions are to operate. Therefore, the definition of a planning package is a prerequisite for carrying out a planning function.

> In the planning level you determine whether a characteristic can be restricted at the subordinate planning package level or not. The system default is that the characteristics are restricted at the planning level; in this case the relevant fields in the subordinate planning packages are not ready-for-entry.

For more information see Planning Level [Page 116].

Copying planning packages

To support you in creating several planning packages with the same or similar structure the system provides the function *Copy planning package*. This function enables you to copy a planning package with a new name into the same planning level as the original planning package.

Structure

A planning package is defined by its name and the restrictions to the characteristics taken from the planning level. For documentation purposes, you can store a textual description for the planning package.

Ad Hoc Planning Package

The system automatically creates an ad hoc planning package for each planning level. This ad hoc package can be used as a user-defined planning package. However, you should note the following points:

- The settings that you make for an ad hoc package are valid only for the duration of the current planning session. When you end the planning session, the settings for the ad hoc package are reset.
  This makes the ad hoc package perfect if you want to display or edit explorative data without having to create permanent planning objects in order to do so. For the last purpose (for example for the delimitation of certain planning areas for recurring tasks), do not use the ad hoc package but create your own permanent package.
- The ad hoc planning package is not locked and can, therefore, be changed simultaneously by all users. For the duration of the planning session, every user has an individual view of the ad hoc package, independent of the settings that other users make in it.
- The ad hoc planning package cannot be deleted from the planning level.

See also:

Planning Environment [Page 90]

Planning Level [Page 116]

4.2.1.2.2.5 Set Up of Planning Architecture

Purpose

In order to carry out planning you must first set up the planning architecture in the Planning Environment [Page 90].

Prerequisites

When setting up the planning architecture note the following navigation principles:

- Elements are created or changed using a context menu. To create an element, use the context menu of the superordinate node. To change an element, use the context menu of the relevant element. To carry out the function that is highlighted in the context menu, double click on the relevant element.

- You select the planning area, planning level, and planning package for processing using the relevant menu function. In this way the related detailed information is shown on the right of the screen and the planning functions are shown on the bottom left. You can create and process planning functions with an activated planning level or an activated planning package. You can only carry out a planning function if you have already activated a planning package. This specifies the data area where the function is to be carried out. You activate a planning package by carrying out any menu function on it, or by double-clicking on it.

- When maintaining the individual elements on the right of the screen you can also have the technical name displayed by choosing .

- A newly created element is only included permanently in the tree if it has been checked and found to be defined in full. Then the element is available for the rest of the planning session. The check is carried out automatically when navigating to another element (at the latest before the element is saved). You can also carry out the check manually using .
  You do not have to save newly created elements or changes to elements immediately. As long as you do not close the planning session all changes are retained automatically. To definitively save an element, use . This includes a check.

  > When saving, there is another alternative: With *Planning → Save model*, you can save changes which you have made for example to planning levels, packages or functions in a planning session. However, the plan data is not saved here. In this way you can test changes to the planning objects without permanently saving the plan data that is changed.

- To carry out a planning function you first activate a planning package. Then, from the context menu of a parameter group of the relevant planning function, choose *Execute*.

  > To carry out a planning function you must always select a parameter group first. A planning function cannot itself be executed itself directly.

- To terminate processing of **one** element without saving the changes made, choose . Changes that you made previously to other elements are not affected.

Process Flow

Creating Planning Objects

1. Create *planning area*

   Create a new planning area to determine the data basis for all the following steps. Under *BW Installation*, choose the option *Local* if you want to plan with an InfoCube from the current system. If you want to plan with data from a different system choose *Remote* and enter the name of an RFC destination in the *RFC Destination* field. You can use this to access data from the other system. Then select an InfoCube from the other system.

> An InfoCube can be assigned to only one planning area. If you select an InfoCube that has already been assigned to another planning area, the system notifies you of this. You can only assign basic InfoCubes to a planning area.

2. Create *planning level*

   In one or more planning levels determine which characteristics are to be the bases for your planning, and which key figures you wish to plan. You can include any characteristics from the InfoCube characteristic list in your *planning level*. On the tab page *Selection* you decide if you want to define the selection conditions for a characteristic for each planning package or in the planning level. In this case, the selection applies for all planning packages that are subordinate to the planning level.

   > A *planning level* serves to specify certain planning topics and to make certain settings centrally for planning that will then be effective globally (for example, the period that is to be planned). For strategic and operational planning, for example, you can create distinct *planning levels* with different characteristics in each case. To determine planning content in more detail, for example if you want to tailor planning for the different people responsible, create different *planning packages*.

3. Create *planning package*

   Based on the characteristics of a *planning level*, you select characteristic values to create one or more *planning packages* for the *planning level*. The *planning package* represents the level that the planning functions refer to.

   > The system automatically provides an *ad hoc planning package* for each planning level. However, the settings that you make for this package are only valid for the duration of the planning session. At the end of the planning session, the ad hoc planning package is reset to its standard settings.

4. Create *planning function*

   You create planning functions for a planning level in order to carry out special planning tasks with the key figures of the planning level. The user-defined planning functions already contain certain default settings (for example, the selection of fields whose values are to be changed) but you cannot execute the planning functions directly. Additional parameters are required. You create these for the planning functions. Similarly, for the predefined functions, you must first create further objects with which to carry out planning (for example, planning layouts for manual planning).

5. Create *parameter group*

   You apply the planning functions using a *parameter group*. You create this for the *planning functions*. In the *parameter group* you determine the parameters that are required to carry out the planning function, for example, the percentage by which plan values are to be revaluated.

Copying Planning Objects

According to the conditions in your company, you can save time by creating planning objects for a special planning task first and then copying the previously created objects to cover similar tasks, and modify them correspondingly. This procedure is particularly effective if you use the option of copying a planning object (for example a planning area) with the appropriate subobjects (for example planning levels and packages). You can copy the following object types of the planning architecture:

- Planning area (with subobjects)
- Planning level (with subobjects)
- Planning package
- Planning function (with subobjects)
- Parameter group

To copy planning objects, proceed as follows:

1. Choose *Copy* in the context menu of the object.

   The system displays a dialog box in which you can enter further entries.

2. In each case enter a unique technical name in the dialog box for the copy to be created.

   If you are copying a planning area [Page 93] note the following: As planning areas have a 1:1 relationship to an InfoCube they cannot simply be copied or two planning areas would reference the same InfoCube.   You have the following options for copying a planning area:

   - First create a new InfoCube that is structured exactly the same as the InfoCube that the planning area to be copied is assigned to. When copying the planning area, assign the copy to the new InfoCube.
   - You decide to create the copy as a Multi-Planning Area [Page 97]. When assigning the basic planning area, select the planning area that you want to copy.

3. With objects for which this is possible, decide whether the copy should also contain subordinate objects. If you set the indicator *With subordinate objects* you will get a list of the subordinate objects after you confirm your entries. All of the objects in the list are selected for copying by default. If you do not want to copy all subordinate objects, you can put together those objects to be copied individually.

   In the selection of subobjects to be copied, it is enough to select the desired object – independently of which superordinate objects it depends on. Therefore, if you only want to copy a planning level with subordinate objects because you want to copy a certain parameter group to a planning function, you only need to select the parameter groups that you require in the list; the planning functions that belong to the parameter group are automatically copied as well.

4.2.1.2.3 Planning Profile

Use

You define planning profiles to set up a user-specific selection of planning objects. A personalized view of planning objects is particularly useful

- if you are working in large planning environments but not all of the planning objects are relevant for you

- if you want to group all planning objects by task-related factors because you, for example, only carry out administrative tasks without actually performing planning yourself.

When you use planning profiles to make administrative tasks easier, make sure that you only include those planning objects in a profile that are useful for all possible employees in the same way and without modification.

It would be meaningful, for example, to include a planning level in a profile with the corresponding functions that contain a selection for a certain planning task.

If you want to include planning packages in the profile that contain a selection for a particular employee, you have to maintain numerous individual profiles.

Instead, SAP recommends that you combine using planning profiles with the option of using variables to define worklists.

Integration

You reach planning profile maintenance from the planning workbench by clicking on *Planning Profile* or via *Goto* → *Planning Profile*.

Features

Create planning profile

You can create multiple planning profiles. For each planning profile you determine which objects are to be visible in the planning environment once the profile is set. You can separate subordinate objects, or parts of these, from the existing planning objects and have these appear in the planning profile.

You can, for example, include a specific parameter group in the planning profile from a planning level for which you have defined many planning functions. When the profile is set, only this parameter group is proposed. All other functions and parameter groups are hidden.

Assigning planning profiles

You can select and activate one of the available planning profiles. In this way you assign the selected profile to your user account. This profile remains in effect until you either set a different profile or display all the planning objects that are available in the system in the planning environment. You do this by choosing *Edit* → *General overview*.

You can assign a planning profile not only to yourself, but also to other users. To do this, you first select the desired profile and then choose *Profile* → *Assign to Other Users*. A dialog box appears with a list of all users who have access authorization for the system. Here you can choose those users to whom you want to assign the profile.

Saving the last work status for the user

Independently of whether you use planning profiles or not, as a user you can set whether the status of the planning objects as it is when you exit the planning session, should be reproduced when the planning session is started again. You do this with *Utilities* → *Settings*. When you save the *Last Work Status* setting the system expands the tree structure of the planning objects up to and including the last edited objects and reproduces the last display of the detail information in the work area.

See also:

Editing and Managing Planning Profiles [Page 125]

4.2.1.2.3.1 Editing and Managing Planning Profiles

Use

You use this function in order to split the entire set of planning objects available in the system into subsets and to assign these subsets to the users based on the different tasks.

Procedure

Creating a profile

1. To create a new planning profile, choose *Profile* → *Create*.
2. Enter a technical name and a description of the profile. Confirm your entries.
3. Define the properties of the profile (see under “editing profile”).

Editing a profile

1. Select the planning profile that you want to edit and choose *Profile* → *Change*.

   Four areas appear in the right-hand side of the screen. Of these four areas, the two on the left show the planning areas and planning functions available in the system and the two on the right show those objects that are assigned to the planning profile.

   In addition you have the option of adding global sequences to the profile or making a selection from the variables which belong to the planning areas contained in the profile. Use *Sequences* or *Variables* to do this.

2. To assign a **planning area**, a **planning level** or a **planning package** to the profile, select it and choose ▶. You can also transfer the object to the profile using Drag&Drop.

   - If you assign an object to a profile and the object's higher-level objects are not yet contained in the profile, the system adds them automatically.
   - If you assign an object to the profile and this object has lower-level objects, you are able to determine how this object is added to the profile by using the *Add to Profile with All Lower-Level Objects* option.
     - If the indicator is set, all selected objects in the first tree are transferred to the profile with all lower-level objects. In this way you can transfer a planning area with all the corresponding planning levels, packages, and parameter groups to the profile.
     - If the indicator is not set, only the selected objects in the first tree are included in the profile. In this way you can transfer a planning area without the corresponding planning levels, packages, and parameter groups to the profile.

You can assign multiple objects to the profile at once. The objects need not be of the same type: For example, you can expand the object hierarchy for two planning areas so that all planning levels and planning packages are visible. Select all expanded objects and transfer them to the profile using Drag&Drop.

3. Transfer the required **planning functions** with their parameter groups to the profile. To get a list of the planning functions to be added, double click on the planning level whose planning functions you wish to add to the profile. Select the desired functions and parameter groups and choose ▶. You can also transfer the object to the profile using Drag&Drop.

Copying a profile

1. Select the planning profile that you wish to copy.
2. Choose *Profile* → *Copy*.
3. In the next dialog box that is displayed, specify the technical name and a description of the copy.
4. Choose ✓ *Execute*.

Setting a profile

1. Select the planning profile that you want to activate and choose *Profile* → *Set*.

   You can also activate a planning profile by double-clicking on it.

2. To go back to the planning environment click on *Planning* or use *Goto* → *Planning*. The planning objects that are displayed and that can be edited are exactly those that belong to the planning profile that you have set.

   In order to reset the planning profile and to have access again to all the planning objects in the system, choose *Edit* → *General Overview* in the planning environment.

Deleting a profile

1. Select the planning profile that you want to delete and choose *Profile* → *Delete*.
2. After you have confirmed the security query that follows, the profile is removed from the system. The planning objects that were assigned to the profile remain intact.

   If you delete the planning profile that you have set for yourself, you automatically receive an overview of all planning objects each time you call up the planning environment.

Setting the sequence of the objects

When you create new objects, the system places these at the top of the relevant planning hierarchy. You can change the sequence that is produced centrally:

1. Exit profile maintenance using the *Planning* pushbutton.
2. Generate the general overview of all planning objects defined in the system using *Edit → General Overview*.
3. Move the objects affected to the desired position by choosing *Edit → Move to Upper/Lower Tree → Move Up/Move Down* (or by using Drag&Drop).

> The sequence that is set in the general overview cannot be changed in the context of a profile; it is automatically included in all profiles. Each profile inherits the sequence of objects that is determined centrally.

See also:

Planning Profile [Page 123]

4.2.1.2.4 anual Planning

Use

You use the function type **manual planning** for the manual entry of plan data and to display data which already exists. You plan values for key figures for a selection of characteristics.

Like other planning functions, manual planning also operates on one planning level [Page 116]. Which key figures and characteristics are actually available for planning depends on the respective planning level. Characteristic values for which plan data can be entered can also be restricted in the planning package [Page 119].

In order to be able to perform manual planning, a planning layout [Page 128]is created in the planning workbench.

Manual planning takes place either using Microsoft Excel spreadsheets, or with the SAP tool ALV Grid.

Planning layouts for manual planning are used as input and output layouts in planning folders (see Planning Folder [Page 269]).

You can reference planning layouts in manual planning with the appropriate components in the Web Interface Builder (see Web Interface Builder [Page 285]).

Prerequisites

You have created a suitable planning layout. For more information see Planning Layout [Page 128].

Features

Function	What you need to know
Change plan data	You use this function when you wish to enter or change plan data. The objects to be planned in a planning session and the time period depend on the settings you have made in the planning level and package. For more information, see Entering Plan Data [Page 169].

Display plan data	With this function, you select the planning objects and display the plan data for a specific combination of characteristic values.
Change plan data automatically	Directly from an open planning layout, you can call up any planning function [Page 196], and see the changed data immediately after execution.
Automatic execution of planning functions [Page 145]	You can assign a planning function or a planning sequence to a planning layout which the system executes during certain events in the layout.

See also:

Planning Layout [Page 128].

Undoing Data Changes [Page 200]

4.2.1.2.4.1 lanning Layout

Definition

A planning layout is a tool for providing data records from an InfoCube for manual planning or data entry.

The data records of an InfoCube can be seen as comprising characteristics that form the key, with key figures forming the data part of the records (see Star Schema [External]). Planning level characteristics and key figures are assigned to the header, lead column, and data column area in a planning layout. At runtime for manual planning the key figure values of the data records for every combination of characteristic values are presented in the header in a matrix spanning the lead columns and data columns.

Structure

A planning layout consists of a header area and the lead and data columns. All characteristics of the planning level have to be distributed to the areas of the planning layout. The planning layout also has to contain at least one key figure.

Header area

The header area only contains characteristics. The values of the characteristics in the header area lie within the selection permitted in the planning level or planning package. SAP recommends you include characteristics in the header area that do not vary greatly.

You can also structure a planning layout without a header area. Normally this is only useful if you are only using a few characteristics and only a few combinations can be formed from the characteristic values.

For planning layouts which should be used in Web applications or in planning folders, the values of characteristics must be restricted to a single unique combination in the header area. Otherwise the layout cannot be executed in Web applications or planning folders.

Data columns

The underlying structure of the data columns is always fixed: The columns are created individually in the layout builder. Which key figure and which characteristic values are to be contained in the column are determined at that point. Each planning layout must have at least one data column.

To keep the number of the planning layout as low as possible, data columns can be parameterized: You can use a variable for a characteristic value or identify a data column for a characteristic as *dynamic*.

In this case the final structure of the data column is only determined at runtime: The system replaces variables and generates, from a dynamic column, as many columns as there are values according to the selection made in the master data. Only then is data from the InfoCube read and structured according to the structure of the planning layout. In this way the data columns for the planning layout always have a fixed structure (at runtime).

You can only use a characteristic in a data column if the selection for this characteristic is already determined in the planning level. This restriction does not apply to a dynamic characteristic in the data column.

Lead columns

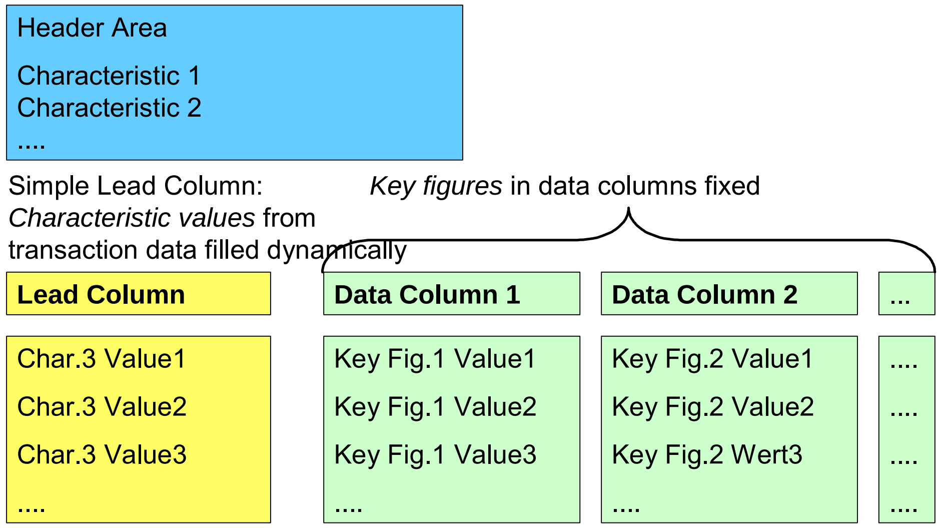
Planning layouts can be divided into two basic classes:

- Those rows that appear at runtime and are **not** determined in the planning layout. The planning layout only describes which characteristics are to be used in the lead columns. The combination of characteristic values results from the data selected from the InfoCube. Lead columns of this kind are called *Simple Lead Columns*.
- As with data columns, you want to define each individual row in the planning layout. In this case the layout has just one lead column. You can determine which key figure and which characteristic values you want to use for each row on an individual basis. Here you have the same formatting options as in the data columns. As in the data columns, you can use a key figure and a characteristic value together. Therefore this lead column is also called a *Complex Lead Column*.

The key figures can either be used in the data columns or in the individually defined rows of a lead column. Each planning layout must contain at least one lead column.

The following graphic shows the ways in which you can structure planning layouts with the permitted layout types, in combination with different uses of characteristics and key figures:

Case 1: Layout type *key figures in data columns*



Header Area

Characteristic 1
Characteristic 2
....

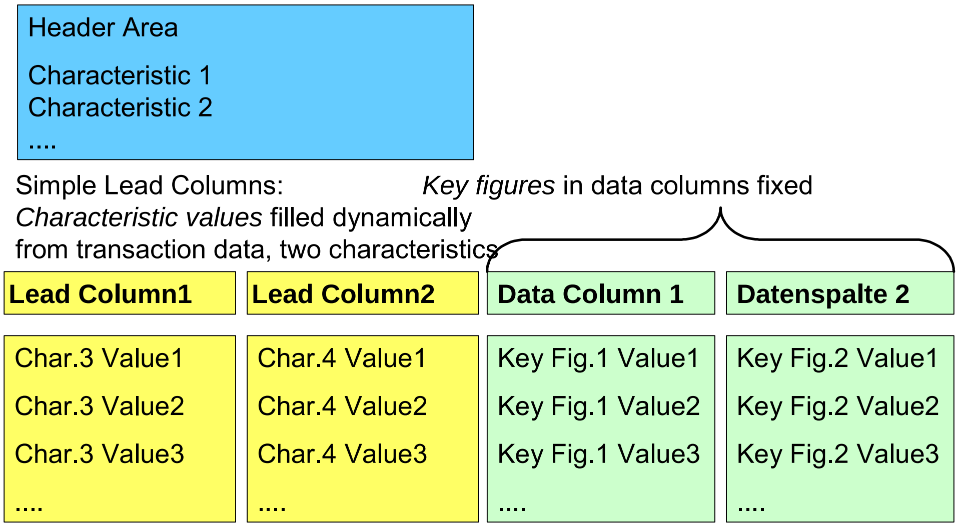
Simple Lead Column: *Characteristic values* from transaction data filled dynamically

Key figures in data columns fixed

Lead Column	Data Column 1	Data Column 2	...
Char.3 Value1	Key Fig.1 Value1	Key Fig.2 Value1	....
Char.3 Value2	Key Fig.1 Value2	Key Fig.2 Value2	....
Char.3 Value3	Key Fig.1 Value3	Key Fig.2 Wert3	....
....	....	....	....

If you display key figures in the data columns, the lead column shows which characteristic values the plan data in the data columns refers to. The number of rows in the lead column is not fixed but is determined dynamically from the transaction data.

Case 2: Layout type *key figures in data columns*



Header Area

Characteristic 1
Characteristic 2
....

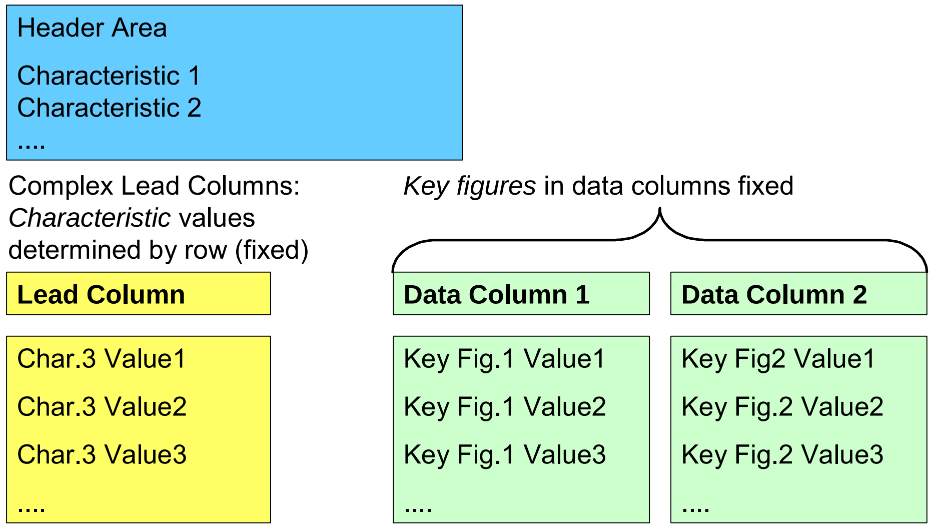
Simple Lead Columns: *Characteristic values* filled dynamically from transaction data, two characteristics

Key figures in data columns fixed

Lead Column1	Lead Column2	Data Column 1	Datenspalte 2
Char.3 Value1	Char.4 Value1	Key Fig.1 Value1	Key Fig.2 Value1
Char.3 Value2	Char.4 Value2	Key Fig.1 Value2	Key Fig.2 Value2
Char.3 Value3	Char.4 Value3	Key Fig.1 Value3	Key Fig.2 Value3
....	....	....	....

This planning layout differs from the first case only in that two characteristics have been included in the lead columns. For every additional characteristic, an additional lead column is generated. Therefore you see in the lead columns the exact combinations for characteristic values for which data records exist.

Case 3: Layout type *key figures in data columns, rows defined individually*



Header Area

Characteristic 1
Characteristic 2
....

Complex Lead Columns: *Characteristic* values determined by row (fixed)

Key figures in data columns fixed

Lead Column

Char.3 Value1
Char.3 Value2
Char.3 Value3
....

Data Column 1

Key Fig.1 Value1
Key Fig.1 Value2
Key Fig.1 Value3
....

Data Column 2

Key Fig2 Value1
Key Fig.2 Value2
Key Fig.2 Value3
....

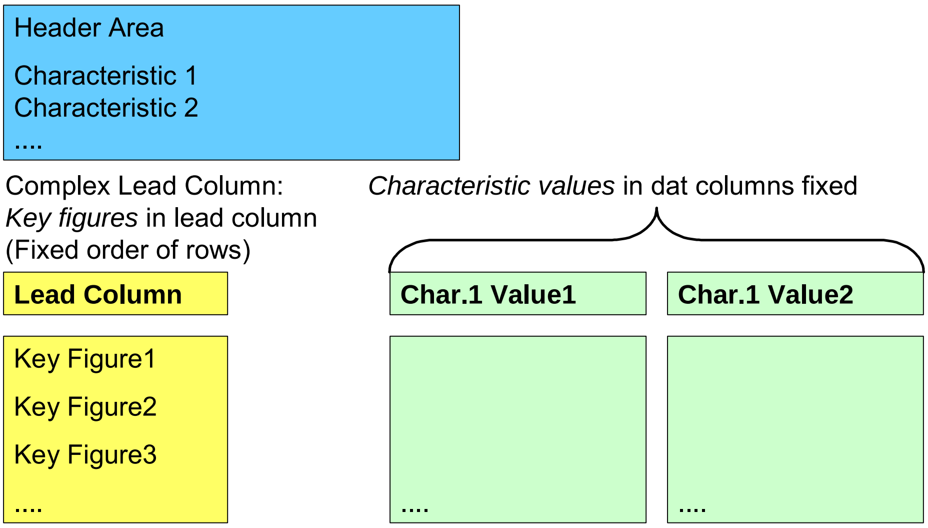
You can also create a planning layout so that you explicitly determine the characteristic values to be displayed in the layout in the definition of the layout.

The following example shows a planning layout with the characteristics fiscal year and business area in the lead columns. The revenue and the sales deductions are output for each fiscal year for the selected business area.

Revenue planning

1. Lead Column	**2. Lead Column**	**1. Data Column**	**2. Data Column**
Fiscal year	Business area	Revenue	Sales deductions
2000	Plant construction: Technical gases	5002100	120350
2001	Plant construction: Technical gases	5502250	130285

Case 4: Layout type *key figures in lead columns*



Header Area

Characteristic 1
Characteristic 2
....

Complex Lead Column:
Key figures in lead column
(Fixed order of rows)

Characteristic values in dat columns fixed

Lead Column

Key Figure1

Key Figure2

Key Figure3

....

Char.1 Value1

....

Char.1 Value2

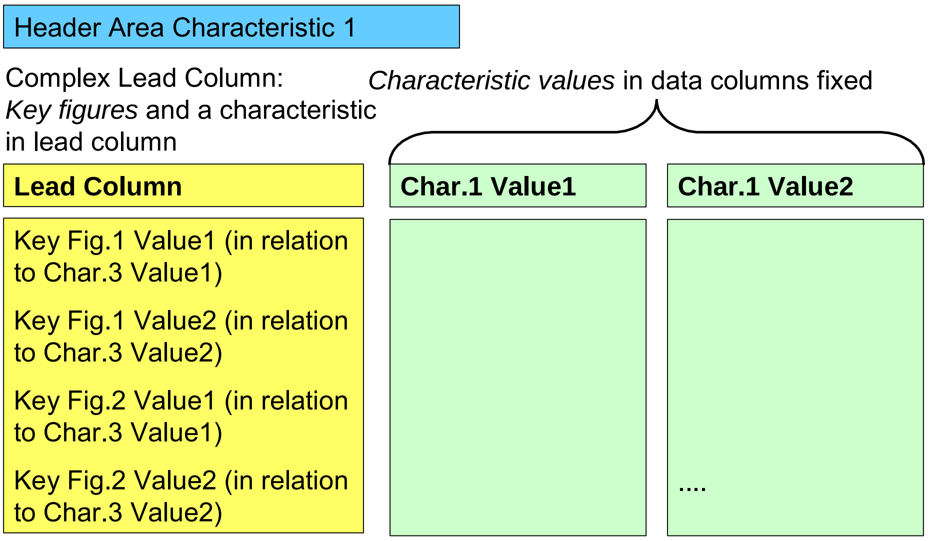
....

The following example shows a planning layout with a number of key figures in the lead column and the characteristics fiscal year and business area in the data columns.

Business area planning

1. Lead Column	**1. Data Column**	**2. Data Column**	**3. Data Column**	**4. Data Column**
	2000: Plant construction	2000: Technical gases	2001: Plant construction	2001: Technical gases
Gross domestic revenue				
Gross foreign revenue				
Stock changes				
Material expense				
Wages and salaries				
Depreciation				
Operating profit (Formula)				

Case 5: Layout type *key figures in lead columns*


Header Area Characteristic 1
Complex Lead Column:
Key figures and a characteristic in lead column
Characteristic values in data columns fixed
Lead Column
Char.1 Value1
Char.1 Value2
Key Fig.1 Value1 (in relation to Char.3 Value1)
Key Fig.1 Value2 (in relation to Char.3 Value2)
Key Fig.2 Value1 (in relation to Char.3 Value1)
Key Fig.2 Value2 (in relation to Char.3 Value2)
....

This planning layout differs from the previous case only in that a characteristic has been included in the lead column.

You are also able to mix key figures and characteristics in the data columns. Complex lead columns and data columns essentially behave in the same manner in the Layout Builder (Exception: Sign attribute as its own data column).

Rules for using characteristics

For the positioning of characteristics, SAP recommends that you include,

- few variable characteristics in the header area
- many variable characteristics in the lead columns
- characteristics that you always want to be able to see in the data columns.

Rules determine whether a characteristic can be displayed in one of the three available areas (header area, lead column, data column):

- According to presentation variants (see Determining Basic Settings [Page 134]).
- According to the selection conditions in the planning level (characteristics can then only be displayed when the characteristic values have been restricted in the planning level).

If you want to determine characteristic values using variables then further conditions apply. For more information, see Variables [Page 108].

Formulas in the data column

To perform advanced calculations with the plan data you can add additional data columns to the data layout for formulas that you have created with Microsoft Excel.

Applications of this kind might be, for example, plan-actual deviation, margins, or key figures you have calculated.

For more information, see Formatting Generated Layout in Preview [Page 155] .

Filling column and row contents dynamically

For data and lead columns, you can set up that the number of columns or rows visible in the layout are determined dynamically at runtime from those characteristic values which are contained in the selection of the planning package. For more information, see Dynamic Data and Lead Columns [Page 146].

See also:

Determining Basic Settings [Page 134]

Determining Detail Settings [Page 151]

Formatting Generated Layout in Preview [Page 155]

Entering Plan Data [Page 169]

Planning Environment [Page 90]

4.2.1.2.4.1.1 Determining Basic Settings

Use

In the basic settings of the planning layout you specify the name, determine the layout type, and make settings for the characteristics. Additional settings are also available.

Prerequisites

In order to create a planning layout, you first have to create the *Manual Planning* function type for the planning level. You do this in the planning functions [Page 196] area.

Procedure

1. In the planning environment, in the *Planning Functions* area, choose *Create Layout* in the context menu of the *Manual Planning* function.

   A dialog box appears in which you can enter the technical name and a description for the layout.

2. In the group field *Layout Type,* specify the layout area in which **key figures** should be displayed. You have the following options:

   Key figures in data columns: You can specify key figures for the data columns. Each key figure of the layout is displayed in its own column. You can use additional characteristics in the data columns so that you only obtain key figure values for certain characteristics. The rows contain all the values of the characteristics included in the lead column for which data is available. The system fills lead columns in accordance with the detail settings (see Determining Detail Settings [Page 151]).

   Key figures in data columns, rows defined individually: You can specify characteristic and key figure combinations for the data columns. You can define the rows of the lead column individually by specifying combinations of characteristic values and texts that you want to use in the detail settings (see Determining Detail Settings [Page 151]).

   - *Key figures in lead columns*: You can only specify characteristic values for the data columns. You can specify a combination of characteristic and key figure values, as well as a text, for every row in the lead column.

If you change this layout type later, the entire layout is reset to the default settings.
If you selected characteristics for the lead column but later want to show key figures in the lead column, the characteristics are automatically transferred to the header area of the layout.

3. Specify the position of the **characteristics** in the *Settings for Characteristics* list:

   For each characteristic included in the planning level, you can specify whether it is to be assigned to the header area, the lead column or the data column.

   Under certain circumstances, the system restricts the assignment of characteristics: A characteristic can only appear in the data column if it has been restricted in the planning level. A characteristic that is only restricted in a planning package however, cannot appear in a data column.

   In the layout in the *Characteristic Text* column, you can specify the length variant that you want to display for the texts of characteristic values.

   For characteristic values that are significant in every length variant (for example, fiscal year), you can use the short text in order to save space. In other cases, (company, for example), it is more advisable to use a longer variant for reasons of comprehensibility.

4. Make further settings as required. For more information, see

   Hierarchically Structured Planning [Page 135]

   Automatic Execution of Planning Functions [Page 145]

   Dynamic Data and Lead Columns [Page 146]

Result

Choose *Continue* to get to the next step of planning layout creation. See Determining Detail Settings [Page 151].

See also:

Planning Layout [Page 128].

4.2.1.2.4.1.1.1 Hierarchically Structured Planning

Use

Depending on particular modeling aspects, you can plan using various hierarchical structures with either a hierarchical or a non-hierarchical data model. This section explains *modeling aspects*, the option of *planning with hierarchical structures*, and the *data models for manual planning*.

Modeling Aspects

In terms of modeling, the following examples of hierarchical structuring are particularly frequent:

- You have defined various characteristics as belonging together in a hierarchical relationship.
- Characteristic values of a single characteristic should be structured in a hierarchy.

However, as data in an InfoCube is stored in a flat structure, an InfoCube does not contain internal hierarchy nodes.

According to the modeling aspects named, you have the following basic options for modeling hierarchies:

- **Determine hierarchical structures in InfoCube modeling**

  Add a characteristic of your own for each level of the hierarchy. For a two-level hierarchy, for example, insert *product group* and *product*. Using appropriate tools (such as characteristic relationships [Page 110]), the system ensures that only those characteristic values that are permitted by this hierarchical structure are stored in the InfoCube.

  For more information on the modeling aspects of hierarchical structures, see Hierarchies [External] and Options for Hierarchical Modeling [External].

- **Hierarchy definition for characteristics (BW characteristic hierarchies, “external hierarchies”)**

  A BW characteristic hierarchy is created for a characteristic. A hierarchy of this type cannot be seen in the InfoCube data but is stored in its own data table. When data is read from the InfoCube, the hierarchy is read too. The data is processed according to the hierarchy structure. Key figure values of hierarchy nodes are always calculated. (They are not in the InfoCube but can, where necessary, be stored redundantly in an aggregate). In this sense, a BW characteristic hierarchy is an ’external hierarchy’.

  As every **postable** node corresponds to a characteristic value from the hierarchy basic characteristic, key figure values for this characteristic value may also have to be stored in the InfoCube. To facilitate this, each postable node implicitly contains a leaf with the characteristic value of the node. This leaf contains the delta.

  In hierarchies that do not have postable nodes, the key figure values for internal nodes only contain totals and cannot be changed.

  For more information on BW characteristic hierarchies, see Hierarchy Nodes [External] and Modeling Nodes and Leaves [External].

Planning with Hierarchical Structures

According to the modeling aspects named, you have the following basic options for planning using hierarchical structures:

- **Mapping hierarchically structured planning using appropriate planning levels**

  A hierarchical structure that is determined in InfoCube modeling can be visualized in a planning layout using subtotals. Use the characteristics – starting with the characteristic for the first hierarchy level – in the lead columns and use the *Total as Hierarchy* setting in the Layout Builder.

- ○ If you want to plan just those values on the first level, only use the characteristic on the first level of the hierarchy in the planning level and the planning layout.
- ○ If you also require data from lower levels you can distribute using a top-down planning function.

Note that a hierarchy that you map using characteristics in InfoCube modeling has to be balanced: Each path from the root to the leaf has to have the same number of hierarchy levels.

- **Mapping hierarchically structured planning with BW characteristic hierarchies**

  You can execute planning with both balanced and unbalanced BW characteristic hierarchies. As the hierarchy structure is not contained in the InfoCube data, BW characteristic hierarchies are more flexible when it comes to changing the hierarchy structure than hierarchical structures that are determined using characteristics in InfoCube modeling. However, processing requires more effort, particularly when node values have to be changed.

Data Model for Manual Planning

For manual planning with hierarchical structures, the important question is whether it should be possible to change key figure values on hierarchy nodes (example: budgeting). If this is the case, the system has to ensure that the key figure values are determined and stored in the appropriate way in the InfoCube. The interpretation of the hierarchy structure is essential. For this reason the following cases are differentiated in the system:

Manual Planning

Data Model	What You Need to Know
Non-hierarchical data model	In this model, all hierarchy nodes only contain totals and cannot be changed. SAP differentiates the following cases: • You are not using hierarchies. • You are using hierarchical structures determined using characteristics in InfoCube modeling. • You are using BW characteristic hierarchies that do not contain postable nodes.
Hierarchical data model	In this model, hierarchy nodes should be ready for input. SAP differentiates the following cases: • If you want to use your own characteristics for the hierarchy levels, you can use BPS characteristic hierarchies. • If you want to structure the characteristic values of a characteristic hierarchically, use BW characteristic hierarchies that only contain postable (inner) nodes.

Hierarchical data models will now be explained further.

Hierarchical Data Model

If you are using a *hierarchical data model* as the basis of your manual planning, you can enter values on the nodes of the hierarchy. In order to use this functionality you have to use a hierarchy in the lead column in the planning layout.

You can select one of the following hierarchy types:

Hierarchy Types in Planning Layouts

Name	What You Need to Know
BW characteristic hierarchy with postable nodes	Hierarchy created for a hierarchy basis characteristic with (inner) nodes that correspond to a characteristic value of the hierarchy basic characteristic. BW characteristic hierarchies are stored on the database and can be used outside of planning in aggregates and reporting. The postable nodes are ready for input in manual planning. At runtime, a leaf with the same characteristic value is always generated for a postable node. This leaf contains the not yet distributed values. (This corresponds to the dummy leaves in BPS characteristic hierarchies).
BPS characteristic hierarchy	Virtual hierarchy that is defined when you structure a planning layout and is not stored on the database. You can build BPS characteristic hierarchies completely freely using the characteristics available in a planning level. At runtime, the system generates a characteristic hierarchy from the sequence of characteristics in the lead column. Each characteristic corresponds to one level of the hierarchy. If you want to prevent a characteristic becoming a part of the hierarchy, you must either move it to the header area or to a data column of the layout, or remove the characteristic from the planning level entirely. For more information, see BPS Characteristic Hierarchies [Page 140].

You **cannot** combine both hierarchy types within one planning layout. You can use either BW characteristic hierarchies or BPS characteristic hierarchies.

Advantages and Disadvantages of Hierarchy Types

Name	What You Need to Know
BW characteristic hierarchy with postable nodes	Advantages: • Both balanced and unbalanced hierarchies are possible. • Greater flexibility with changing hierarchies. Disadvantages: • As the values on the nodes of the hierarchies can only be calculated at runtime, SAP recommends that you only work with small hierarchies or subtrees (less than 1000 nodes) in manual planning. • The system can only ever work with a whole subtree for a hierarchy.

BPS characteristic hierarchy	Advantage: • No restrictions in terms of hierarchy size. Disadvantages: • Only balanced hierarchies are possible. • Greater effort involved in changing hierarchies. As the connection between the hierarchy levels is dependent exclusively on the order in the planning layout, SAP recommends that you do not change the order of the characteristics after they have been posted the first time.

Functions

You open the *Hierarchy in Lead Column* dialog box by clicking on *Hierarchy* on the screen for determining the basic settings of the planning layout (see Determining Basic Settings [Page 134]). You then make the following settings:

Choice of Data Model

- *Non-hierarchical data model*
    - *Calculate Data Locally*

      If you want to use the local SAP functions for *Microsoft Excel*, activate this setting. See also, Local Functions [Page 174].
- *Hierarchical data model*

If you have chosen a *hierarchical data model*, the system makes the following settings eligible for selection or ready for input:

Hierarchy Type

- *BW hierarchy with postable nodes*
- *BPS characteristic hierarchies.* For more information, see BPS Characteristic Hierarchies [Page 140].

Summation Logic

- *top-down*

  When the value of a node is changed, the change is deducted from the leaf of the higher-level node that contains the not yet distributed value.
- *bottom-up*

  When the value of a leaf is changed, the values of the leaves are simply summarized. The value of the leaf with the not yet distributed value does not change. However, if the value of a node with lower-level elements is changed, the system updates the change to the leaf with the not yet distributed value that belongs to the node.

Check Entries

If you switch on the validation function, the system checks if the sum of the values entered in the hierarchy branches is smaller than or equal to the value of the higher-level node. If the sum is greater than the node value, then the data cannot be posted.

Activities

You can execute the following activities on the screen for determining the basic settings of the planning layout:

1. If you want to structure your planning layout hierarchically, the following conditions have to be fulfilled:
    - In order to use BPS characteristic hierarchies, you need to include at least one characteristic in the lead column of the planning layout.
    - In order to use BW characteristic hierarchies, you can include a maximum of one characteristic in the lead column of the planning layout. A hierarchy must have been defined in SAP BW for this characteristic and you must have assigned one of these hierarchies in the planning level or package to the characteristic as a selection.
2. You open the *Hierarchy in Lead Column* dialog box by clicking on *Hierarchy*. Make the required settings.
3. Under *Additional Settings* → *Characteristic Combinations*, decide which characteristic values are to be displayed in the lead column. For more information, see Characteristic Combinations [Page 144]

Further Notes

A *BW characteristic hierarchy* is displayed in the planning layout using the Microsoft Excel outline function. Therefore, you can also use this function when displaying and changing plan data (certain areas of the hierarchy can be expanded and collapsed individually).

> The outline function is only available in the display or change mode of the executed layout. However, when processing the layout, only some sample values are displayed in order to give you an impression of the executed layout.
>
> In order to actually display a BW characteristic hierarchy in a planning layout as a hierarchical structure, choose *Lead Column Settings* in step 3 of the layout definition, and set the *Hierarchy* indicator. Otherwise the characteristic values contained in the hierarchy are output in the layout as a single-level list.

Entering plan data in a planning layout for *BPS characteristic hierarchies* is only slightly different from entering plan data in a planning layout without hierarchies. For more information, see Entering Plan Data [Page 169].

4.2.1.2.4.1.1.1.1 BPS Characteristic Hierarchy

Definition

Virtual hierarchy generated by the system from the characteristics in the lead columns at runtime. You can enter values on the nodes of the hierarchy after it has been generated.

Use

For planning layouts with a hierarchical data model, a hierarchy has to be consulted along with the transaction data in order that the data is correctly interpreted and processed. If you set the indicator *BPS Characteristic Hierarchies* in the *Hierarchy in the Lead Column* dialog box, manual planning switches to a special mode. In this mode the system generates a characteristic hierarchy at runtime from the sequence of characteristics in the lead column. Each characteristic corresponds to one level of the hierarchy.

Structure

The structure of the hierarchy can be illustrated in this example: You have the characteristics *article group* and *article* (in this order) in the planning layout. You have the key figure *value* in the data column. The characteristic combinations [#, #], [AG1, #], [AG1, A1], [AG1, A2], and the combinations [AG2, #], [AG2, A3] and [AG2, A4] have been entered. Here '#' indicates the initial characteristic value *Non-Assigned*.

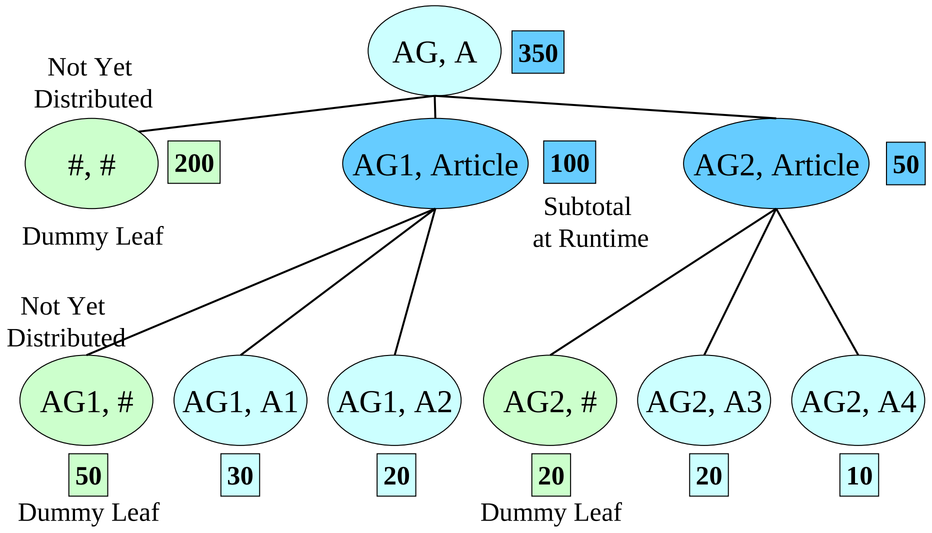
Example of BPS characteristic hierarchy

Product group	Article	Value
Article group	Article	350
#	#	200
AG1	Article	100
	#	50
	A1	30
	A2	20
AG2	Article	50
	#	20
	A3	20
	A4	10

The system also generates the hierarchy nodes for the characteristic hierarchy as additional combinations. For each hierarchy node there is one combination of characteristic values that bears the not yet distributed values. This kind of combination is called a *dummy leaf*.

- Combination [#,#] belongs to the total [*article group*, *article*]. The corresponding value 200 has not yet been distributed on the article group, article level.
- Combination [AG1, #] belongs to the subtotal [AG1, *article*]. The corresponding value 50 has not yet been distributed on the AG1, *article* level.
- Combination [AG2, #] belongs to the subtotal [AG2, *article*]. The corresponding value 20 has not yet been distributed on the AG2, *article* level.

The following graphic illustrates the structure of the BPS characteristic hierarchy in this example:



Integration

If you have set the *BPS Characteristic Hierarchies* indicator, the system generates the subtotals. These subtotals are only available at runtime. They are not stored in the database. If the planning layout is displayed in manual planning using Microsoft Excel, the hierarchy is visualized with the Excel group levels.

System behavior when entering plan data subject to summation logic

Dummy leaves for the subtotals are **not** ready for input.

- If you choose the *top-down* option, the first postable node is also **not** ready for input.
- If you choose the *bottom-up* option, the postable nodes are all ready for input. There are exceptions to this. These are mentioned in the following points.

A **subtotal** is ready for input when an associated combination of characteristic values has been entered that contains the not yet distributed values. That is, when the corresponding dummy leaf is available.

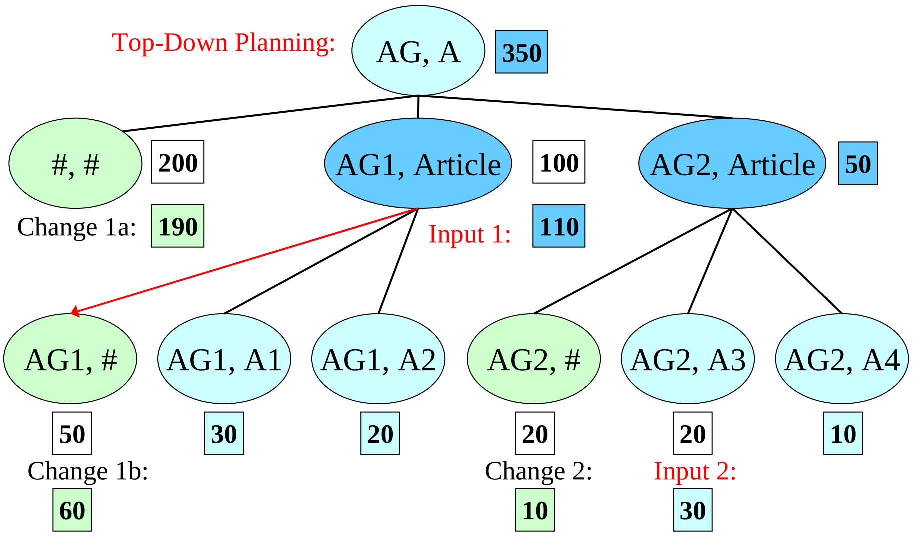
- If you choose the *top-down* option, any change to the value in the subtotal causes a corresponding change to be made to the value of the dummy leaf.

  Example

  Input 1: If the value of the subtotal [AG1, article] is increased from 100 to 120, the system adjusts the value of [AG1, #] to 60 and the value of [#, #] to 190.

  Input 2: An increase in the value on one leaf – for instance [AG2, A3] – reduces the value of the dummy leaf [AG2, #] accordingly.

  The following graphic illustrates this example:

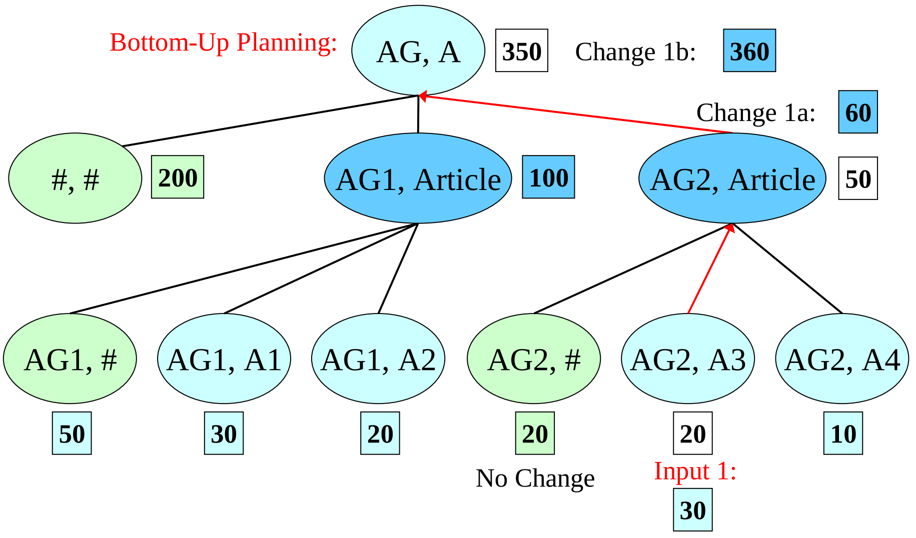


- If you choose the *bottom-up* option, input on one leaf is not deducted from the lower-level leaf. Instead, the value of the higher-level node is changed accordingly. Only an input on a node with lower-level elements causes a change to be made to the corresponding dummy node.

  Example

  Input 1: If the value of the leaf [AG2, A3] is increased from 20 to 30, the system changes the value of [AG2, article] accordingly from 50 to 60 and the value of [AG, A] to 360.

  The following graphic illustrates this example:



In the top-down mode, if the appropriate dummy leaf for a subtotal is not there or is locked in a data slice for the planning, the corresponding subtotals and lower-level elements of this subtotal from the subsequent level are **not** ready for input. In bottom-up mode it is only the corresponding subtotal that is not ready for input.

Further notes

- In order to assort newly entered rows (that is, new combinations of characteristic values) automatically in manual planning, choose *Check*.
- Characteristic combinations that may contain initial values for some characteristics but that do not fit in the characteristic hierarchy are **not** displayed in manual planning. It is also not possible to re-enter these combinations.

  Example: A combination of this kind could be [#, A1].
- If you have chosen the option *Check Input* in the *Hierarchy in the Lead Column* dialog box, the system checks whether the values for the dummy leaf are greater than or equal to zero. For more information, see Hierarchically Structured Planning [Page 135].

4.2.1.2.4.1.1.1.2 Characteristic Combinations

Use

Normally in manual planning those exact combinations of characteristic values are presented for which transaction data is already available. Alternatively to this transaction data-dependent view of the planning layout, you can also determine that all valid combinations of characteristic values (taking account of the selections in the planning level or package) are displayed and are offered for data entry. The layout is then dependent on master data. For combinations where no data has yet been entered, the value 0 is displayed for every key figure.

> When you choose that the layout is presented depending on the master data in a planning layout, you should make sure that the selection in the planning level or package (and if required the characteristic relationships) provides for a sufficient restriction of the possible characteristic combinations. Otherwise multiplying all the characteristic values with each other can result in overdimensional data sets that are no longer easy to manage.

Integration

As an alternative to the presentation dependent on master data based on the current selection, you can formulate differentiated rules for valid combinations of characteristic values. You achieve this with the help of characteristic relationships [Page 110] that you define in the context of the planning area. Here the system can only propose those characteristic values that fall within the selection.

Features

In determining basic settings for the planning area you can choose between a master data-dependent presentation and a transaction data-dependent presentation.

Activities

On the screen where you determine the basic settings for the planning layout, choose which characteristic values are to be displayed by selecting *Additional Settings → Characteristic Combinations.*

- Choose *From Transaction Data* to only display those characteristic values for which data has already been posted. In this case you can enter new lines for additional characteristic combinations in the executed layout.
- Choose *All Possible Characteristic Combinations* in order that the system generates all characteristic combinations according to selection. In this case you cannot enter **any** new lines in the executed layout.

> Note that this setting affects all characteristics in the planning layout because the characteristic relationships can, for example, also restrict the possible combinations for header characteristics.

4.2.1.2.4.1.1.2 Automatic Execution of Planning Functions

Use

You use this function, in the context of an executed planning layout, to make sure that a function that you have determined is always executed with particular actions.

> This allows you, for example, to assign a function of type *Currency Translation* to a layout that you are using to enter amounts in the local currency. The function automatically converts the entered amounts into the group currency. This means that the amounts are available in both currencies at all times.

Prerequisites

You have already created planning functions and parameter groups which you want to assign to the layout.

Features

You can assign any planning function to be executed automatically to a planning layout. However, *documents* and *manual planning* **cannot** be used here.

You can use functions from the context of the same planning area to which the layout also belongs, or you can enter a function from any other planning area. In this way, you are able to synchronize data in different InfoCubes with each other.

In addition to the usual function types such as *Copy* and *Repost* you can also specify a planning sequence to trigger more complex actions. For this you can use planning sequences of any particular planning area or global, cross-area sequences.

A function which you assign to a planning layout for automatic execution is automatically executed on the following events, that is **before** the action begins which is triggered by the event:

- Saving
- Navigation from the planning layout to another object of the planning environment
- Executing a planning function with a planning layout open

This is referred to as the *commit event*: This is the time at which manual planning transfers the changed data back to the BW-BPS transaction data buffer.

Each function uses the buffer. Data to be validated may already exist in the buffer. For this reason a function that is executed automatically after the commit event is **not** suitable for data validation.

Activities

1. In the first step of the layout definition choose *Automatic Plng Funct*. The system displays a dialog box.
2. Decide whether you want to assign a planning function, an area-specific planning sequence, or a global planning sequence.
3. Make the entries required by the system one after the other from left to right. Through every input field which you fill in, you simultaneously control the possible values of the F4 Help in the following input field.
4. In the input field *Package*, you can decide whether you want to enter a package or not:
    - If you enter a package, the function is executed for that data that is contained in the selection of the package entered.
    - If you do not enter any package, the function is executed for that data that is contained in the selection of the package that you are editing with the planning layout at the time of execution. The prerequisite for this is that you have entered the same planning area and the same level which the planning layout itself also belongs to.

See also:

Planning functions in planning folders

- Automatic Execution of Planning Functions [Page 279]
- Automatic Execution of Global Functions [Page 282]

4.2.1.2.4.1.1.3 Dynamic Data and Lead Columns

Use

Alongside variables, dynamically created data and lead columns are an additional way in which you can parameterize the structure of a planning layout. This functionality allows you to define data columns or rows in the lead column as a template for a characteristic.

At runtime the system generates the completed planning layout. In this planning layout the system first determines the possible characteristic values from the planning package and then inserts these values in the dynamic columns and rows sequentially. The number of rows or columns in the planning layout is calculated from the number of characteristic values in the selection for the characteristic that is flagged as dynamic. In this way, a number of columns or rows are generated dynamically from one column or row.

Dynamic columns enhance your options to control the characteristic values to be displayed in the following way:

Lead column

For just one of the characteristics in the lead column you specify that the rows should be displayed dynamically corresponding to the characteristic values contained in the planning level or package, in accordance with the current selection conditions.

> You select the characteristic *article* for the dynamic lead column and in addition include the characteristic *country* in the lead column. In the second step of the layout definition you first combine *article* with the (static) value **DE** and in a further row with the value **USA** of the characteristic *country*. With a selection of the articles **A1**, **A2** and **A3**, you get six rows for the following combinations: **A1/DE**, **A2/DE**, **A3/DE**, **A1/USA**, **A2/USA**, **A3/USA** when executing the layout in the lead column.

Data column

For just one of the characteristics in the data column you specify that the columns should be displayed dynamically corresponding to the characteristic values contained in the planning level or package, in accordance with the current selection conditions.

> You select the characteristic *fiscal year* for the dynamic data column and in addition include the characteristic *country* in the data column. In the second step of the layout definition you first combine *fiscal year* with the (static) value **DE** and in a further row with the value **USA** of the characteristic *country*. With a selection of the fiscal years **2001**, **2002** and **2003**, you get six data columns for the following combinations: **2001/DE**, **2002/DE**, **2003/DE**, **2001/USA**, **2002/USA**, **2003/USA** when executing the layout.

> The selection of a characteristic in the data or lead column that should be handled dynamically can, apart from fixed values, also take place with the help of a **variable**. In this case, you can build a scenario for rolling planning with the joint use of dynamic columns and variables that is particularly easy to maintain. For more information see Rolling Planning [Page 149].

Prerequisites

Compound characteristics

A characteristic that you want to present in dynamic columns is not allowed to be contained as a higher-level characteristic in a compound characteristic. You can use compound characteristics in dynamic columns if the higher-level characteristic is restricted to a single value in the planning level.

Layout types

You can only create dynamic lead columns for the layout types *key figures in data columns, rows defined individually*, and *key figures in lead columns.* Dynamic data columns, on the other hand, are supported for all layout types.

Features

From the characteristics contained in the planning level, you can choose a characteristic that should be handled dynamically in the data or lead column. The system offers you the available characteristics in a selection list. In this way you can determine whether the characteristic value, or a text, should be used in the header of the dynamic data column (or in the rows of the lead column). With texts you can choose either short, medium or long texts.

In the data or lead columns you can add further (static) characteristics in addition to a dynamic characteristic. For such additional characteristics, you must determine the desired values explicitly as usual. You can combine the dynamic characteristic with as many explicitly entered values of the remaining characteristics as you like.

In addition, it is possible to add comparison columns that contain such predetermined values for the dynamic characteristic which are outside the dynamically determined selection.

> If, for example, the characteristic fiscal year appears dynamically with the characteristic values 2001, 2002, and 2003 in the planning layout, you can add a comparison column for the (static predetermined) fiscal year 2000 that always appears in the layout independent of the dynamic value determination.

When generating the new rows or columns, adjacent rows or columns are correspondingly moved either down or to the right.

> If you are using Microsoft Excel for manual planning and have your own areas in an Excel worksheet with your own formats or formulas, these areas may be overwritten if you use dynamic rows or columns. This is because the final structure of the planning layout depends on the number of characteristic values for the dynamic characteristic. If you have special requirements as to the structure of the entry screen, you can create these on your own worksheet. You can determine the layout structure using VBA macros and copy data between worksheets.

Activities

1. Choose *Additional Settings* in the first step of the layout definition. The system displays further setting possibilities.
2. If you want to create a dynamic lead column ensure that you are using a suitable layout type (*key figures in data columns, rows defined individually* or *key figures in lead columns*).
3. Choose *Dynamic for* in the group field *Lead Columns* or *Data Columns* for the characteristic that should be treated dynamically in the corresponding layout area.

   In the *Settings for Characteristics* table the system modifies the settings automatically according to your choices. The system inserts the symbol ▦ in the *Restriction* column to indicate the characteristic that is to be treated as dynamic.
4. Choose the desired *Text Replacement*.
5. Choose *Continue*. You get to the second step of layout definition. From this point, the process for dynamic lead columns and data columns differs:

   a. If you are defining dynamic lead columns, choose the tab page *Lead Columns*.

      In a row of the table *Configuration of the lead columns* choose the indicator *Dyn*.

The system automatically enters in the column intended for the dynamic characteristic its technical name as well as in the column *Text*. The technical name in the column *Text* is replaced by the dynamically determined value of the characteristic when executing the layout.

To display the respective value of a dynamic characteristic in the lead column (or in the column header of a data column) as a label, you must always enter the technical name of the characteristic in the layout definition. This also applies when the characteristic values are determined at runtime using a variable: You also use the technical name of the characteristic in this case (and not that of the variable).

If necessary, include more information on the key figure and further (static) characteristics.

b. If you are defining dynamic data columns choose the tab page *Data Columns* and proceed as with dynamic lead columns.

6. Continue the layout definition as usual.

4.2.1.2.4.1.1.3.1 Rolling Planning

Use

You execute rolling planning in order to replace plan values with the actual values of the previous period within a fixed time context after every completed period. You can protect these actual values against changes and if necessary fix the plan data again for the remaining periods in the predefined time frame.

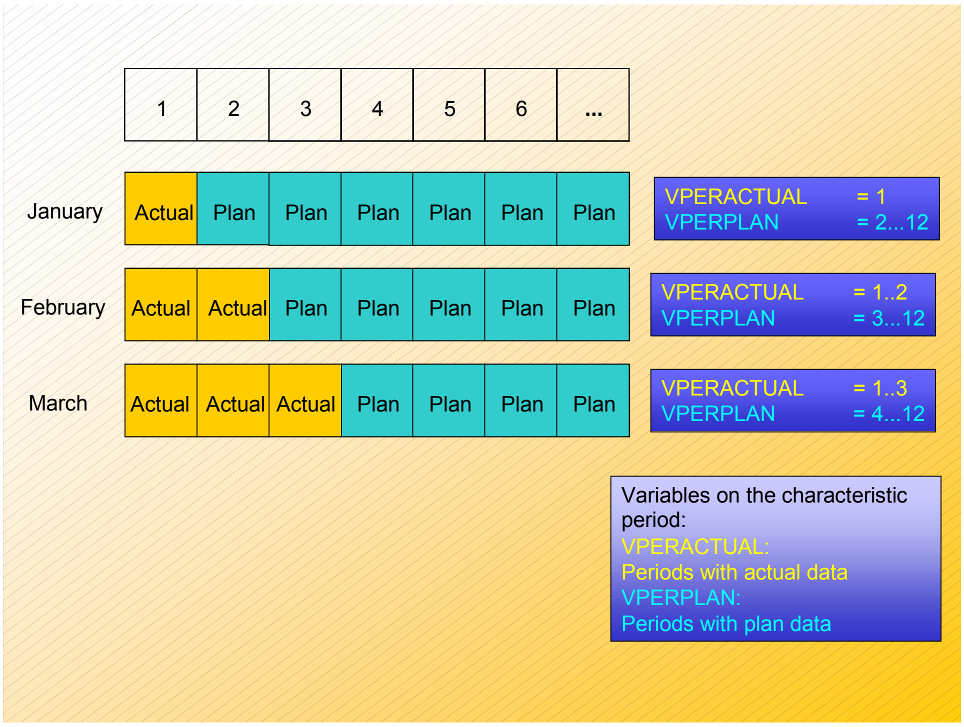
A typical example of such a scenario is the planning of sales figures on a monthly basis for the current fiscal year.

In order to model such a scenario, you require a planning layout for manual planning in which the consecutive listed periods are divided into two areas:

- Completed periods which contain actual data and are protected against changes
- Open periods which contain plan data and can be changed

In addition, the requirement exists that the boundary between both of these areas should be able to be changed with the least possible administrative effort.

The following graphic presents the procedure aimed for:



	1	2	3	4	5	6	...	
January	Actual	Plan	Plan	Plan	Plan	Plan	Plan	VPERACTUAL = 1 VPERPLAN = 2...12
February	Actual	Actual	Plan	Plan	Plan	Plan	Plan	VPERACTUAL = 1..2 VPERPLAN = 3...12
March	Actual	Actual	Actual	Plan	Plan	Plan	Plan	VPERACTUAL = 1..3 VPERPLAN = 4...12

Variables on the characteristic period:
VPERACTUAL:
Periods with actual data
VPERPLAN:
Periods with plan data

The whole planning period is moved.

The general approach to fulfill this requirement is based on the use of variables in connection with dynamic data columns. You create the variables VPERACTUAL and VPERPLAN for the characteristic period and set the selection of variables so that the characteristic values cover the area from 1 to 2 over both variables and that the initial value of VPERPLAN is the same as the end value VPERACTUAL plus 1.

You use the variable VPERPLAN for the selection of the characteristic *period* in the planning level. In addition, you must restrict the characteristic *fiscal year* variant compounded to the period to a single value, since *period* is otherwise not available as a dynamic characteristic.

For the planning layout, choose the type *Key figures in data columns* and enter that the characteristic *period* should be presented in dynamic data columns. In addition, you include the characteristic *version* in the data column in order to distinguish between plan and actual data.

You take the further procedure from the following section.

Procedure

1. In the second step of the layout definition, choose the tab page *Data columns*.
2. In the table *Configuration of data columns*, define the data column for the presentation of the actual from completed periods. For this you make the following settings:

   - ...

   a. Choose the key figure for which you want to display or enter transaction data.

   b. Set the indicator in the column *Comparison column* to protect the actual data displayed here from changes.

   c. Set the indicator in the column *Dyn.* to indicate the data column as dynamic.

   d. Select the field in the column for the characteristic *period*. Choose *Variable*. The system displays a dialog box.

    e. Choose the option *Variable* and enter the *Name* VPERACTUAL.

    f. Enter in the column for the characteristic *version* the characteristic value for actual data.

3. In the table *Configuration of data columns*, define the data column for the presentation of the plan data for the periods that are still open. Make the same settings for this as for the actual data, however, with the following differences:

   ▪ ...

   a. Make sure that the indicator *Comparison column* is **not** set.

   b. Because the indicator *Dyn*. is set, the system has entered the technical name of the characteristic in the column for the characteristic *period*. Do not change this setting.

      This setting ensures that those values are used for the periods which are set in the selection of the planning level. In the scenario presented here, these values result from the definition of the variable VPERPLAN which is used in the planning level as a selection.

   c. Enter the characteristic value for plan data in the column for the characteristic *version*.

4. Save the planning layout and execute it.

Result

The executed planning layout contains as many data columns as there are characteristic values in the selections of both variables VPERACTUAL and VPERPLAN, therefore, in the example presented 12. After a change of month, you as the administrator only have to extend or reduce the selections of both variables by one period in order to adjust the division of the period range in a display area for actual data and an entry area for plan data.

4.2.1.2.4.1.2 Determining Detail Settings

Use

In the planning layout detail settings you are able to determine the sequence of characteristics in the header area, and configure the lead and data columns. Furthermore, you can select the display tool for previewing the planning layout in the next processing step.

Prerequisites

You have determined the basic settings for the planning layout (see Determining Basic Settings [Page 134]).

Procedure

The system dynamically offers you the tab pages on which you probably want to make changes. According to the layout type that you have chosen, enter the key figures that you are displaying in the planning layout and for which you want to enter plan data on either the *Data Columns* or *Lead Columns* tab page. In detail, you have the following options:

Tab page *Header Area*

Here you can change the order in which the system suggests characteristics in the header area are to be listed. In addition, you can decide whether each characteristic in the header area should be hidden or displayed in the output in the front end. In this way you can increase the visible area for data entry if the layout has a large number of header characteristics.

Tab page *Data Columns*

The structure of this table depends on the layout variants and the key figures and characteristics that you have chosen in the previous step when editing the planning layout:

- For the layout variants *key figures in data column* and *key figures in data column, rows defined individually* you get one column for selecting the *key figures*.

  If characteristics have also been selected for the data columns, an extra column appears for each characteristic. This has the name of the relevant characteristic as the title of the column.

- For the layout variant *key figures in lead column* you get, for each characteristic, a column with the name of the respective characteristic as the title of the column.

In the fields for key figures, select the required key figures from those key figures that were included in the planning level. In the fields for characteristic values, determine these values either with fixed values or by variable replacement.

In the *Heading* column you are able to edit the heading of the data column. If no text has been determined for a column heading yet the system uses the name of the key figure or characteristic that you select.

If there are characteristics among those chosen for the data column for which an attribute for sign conversion has been determined, you get another column: sign attribute (see Automatic Sign Assignment [Page 172]).

For more information, see Configuring Data Columns [Page 153].

You can choose whether the layout should be displayed with the SAP List Viewer (ALV) or with Microsoft Excel using the *Display Tool* function. In the third processing step different formatting options are available to you depending on the display tool (see Formatting the Generated Layout in the Preview [Page 155]).

Tab page *Lead Columns*

The structure of this table depends on the layout variants that you have chosen in the previous step when editing the planning layout: You have to distinguish between the following cases (for the arrangement, see Planning Layout [Page 128]):

Simple lead column(s): Here you define the sequence of characteristics in the lead columns for the *key figures in lead column* layout variant. You also determine whether the characteristics are to be displayed with their technical name (key), their description, or a combination of both.

If attributes are defined for a characteristic you can choose, in the *Attr.* column, whether and for which attributes of the characteristics additional columns are to be created in the lead columns. The system displays the relevant attribute values in these additional columns.

Complex lead column(s): The structure of this table depends on the layout variants and the key figures and characteristics that you have chosen in the previous step when editing the planning layout: Determine the characteristic values and, where appropriate, the key figures. This process is described in the section on the *Data Columns* tab page.

Result

You have made the basic and detail settings for the planning layout. To see the result in a preview with sample data, choose *Continue*. For more information, see Formatting Generated Layout in Preview [Page 155] .

4.2.1.2.4.1.2.1 Configuring Data Columns

Use

You determine the properties of data columns on the *Data Columns* tab page in the *Configuration of Data Columns* table. Here you determine which key figure or characteristic values are contained in the respective data columns and how these values are to be displayed (for example, with regard to ready-for-input status, display of units, display decimals, scaling factors).

> Note that one **row** in the *Configuration of Data Columns* table represents one **column** of the data column area on the planning template.
>
> You can find more information on structuring the table under Determining Detail Settings [Page 151].

Prerequisites

You have fully executed the basic steps for creating and editing the planning layout (see Determining Basic Settings [Page 134]).

Procedure

1. In the *Key Figure* column, enter the technical name of the key figure that you want to include in the layout. An input help is available.
2. In the column *Heading*, enter the text that is to be used as the column heading for the key figure.

> It makes sense to change the *Heading*, for example, if you have broken down the data columns by characteristic value and the same key figure appears several times in the layout, for example, *sales* broken down by region.

> If you want to use a variable within a text, enter the technical name of the variable in capital letters. In order to provide a variable with an offset, use a character string of form `VARIABLE+1` or `VARIABLE-1` (without blank characters) in the text. You are also able to combine “static” and “variable” text in one heading, for example, “sales in year VARIABLE+1".
> At the time of execution (that is, when displaying or entering the data), the variables are replaced with the corresponding value.

⚠

If you replace a key figure (that you have already added to the data columns) with another key figure, the system only enters the name of the new key figure in the field *Heading* if the field was previously blank. Otherwise the column heading retains the original key figure.

3. In the column *Position,* enter a figure that specifies in which position the current column should appear among the data columns. If you do not enter a position the system automatically numbers the columns in the order of the rows in the table.

   💡

   In certain cases you can simplify the entry: For example, to swap the positions of the last two columns, you can simply enter a value for the second last column that is greater than the value for the last column. The system then rearranges the columns and ensures correct sequential numbering.

4. Determine whether the key figure that you are processing is to appear as a *comparison column*. Comparison columns are **not** ready for input in manual planning. Furthermore, a column must also be specified as a comparison column if you want to allow a fixed characteristic value in the planning layout for a characteristic whose value is determined by a variable in the planning level.

   💡

   Key figures that you include in a comparison column are **not** subject to the characteristic value restrictions in the planning level or the planning package. Therefore, you can display key figures for any combination of characteristic values for purposes of comparison. You can use only those key figures that were included in the planning level. Comparison columns are also locked.

5. If the unit or currency should be suppressed when showing the values of a key figure in the planning layout, set the necessary indicator in the *Suppress Units* column. The currency unit or unit of measure is often already included in the header area. In such cases, it is not necessary in the data area.

6. Where necessary, determine a scaling factor that is a rounding factor, for displaying the key figure values. This is particularly useful if you want to display very large amounts. If you enter a new row in the table but do not enter anything in this field, the default settings of the BW system are used.

7. An input help is available in the *Decimals* column for determining the number of decimal places. The representation of decimal places in the planning layout depends on the following settings:

   - If you determine decimal places in the properties of the planning layout this overwrites all other settings. The settings in the layout **cannot** be used to increase the accuracy of those values saved in the database.

   - If you do not determine anything in the planning layout the system adopts those settings that are maintained for the key figure in InfoObject maintenance.

   - If you have not determined anything for the key figure either, those settings that are maintained for the assigned unit of measure or currency unit become effective. For key figures of type figure, the technical decimals become effective.

8. If you have included characteristics in the data columns, enter the desired characteristic value for each column.

   If in the planning level you have set the selection condition for a characteristic by variable replacement, you can transfer the value calculated by the variable at the execution time, instead of a fixed entered value, using the *Switch Variable On/Off* function. The function is only available if a corresponding characteristic is used in the planning layout. For more information, see Using Variables [Page 108].

9. If you want to create an attribute column for signs you also have to select the corresponding attribute key figure in the field for the *sign attribute*. You can find more information on the sign attribute under Automatic Sign Assignment [Page 172].

   Attribute columns are not ready for input and are displayed without units. For this reason the system sets the indicators for the comparison column and the suppression of units automatically.

Result

You can view the results of the settings you have made as a spreadsheet and then make further changes if you wish by choosing *Continue* to switch to the last processing step of the planning layout.

The settings that you make for the presentation of values in the layout also have an effect on how these values are transferred using the Application Programming Interface [Page 351].

If, for example, you use a particular scaling factor the value is delivered exactly as it appears in the planning layout (and not with the actual value) when a value is accessed using the programming interface. For an application that requests values using the programming interface it is not possible to determine whether a value has been modified.

See also:

Formatting Generated Layout in the Preview [Page 155]

Hierarchically Structured Planning [Page 135]

4.2.1.2.4.1.3 Formatting Generated Layout in the Preview

Use

In the last three processing steps you see the finished planning layout as a preview with your own sample values. Depending on the display tool that you chose in the second editing step you can modify the layout to fit your individual formatting requirements.

Integration

BW-BPS generates the planning layout anew whenever you make changes in the previous layout-processing steps, or in the data basis, that affect the layout (for example, when selecting the characteristics to be used in the planning level). For this reason, it is a good idea to make the final configurations to the layout when changes like these are no longer necessary.

Features

The automatically generated layout contains the header area and the lead and data columns. The system also generates some random data for the lead and data columns so as to give you a better idea of how the planning layout will look at the time of data entry or data display.

You can find further information under:

- MS Excel-Specific Formatting Options [Page 157]
- Editing Styles [Page 161]
- Editing Layout Template [Page 163]
- Setting Totals Rows [Page 165]
- Documents in the Planning Layout [Page 167]

The following function keys are available depending on the display tool that you selected:

Function keys for SAP ALV grid

SAP ALV Grid	Further Information
Information	You obtain statistical information on the last change made to the layout.
Remove Fixing	In the ALV layout the lead columns are fixed by default. This means that you can only scroll through the layout as of the first data column. The lead column is always visible when you scroll down. However, if the number of lead columns exceeds the number of columns that can be displayed on the screen and the first data column falls outside of the visible area, **no** scroll bar appears. With layouts of this type you need to choose the *Remove Fixing of Lead Columns* option. The scroll bar then appears above the lead columns.
Display Documents	See Documents in the Planning Layout [Page 167]
Lead Column Settings	Setting Totals Rows [Page 165]

Function keys for Microsoft Excel

Microsoft Excel	Further Information
Information	You obtain statistical information on the last change made to the layout.
Excel Settings	See MS Excel-Specific Formatting Options [Page 157] (section “Influencing system behavior using macros“).
Column Width	See MS Excel-Specific Formatting Options [Page 157] (section “Controlling column width“).
Restore Standard Positions	See MS Excel-Specific Formatting Options [Page 157] (section “Moving and newly creating cells“).
Display Documents	See Documents in the Planning Layout [Page 167]
Lead Column Settings	Setting Totals Rows [Page 165]

4.2.1.2.4.1.3.1 MS Excel-Specific Formatting Options

Use

All of the formatting options described here refer exclusively to planning layouts for which you have chosen Microsoft Excel as the display tool. These options are **not** available for the display with the SAP List Viewer.

You can use the following formatting options:

- Formatting Cells
- Controlling Column Width
- Moving Cells and Creating New Ones
- Creating New Sheets
- Influencing System Behavior with Macros

Prerequisites

Microsoft Excel display tool

In the second editing step you selected Microsoft Excel as the display tool.

MS Excel certificate

The macros in the central layout template for the SAP MS Excel master template are certified. To be able to use the certificate you have to accept it. You only have to do this once. When you open an MS Excel document (in the Layout Builder, in the third processing step, for example, or when you start manual planning) choose the option *Always trust macros from this source* and then *Enable Macros* in the *Security Warning* dialog box. You will not be asked again whether macros should be executed.

> ⚠
>
> If you are using the certificated templates with MS Excel97 problems occur because of a Microsoft error. The functionality of the certificate is not recognized in this version. This causes a visual basic error to occur when you execute a layout in manual planning. You can find more information on resolving this problem in SAP Note 611090.

> 💡
>
> If you are using your own template instead of the central layout template delivered by SAP, the certificate for the SAP MS Excel master template is still valid as long as no changes are made to the macros. All newly created layouts are also certified as copies of the certified template.
>
> The SAP certificate is no longer valid if you change or insert macros in your own planning layouts or in the template. In this case SAP recommends that you purchase the license for an appropriate certification software and certify the Excel files for your enterprise. You can find more information on certificates for MS Excel documents on the Internet on the Microsoft homepage.

Functions

Formatting cells

For every planning layout that you create using Microsoft Excel, BW-BPS administers a list of **styles**. Every style represents a complete record of predefined formatting settings (for example font settings, colors, alignment, frames) for a certain cell range. Using this function you can use all the formatting possibilities in the planning layouts which Microsoft Excel also offers in stand alone operation.

> The possibilities offered by Microsoft Excel for the presentation of decimal places and units are overwritten in planning layouts by the appropriate settings. For more information, see Configuring Data Columns [Page 153].

The styles are contained in the SAP MS Excel master template **layout template** which is stored centrally. This serves as a copy template when you are creating a new planning layout. It also contains two predefined macros that you can use for special purposes (see below in the section "Influencing System Behavior with Macros").

You can decide whether one of your desired changes is to apply to one concrete planning layout or whether the change is to be adopted system-wide in all planning layouts that are created in the future (such as inserting the company logo in the layout). For more information, see Editing Styles [Page 161] and Editing Layout Templates [Page 163].

The system generates a style for the following layout areas:

- Header area
- Column title for lead column
- Column title for data columns
- Values of lead column (not ready for input)
- Values of data columns (not ready for input)
- Ready for input cells
- Two different subtotal rows/hierarchy levels
- Totals rows/upper most hierarchy level

In addition, you can define further styles and assign certain cell ranges, for example in order to format every data column individually. If subtotals are to be displayed the subtotal format is prioritized in the columns. Subsequently, formatting is automatically continued with the style for the columns.

Controlling column width

Using the *Column Width* function you can determine if the width of the lead or data columns is to be automatically fitted to accommodate the largest value or the text length of the column heading. If this accommodation is active (default) column values and headings are always visible in their full length. As this can lead to columns becoming extremely wide you are also able to deactivate this function.

> SAP recommends that you deactivate the automatic adjustment of column width if a column with a long heading is not followed by other columns. In this case the

text is automatically distributed over the available space; the column width itself remains unchanged.

Exactly how the text is distributed and displayed is determined by the underlying MS Excel functionality. The BW-BPS function described here serves simply to activate or deactivate this functionality in general.

This setting always affects all the columns in a layout (independent of whether certain columns are selected or not). You **cannot** determine the width of individual columns.

Moving cells and creating new ones

You can move or regroup the automatically generated cell ranges. You can also create cells with your own content (for example with formulas that refer to the plan data and perform further calculations). Note the following points:

- BW-BPS manages the automatically generated cells in several related, named cell ranges and these may only be moved in one piece. If you happen to break up such a cell range by moving individual cells, BW-BPS will rearrange the range if necessary and undo the changes you have made. The following elements of the layout are managed as related cell ranges:
    - Each row in the header area
    - Each column heading
    - Each column including the cell that belongs to the totals row If both the technical name and a name or attribute are also given for the lead column these appear in abutting columns and are treated as one cell range.

      When moving cell ranges, you must only use the Windows standard functions for cutting, copying, and pasting of selected ranges (Ctrl+X and Ctrl+V or Shift-Del and Shift-Ins). You can also use the mouse to move selected cell ranges to different locations (Drag&Drop). If you use different procedures there is a danger that the relationship between the column heading and the columns that belong with it will be distorted or lost. The same applies to a named range and the range where the data is actually stored. In such a case, the planning layout cannot be used and it must either be restored or newly constructed.

      If you have made changes in the layout by mistake and these could damage the relationship between the cell ranges, you can return the layout to the one which BW-BPS had generated as a default setting using the *Restore Standard Positions* function.

- When you open the planning layout for data entry, BW-BPS automatically generates 20 new blank rows under the last row where data was entered. If you have placed anything in these rows the content is overwritten by the blank lines without any warning.

  If you have entered content below the data entry range, you can protect it from accidental deletion by inserting a sufficient number of blank rows between the data entry area and your content and then hiding these blank rows. This way, your content appears close to the data entry area but cannot be overwritten by automatically generated data entry rows.

When BW-BPS opens the planning layout for data entry, the plan data is inserted in the positions that you defined when creating the layout. When saving the data is read from the same positions without further checks and is written to the data basis. Therefore avoid executing actions in data entry that change the positions of the cell ranges in the layout (inserting or deleting rows or columns or sorting columns).

Creating new sheets

For each planning layout you create, BW-BPS generates a Microsoft Excel workbook with exactly one sheet. This sheet is filled with the various layout areas according to your settings. This **standard worksheet** is called **`SEM-BPS 1`**. If you need additional complex calculations, want to create detailed comments or want to save other data related to the plan data, you can create new worksheets in the workbook and use them as you wish. Note the following:

- The data transfer between the data basis and the planning layout always refers to the standard worksheet **`SEM-BPS 1`**. Changes that you make in this worksheet are written in the data basis when saving, but are also overwritten when reading data again from the data basis. On the other hand, all data in further worksheets created by you is **not** taken into consideration when saving, however, it remains unchanged when reading again from the data basis.

- In order to save data that you have calculated or changed in additional worksheets in the data basis, it is therefore necessary that this data is transferred to the standard worksheet **`SEM-BPS 1`** immediately before saving (and after reading from the data basis from there to the worksheet that you have created). You can achieve this by using macros to ensure that whenever data is exchanged between the system and the worksheet, the available worksheets contain the data required according to the named request.

Influencing system behavior with macros

You have the facility to influence the system behavior in the context of Microsoft Excel whenever data is exchanged between the workbook and the system. You do this by running the macros you have created for this purpose. Using the *Excel Settings* function you can determine if further operations are to be carried out when data is exchanged. You can plan operations for the following times:

- The time just after BW-BPS has inserted data from the data basis into the layout (predefined macro name: **`Modul1.SAPAfterDataPut`**)

- The time just before BW-BPS transfers data from the layout into the data basis (predefined macro name: **`Modul1.SAPBeforeDataGet`**)

Every Microsoft Excel template for a planning layout already contains the macros **`SAPBeforeDataPut`** and **`SAPBeforeDataGet`** in module **`Modul1`**. With the *Excel Settings* function you can determine which of the two macros is to be called by the system. Implement your macros, preferably in a module of your own, and insert just one call up of your macros in the specified places in **`SAPBeforeDataPut`** or **`SAPBeforeDataGet`**.

The macros that you can activate are started only at the specified times and then only in conjunction with the functions *Display Plan Data* or *Enter Plan Data* (however, not with *Display Layout* or *Change Layout*). Test your macros in the VBA development environment.

Example

Here is an application of the "Influencing system behavior with macros" facility described above:

You want to carry out special formatting so as to adhere to your enterprise guidelines for designing forms. One specification is that blank spaces are to be inserted in certain places in order to improve clarity. This cannot be achieved with an automatically generated layout. You also want to use formulas in a spreadsheet to further process the plan data presented in the layout.

Therefore, you decide to move the presentation of the plan data to a new worksheet that you have inserted. You need a solution for the requests described above in the section "Creating new sheets". Proceed as follows:

1. Ensure that the layout generated by BW-BPS contains all the data that you require for further processing.
2. Insert a new worksheet.
3. Activate the macros for both times using *Excel Settings*.
4. Choose *Tools* → *Macro* → *Visual Basic Editor*. Open the predefined VBA module **Modul1**.
5. Insert your code in the **SAPAfterDataPut** macro where indicated. When this macro is executed your code ensures that the plan data taken from the data basis is transferred from the automatically generated layout **SEM-BPS 1** into the worksheet you have created. It positions the data according to your requirements, formats it and uses predefined (or newly created) formulas to calculate it.

   If the macro become too complex due to your code, you can share the required functionality with other macros that you create and then call them up from the predefined macro. Note that your own module cannot begin with **SAP_**.
6. Enter **SAPBeforeDataGet** statements in the predefined macro that copies the plan data from the worksheet you created back into the **SEM-BPS 1** worksheet with the automatically generated layout. This allows them to be transferred into the database again from there.

4.2.1.2.4.1.3.2 Editing Styles

Use

You edit styles of a planning layout, or create additional styles, in order to achieve an individual structure that is different from the central style for a particular layout.

You include an additional column in a planning layout in which further calculations of the plan data presented in the data column are executed with the help of Microsoft Excel formulas. You want it to be identifiable in the layout that the values have a particular status in this additional column and are not saved in the data basis. To do this, you define an additional style in which you define a different background color for the additional column.

Note that when editing styles, you are only allowed to change the format-specific properties of a template, but not the names of a delivered template. BW-BPS is dependent on the delivered styles being available with the names determined by SAP for the correct processing of planning layouts.

The possibilities offered by Microsoft Excel for the presentation of decimal places and units are overwritten in planning layouts by the corresponding settings in BW-BPS. For more information see Configuring Data Columns [Page 153].

Procedure

In order to use styles for cell formatting, proceed as follows:

1. Choose *Change Layout* in the context menu of a planning layout.
2. Choose *Continue* until you reach the third step of the layout definition. The worksheet is displayed with the design of the layout
3. In the layout, select a cell area whose formatting you want to change
4. Choose *Format* → *Style*. A dialog box appears in which the valid styles are displayed for the selected cell range with the current settings.
5. Choose *Change* to edit the style settings. A dialog box appears in which you have access to all formatting settings.
6. After you have made the changes, choose *OK*. You return to the first dialog box.
7. You can determine whether you want to create the changed style under a new name as an additional format to the ones which already exist, or whether you want to overwrite the underlying format with the changes. What you decide upon here will affect the layout in different ways:
   - Creating style under a new name: The new style is assigned to the cell area that was selected in the third step. All remaining areas of the layout remain unchanged.
   - Overwriting the underlying format with the changes: All cells of the layout which are assigned to the style adopt the changes you made.

   Because of an error in Microsoft Excel 2000, you should avoid creating styles whose names contain other characters than letters or figures. In particular, the underscore "_" leads to problems in style names when saving the template.
8. Choose *Ok* again. The system adopts the changes.

Result

The changes are visible in the layout design and have an effect when calling up the layout to display and enter data.

See also:

Formatting Generated Layout in the Preview [Page 155]

4.2.1.2.4.1.3.3 Editing the Layout Template

Use

To ensure that changes you have made are effective in all planning layouts created after the layout template is changed, you can make changes in this central layout template, the SAP MS Excel Master Template. The layout template is used as a standardized template for all planning layouts in BW-BPS manual planning.

> You can use this function, for example, to achieve a unified corporate design by including a graphic with the company logo in the header area of the layout.

Prerequisites

SAP recommends modifying the central layout template to the desired form at an early stage so that no deviating layouts occur. Changes to the layout template always only have an effect on planning layouts which are created later. The planning layouts which already exist remain unchanged, and if required, must be manually adjusted to the changed template.

> The central layout template must always contain one worksheet with the predefined name **`SEM-BPS 1`**. In addition, this worksheet cannot be hidden.
>
> Note that when editing styles, you are only allowed to change the format-specific properties of a template, but not the names of a delivered template. BW-BPS is dependent on the delivered styles being available with the names determined by SAP for the correct processing of planning layouts.

> If you do not make any changes to the macros, the certificate for the SAP MS Excel Master Template is still valid. If you insert or change macros this invalidates the certificate. You can find more information under MS Excel-Specific Formatting Options [Page 157] in the “Prerequisites“ section: MS Excel certificate.

Procedure

The system is delivered with an SAP MS Excel master template that is used as a copy template for each new planning layout that uses Microsoft Excel.

In order to deal with central templates for newly created planning layouts using your own master template it is advisable to copy the SAP MS Excel master template to a local computer using report **UPP_GET_MASTER_COPY**. You can edit the template there and then transport it back to the system using report **UPP_MASTER_CHECKIN**. Proceed as follows:

1. From the *SAP Easy Access* screen, choose *Tools* → *ABAP Workbench* → *Development* → *ABAP Editor*.
2. In the *Program* field, enter the name **UPP_GET_MASTER_COPY** and choose *Execute*. This report makes it possible to store a copy of the SAP MS Excel master template locally on your PC.
3. Choose the directory path in the *Save SAP Template* dialog box and enter a name for the copy.
4. Save your entries.
5. Make the required changes in the copied version of the SAP MS Excel master template:
   - Make the required changes to the styles without renaming them. For more information, see Editing Styles [Page 161].
   - Make all further changes (for example inserting graphics, logos, background colors).
6. From the *SAP Easy Access* screen, choose *Tools* → *ABAP Workbench* → *Development* → *ABAP Editor*.
7. In the *Program* field enter the name **UPP_MASTER_CHECKIN** and choose *Execute*. You can load the changed file back into the system as a custom master template with this report
8. In the selection screen, enter the complete qualified name of the Microsoft Excel file you have edited.
9. Check whether the new master is also needed in other systems. If you want to make sure that the changed layout template is also transported to down-stream systems, choose *Transport*.
10. Choose *Execute*.

Alternative procedure

1. Create a new planning layout in any planning level. You only assign characteristics and key figures here pro forma.
2. Switch to the third step of layout definition where the layout design is displayed with Microsoft Excel.
3. Save the worksheet under a new name as a Microsoft Excel file in a local directory.
4. End the planning session without saving the newly created planning layout.
5. Start Microsoft Excel and open the file you saved locally before.
6. Remove all elements from the file that point to the characteristics and key figures that have been assigned pro forma when creating.
7. Make the desired changes to the styles.
8. Make all further changes (for example inserting graphics, logos, background colors).
9. Save the file.
10. To load the Microsoft Excel file that you have edited back into the system as a custom master template, use report **UPP_MASTER_CHECKIN** again. To do this, perform steps 6 – 10 of the procedure described above.

Result

If a custom master template exists, this is used as the template for new planning layouts. The SAP MS Excel master template itself is not overwritten.

As report **`UPP_GET_MASTER_COPY`** saves both the SAP MS Excel master template and, if available, the custom master template locally, it can also be used if you want to find errors in the master templates stored in the system.

You can remove your custom master template from the system again using report **`UPP_DELETE_CUSTOM_MASTER`**. If you have executed this report, the SAP MS Excel master template is used again when you create a new planning layout.

See also:

Formatting Generated Layout in Preview [Page 155]

4.2.1.2.4.1.3.4 Setting Totals Rows

Use

You can use the *Lead Column Settings* function to summarize the plan data values displayed in the data columns in totals or subtotals. You can use this function both for setting up the layout (*Change Layout*) in the sense of default values and when displaying the plan data (*Display Plan Data*).

In addition, you can determine whether the characteristic values of the lead column which are contained in a hierarchy you have selected should be presented hierarchically (*Totals as Hierarchy* option) or in the form of a “flat” list.

Prerequisites

Totals rows can only be displayed if you have selected the option *Key Figures in the Data Columns* in the basic settings of the layout (see Determining Basic Settings [Page 134]).

However, if key figures are displayed in the lead column no totals rows can be generated. In this case the function is not available.

Features

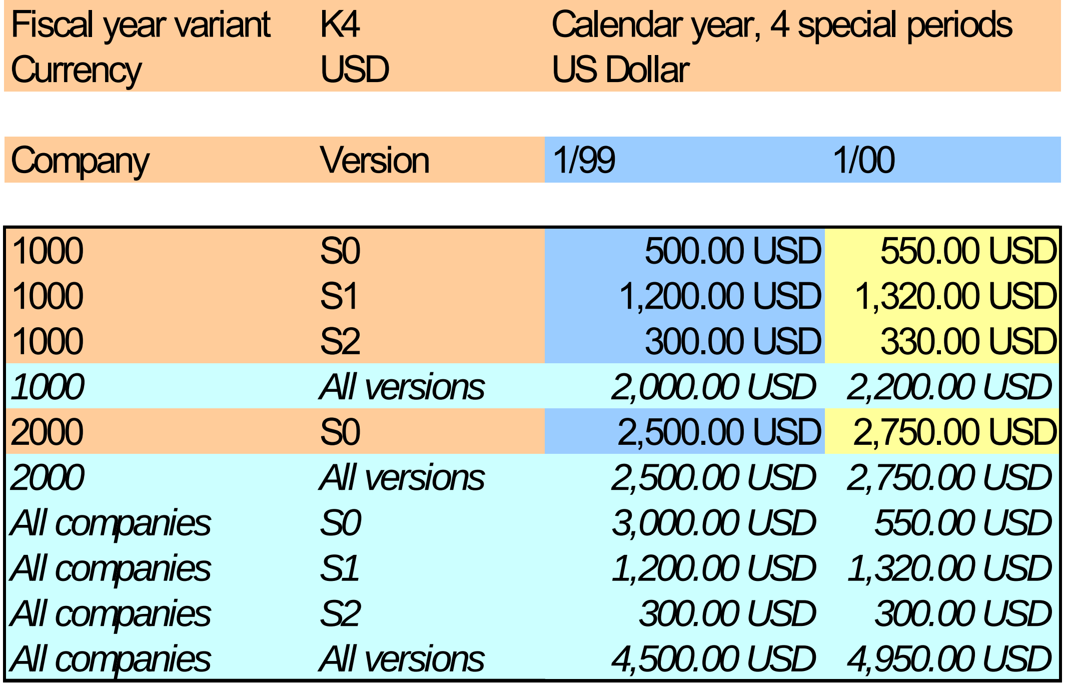
For every characteristic that is contained in the lead column you can define whether a subtotal should be generated or not. In addition you can determine whether as the last row a total – aggregated over all values of all characteristics – should be disclosed.

If only one single characteristic is in the lead column, the additional total is not offered, as the sum of a single characteristic is already identical to the total.

Examples

The following graphics show various possibilities of using totals rows in the planning layout. For better distinction, the totals rows have a light blue background and are italicized. As an example we have a planning where a key figure from the first quarter of 2003 is taken on with a 10% markup in each case in the first quarter of 2004. In the key block, it is broken down according to the characteristics *company* and *version*.

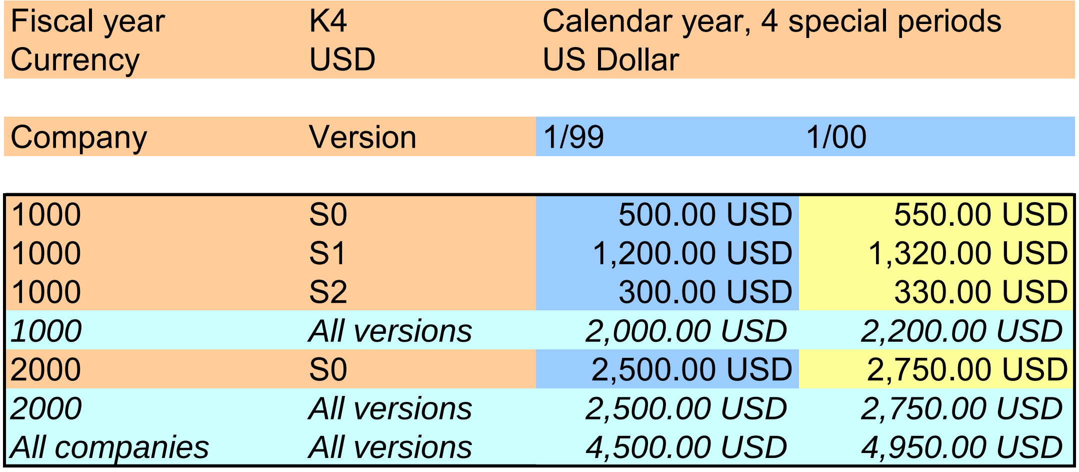
The following graphic shows the use of all totals rows that are available for the data to be displayed: Sum for company 1000 (or 2000) over all versions, sum for version S0 or S1 or S2 over all companies and finally the total over all companies and versions:



Fiscal year variant	K4	Calendar year, 4 special periods
Currency	USD	US Dollar

Company	Version	1/99	1/00
1000	S0	500.00 USD	550.00 USD
1000	S1	1,200.00 USD	1,320.00 USD
1000	S2	300.00 USD	330.00 USD
1000	*All versions*	*2,000.00 USD*	*2,200.00 USD*
2000	S0	2,500.00 USD	2,750.00 USD
2000	*All versions*	*2,500.00 USD*	*2,750.00 USD*
All companies	*S0*	*3,000.00 USD*	*550.00 USD*
All companies	*S1*	*1,200.00 USD*	*1,320.00 USD*
All companies	*S2*	*300.00 USD*	*300.00 USD*
All companies	*All versions*	*4,500.00 USD*	*4,950.00 USD*

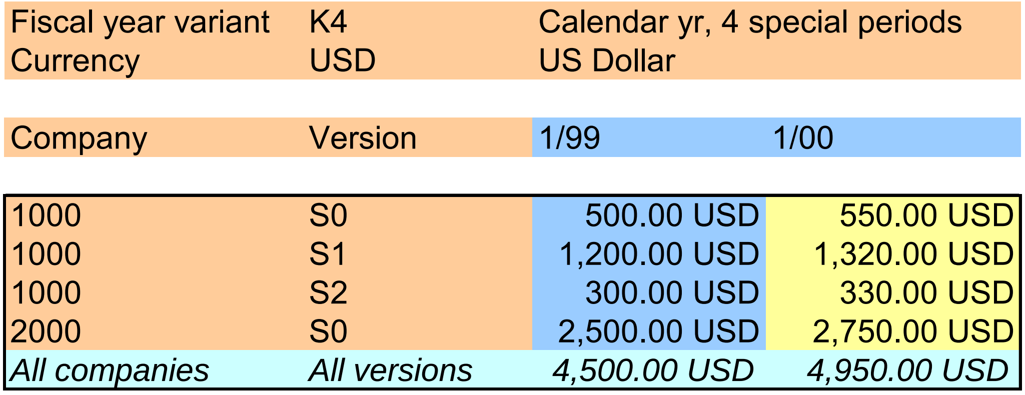
The following graphic shows a combination of subtotals for the characteristic *company* and totals over all companies and versions:



Fiscal year	K4	Calendar year, 4 special periods
Currency	USD	US Dollar

Company	Version	1/99	1/00
1000	S0	500.00 USD	550.00 USD
1000	S1	1,200.00 USD	1,320.00 USD
1000	S2	300.00 USD	330.00 USD
1000	*All versions*	*2,000.00 USD*	*2,200.00 USD*
2000	S0	2,500.00 USD	2,750.00 USD
2000	*All versions*	*2,500.00 USD*	*2,750.00 USD*
All companies	*All versions*	*4,500.00 USD*	*4,950.00 USD*

The following graphic shows the exclusive use of a total without subtotals:



Fiscal year variant	K4	Calendar yr, 4 special periods	
Currency	USD	US Dollar	
Company	Version	1/99	1/00
1000	S0	500.00 USD	550.00 USD
1000	S1	1,200.00 USD	1,320.00 USD
1000	S2	300.00 USD	330.00 USD
2000	S0	2,500.00 USD	2,750.00 USD
All companies	*All versions*	*4,500.00 USD*	*4,950.00 USD*

4.2.1.2.4.1.3.5 Documents in the Planning Layout

Use

You can use documents to provide additional information on a particular value.

For example, in a layout that you are using to present the income statement, you could assign a document to the field for the disclosure of the transferred tax on earnings, in which you report about fiscal developments in the previous fiscal year.

You could add a graphic with the sales progress of the last five years to a field that shows the current sales figures for a specific product as a document.

You determine whether the document function should be active at the time of execution in the **Customizing of the planning layout**. In the **executed planning layout** you can open, edit, and delete existing documents.

Integration

- This function uses SAP BW functionality to edit documents on InfoProvider data.
- You can also access documents that you create in the context of a planning layout with the documents [Page 191] planning function.

In contrast to documents which you create there, documents created in the context of a planning layout always have a key figure assigned. The system automatically determines which key figure is concerned when creating the document because of the position of the cell in the planning layout for which you create the document.

Prerequisites

- Documents in planning layouts are always stored for a characteristic/key figure combination. For this reason you can only maintain documents for a particular cell in the layout if the storage of documents is permitted for the characteristics involved. The appropriate option is determined on *Tab Page: General* in InfoObject maintenance. For more information, see Characteristic is Document Property [External].

If not all the characteristics in a planning layout have the comment facility, it can happen that one and the same document is assigned to several cells in the planning layout.

- If the documents that you want to define in the layout were created with a Microsoft Office application, the corresponding application must be installed locally on your PC in order to be able to use the function.

Features

Customizing the planning layout

Using *Display Documents* you can determine that the system is to check whether documents are defined on the values in the data columns when the planning layout is executed. If this is the case, the cells for which documents exist are indicated in the executed planning layout either by a small shape symbol (MS Excel) or a blue background (SAP ALV grid). If the function is not active then there is no indication in the executed planning layout as to whether documents exist for the data displayed. In addition, you control whether new documents can be created in the layout with this function.

In the **executed planning layout** the following functions are available: *Create Documents*, *Open Documents* (and edit), *Delete Documents*. You are also able to create several documents for one cell.

The following document types are supported:

- Microsoft Excel
- Microsoft PowerPoint
- Microsoft Word
- Text document

Documents act like transaction data: They are locked with transaction data. Documents for cells that are not ready for input can only be displayed. They cannot be changed or deleted. Neither can new documents be created for cells of this kind.

Activities

Activating documents for planning layout

1. Choose *Change Layout* in the context menu of a planning layout.
2. Choose *Continue* until you reach the third step of the layout definition. The preview is displayed with the design of the layout
3. Choose *Display Documents*. A dialog box appears in which you can determine whether documents should be read. This option is normally **not** selected so as to optimizing performance.
4. Highlight the option *Read Documents*.

This setting has the effect that the system searches for existing documents when executing the planning layout, assigns them to the cells of the planning layout, and activates the processing function for documents.

Using documents in the planning layout

1. Execute the planning layout.

   If documents exist for the current characteristic selection and the key figures used in the layout, the system adds a small symbol in the upper right cell corner for the cells concerned. In SAP ALV the cell is highlighted in color in this case.

2. To create a new document for a cell, highlight this and choose *Create Document*. A dialog box appears where you enter the title and type of the document.

   Enter the *Title* of the documents and determine its *Type* (Microsoft Excel, Microsoft PowerPoint, Microsoft Word, Text document). The system starts the assigned editor with a new document in accordance with the type you selected.

   > If you have already opened a document with the same application, this document is closed automatically and stored temporarily in a document buffer.

   Edit the new document.

3. To open a document that is assigned to a cell, highlight it and choose *Open Document*. The system starts the appropriate application with the document.

   If more than one document is assigned to the cell, the system displays a dialog box with the list of available documents from which you can choose the desired document.

   > You can access the documents displayed in the layout in Excel by simply clicking on the document symbol in the upper right corner of the cell.

4. To delete a document that is assigned to a cell, highlight it and choose *Delete Document*.

   If more than one document is assigned to the cell, the system displays a dialog box with the list of available documents from which you can choose the document to be deleted.

5. To save changes to documents, choose *Save All*.

4.2.1.2.4.2 Entering Plan Data

Use

You use this function to enter data manually with a previously created planning layout. The plan data that you enter is valid for those characteristic value combinations that you determined by making specific selections in the current planning level or planning package.

> It is only possible to enter plan data within the context of the lock concept implemented in BW-BPS. For more information, see Locking Transaction Data [Page 184].

When you open a planning layout for data entry, BW-BPS automatically generates 20 new blank rows under the last row in the data entry area. You can use these blank rows to enter plan data for new combinations of characteristic values.

> If data records already exist for the characteristic combination that is currently active, the characteristic values are write-protected in the lead column. Therefore, new plan data is always entered under the existing data records. You can enter the new data records in any order. The newly entered data is sorted and moved to the correct place when you save (or if you choose the function 🔄).

Within a non-hierarchical data model you also have access to local SAP functions for *Microsoft Excel*. These can be used as a data entry tool. See also, Local Functions [Page 174].

Notes on Entering Data

- **Avoid** using the **Excel functions** for **deleting**, **inserting** and **sorting**. If you use these functions, it is not possible to read data into the SAP system correctly.

  If you want to **delete** data, select the data and then choose 🗑.

  > The delete function works as follows: The system uses the selected cells to calculate the data records that lie behind these cells. (A cell selected in the lead column makes it look as if the whole row has been selected). The system then sets all key figure values that are ready for input in these data records to 0. An entire row is only deleted if all the key figures in all the data records displayed in the layout that belong to one row have the value 0.

- On a second spreadsheet page you can carry out supplementary calculations (with MS Excel formulas) or insert diagrams. These are only retained temporarily during the planning session. All MS Excel objects that you want to use on a permanent basis have to be defined when you create the layout.

- If you make an entry in MS Excel formula in the SAP data area, this is overwritten as soon as the data is updated in MS Excel, without a message being produced. This generally happens after executing functions that require data to be updated in MS Excel.

- MS Excel must not be in edit mode if you want to save, use a planning function or execute the *Check* function (in other words whenever the system has to read data from MS Excel). MS Excel cannot be accessed "externally" when in edit mode and you will receive an error message.

  > The easiest way of ensuring that cells can no longer be edited in MS Excel is to highlight any cell (but not by double-clicking on it).

Percentages in Manual Planning

To display percentage rates in the planning layout, you must restrict the unit-bearing characteristic to the value "0%". Unlike when you use MS Excel outside BW-BPS with the cell format "percent", **no** internal conversion of values entered is executed with manual planning.

If you enter the value 50 in a cell formatted with the "percent" format of an MS Excel worksheet and confirm, the value "50 %" appears in the cell and the value 0.5 is derived internally for the cell. If you multiply two cells with the value "50 %", Excel displays the value "25 %", and internally derives the value 0.25 for the cell. On the other hand, if you set the "%" unit in a planning layout, the values entered are displayed with an adjusted percent character but are posted with the exact amount that is displayed. In the example cited this means that for a cell which displays the value "50 %", the value 50 is saved in the data basis (and not 0.5). The multiplication of two such cells results in the value 2500.

The following consequences result from dealing with plan data that is entered as a percentage:

- Key figure values entered as percentages must always be divided by 100 in formula function calculations.
- In all planning layouts that display such key figure values or offer them for entry, you have to make sure that the corresponding values are always displayed with the unit, in order to avoid misunderstandings.

Procedure

1. Select an empty cell in the lead column and enter the required value. You can use the input help for characteristic values and key figures that have the type ‘date’.
2. Select an empty cell in the same row of a data column and enter the required plan value.
3. Proceed in the same way for the row's other data columns.
4. Repeat the operation for all additional values in the lead column for which you want to enter plan data.
5. Choose *Sort Newly Entered Data Records*. If necessary, BW-BPS will also set the totals row to the correct position. Sorting the data also includes checking its formal accuracy.
   If you choose the *Check* function, the system checks the data you have entered (for example for valid characteristic values).
6. Use ▼ or ▲ to get to the next or previous value combination for the characteristics in the header area. You can also use to select a specific combination.
7. When you have entered all plan data, choose *Save All* to save the plan data in the database.

   It is not absolutely necessary to explicitly save your entries as BW-BPS saves all changes on a temporary basis. When you end the planning session (at the latest), the system asks you whether you want to save the changes made in the session.

Special Case: Entering Plan Data with *Hierarchical Data Models*

If you want to enter plan data for BPS or BW characteristic hierarchies in a planning layout, system behavior differs slightly to when entering data without hierarchies. In the data entry for a specific hierarchy node on a certain hierarchy level, the system distinguishes the following cases:

- No values have been entered yet for hierarchy nodes on a particular hierarchy level. After checking the new entry, the system generates an additional entry, on the same hierarchy level, for the amounts that are still to be distributed (characteristic value # or a leaf with the characteristic value of the node for a BW characteristic hierarchy with postable inner nodes). The system uses these nodes in budgeting to be able to save difference amounts between a higher-level hierarchy node (for example *product group*) and the total amounts of the lower-level node (for example *product*).
- Values have already been entered for hierarchy nodes on a particular hierarchy level. You enter values for the hierarchy nodes in exactly the same way as in a layout without hierarchies. The system modifies the key figure value for the amount still to be distributed (characteristic value # or, for BW characteristic hierarchies with postable internal nodes, a dummy leaf) when checking or posting it, so that the total of all characteristic values is equal to the amount of the higher-level hierarchy node.

The *Check* function sorts the newly generated rows immediately. It works like the *Check* and *Assort* functions with a non-hierarchical data model.

Result

You can now look at your plan values and compare them with plan values from other areas or periods. You can view your recorded plan data as follows:

- Display in the planning layout
- Display using a query

> You can also execute a planning function directly from manual planning without having to leave the planning layout first. After the function has been executed, the system reads the updated data from the data basis and displays it in the planning layout immediately.

See also:

Automatic Sign Assignment [Page 172]

Local Functions [Page 174]

4.2.1.2.4.2.1 Automatic Sign Assignment

Use

The account-based SAP applications store credit balances (revenue, liabilities and owner's equity) with negative signs in the data basis, and also transfer these values in this form to SAP BW. SAP BW is used as the data basis for BW-BPS. In this way it is guaranteed that debit and credit balances are not implicitly added up. To simplify plan-actual comparisons, and the reference to actual data, it is also useful to keep this convention in BW-BPS.

Automatic sign assignment enables the user to always enter and display plan data without entering a +/- sign (therefore always appearing as a positive amount), however, the system determines the correct sign on the basis of the current characteristic value, and stores it in the data basis. This type of number treatment is frequently used in the presentation of balance sheet or income statement data, which is usually displayed without signs.

Prerequisites

In order to be able to use this function, in the Administrator Workbench in SAP BW, you have to assign a key figure of type number or integer as a display attribute to a characteristic that you have specified. Subsequently you maintain the corresponding value of the attribute key figure for each characteristic value. Only the values **1** and **-1** are possible. In the context of manual planning, the system multiplies the value of a key figure contained in the layout with these values, taking account of the existing characteristic values.

The key figures that you want to use for entering plan data in connection with automatic sign assignment must be of type quantity or amount.

Features

As soon as you have made the preparations described in the *Prerequisites* section, in manual planning [Page 127], you can enter plan data on the corresponding characteristic values without entering the +/- sign. The system uses the value of the attribute key figure to determine the +/- sign for the current characteristic value and takes this into account when saving data in the data basis and when displaying the data later.

> To use this function, which has already been mentioned in the scope of balance sheet and income statement planning, you enhance the account characteristic by an appropriate attribute key figure. This specifies, for each account in the chart of accounts, whether the values posted here have a positive or negative sign.

The system offers the option of displaying the values of an attribute in an individual column. Therefore you can decide whether you want to display the amount only in the layout without signs, or whether you want to display the signs used internally in an additional attribute column.

Activities

In order to be able to use the automatic sign assignment when entering and displaying plan data, proceed as follows:

1. Choose *Change Layout* from the context menu of the planning layout you require.
2. Choose *Continue* to display the second screen of layout definition.
3. Check whether the planning layout contains a characteristic to which an attribute key figure has been defined for sign assignment.

   This is indicated by an additional *Sign Attribute* column on the *Data Columns* tab page.
4. Position the cursor on the *Sign Attribute* field in the row that contains the key figure for which you want to activate sign assignment
5. Transfer the name of the attribute key figure from input help (F4).
6. In order to output the sign itself in a separate column, transfer value **2403** *Sign attr.* from the input help (F4) to the *Attribute column* column.

Result

When data is entered the system determines, for each characteristic value, whether the data entered has to have a positive or negative sign and adds this sign to the key figure value when saving to the data basis.

The system calculates the value of the sign as follows:

- A key figure for the sign is assigned to a key figure in the planning layout. This sign key figure is an attribute of a characteristic that appears in the planning layout. The system determines all the characteristics of the layout that have the sign attribute as an attribute. The product for the sign attribute of this characteristic value is then the sign attribute of the corresponding key figure values.

  In the lead columns, the planning layout contains the characteristics *item* and *subitem* which each have the attribute *SignAttr*. **Key figure value** and **sign attribute** are displayed in the data columns. The *item* 1000 has the value +1 for *SignAttr*, 2000 has the value -1 for *SignAttr*. The *subitem* 100 has the value +1 for *SignAttr*, 200 has the value -1 for *SignAttr*. The following values are then produced for the sign attribute:

Item	**Subitem**	**Sign Attribute**	**Key Figure Value**
1000	100	+	100
1000	200	-	200
2000	100	-	300
2000	200	+	400

- With totals rows [Page 165], subtotals and hierarchy nodes, the key figure values that are being used determine the sign.
  - If all the contributory cells have the same sign, the relevant cells inherit this sign.
  - If they do not, the sign is '+'.
- For planning layouts with budget hierarchies, which also have key figures with signs, the signs should always be the same.

4.2.1.2.4.2.2 Local Functions

Use

When entering plan data, a special help tool is available using the *Microsoft Excel* display tool with integrated SAP-specific local functions. When working in manual planning or when entering data, it is of the utmost importance that the response times from the BW system are as short as possible. By using SAP-specific local functions for *Microsoft Excel*, you are able to run basic functions for manual data entry locally on your client PC. This means you do not have to go via the BW system. These local functions are displayed in *Microsoft Excel* in their own toolbar, under the name *SAP Menu*.

In the following, when local functions in *Microsoft Excel* are referred to, these are SAP-specific functions **only**; that is, local functions that are integrated into *Microsoft Excel* by SAP within BW-BPS.

Integration

Every planning layout in the *Microsoft Excel* display tool is based on a so-called *Excel template* which contains VBA macros delivered by SAP (for example, the standard exits

SAPAfterDataPut and *SAPBeforeDataGet*). The local functions are delivered as of support package 04 in release *SAP BW 3.5*. As of this support package, a new master template for Microsoft Excel exists with the up-to-date SAP VBA macros. If you are using a customer-specific master template, make sure that this corresponds to the following settings:

Using Local Functions in Old Planning Layouts

If you want to use an existing planning layout with the local functions in *SAP-BPS*, make sure that the Excel template you use is up-to-date:

- The *Modul1* module must contain the VBA subs *SAPCreateCode* and *SAPCreateCode2*.
- The *SAP_03_calc* and *SAP_04_prot* modules have to be available. Module *SAP_03_calc* must contain the empty sub *SAPWorksheetChange*.
- In the third screen of layout definition, call the *Visual Basic Editor* (**ALT+F11**) and choose *Tools* → *References*. Under *Available References*, make sure that the *Microsoft Scripting Runtime* selection is activated. Otherwise the local functions will not work properly, for example, you will not be able to enter totals. In old planning layouts the *SAP Table Factory* selection may have been active. This reference is no longer used for executing the local functions described here and should be deactivated.

Buffer for Excel Templates and VBA Macros

Excel templates for planning layouts are buffered locally on your PC in the TEMP directory.

You find this directory, or its equivalent in the various versions of *Microsoft Windows*, using the relevant environment variables. Choose *Start* → *Control Panel* → *System* → *Environment Variable*.

So that bug fixes can be delivered for the VBA implementation of the local functions, the VBA source code is not placed in Excel files but in an ABAP include. If a planning layout uses local functions, the BW system loads the VBA source code into the TEMP directory and loads the macros to *Microsoft Excel* during runtime. The file name for release *SAP BW-BPS 3.5* is *SAP_BW_CODE_CALC_350_01.txt*. Bug fixes contain the corresponding number of the bug fix in the postfix information, for example, *_02.txt*.

In order to load macros to *Microsoft Excel XP/2003* during runtime, access to VBE has to be permitted in the security settings:

i. Open any Excel file.

ii. Choose *Tools* → *Macro* → *Security* → *Trusted Sources.*

iii. Activate the *Visual Basic Project.*

Also see SAP Note 429183 on *Microsoft Excel* security settings.

Prerequisites

The activities and modifications that have to be made so that local functions can be implemented are now described.

Activate Local Functions

The local functions are **only** available in a *non-hierarchical data model* because in a *hierarchical data model*, communication to the server has to be maintained so as to ensure data consistency. However, when local functions are used, the values of the hierarchy nodes can be calculated locally at runtime.

Proceed as follows to perform calculations locally:

1. You are in the first screen in layout definition.

   Choose *Hierarchy*.

2. The *Hierarchy in the Lead Column* dialog box appears.

   Choose *Non-Hierarchical Data Model.*

3. Activate the selection *Calculate Data Locally*.

Predefine Totals in Rows and Columns

You are able to determine settings for totals in the data columns or self-defined rows. Depending on the planning layout, you do this in either the *Data Column* area or the *Lead Column* area in the second screen in layout definition.

1. Indicate a data column or lead row that is to be used as a total. To do this, select the appropriate selection in the left-hand totals column (*Total*).
2. Define a total in the right-hand totals column (sum). The totals for the columns or rows can consist of the following elements:

   - **`C(x)`** or **`R(x)`** indicates the column or the row and the number x
   - **`C(x):C(y)`** or **`R(x):R(y)`** indicates a total using an interval of columns or rows x to y (inclusive)
   - Elements of the type shown above can be combined and used for any total, however, they must not contain circular references, for example, **`C(1)+C(3)+C(7):C(9)`**
   - You are able to use totals columns in other totals.
   - If you are using dynamic columns, these are replaced at runtime by the corresponding totals. The elements of the columns affected by a dynamic column are modified accordingly: If x is a dynamic column, the total **`C(x)`** represents the modified columns.

     Column number 3 is dynamic and is replaced at runtime by two columns. As column 4 now becomes column 5, the total **`C(3)+C(4)`** is replaced at runtime with **`C(3):C(4)+C(5)`**.

     These predefined totals in columns and rows can be used in any planning layout with any display tool (*Microsoft Excel*, *SAP ALV Grid* and *Web Interfaces/Web Interface Builder*). These totals are calculated on the server.

Influence System Behavior with Macros

The local functions are normally incompatible with the predefined VBA macro exits *SAPAfterDataPut* and *SAPBeforeDataGet*. For more information, see MS Excel-Specific Formatting Options [Page 157] → *Functions* → *Influencing System Behavior with Macros*.

SAP strongly recommends that you do not use the predefined macros referred to above with the local functions.

Functions

Planning layouts contain some general characteristics for calculating the values of hierarchy nodes, totals and sub-totals. In addition, you are able to define totals in the static part of the planning layout or in the data columns or the individually defined rows. These predefined totals are then retained in the layout description and are available for calculating other totals.

The Excel functions *Group and Outline* (from the main menu: *Data* → *Group and Outline*) are also used in the static part of the planning layout: If the predefined totals form a hierarchy, *Microsoft Excel* reproduces this hierarchy automatically using *Group and Outline*.

If there are gaps in the columns or rows in the planning layout, this automatic group and outline function may not work.

The following list represents an overview of the local functions:

- Manual data entry [Page 178]. After manual changes to data or the execution of local functions such as distribution or revaluation, data is recalculated locally.
- Locking cells, unlocking cells [Page 179]. Local locks and freezing cells: If you highlight and lock cell areas, these cells cannot be changed manually or by local functions.
- Top-down distribution [Page 179]. Changes to totals are passed down to lower levels by top-down distribution; cells that have been locked are not affected by this distribution.
    - The distribution methods *Analog* and *Evenly* are available.
    - You can determine the order in which values are distributed as columns then rows, or as rows then columns.
    - You are also able to define whole reference columns and rows for the distribution method. You use this function to map the existing distribution schema from one column or row to other columns or rows.
- Revaluate [Page 183]. You can revaluate specifically-selected cell areas. You can do this using either absolute or percentage revaluation factors.

The local functions have to be initialized by the back-end system. For example, *Microsoft Excel* transaction data and information has to be initialized by the existing structure of the planning layout so that local functions can be executed. So that the number of data flows to be transferred for the back-end functions (such as *Check*, *Delete*,… from the system menu) can be restricted, settings for local functions are not included in this transfer. A recourse to the server will cause the local functions to be reinitialized. This produces the following symptoms:

- The area for new rows is not affected when local functions are executed. In general, the back-end system has to align the information on these new rows, for example, if the new rows are part of a hierarchy.

  Use the back-end function *Refresh* to sort the new rows in the data area and reinitialize the local functions for the data area.

- Local locks are lost, for example, after changes are made to the settings for totals and after the functions *Check* (after entering new rows), *Delete*, *Previous/Next/Other Combination*, *Refresh*. The only exceptions to this are the back-end functions *Input Help* (**F4**), *Create/Open/Delete* and *Save Document*.

- Note that in planning folders, the *Save* function can trigger planning functions. If these planning functions change the data on the screen, the local locks are removed. Therefore SAP recommends that you follow this procedure instead:

  a. Add a new row.

  b. Use the *Refresh* function.

  c. Use local locks and other local functions.

- A planning layout can have gaps in the data columns or the self-defined rows. These gaps are not part of the SAP data area in *Microsoft Excel*. Therefore, the local functions do not change the data in these gaps. Usually these Excel formulas for gaps are included for displaying calculated key figures. The values of these key figures are refreshed after changes are made to data in the SAP data area.

- Data slices can prevent you from making entries in some cells or in whole columns and rows. Totals can also be affected if the cells involved are locked.

  Data slices and characteristic relationships (“back-end locks“) are passed on to totals. Intentionally, this does not apply to local locks; as a result, the *Lock Cells* and *Unlock Cells* functions represent a reversible, symmetrical process.

The list of revaluation amounts in the *SAP Menu* toolbar is not changed after a recourse to the server. This is also true for the list of distribution methods.

4.2.1.2.4.2.2.1 Manual Data Entry

Use

After manual data entry, the system recalculates the totals and Excel formulas. You are also able to use *Copy&Paste*, with one restriction: In order to paste an area that contains a total, this area can only contain one totals cell. The reason for this is that the cell values of an area can be inconsistent if this area contains totals and basic key figures. In that case, it is not clear whether the basic key figures determine the totals values or vice versa.

Automatic sign assignment [Page 172] does not work with the local functions because *Microsoft Excel* does not have any information on the settings in the planning layout that are affected. If automatic sign assignment is used, the system simply treats the affected totals as carryforward. *Automatic sign assignment* only works if all signs are consistent. *Microsoft Excel* calculates totals once using the displayed values.

Columns/rows with time-related key figures are ignored by local functions.

4.2.1.2.4.2.2.2 Locking Cells, Unlocking Cells

Use

Local locks (*Lock Cells* function) always apply for the area you have selected to be locked. Cells with “back-end locks“ (see below) or cells from outside the SAP data area are exceptions to this. The *Unlock Cells* function works in the same way.

Within the SAP data area, cells are either totals or basic key figures. In the layout definition, you are able to define data columns or self-defined rows that cannot be changed manually. Data slices or characteristic relationships may also mean that cells are not be ready for input. These locks are normally due to user-independent settings. These “back-end locks” cannot be removed with the local *Unlock Cells* function.

4.2.1.2.4.2.2.3 Top-Down Distribution

Use

The distribution functions allow you to determine the way in which totals are distributed to the line items. The following functions are available:

- *Distribution Methods*

  In the appropriate selection list you can first choose between the following methods:

  - *Analog*

    In this method, values are distributed according to the previous percentages by which the total had been split amongst the line items (addends). This method is set as the default.

    The following table represents a cell area in *Microsoft Excel*. The top cell is the totals cell, the four cells below are the line items. The total is 10. This is distributed to the line items in a ratio of 20%, 50%, 30%, 0%.

Totals Cell	10,00
Cell 1	2,00
Cell 2	5,00
Cell 3	3,00

Cell 4	0,00

If you enter the value 13 as the total, the *Analog* method distributes the new values according to the previous ratios:

Totals Cell	13,00
Cell 1	2,60
Cell 2	6,50
Cell 3	3,90
Cell 4	0,00

- *Evenly*

  In this method, values are distributed evenly to all the line items of the total.

  If the *Evenly* method is applied to the table above, the previous ratios are ignored and the values are now distributed evenly.

Totals Cell	13,00
Cell 1	3,25
Cell 2	3,25
Cell 3	3,25
Cell 4	3,25

It is also important to remember that the *Lock Cells* and *Unlock Cells* functions influence this. If, for example you return to the previous table:

Totals Cell	13,00
Cell 1	2,60
Cell 2	6,50
Cell 3	3,90
Cell 4	0,00

You lock cell 4 using *Lock Cells* and then apply the *Evenly* distribution method. The locked cell, cell 4, is not affected by the redistribution of values:

Totals Cell	13,00
Cell 1	4,33
Cell 2	4,34
Cell 3	4,33
Cell 4, locked	0,00

In the next step, you remove the lock for cell 4 (*Unlock Cells*) and again apply the *Evenly* distribution method. Values are distributed to all cells evenly:

Totals Cell	13,00
Cell 1	3,25

Cell 2	3,25
Cell 3	3,25
Cell 4	3,25

In order to avoid rounding errors, the distribution methods use the number and unit formats of the individual cells. Therefore you should always make sure you use consistent formats.

- *Redistribute*

  Redistribution normally only takes place when a new value is entered in the totals cell. The *Redistribute* function is also useful if:

  d. No new total has been entered, but

  e. The distribution method has been changed.

  This function replaces the need to re-enter the same total just so as to perform redistribution.

- *Reference On/Off*

  Determination of whole reference columns or rows for the distribution method. You use this function to map the existing distribution schema from one column or row to other columns or rows.

  Depending on the structure of the planning layout, you are able to highlight a whole column and/or row as a reference for distribution, that is, as its own distribution method. Proceed as follows:

  a. Select a complete column or row

  b. Choose *Reference On/Off*

  The selected column or row is now displayed in the distribution methods selection list. You are then able to use it as an additional distribution method at any time.

  The following table represents a cell area (two columns) in *Microsoft Excel*. The top cell is the totals cell, the cells underneath it are two sub-totals cells, each with four line items. The total of column 1 is 13, and is distributed to the other cells as shown in the table. The total of column 2 is 0.

Totals Cell (A+B)	13,00 (Column 1)	0.00 (Column 2)
Sub-Total A	**10,00**	**0,00**
A – Cell 1	2,00	0,00
A – Cell 2	5,00	0,00
A – Cell 3	3,00	0,00
A – Cell 4	0,00	0,00
Sub-Total B	**3,00**	**0,00**
B – Cell 1	0,75	0,00
B – Cell 2	0,75	0,00
B – Cell 3	0,75	0,00

B – Cell 4	0,75	0,00

You want to copy the existing distribution schema from column 1 to column 2.

i. Select all of column 1

ii. Choose *Reference On/Off*

Column 1 is included in the distribution methods selection list and is automatically selected as the active method.

iii. Enter the required values in column 2 for the total or sub-totals. In this case, 26 is entered as the total. Values are now distributed in column 2 with reference to column 1. Column 2 now looks like this:

Totals Cell (A+B)	13,00 (Column 1)	26.00 (Column 2)
Sub-Total A	**10,00**	**20,00**
A – Cell 1	2,00	4,00
A – Cell 2	5,00	10,00
A – Cell 3	3,00	6,00
A – Cell 4	0,00	0,00
Sub-Total B	**3,00**	**6,00**
B – Cell 1	0,75	1,50
B – Cell 2	0,75	1,50
B – Cell 3	0,75	1,50
B – Cell 4	0,75	1,50

If you want to delete particular columns or rows as references from the distribution method selection list, proceed as follows:

a. Select the appropriate original column or row in *Microsoft Excel*

b. Choose *Reference On/Off*

When you use back-end functions (for example, planning functions or *Previous/Next/Other Combination*), reference columns and rows may no longer be valid. The system checks whether these references still exist as before in the new screen in the same columns or rows with the same characteristic values. Where necessary, the reference is simply deleted from the distribution methods selection list.

- *Calculation Order*

  Determine the distribution order by columns then rows or by rows then columns.

  A planning layout can contain column totals and row totals at the same time. If you change the values of “cross-totals” of this type, all line items affected are redistributed, or recalculated, according to the top-down principle. As the default, values are first distributed by column and then by row. However, if it makes more sense for values to be calculated for the rows first and then the columns, you can reverse this order in the selection list of the function.

  If the active distribution method is a reference column (see *Reference On/Off*), distribution is always carried out as follows:

a. According to the reference column for the column

b. According to the *Analog* method for the row.

The exact opposite is true if the active distribution method is a reference row.

4.2.1.2.4.2.2.4 Revaluate

Use

You are able to revaluate several existing entries simultaneously; the restrictions are the same as those that apply for *manual data entry*. Use the selection list for the *Revaluate* function:

1. Initially, the selection list is always empty. Enter the required revaluation amount either as a simple decimal number (+/-) or as a percent (+/-) and confirm your entry by pressing *enter*. Your entry is transferred to the selection list. You can make as many entries as required in this way.
2. Select a cell or a cell area.
3. Select the appropriate revaluation amount from the selection list and execute the *Revaluate* function. Each individual value within the highlighted cell area is increased or reduced by the revaluate amount.

The *Revaluate* function does not affect:

- locked cells
- cells that lie outside the SAP data area.

4.2.1.2.4.3 Planning Layout: Environment-Specific Attributes

Planning layouts that you create for manual planning have numerous attributes and functional possibilities that you can use independently of which environment you are working in. However, in addition to that, a series of attributes and functions exist, the usage of which depends on in which way and with which basic technologies the layout is displayed at runtime. The following table gives information about the possible uses of the environment-specific attributes of planning layouts:

Attribute or Function	XL	HTML	XWC	ALV	M1	M2
Hierarchy in lead column (expand and collapse, totals row)	X	X	Display only	X	X	X
Pushbutton “Delete Row”	X	X	X	X	X	X
Navigation within the characteristic values in header area	X	–	X	X	X	X
Dialog field to enter another characteristic value combination	X	–	–	X	X	X
F4 help for characteristic values in new row	X	X	X	X	X	X

Dropdown list box for characteristic values in new row (for uncompounded characteristics or in header area)	–	X	–	–	–	–
Enter new data rows also when only the description of the characteristic value is visible	–	X	–	–	–	–
Customer-specific Excel formulas (client)	X	–	X	–	X	–
Automatically executed planning functions	X	X	X	X	X	X
Fill variables with characteristic value combination of a cell	–	X	X	–	–	–
Access to documents on transaction data (several per cell)	X	Text document only	Text document only	X	X	X
Access to documents on transaction data (one per row)	–	X	X	–	–	–
Display master data attributes in lead column	X	X	X	X	X	X
Sign attribute in data columns	X	X	X	X	X	X

Legend:

XL: Microsoft Excel (in place activation in the SAP Container)

HTML: HTML-based planning layout (for example, generated with the Web Interface Builder [Page 285])

XWC: Microsoft Excel Web Component (as of Microsoft Office 2000)

ALV: SAP List Viewer

M1: One-piece Planning Folders [Page 269] with combined input/output layout

M2: Two-piece planning folders with separate input/output layouts

4.2.1.2.4.4 Locking Transaction Data

Concept

If a planner requests transaction data in change mode, the system locks the requested data exclusively for this user. BW-BPS uses the SAP lock server to do this. However it uses it in a particular way so that multidimensional data can also be locked using selection tables. This lock concept will now be explained in more detail.

Locks are created independently of the existing transaction data. The system locks exactly that data that is covered by the selection in the planning package and, where applicable, the planning layout (see below in the section “examples”).

The data that is to be locked is specified in a **selection table**. The selection table is created from the planning package and, where applicable, from selections from the planning layout. Every characteristic that does not exist in the planning level is therefore locked generically because summarization is performed on this characteristic. The selection table contains selections for the characteristics from the planning level. These are single values or intervals of characteristic values. However the selection cannot be restricted for one characteristic. The following rules apply:

- If variables are contained in the planning level, in the parameter groups of functions or in the planning layout, these are replaced when the selection is determined.

  Replacing variables always leads to a (potentially empty) set of single values or intervals. In Customizing, take into account the fact that replacing variables of type *hierarchy node*, *attribute*, and *authorization* can produce an extremely large number of characteristic values:

  ○ *Hierarchy nodes*: The system calculates all characteristic values from the leaves of the selected subtree for hierarchy basis characteristic und puts these in the selection.

  ○ *Attribute*: The system reads all characteristic values for attribute selection and puts these in the selection.

  ○ *Authorization*: The system determines the authorized characteristic values. These are included in the selection table individually.

- In manual planning, the selection of characteristics in the header area is taken from the planning package. This is also the case for the characteristics in the lead columns, as long as the rows in the lead columns are not defined individually.

  For data columns and individually defined rows in the planning layout (and also for comparison rows or columns) the selection comes from the planning layout.

- Key figures that are used in manual planning or in a planning function are also relevant: The key figures are seen as characteristic values for characteristic '_BKENNZAHL'.

Lock Management

BW-BPS uses the BI Integrated Planning lock management concept.

Depending on the number of users working in planning in parallel and the complexity of lock requests necessitated by the modeling, you can use various techniques to manage the lock table.

- Shared object memory of an application server (default setting): better performance than the lock server.

- Lock server (present implementation in BPS): particularly suited to small and mid-size installations. This option requires the least administrative effort.

- SAP liveCache: suitable for very large installations. An SAP liveCache server has to be installed and connected. If you are considering this option for your project, first contact the development team at SAP.

On the *Planning Lock Settings* screen (transaction RSPLSE) you can run all the lock server settings for planning. On the *Locks* tab page you can monitor the current lock entries.

For more information, see transaction RSPLSE or the Implementation Guide under *SAP NetWeaver* → *Business Intelligence* → *BI Integrated Planning* → *Settings for Lock Management*.

For more information, see Customizing with Reference to the Locking Concept [Page 187].

Examples

The following examples illustrate the rules for locking transaction data.

The example data in the following table is used as the basis:

Example data

Year	Customer Group	Customer	Sales Revenue	Costs
2004	IT	SAP	1000	500
2004	IT	IBM	2000	1000
2004	CARS	BMW	3000	2000
2004	CARS	VW	4000	3000

Example 1

In the definition of the planning level the following values have been determined:

Characteristic:	Key Figure:
Year = 2004	Revenue

The result would be as follows:

Results table for example 1

Year	Sales Revenue
2004	10000

The system has locked the key figure revenue for all four records (blue area in the example data table) as all records are required to calculate the total value. Any change to one data record will cause the total value to be changed. For this reason, a data record may not be used by another user simultaneously.

Example 2

In the definition of the planning level the following values have been determined:

Characteristic:	Key Figure:
Customer group = CARS	Costs

The result would be as follows:

Results table for example 2

Customer Group	Costs
CARS	5000

The system has locked the key figure *costs* for the two records for customer group "CARS" as these records are required to calculate the total value. Any change made to a data record causes the overall total to be changed. Therefore it must not be used by another user simultaneously.

In the examples given, the selection criteria are concordant with the existing records. As this is not normally the case (for example, no transaction data is available), the system creates locks on the basis of the selection criteria. This means that new records that are created during planning are also locked. In this way the system ensures that each user in an exactly specified area can change and create transaction data.

4.2.1.2.4.4.1 Customizing with Reference to the Locking Concept

Note that when creating and designing planning levels, several users will be planning simultaneously. In Customizing you therefore have to ensure that user-specific selections do not interfere with each other. As soon as a user executes a planning step and the system creates the corresponding locks, these locks remain in place until the end of the planning session. This is so that data consistency can be ensured. Note that the requested transaction data is normally connected in complex ways, even when data is being edited in a planning session with manual planning and planning functions in different planning levels. This means that you cannot normally only save and lock some data packages. The other data packages will also be locked.

> You can alter this behavior by using the user parameter SEM_BPS_SAVE_UNLOCK. This parameter allows you to save and unlock all edited data. The detailed application that was active when data was saved is then instantiated again. This means that only data required for this detailed application is reselected and locked (in change mode). For more information, see SAP Note 635244.

The following sections contain notes that should be taken into account during Customizing in order to avoid unwanted lock situations.

Selections with No Interference

To avoid users being hindered by other users, user-specific selections should not interfere with each other. The following rules apply:

- There is no interference between the selections if users are using different key figures. This is demonstrated by the table showing the sample data records in Locking Transaction Data [Page 184]: The colored areas derived from the examples do not overlap.
- As transaction data is always locked in the basic planning area, two selections can only interfere with each other if they access data from the same basic planning area. If different user selections are parts of different basic planning areas, or if the selection of at least one characteristic has no intersection, there is no interference between users.

  With multi-planning areas note that locks are implemented in the basic planning areas contained in the multi-planning area.
- No lock entries are created if a planning layout is executed in display mode.

Selections with Interference

There is interference between the selections if there is at least one common data record that is required to create the result values. This is shown in the following example:

User A and user B want to perform manual planning. User A includes the characteristic *year* and the key figure *revenue* in their planning level. User B includes the characteristic *year*, *customer group* = IT and the key figure *revenue* in their planning level. The results are as follows:

Results table for user A

Year	Sales Revenue
2004	10000

Results table for user B

Year	Customer Group	Sales Revenue
2004	IT	3000

The records highlighted in red are used in both sets of results:

Year	Customer Group	Customer	Sales Revenue	Costs
2004	IT	SAP	1000	500
2004	IT	IBM	2000	1000

As these selections interfere with each other they cannot be used by two different users simultaneously. The system produces a lock message accordingly.

Handling Characteristics Not Contained in the Planning Level

The following rule applies: If a characteristic is **not** part of a planning level, it can be accounted for in selection '*' (all values). We recommend that you adhere to this rule as selections can then be compared more easily and potential locking problems are easier to identify.

Example: Conversion rule

As far as locks are concerned, the following selections are the same, since all the records for the year 2004 are locked:

Characteristic:	Key Figure:
Year = 2004	Revenue

and

Characteristics:	Key Figure:
Year = 2004	Revenue
Customer group = *	

Example: Determining an intersection

You have to determine the intersection of the two selections shown above:

Selection A:

Characteristic:	Key Figure:
Year = 2004	Revenue

Selection B:

Characteristics:	Key Figure:
Year = 2004	Revenue
Customer group = IT	

Convert selection A in accordance with the rule mentioned above:

Characteristics:	Key Figure:
Year = 2004	Revenue

Customer group = *

If you compare the selections characteristic by characteristic you see the following intersection:

Characteristics:	Key Figure:
Year = 2004	Revenue
Customer group = IT	

Example: Checking whether selections interfere with each other

Selection A:

Characteristic:	Key Figure:
Year = 2004	Revenue

Selection B:

Characteristic:	Key Figure:
Customer group = IT	Revenue

In order to compare selections, they should contain the same set of characteristics. Based on the conversion rule, we add the *year* to selection A and *customer group* to selection B. The result looks like this:

Selection A:

Characteristics:	Key Figure:
Year = 2004	Revenue
Customer group = *	

Selection B:

Characteristics:	Key Figure:
Customer group = IT	Revenue
Year = *	

If you compare the selections characteristic by characteristic you see the following intersection:

Characteristics:	Key Figure:
Year = 2004	Revenue
Customer group = IT	

Budgeting as an example of this scenario

You want to implement a budgeting process and simultaneously facilitate a bottom-up and a top-down approach: The central planner enters the revenue values for all customers for the budget year. The person responsible for customer contacts enters the revenue values for their customers. The results are compared at group level.

The characteristics *year*, *customer group* and *customers* are added to the planning level. The revenue is stored with the key figure *revenue*.

The selection for the central planner is:

Characteristic:	Key Figure:
Year = 2004	Revenue

The selection for the person responsible for customer contact with SAP is:

Characteristics:	Key Figure:
Year = 2004	Revenue
Customer group = IT	
Customer = SAP	

When both planners are working, the second planner will receive a message saying data is locked.

Proposed solution: To avoid this locking problem, the data model has to be changed so that both groups are working with independent data records. You have the following options:

- Add a key figure: One key figure is used for the top-down direction, the other key figure is used for bottom-up.
- Add a characteristic (such as a budgeting version). This characteristic can be used to indicate the budgeting direction.

Comparison Columns in the Planning Layout

Lock-relevant characteristics are normally contained in the header or in the rows of the planning layout. As long as there are no overlapping selections in these areas, lock problems do not occur.

The following example deals with the special case that the data columns contain the users' different selections and you want to display the same reference data for all users.

Comparison columns are **not** input ready in manual planning. We differentiate between the following cases:

- The data in the comparison columns is in planning packages. In this case it is locked.
- The data in the comparison column is not in planning packages. In this case you have reference data that is not locked.

Individually defined comparison rows behave in the same way.

One example of using reference data is cost center planning where different users are assigned to the same cost center and are differentiated between using the characteristic *Version* .

Planning layout with reference data

Fiscal Year 2005		
Cost Center	**Costs/ Version: User Version**	**Current Cost: Version 0**
4711	90	100
4712	190	200
Total	280	300

If the planning package contains characteristic *Version 0*, you have a comparison column that is locked. If not, you have reference data that the different users can access without locking other users.

In the case of reference data, the system splits the selections for the planning layout into two disjunctive selections; each requires its own database request:

1. The selection for the plan data is: Fiscal year: 2005, cost center: 4711,4712, version: 'user version', _BKENNZAHL: 'Costs'. BW-BPS is called in change mode for this selection (the selection is locked).

2. The selection for the reference data is: Fiscal year: 2005, cost center: 4711,4712, version: Version0, _BKENNZAHL: ‘Costs’. BW-BPS is called in display mode for this selection (the selection is not locked).

For more information and examples of special features of using MultiProviders, see SAP Note 555849.

This function has to be activated in the system: Add a new record to database table UPC_DARK2 using param = SPLIT_SELECTION and value = X.

This function requires a high number of database accesses and may therefore affect performance in manual planning.

It is only possible to split the selection into two disjunctive selections if reference data is used either in the rows or in the columns. If there is reference data in the rows and columns, the system does not split the selection and all of the data in the planning layout is locked.

4.2.1.2.5 Documents

Use

You use this function to enable access to qualitative information that is related by content to the characteristics and key figures in a planning package. This way, you can store information close to the plan data, which may refer to the data itself, or to the way the data is to be handled.

Examples of using documents:

- The basis for the determination of particular plan data (for example, through reference to economic growth forecasts, or industry indicators such as order book levels).
- The graphical presentation of plan data in slides
- The preparation of planning manuals with details on how planning objects are to be set up, which planning tasks are to be carried out, and the planning intervals to be used.

Integration

If you have created documents for planning objects, you can maintain these in the *InfoProvider Data* area of document management in the Administrator Workbench. Documents for planning objects belong to document class TRAN.

You can find more information in the documentation on documentation management in the Administrator Workbench under Documents [External].

Prerequisites

At least one of the characteristics contained in the planning level must be checked for the “characteristic is document property” facility. This option is determined on *Tab Page: General* in InfoObject maintenance. For more information, see Characteristic is Document Property [External]. Otherwise, the *Documents* function cannot be accessed.

Functions

Creating and importing documents

You can create documents or import locally saved documents.

Documents refer to a combination of characteristic values. Additionally you can assign documents to a particular key figure from the current planning level.

When you have assigned a document to a key figure, the system evaluates this additional information when displaying planning layouts and queries so that the document is offered for display where the corresponding key figure exists.

> To create documents a characteristic must be restricted to one or several single values. The value # ("not assigned") is also allowed for this. With a restriction of characteristics to value sets, those documents can be displayed whose assigned characteristic value is contained in the value set. However, in this case you cannot create any new documents.

Display documents

The system offers documents to be displayed when the selection of characteristic values agrees with the characteristic values assigned to the document. This is also the case if the additional key figures assigned to the document are not available.

> To display a document in a planning layout it is necessary that a key figure is also assigned to the document. Only in this way can the system determine the cell in the planning layout in the field matrix of characteristics and key figures for which an existing document should be offered.

Example

If you have assigned a document to different combinations of characteristic values and the combinations partially overlap, the system proceeds in the following way: The system only displays those documents where the assigned characteristic value combinations are not more complex than the combination in the current selection. This avoids a situation where, in a selection at a superior hierarchical level, all documents are offered that have been assigned at lower hierarchical levels.

The following example illustrates this procedure:

Document assignments

Document	Customer Group	Customer
1	Wholesale trade	
2	Wholesale trade	IDES

3	Retail trade	ABC GmbH
4	Retail trade	

The following table shows which documents are offered with different selections:

Documents available by selection

Row	Selection: Customer Group/Customer	1 Available Document	2 Available Documents	3 Available Documents	4 Available Documents
1	Wholesale trade/*	Yes	Yes	No	No
2	Wholesale, retail trade/*	Yes	Yes	Yes	Yes
3	Wholesale trade/IDES	No	Yes	No	No
4	*/IDES	No	Yes	No	No
5	*/ABC Ltd.	No	No	Yes	No

Explanations:

- Row 1: The selection of customer group "wholesale trade" (without selection of individual customers) delivers all documents that are assigned to this customer group – independent of whether an assignment to a customer has been made.
- Row 2: The extension of the selection to two customer groups delivers all documents that were assigned to one of these customer groups, again independent of whether an assignment to a customer has been made.
- Row 3: For a selection with a combination of customer group and customer, the system delivers the documents with exactly this assignment; document 2, in the example.
- Rows 4 and 5: The selection by customer only delivers the corresponding documents.

Generally, in order to determine which documents should be displayed with a specified package selection, the system generates an internal query in which all characteristics of the planning level are included. Only those documents are delivered which fulfill the following conditions:

- All the characteristics contained in the planning level are assigned to the document.
- The characteristic values assigned to the document represent a subset of the package selection.

In the example above, documents 1 and 2 (which are only assigned to a customer group, but not to any customers) would not be displayed if the characteristic *customer* were also contained in a planning level.

See also:

Documents in the Planning Layout [Page 167]

4.2.1.2.5.1 Creating a Document

Use

You can create and edit documents in the planning workbench so that they can be accessed in the planning layout.

Prerequisites

At least one characteristic has to be contained in the planning level that is identified as “characteristic is document property” (see Documents [Page 191]).

Procedure

1. Select the required planning package.
2. Choose *Create Documents* from the context menu of the planning package.

   The system generates the *Documents* entry in the navigation area.
3. Choose *Call* from the context menu of the *Documents* entry.

   The system shows a detail screen that is divided in two:

   - List of the documents available for the characteristic value selection of the planning area.
   - Editing area for a document.
4. Choose *Create* from the context menu of the document list.

   The *Create/Import Document* dialog box appears. Enter the title and type of the document.

   The following document types are supported:

   - Microsoft Excel
   - Microsoft PowerPoint
   - Microsoft Word
   - Text document
5. The system displays all the characteristics in the current planning level that have documents that can be assigned. Enter one or more values for those characteristics to which you want to assign the document.

   > The system preassigns all characteristics for which the planning package contains a unique selection with the allowed value. You must enter characteristics for which several values are allowed appropriately.
6. Decide whether you want to assign a key figure to the document in addition to the characteristic values.
7. Confirm your entries.

   The system creates a document and adds an appropriate entry to the document list. The document is loaded into the assigned editor in the editing area.

Result

The newly created document is available for display in planning layouts, provided that displaying documents is activated in the layout.

See also:

Documents [Page 191]

4.2.1.2.5.2 Editing a Document

Use

You edit documents in order to comment on the plan and actual data entered for a combination of characteristic values as regards content. In addition, you can inform yourself about the existing characteristic value assignment of a document or delete a document.

Procedure

Change document

1. Choose a planning package in the desired planning environment.
2. Call up the list of available documents by double-clicking on the *Documents* entry in the *Planning Functions* screen area.
3. Choose *Change* from the context menu of the desired document. The system opens the document with the assigned editor in the editing area.
4. Make the desired changes.

Display characteristic value assignment of document

1. Choose a planning package in the desired planning environment.
2. Call up the list of available documents by double-clicking on the *Documents* entry in the *Planning Functions* screen area.
3. Choose *Display Assignment* from the context menu of the desired document. The system displays the characteristic values which are assigned to the document.

   You can display the assigned characteristic values, but **not** change them.

Delete a single document

1. Choose a planning package in the desired planning environment.
2. Call up the list of available documents by double-clicking on the *Documents* entry in the *Planning Functions* screen area.
3. Choose *Delete* from the context menu of the desired document.

Note when deleting documents that a document is only indirectly assigned to the current planning package. However, the primary assignment takes places exclusively with characteristic values. This means that if you delete a document in the context of a particular planning package, it is also deleted in all other planning packages and layouts at the same time.

In contrast to deleting documents, the *Delete* function in the context menu of the *Documents* entry in the *Planning Functions* screen area, is only used to remove the planning function type from the planning level. Here, the documents remain unchanged in the data basis. If necessary, the documents are also displayed in planning layouts of the current planning level, if these are set up for this, and a suitable combination of characteristic values is set up.

See also:

Documents [Page 191]

4.2.1.2.6 Planning Functions

Use

Planning functions change the transaction data in a planning package. Each planning function is assigned to just one planning level. There are various types of planning function (for example, copy, distribute, delete, formula). Parameter groups are created for a planning function. Those fields that are to be changed are determined when a planning function is defined The values of those fields are determined when a parameter group is defined.

You want to copy data from the actual version to the plan version. When you define the planning function you can determine which fields are to be changed during the copy operation. Select the *Version* field. When determining the parameter group you determine that version 001 is to be copied to version 002.

Integration

You can subsequently process the plan data created with these functions with all other kinds of planning functions including manual planning. In addition, in planning sequences [Page 267], you can group together any number of functions. These are always executed together in a predefined sequence.

Features

You design the planning functions according to a standard schema. To do this you select key figures and characteristics for each planning function for particular tasks (as change or condition fields or as reference data). The field types that are available and can be used in the corresponding **parameter group** depend on the function type (see the **functions and field types** table below and Editing Plan Data Automatically [Page 201]).

Field Type *Fields to Be Changed*

Here you select characteristics or – using the generic entry "key figure name" – key figures whose values you wish to change. When you execute a planning function using a parameter group, you determine which characteristic values (for example, in reposting) or key figures (for example, in revaluation) are to be changed in which way.

When you change characteristic values both the initial values and those created after the planning function was executed must lie within the planning package. If, for example, for the characteristic *company* there is a restriction to company codes 1000 to 2000, you cannot repost from company code 1000 to 3000.

Characteristics, which are restricted to a single value in the planning level are **not** displayed in the list of fields to be changed.

The planning functions formula [Page 251] and copy [Page 206] are the exception to this rule. In both these cases you are able to activate user-defined values that lie outside of the value range of the package.

Field Type *Fields for Conditions*

Here you select the characteristics whose values you wish to change in order to enter deviating processing rules per specification for the planning functions. You can also use the generic entry "key figure name" to determine that the planning function does not change the values of all key figures in the planning level (this is the default setting), but only the values of specific key figures.

For a planning function, you include the characteristic *version* and the generic entry "key figure name" in the *Fields for Conditions*. When you define a parameter group for the planning function, you choose the key figure *Sales* for version S1 and for "key figure name". This means that the planning function only processes those sales values that are assigned to version S1. *Sales* is the only key figure that is changed.

Restrictions that you can define for a function using condition fields are not an alternative to a characteristic value restriction in planning levels or packages.

The best way to use conditions is so that within the planning package the valid value ranges of a characteristic are completely covered by conditions you defined. If this is not the case, the whole data quantity defined by the package is transferred to the function for processing, although it is certain from the start, that not all of the data will be required. This can be disadvantageous in terms of performance.

Therefore SAP generally recommends that you reconcile planning packages and planning functions to each other as exactly as possible. If you determine, on the basis of the conditions of a function, that not all the data of a package is to be processed, you should either restrict the characteristic value of the package further or create a new package with the corresponding restriction and execute the function in this package.

If you use fields for defining conditions, those fields cannot simultaneously be included in the fields to be changed.

Field Type *Reference Data*

Here you select characteristics or – using the generic entry "key figure name" – key figures. You can then use their values, when executing the planning function, to relate the new plan values to existing values.

A typical example of the use of reference data is the distribution [Page 207] of plan data. When you execute such a distribution function, the first step is to read the reference data (usually the previous year's actual data) and to determine it's distribution. The distribution key that is determined automatically here is used in the second step to distribute the plan data to the selected characteristics. In this case, you would choose *version* as the reference characteristic and enter the value that identifies the actual data in the parameter group.

Value restrictions that are made in the planning level or planning package do not apply to characteristics that are used as reference data. Therefore, you can take all values that are defined for a reference characteristic, and use them as a reference.

The following table provides an overview of the various function types and the field types available in each case:

Function types (in alphabetical order) and the corresponding field types

Function Type	Available Field Types
Exit function	Change fields
Formula	Change fields, condition fields
Copy	Change fields, condition fields
Copy to several target objects	Change fields, condition fields
Delete	Condition fields
Deleting (invalid combination)	Condition fields
Forecasting	Condition fields, reference data, change field is the time characteristic
Reposting	Change fields, condition fields
Reposting (characteristic relationships)	Change fields, condition fields
Revaluation	Condition fields
Distribute with reference data	Change fields, condition fields, reference data
Distribute by reference data from sender to receiver	Change fields, condition fields, reference data
Distribute by key	Change fields, condition fields

Distribute by key from sender to receiver	Change fields, condition fields
Currency translation	Condition fields; the change fields (fields with currency keys) are calculated automatically
Unit conversion	Condition fields; the change fields (fields with unit keys) are calculated automatically

You can find more information on the individual **planning functions** in the following sections:

- Copy [Page 206]:
  - Copy
  - Copy to several target objects
- Distribute [Page 207]:
  - Distribute by key
  - Distribute by key from sender to receiver
  - Distribute with reference data
  - Distribute by reference data from sender to receiver
- Repost:
  - Repost [Page 211]
  - Repost (characteristic relationships) [Page 212]
- Delete:
  - Delete (for deleting data records from the planning package that fill the specified conditions).
  - Delete (invalid combinations) [Page 214]
- Revaluation [Page 215]
- Forecast [Page 216]
- Formula [Page 251]
- Exit functions [Page 257]

Functions for conversions:

- Currency translation [Page 259]
- Unit conversion [Page 264]

You can also execute a planning function directly from manual planning [Page 127] (for example revaluation, distribution), without having to leave the planning layout beforehand. After the function has been executed the system reads the updated data and displays it immediately in the planning layout.

See also:

Undoing Changes to Data [Page 200]

4.2.1.2.6.1 Undoing Data Changes

Use

Executing planning functions causes data of the planning package to be changed, generated again or deleted. When testing newly created functions and the corresponding parameter groups (in particular the Customizing phase) data changes may occur unintentionally.

> Note that double-clicking on the parameter group of a planning function causes the immediate execution of the function, while a double-click on all other planning objects merely displays the detail screen of the object.

To undo unintentional changes if necessary you have the following options within the BW-BPS planning environment:

- You can exit the planning session without saving. This means that all changes (both to data as well as to the planning objects, such as planning levels and functions) which you have made since saving the last time will be lost.
- You can deliberately undo the data changes which were caused by the last execution of a planning function. All remaining changes that you have executed during the current planning session are not affected. You will find more information on these options in this section.

Prerequisites

In order to be able to undo changes to data you must **not** have performed the following activities since executing the function that is to be reversed.

- You have not executed any further planning functions.
- You have not navigated to any other element (such as, for example, displaying a planning area or changing a parameter group) within the planning environment.
- You have not saved the data.

Features

When undoing data changes, the system restores the status for transaction data which it had before the planning function was executed. You can see the effects of the function in the following table for sample operating processes. The following cases are to be decided between with regard to manual changes:

1. The function to be undone was executed **after** a manual data change.
2. The function to be undone was executed **before** a manual data change.

1. Behavior when undoing a function after manual change

Data origin or interaction Step	Value
1. Persistently saved value from database	1000
2. Manual change in planning layout	800

3. Overwriting change by copy function	1200
4. Undoing planning function	800
5. Manual change in planning layout	900
6. Navigation to other planning object	900
7. Overwriting change by copy function	1200
8. Undoing planning function	900

After the last executed planning function was undone (step 4 and 8), the system restores the value which resulted from the manual change of data before executing the undone function.

2. Behavior when undoing a function which follows a manual change

Data origin or interaction step	Value
1. Persistently saved value from database	1000
2. Overwriting change by copy function	1200
3. Manual change in planning layout	800
4. Undoing planning function	1000

The system does **not** take manual changes that have been made in the mean time into account. The data is restored in the form in which it was before executing a planning function – independent of whether manual changes were made in the mean time.

Activities

Proceed as follows to undo the last executed planning function:

1. Choose *Edit* → *Undo Planning Function*.

   The system reports that the planning function will be undone.

2. Confirm the message.

   If you have opened a planning layout in which the affected data is displayed, you will immediately see the original data restored by the undo.

4.2.1.2.6.2 Editing Plan Data Automatically

Prerequisites

In order to edit plan data automatically, you must have already created a planning function and a parameter group. You must also have selected the planning package that contains the data the function will access.

Procedure

The use and operation of the various planning functions is to a large extent identical. The procedures and various details for the predefined planning functions are described below:

1. Add the characteristics or key figures, whose values are to be changed by the planning function, to the ***Fields to Be Changed*** list.

> Characteristics and key figures that have already been included in a planning function's list of condition fields cannot be changed by the same planning function.
>
> Depending on the type of planning function, it may be the case that the definition of the fields to be changed is not necessary. In this case, the list of these fields does not appear.

2. Decide which characteristics you want for the definition of conditions and add these to the list of ***Fields for Conditions***. Which characteristics are available depends on which characteristics were included in the planning level to which the planning function belongs.

   With some planning function types you have the additional option of using key figures for the definition of conditions.

3. Add the characteristics or key figures, the values of which are to be used as reference data by the planning function when generating new data, to the ***Reference Data*** list.

   > Depending on the type of planning function, it may be that the definition of the reference data is **not** necessary. In this case, the list of these fields does not appear. For an overview of the available field types per function type, see Planning Functions [Page 196].

4. Create a **parameter group**. In the parameter group you enter values for the characteristics and key figures that you selected as condition fields, fields to be changed or reference fields in the definition of the planning function.

   Editing the parameter group varies depending on which objects you have selected for the planning function:

   - If you have selected condition fields, first enter the necessary data for the conditions. With some planning functions (such as “distribution with reference data”) it is also possible to maintain all other settings here.
   - Choose *Detail Screen*. A table appears in which you maintain the fields of those values that are being changed. With planning functions of type “distribute by key”, those tables are ready for input in which you have to enter values for the characteristic that is being distributed to, and the key.
   - If reference data is required, you can also enter values for the reference data on the detail screen. In general, one value only can be entered for the reference data.

   According to the type of planning function you can also make additional technical settings in the parameter group. For example, you can decide for planning functions of type “copy” whether the plan data generated by the planning function overwrites existing data records with the same key, or whether it is added to these.

5. Execute the planning function. To do this, choose *Execute Planning Function* from the context menu of the parameter group.

   > When executing a planning function the system must be able to determine for which planning package in the current planning level the function is to be carried out. If the system cannot determine this, it will issue an appropriate message. In

this case, select a planning package by double clicking on it or select any function from the context menu of the package.

With the *Active Objects* function you can make the system display a list of those objects that are active at the current point in time and which your actions refer to.

Result

The plan data is generated automatically depending on the function you have selected. After the planning function has been executed, a message appears informing you whether execution was successful or if any errors have occurred. You can review the generated plan data using a planning layout with the function manual planning [Page 127].

If you carry out a planning function while a planning layout from manual planning is open, the system loads all new or changed data into the planning layout as soon as the function is carried out. This means that you get immediate feedback on the result of the function.

To save the generated data permanently in the database, choose *Save All*. This way, not only the plan data is saved, but also all other changes made to the planning architecture objects, the planning functions, and the parameter groups since the last time you saved.

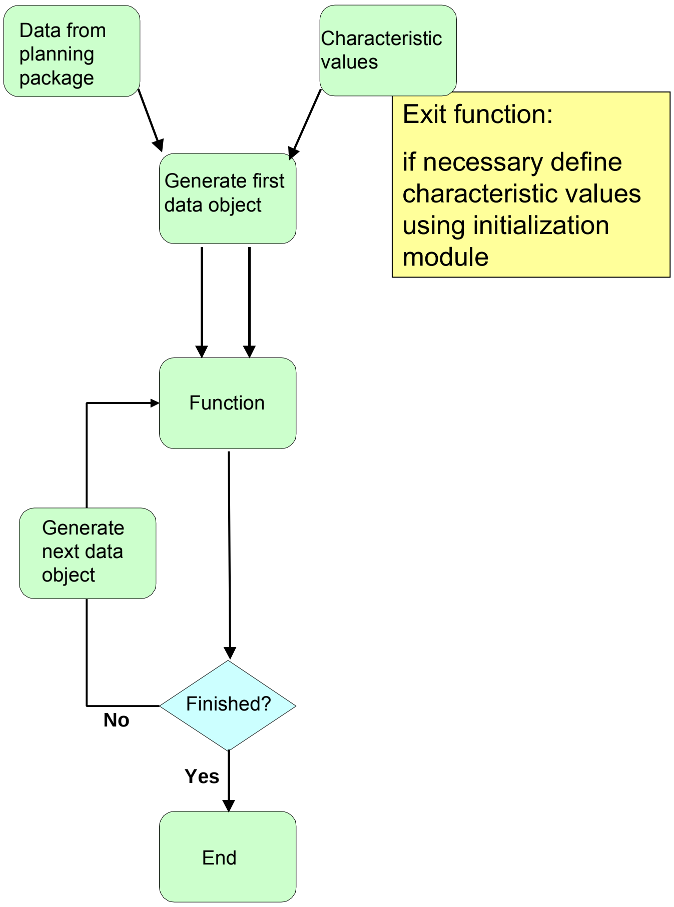
See also:

Planning Functions [Page 196]

4.2.1.2.6.3 Data Transfer to Planning Functions

In this section you learn how SEM-BPS executes the data transfer to this function internally when executing a planning function. The following information is valid for planning functions of all types, and is normally of no importance for the execution. However, especially for the use of Exit Functions [Page 257] and Formula Functions [Page 251], it is important that you are aware of the type of internal data preparation, which depends on which data you can access when executing such functions, and which data you cannot access.

SEM-BPS forms internal data objects which are transferred to this function for processing when a planning function is called up. The number of these data objects depends on whether or how many characteristics you have included in the list of fields to be changed for the function. The following processing logic arises from this:


Data from planning package
Characteristic values
Exit function:
if necessary define characteristic values using initialization module
Generate first data object
Function
Generate next data object
Finished?
No
Yes
End

A planning function is actually called up as often internally with one-off execution (from the user's view) as corresponds the number of formed data objects. The selection of characteristics to be changed has the following consequences:

- If you do not select a characteristic, then every data object contains exactly one data record that contains the values of all key figures for an internally defined unique combination of characteristic values. Therefore, to execute changes to all data records, the function is called up as often as data records exist in the data basis. In addition in this case, no internal processes can be made independent of data in other data records, since at the runtime of the function, access only exists to one data record.
- If you select all characteristics, exactly one data object is formed that contains all the data records of the data basis (corresponding to the selection conditions in the planning package and level). Therefore, you have access simultaneously to the values of all key figures for all entered combinations of characteristic values.
- If you select a selection from the characteristics available, each data object contains as many data records as result from the characteristic value combinations, that are formed from these characteristics and for which data is already entered. At the same time for every data record of the data object, an internally defined unique combination of values is generated, for those characteristics that you have **not** selected. The number of times that the function has to be called up, then corresponds to the number of data objects formed by SEM-BPS. If you produce internal references to other data records, then you must make sure that these data records are in the current data object at the runtime of the function.

See also:

Exit Function [Page 257]

Formula [Page 251]

4.2.1.2.6.4 Trace Mode for Planning Functions

Use

The trace mode allows you to obtain detailed information about which data objects have been changed and how when a planning function was executed. This function is especially useful when you set up new planning functions and parameter groups and want to check whether the function has the desired effect on the dataset.

Integration

The execution of planning functions in trace mode is available with all planning functions [Page 196]. (Although, the trace mode is not supported for manual planning [Page 127].)

Functions

When you execute a planning function in trace mode, before the function is executed, the system logs the values changed by the planning function for every internal data object, as well as the corresponding values. Internal data objects that are edited when the planning function is executed are subsets of the transaction data contained in the planning package (you execute a planning function by selecting a package and a parameter group). This package is divided into smaller transaction-data data objects. The only difference between the data objects is in the characteristic values of the characteristics to be changed.

After the function has been executed, you see the *Log* tab page in the right-hand area of the screen. Here you can find all the log information in the form of an overview of the transaction data records for which messages were triggered. For all messages that are not just *Information* messages (green status), you can display further *Details* on the selection conditions that caused the message to be triggered. These are those data records whose characteristic values or characteristics are the same and have **not** been selected as characteristics to be changed.

On the *Trace* tab page (screen *Edited Objetcs*) you can see an overview of all the edited internal data objects. For each internal data object you are able to:

- Call a detailed view that displays the key figure values that have been changed by the planning function, broken down by the values of the characteristics to be changed. If reference data is used when executing a planning function (for example, with the planning function type *distribution*), this is also presented in a separate screen area.
- Call a log that displays the messages triggered for the data records that are part of the data objects.

For more information about the internal data objects in which data from the package selection for the function execution is broken down, see Data Transfer to Planning Functions [Page 203].

Activities

To execute a planning function in the trace mode, proceed as follows:

1. In the planning environment, select the planning area, level, and packages for which you want to edit data.

2. Expand the list of planning functions for the parameter groups with which you want to execute a planning function.
3. From the context menu of the parameter group, choose *Execute with Trace.*

   The planning function is executed with the parameter group.
4. In the right-hand area of the screen, switch to the *Trace* tab page.
5. In the *Detail* column, choose to see the changes that the planning function has made to this data object. In the detail screen, the key figure values **before** executing the planning function have a light background, the values **after** executing the function have a dark background.

   > In certain cases it is possible that only the values for the time **before** the function was executed are displayed in the detail screen for an internal data object, or only the values for the time **after** the function was executed. For example, this is the case when data records have been deleted (function type *delete*), or regenerated (function types *copy*, *forecast*).
6. Return to the overview.
7. In the *Messages* column, choose to display the message texts for the data objects to be edited.

See also:

Planning Functions [Page 196]

4.2.1.2.6.5 Copy

Use

You use this function if you wish to copy plan or actual data from one characteristic value to another characteristic or number of characteristics (for example from one version to another version, or from one time frame to another time frame). You can take reference data from more than one year, for example, from 010.2003 to 004.2004, and also copy to more than one year. The number of periods in the reference and plan area must, however, be the same. The reference data periods must be selected as individual values (not as an area).

When updating the values you determine whether existing plan data is to be overwritten, or if the values are to be added to existing records on the database.

Features

BW-BPS offers the following copy functions:

Copy

Use this function if you want to copy plan or actual data from one characteristic value to another.

Copy to several target objects

This function is intended for particular cases where a template has to be copied several times. This is possible with the basic copy function but is made easier by this function.

Activities

Copy

For example, you want to enter plan values for three customers in a package and have already entered the values for the first customer. You want to copy the values for the other two customers from the values for the customer that has already been planned using entry help. To do this, select the customer number as the field to be changed. In the parameter groups, specify which new values are to be generated from which old values.

You can still proceed in the manner described above, even if the values that you want to copy are not in the package or level. After the planning function has run, the system checks whether the generated records are contained in the package.

If you do not want to copy all the key figures but just those that you have selected, select the fields *Customer* and *Key Figure* as fields for conditions. In the parameter groups, define which key figures and customers are to be copied.

Copy to several target objects

When you define the parameter groups the template can only be used once. When you define the target you can specify intervals.

See also:

Planning Functions [Page 196]

4.2.1.2.6.6 Distribute

Use

You use this function when you want to change the distribution of plan data to various characteristic values. The sum of the plan values for the chosen selection is kept constant and the total amount is newly distributed to the characteristics you want. You are able, for example, to determine a specific yearly sales figure as the total and then distribute this to the individual periods in the fiscal year. Typical application areas for the distribution of plan data are:

- Allocation of capital
- Budgeting
- Seasonal distribution of quantities and amounts

Prerequisite

At least one data record must exist for the corresponding selection conditions. The data from this data record is to be distributed using the selected distribution function.

Features

BW-BPS offers the following types of distribution:

- Distribute by **key**
- Distribute by key from sender to receiver
- Distribute with **reference data**
- Distribute by reference data from sender to receiver

You use appropriate parameter groups to define distribution keys. You create these parameter groups for a distribution function. You can carry out the distribution for all data records or for a specific combination of characteristics and key figures.

You can use reference data from more than one year, for example, from 010.2003 to 004.2004, and also plan for more than one year. The number of periods in the reference and plan area must, however, be the same. The reference data periods must be selected as individual values (not as an area).

You can decide whether the data to be distributed should be kept at the outgoing data records or deleted after distribution. In addition, you can determine that only data is distributed that was previously contained in data records with the initial characteristic value “non-assigned”.

The types of distribution differ as follows:

Distribute by Key

When distributing by key, you provide characteristics values and the shares by which distribution is to take place. For the fields to be changed you select the characteristics that you want to use and distribute the plan data using their values.

To distribute plan data within periods of a fiscal year in order to seasonally distribute the planned key figure, include, for example, the characteristic *period/fiscal year* in the list of fields to be changed.

In addition, you determine characteristics for conditions that must be fulfilled so that the planning function can be carried out.

To carry out planning only for those divisions that you specify, include the characteristic *division* in the list of conditions fields.

To be able to carry out a planning function defined in this way, you must finally define a parameter group. Enter the values necessary for your concrete planning task for the previously determined characteristics.

You could, for example, define as a condition that plan data only is to be generated for the "pharmaceuticals" division. In the parameter group you could determine that the key figure to be planned has a normal distribution over the fiscal year.

Distribution factors

The distribution factors with which you distribute the total amount to be distributed in the parameter groups represent relative values. These relate to the sum of the factors used in the parameter group: The system views the total that is formed from the addition of all factors within a parameter group as 100 % and then determines the relative weight of every factor for distribution. The following table shows the treatment of distribution factors:

Values *before* distribution

Period	Amount	Distribution factor (user entry)	Relative weight (internal calculation)
001.2003	100	1	12,5 %
002.2003	100	2	25,0 %
003.2003	100	2	25,0 %
004.2003	100	3	37,5 %
Total:	**400**	**8**	**100 %**

Values *after* distribution

Period	Amount	Distribution factor (user entry)	Relative weight (internal calculation)
001.2003	50	1	12,5 %
002.2003	100	2	25,0 %
003.2003	100	2	25,0 %
004.2003	150	3	37,5 %
Total:	**400**	**8**	**100 %**

The system proceeds as follows during distribution:

1. The total amount to be distributed is determined (400 in the example).
2. The distribution factors, which are assigned to the individual characteristic values (in the example the 4 periods), are added to a total (8) and then set in proportion to each other relatively. This results in the percentages which are reported in the “relative weight” column and always add up to 100 %.
3. The amounts, which are assigned to the individual characteristic values, are multiplied by the relative weights determined and thus distributed again.

Note the following special features: If the total of the keys is 0, distribution is carried out equally. Negative keys remain negative. The system tries to distribute rounding differences as equally as possible. Rounding differences usually occur because key figures are stored with two or three decimal places. Non-assigned characteristic values are disregarded when the key is created.

Distribute with Reference Data

When distributing with reference data, you make the same settings as when distributing by key. In addition, you select one or more characteristics that are to be used to determine reference data.

For example, you want to generate plan data for the "plastics" division. The data is to have the same seasonal distribution as the (existing) plan data of the "pharmaceuticals" division. Choose the characteristic "fiscal year/period" as the field to be changed and the characteristic *division* as the reference characteristic. You create a parameter group in which you enter the “pharmaceuticals” division under reference data. When executing the function, the system distributes the values to the periods so as they are already available for the values with the characteristic value “pharmaceuticals”.

Distribution with reference data is suited to two-level sales planning. First the revenue and sales quantities that are to be realized by different customer groups are planned. In a second level you can plan which revenue and sales quantities can be realized by the customers. To obtain proposals for this detailed planning, distribution is carried out with reference to the actual data for the previous year. The field that is supplied with values during this process is the *Customer Number* field. Plan values for revenue and sales quantity are thereby allocated as in the actual values of the previous year.

Further Settings

Only distribute non-assigned values

For the basic function types *distribute with reference data* and *distribute by key* you can also specify that only those data records should be included in distribution that contain the initial value "not assigned" (#) for the characteristics to be changed.

In the example above, this would mean that the values of all data records, to which no seven-digit period (= characteristic to be changed) was assigned, would be distributed to the four periods 001.2003 to 004.2003.

Retain values at sender data record

For the extended function types *distribute with reference data from sender to receiver* and *distribute by key from sender to receiver*, you can decide whether the values that you distribute to the receiver data records should remain posted at the sender data records, or whether the appropriate sender data records should be deleted after distribution.

Enhanced distribution function types

Function type	What you need to know
Distribution by key from sender to receiver	This function distributes values from sender to receiver within a characteristic (or several characteristics). When you define the planning function you have to specify within which characteristics you want distribution to take place. Specify the sender, receiver and ratio for the distribution in the parameter groups.

Distribute by reference data from senders to receivers	This function distributes values from sender to receiver within a characteristic. When you define the planning function you have to specify within which characteristic you want distribution to take place and the characteristics for reference data. Specify the sender and receiver in the parameter groups. The ratios that distribution is to be carried out by are determined from the reference data.

See also:

Planning Functions [Page 196]

4.2.1.2.6.7 Repost

Use

You use this function when you want to change the characteristic values of data records.

Reposting differs fundamentally from most of the other planning functions: Whereas the other planning functions make changes to **key figure values** you use reposting to change the specific combination of **characteristic values** that a data record belongs to (for example, to a specific division or region).

Reposting largely corresponds to the *Copy* function. However it is different in the following respects:

- You can only perform reposting within a planning package.
- You select the fields that are to be changed. For the planning parameters (parameter groups) you specify the old and the new value. The system reposts the old value as the new value.
- The key figure in the old data record is deleted.

Prerequisites

The target values that you want to plan as a result of the reposting must be maintained in the relevant master data tables.

Features

You can determine that reposting is only carried out for those data records that have a specific characteristic combination. You define these combinations by including the characteristic affected as a condition field. Apart from characteristics, you can also use the generic entry "key figure name" so that the function is carried out for certain key figures only.

When you create a parameter group for the function, two columns appear for each of these characteristics. Here, you enter the old value to be changed and the new replacement value. If you have also included key figures in the condition fields, you can make this setting separately for each key figure you have selected.

The system proceeds as follows during reposting: The existing data record is deleted. If there is not yet a data record that contains the new characteristic value combination and the same key figure selection, then a new data record is generated for the new characteristic value to which it is reposted. However, if such a data record already exists, then the key figure values of the data record to be reposted are added to the key figure values of the existing data record. Therefore, in this case no new data record is formed in the database, instead only the key figure values of the existing data record change.

The data records with the original characteristic values that were changed in reposting are no longer available.

The new characteristic values that are generated by reposting must be covered by the characteristic value restrictions that are valid for the planning level or planning package. Otherwise, no changes are made.

See also:

Planning Functions [Page 196]

4.2.1.2.6.8 Reposting (Characteristic Relationships)

Use

Characteristic relationships [Page 110] allow you to determine that, within a planning area, the system automatically derives the values of certain characteristics when generating transaction data and enters these derived values in the key of the data record. Apart from the possibility to determine characteristic values with exits, the way of determining characteristic values complies with conditions which are specified by data modeling: Attributes, hierarchies, and the link to reference data can be selected here.

If there are changes in the area of data modeling after transaction data has been generated using characteristic relationships (for example, a new assignment of attribute values or the reorganization of a hierarchy), the transaction data has to be made consistent.

You have generated transaction data for the article “light bulb”. Because of a characteristic relationship defined in the planning area, the appropriate article group “electrical appliance” was derived for the article, and the data record was saved in the database with the characteristic value combination “light bulb”/”electrical appliance”.

Later, in the context of a reassignment, the article “light bulb” was assigned to the article group “household goods”. Thus the data record saved with the characteristic value combination “light bulb”/”electrical appliance” is invalid.

In our case you have to define a planning function of type *Repost (Characteristic Relationship)* that contains the *article group* characteristic as a field to be changed.

Integration

Implementing this function is only advisable under the following conditions:

- Within a planning area in which characteristic relationships of the type *derivation* are defined.
- After data records have been generated whose characteristic value combinations may have been affected by these relationships.

Prerequisites

To make sure that all possible data records are entered by the function, you can only create and execute functions of the type *Repost (Characteristic Relationships)* in a planning level that contains all characteristics and key figures of the InfoCube it is based on.

Planning functions of type *Repost (Characteristic Relationships)* can only be created in a basic planning area and **not** in a multi-planning area [Page 97].

The function type *Repost (Characteristic Relationships)* differs from all other planning functions in so far as the system checks whether the planning level contains all characteristics and key figures of the InfoCube.

The reason for this lies in the function's mode of operation. To repost the existing transaction data records from the previously assigned characteristic values to the values that are valid now, the existing key figure values must be assigned internally to the characteristic value combinations that are valid now. This is only possible if the transaction data records are selected with all characteristics for which data was posted at any time. But this condition is only fulfilled when the quantity of characteristics in the planning level in which reposting is executed, corresponds to the upper quantity of all characteristics from all levels of the planning area. The same applies to the quantity of key figures. But even under these circumstances, it is possible that a level contained further characteristics for which data was also posted at an earlier time. Therefore, a secure and consistent reposting of data can only be guaranteed if this happens taking all characteristics and key figures of the planning area into account.

Features

You make a selection of the fields to be changed where the system (different to all other planning function types) does **not** offer all characteristics of the planning level, but only those which occur as a derived characteristic (target characteristic) in the characteristic relationships of the planning area. If necessary, you can define the conditions to restrict the data space within the function.

The parameter group that you create for a function type *Repost (Characteristic Relationship)* has a formal character. You only enter the names of the parameter group here. Further functions in relation to the fields to be changed are not possible.

If you execute the function with parameters set in this way, the system determines all data records in accordance with the selection in the planning package that are saved for characteristic relationships, in which at least one of the target characteristics occurs from characteristic relationships of the type *derivation*. If necessary, the function changes the values of the derived characteristics in accordance with the conditions, which apply in the data model at the execution point of the function.

See also:

Delete (Invalid Combination) [Page 214]

Planning Functions [Page 196]

4.2.1.2.6.9 Delete (Invalid Combination)

Use

Under certain circumstances executing the *Reposting (Characteristic Relationships)* function can mean that data records exist within a planning area that are inconsistent regarding their characteristic combination because of changes made in the area of data modeling (see Reposting (Characteristic Relationships) [Page 212]).

To delete data records of this type from the database, use the *Delete (Invalid Combination)* function. This is recommended when the change in the characteristic values reflects a business activity that does not allow a sensible conversion of the data records using the *Reposting (Characteristic Relationships)* function.

> You have entered transaction data for branch A100 of your company. Because of a characteristic relationship defined in the planning area, the appropriate region, “North,” was derived for the branch and the data record was saved in the database with the characteristic value combination “A100”/“North.”
>
> Later while restructuring your company, you dissolve the branch and the characteristic value A100 of the characteristic branch is deleted. Thus the data record saved with the characteristic value combination “A100”/“North” is invalid. You can delete the data record with the function described here, provided that you no longer require the data records for summary reporting at regional level.

Integration

Implementing this function is only advisable under the following conditions:

- Within a planning area in which characteristic relationships of the type derivation are defined.
- After data records have been generated whose characteristic value combinations may have been affected by these relationships.

Prerequisites

To make sure that all possible data records are entered by the function, you can only create and execute functions of the type *Delete (Invalid Combinations)* in a planning level that contains all characteristics and key figures of the InfoCube it is based on.

Features

You can use conditions to restrict the number of data records among which the system should search for invalid characteristic combinations. No further settings are possible.

After you have created a parameter group for a function, you can execute it. The system then searches for the data records, the characteristic values of which are invalid, and deletes the appropriate data records. Characteristic values can be invalid for the following reasons:

- A value is entered for a characteristic in a data record that is not contained in the master data of this characteristic.
- Values occur in a combination of characteristic values, which are linked by a hierarchy, but do not occur in the hierarchy.

See also:

Reposting (Characteristic Relationships) [Page 212]

Planning Functions [Page 196]

4.2.1.2.6.10 Revaluation

Use

When revaluating your plan data you specify a percentage by which the key figures of a planning package are to be increased or decreased.

This enables you, for example, to have your plan values for 2005 based on the actual values from 2004 plus 10%.

Features

Revaluation for selected key figures

The standard setting for revaluation functions is that the revaluation is carried out for **all** key figures that were included in the planning level. You can, however, select specific key figures from the total amount and have only these revaluated. To do this you must include the generic entry "key figure name" in the list of condition fields when you are defining the revaluation function. When you create parameter groups for the function, an additional column appears. Here, you enter those key figures for which the revaluation is to be carried out.

Revaluation for selected characteristic value combinations

You can perform revaluation either for all data records or only for those data records for which a particular characteristic value combination exists. You define these combinations by including the characteristic affected as a condition field for the formula function. When you create a parameter group for the function, a column appears for each of these characteristics and you can enter the values here.

This means that within the parameter group of a revaluation function, you can specify a percentage rate for each combination of values of all characteristics that were included as condition fields, and (if applicable) for each key figure that exists in the planning level. Then, when you carry out the planning function, all these percentage rates become effective.

Example

The planning level contains the key figures *revenue* and *sales*. You create a revaluation function that contains the generic entry "key figure name" and also the characteristics *country* and *division* as condition fields. You also define a parameter group for this function. The parameter group contains the following entries:

Parameter group

Key figure	Country	Division	Revaluation
Revenue	DE	Pharmaceuticals	10
Revenue	DE	Plastics	8
Revenue	G.B.	Pharmaceuticals	12
Revenue	G.B.	Plastics	2–
Sales	DE	Pharmaceuticals	8
Sales	DE	Plastics	15
Sales	G.B.	Pharmaceuticals	10
Sales	G.B.	Plastics	5

In this way you define plan data that results in differing sales growths for both divisions in both countries, and a drop in profits for the German plastics division. In addition, an increasing return on sales can be derived from the planned figures for the pharmaceuticals division, whereas it is decreasing in the plastics division

See also:

Planning Functions [Page 196]

4.2.1.2.6.11 Forecasting

Use

Use a forecast procedure to predict the future development of key figures. The forecast function uses historic data to calculate expected forecast values with statistical procedures.

This involves simple univariate forecast procedures.

Integration

The forecast function has been newly implemented in BW-BPS. In standard cases the previous forecast function used within the bounds of SEM-BPS is **no longer** available. For compatibility reasons the system still provides the old function if you were using it within SEM-BPS.

SAP recommends you use the newly implemented forecast function in BW-BPS for new developments. This explicitly enhances the functionality of the old (SEM-BPS) forecast function, even though it does not cover all aspects of it. (For example, you do not obtain forecasts according to Croston’s algorithm). For more information on the old (SEM-BPS) forecast function, see Forecasting (Old) [Page 227].

Prerequisites

- Historic data that can serve as reference data for the forecast must be available.
- The planning level, in the context of which you create a forecast function, must contain at least one characteristic with a time reference (for example, fiscal year).
- The forecast period is copied from the selection in the planning package or planning level and therefore must be restricted there.

Features

Forecast strategies

The following forecast strategies are available to you for performing calculations:

- *Automatic model selection*
- *Average*
- *Moving average*
- *Weighted moving average*
- *Simple exponential smoothing (constant model)*
- *Linear exponential smoothing (trend model)*
- *Seasonal exponential smoothing (seasonal model)*
- *Trend-seasonal exponential smoothing (multiplicative seasonal component)*
- *Trend-seasonal exponential smoothing (additive seasonal component)*
- *Linear regression*

The *automatic model selection* forecast strategy allows you to let the system select the forecast model that best fits the trend of the historic data (see Automatic Model Selection [Page 221]).

If you already know that a particular forecast model is well matched to the time series trend, or if you explicitly want to use a forecast model for other reasons, you can select a particular forecast model (see Forecast Strategies [Page 219]).

Optional functions for the forecast strategies

The forecast strategies offer the following additional functions and possibilities:

- Outlier correction [Page 223]
- Setting negative forecast values to zero [Page 224]
- Logging statistical key figures [Page 225]
- Ignoring initial zeros [Page 226]

For exponential smoothing

- Optimization of smoothing factors for exponential smoothing [Page 222]

For forecast models with trend components:

- Trend dampening [Page 224]

Activities

To create a planning function of type *Forecast*, the following steps are necessary:

1. Creating a planning function of type *Forecast*

   Time characteristic

   A time characteristic is required for each forecast function in order to represent the time dimension of the forecast.

   ○ The selected time characteristic is included automatically in the set of reference characteristics.

   ○ The characteristic has to be restricted for the package. Values for this period are forecast.

   Fields for reference data

   The characteristics selected as reference data serve to determine the historic data on the basis of which forecast values are to be calculated.

   Fields for conditions

   You cannot include the selected time characteristic in the set of characteristics for conditions.

   For more information on fields for conditions, see Planning Functions [Page 196].

2. Creating parameter groups

   Select a forecast strategy on the ***Forecast Parameter* tab page**. This determines how forecast values are to be calculated. Certain parameters have to be entered depending on the forecast strategy.

   The system proposes *Automatic Model Selection* as the default forecast strategy and, in comparison to other forecast strategies, makes available the greatest number of parameters. Note that calculation is accordingly time-consuming.

   On the ***Reference Data* tab page**, determine the period in the past on which the forecast is to be based.

   Make sure to exclude non-posted special periods from the past period as otherwise these periods go into the forecast calculation with the value 0.

3. Executing the planning function

   In comparison to the *Execute Planning Function* option, the *Execute with Trace* option is more time-consuming. As well as the tab page *Log*, the system also adds the tab page *Trace*.

   ○ You can display the log for each data object for which the forecast was generated by clicking on in the *Messages* column.

   ○ You can display the reference data and the values of the processed data objects before and after execution by clicking on in the *Detail* column.

4.2.1.2.6.11.1 Forecast Strategies

Use

The forecast strategy determines how forecast values will be calculated.

All forecast strategies are based on statistical forecast procedures and, therefore, on forecast models that mathematically qualify the period of historic data. The exponential smoothing methods (exponentially weighted moving average) are currently the most widely used time series methods (see Exponential Smoothing [External]).

If you expect historic values to continue to develop as they have in the past, choose a forecast model that fits the previous trend well.

The *automatic model selection* strategy allows you to let the system select the forecast model that best fits the trend of historic data (see Automatic Model Selection [Page 221]).

Features

The following forecast strategies are available:

Average

The forecast value is calculated from the medial average of the historic values.

Optional forecast parameters: Outlier Correction [Page 223], Setting Negative Forecast Values to Zero [Page 224], Logging Statistical Key Figures [Page 225] , Ignoring Initial Zeros [Page 226]

Moving average

The forecast value is calculated according to the order.

- Obligatory forecast parameter: *Order of Moving Average*

  The order of the moving average is a number N that determines the length of the time interval for calculating the average. This is the number of chronologically sequential historic values. The forecast value is calculated as the average of the last N historic values.

  Do not enter a negative number for the order.

- Optional forecast parameters: Outlier Correction [Page 223], Setting Negative Forecast Values to Zero [Page 224], Logging Statistical Key Figures [Page 225] , Ignoring Initial Zeros [Page 226]

Weighted moving average

To calculate the moving average, each historic value obtains a defined weight in the weighting group.

- Obligatory forecast parameter: *Weighting Group Moving Average*

  The weighting group for the moving average is a key that specifies

  - how many historic values are included in the calculation of the forecast value
  - what weight the individual historic values carry in the calculation

By clicking on *Weighting Groups* you will come to a screen for displaying and maintaining weighting groups.

- Optional forecast parameters: Outlier Correction [Page 223], Setting Negative Forecast Values to Zero [Page 224], Logging Statistical Key Figures [Page 225] , Ignoring Initial Zeros [Page 226]

Simple exponential smoothing (constant model)

Simple exponential smoothing is suitable if the historic data displays a horizontal trend.

- Smoothing factor settings: *Alpha* (Basic Value [External])
- Optional forecast parameters: Outlier Correction [Page 223], Setting Negative Forecast Values to Zero [Page 224], Logging Statistical Key Figures [Page 225] , Ignoring Initial Zeros [Page 226]

Linear exponential smoothing (trend model)

The forecast is calculated according to Holt’s method and is suitable if historic values display a rising or declining trend.

- Smoothing factor settings: *Alpha* (Basic Value [External]), *Beta* (Trend Value [External]).
- Optional forecast parameters: Outlier Correction [Page 223], Trend Dampening [Page 224], Setting Negative Forecast Values to Zero [Page 224], Logging Statistical Key Figures [Page 225], Ignoring Initial Zeros [Page 226]

Seasonal exponential smoothing (seasonal model)

Choose this strategy if your historic values display seasonal fluctuations (for example, annual fluctuations) from a constant base value.

- Obligatory forecast parameter: *Periods per season*
- Smoothing factor settings: *Alpha* (Basic Value [External]), *Gamma* (seasonal components).
- Optional forecast parameters: Outlier Correction [Page 223], Setting Negative Forecast Values to Zero [Page 224], Logging Statistical Key Figures [Page 225] , Ignoring Initial Zeros [Page 226]

Trend seasonal exponential smoothing (multiplicative seasonal component)

The forecast is calculated according to Winter/Holt’s multiplicative method and is suitable if historic values display seasonal fluctuations from a rising or declining trend. Here the extent of the fluctuation depends on the strength of the trend.

Ice cream sales in summer: It is taken for granted that ice cream sales rise by a trend of 10% annually. A seasonal increase of 30% each summer then leads to ever greater absolute fluctuations.

- Obligatory forecast parameter: *Periods per season*
- Smoothing factor settings: *Alpha* (Basic Value [External]), *Beta* (Trend Value [External]), *Gamma* (seasonal components).

- Optional forecast parameters: Outlier Correction [Page 223], Trend Dampening [Page 224], Setting Negative Forecast Values to Zero [Page 224], Logging Statistical Key Figures [Page 225], Ignoring Initial Zeros [Page 226]

Trend seasonal exponential smoothing (additive seasonal component)

The forecast is calculated according to Winter/Holt's additive method and is suitable if historic values display seasonal fluctuations from a rising or declining trend. Here the extent of the fluctuation is independent of the strength of the trend.

An example of this is a product, 1000 units of which are sold every July over a long-term trend.

- Obligatory forecast parameter: *Periods per season*
- Smoothing factor settings: *Alpha* (Basic Value [External]), *Beta* (Trend Value [External]), *Gamma* (seasonal components).
- Optional forecast parameters: Outlier Correction [Page 223], Trend Dampening [Page 224], Setting Negative Forecast Values to Zero [Page 224], Logging Statistical Key Figures [Page 225], Ignoring Initial Zeros [Page 226]

Linear regression

Simple linear regression (ordinary least squares).

- Optional forecast parameters: Outlier Correction [Page 223], Trend Dampening [Page 224], Setting Negative Forecast Values to Zero [Page 224], Logging Statistical Key Figures [Page 225], Ignoring Initial Zeros [Page 226]

4.2.1.2.6.11.2 Automatic Model Selection

Use

Automatic model selection allows you to let the system determine which forecast model best fits your historic data.

SAP recommends automatic model selection if you do not know the trend of your historic data, if you cannot estimate how your data will develop, or if you do not want to specify a model.

Features

The system computes and compares the forecast results for all exponential smoothing methods. It selects the best of these models. Note that the system suggests the optimization of smoothing factors.

Note that optimization requires a great amount of calculation. This is particularly true of the trend-seasonal model. The calculation cost also depends on the size of the search space and the precision of the step sizes you set (see Optimizing Smoothing Factors for Exponential Smoothing [Page 222]).

The suitability of a model is measured by Akaike's Information Criterion (*AIC*). This takes into account the fitting error and the number of parameters for the model. The fitting error results from the residuals between the calculated forecast values and the observed values. The smaller the AIC, the better the model fits.

4.2.1.2.6.11.3 Optimization of Smoothing Factors for Exponential Smoothing

Use

If you do not want to enter smoothing factors for exponential smoothing yourself, or you are unable to, you can let the system determine the best smoothing factors by means of optimization.

The optimization of smoothing factors is a mathematical process to determine the best possible smoothing factors. The aim is to fit the underlying forecast model to the historic data as well as possible by varying the smoothing factors. In this way you can generally expect to find the best quality forecast for the forecast model.

The variation of smoothing factors takes place within certain limits. These are defined by the search space.

Features

Optimization consists of two steps:

- Performing a grid search which variegates the smoothing factors within the respective limits with the respective step sizes. The error is determined for each combination of smoothing factors (in accordance with the optimization variable). The result of the grid search delivers the starting value for the second step.
- Executing the direct search according to the Hooke-Jeeves algorithm.

The system **cannot** guarantee that this optimization algorithm finds the absolute optimum as the direct search can converge to a local minimum for the error measure.

Activities

Choose this option to obtain the highest quality forecast.

As with *automatic model selection*, the system proposes the optimization of smoothing factors supplied by the system with exponential smoothing forecast models (see Automatic Model Selection [Page 221]). This can be costly in terms of time when using trend-seasonal exponential smoothing. In this case SAP recommends:

- Speeding up the search by reducing the search area for the smoothing factors or increasing the step size.

- Entering suitable smoothing factors yourself. This is only advisable if, for example, you have recognized suitable values for the smoothing factors from past experience.

Optimization variables

The optimization variable is an error measure that is used for optimizing smoothing factors. Error measures allow a judgment to be made on the quality of the forecast. Optimization variables determine when one combination of smoothing factors is to be preferred to another combination.

You can select one of the following optimization variables:

- *Mean absolute error*
- *Mean absolute percentage error*
- *Mean squared error (MSE)*
- *Exp. smoothed absolute error*

Search area

You can restrict the search space for smoothing factors by defining an upper and a lower limit for each smoothing factor.

Furthermore, you can set a *step size* for each smoothing factor with which the system should begin to search, within the appropriate search space.

4.2.1.2.6.11.4 Outlier Correction

Use

“Outliers“ are atypical values that cannot be explained by the forecast model that is being used. The results of the forecast can be heavily influenced by outliers.

The system is able to identify and replace outliers in the historic data. To do this the forecast procedure calculates forecast values in the past period and compares them to the observed values. If the difference (the residual) exceeds a specific value, the observed value is replaced by the ex-post-forecast value for the corresponding point in time. After this correction, the forecast calculation is performed again with the amended historic data.

You determine a Sigma factor in order to define the threshold value.

Integration

Outlier detection depends on the respective forecast model because the above mentioned forecast value is computed according to a model and its related algorithm.

Features

In the context of the forecast function, outliers in the historic values are defined by the Sigma factor. The greater the Sigma factor the more tolerant the system of atypical values, and the fewer outliers are determined.

The Sigma factor *fac* controls outlier detection in the following way: An observed value *y* is declared to be an outlier if the difference to the forecast value *e* is greater than *fac* *s. *s* denotes the standard deviation of the residuals.

Activities

Activate outlier correction if you think that outliers are having an unfavorable effect on the forecast result in accordance with the forecast function definition.

Do not select outlier correction if atypical values are always to be taken into consideration by the forecast.

4.2.1.2.6.11.5 Trend Dampening

Use

For forecast models with a trend component (*linear regression*, the trend models for *exponential smoothing with or without seasonality,* and, in some cases, *automatic model selection*), you can dampen the trend for future forecast values by specifying a trend dampening factor.

Use the trend dampening factor if you expect that the past growth rate will either slow down or intensify in the future.

Features

The trend dampening factor is a number that is multiplied by the trend value (growth rate) for the calculation of each forecast value at the respective time. In this way, growth in a long-term trend is further slowed or intensified:

- With a trend dampening factor that is less than 1, a type of saturation effect is produced. The trend will exponentially converge to 0.

  A trend dampening factor of 0.9 decreases the growth rate for each period by 10% recursively.

  If, for example, the number of cars sold currently grows by 1000 per period, the growth in the number of cars sold in the next period would be 0.9 * 1000 = 900, and would be 0.9 * 900 = 810 in the period after that and so on.

- To achieve the reverse effect, you can enter a trend dampening factor that is greater than one.

Activities

To perform trend dampening set this indicator and determine a trend dampening factor that corresponds to your expectations.

4.2.1.2.6.11.6 Setting Negative Forecast Values to Zero

Use

Negative forecast values can be useful but this depends on the significance of the forecast key figure.

For example, negative forecast values occur when the historic data shows a declining trend and you have selected a forecast model with a trend component.

In other cases negative forecast values are not desirable.

Negative forecast values that are not meaningful include, for example, certain quantity key figures such as a quantitative warehouse inventory in units.

Features

Negative forecast values are replaced with the value 0.

Activities

Choose this option to make sure that all forecast values are always greater than or equal to 0.

4.2.1.2.6.11.7 Logging Statistical Key Figures

Use

This function logs statistical information about the forecast result.

After the forecast function has been executed the system displays the corresponding statistical information on the *Log* tab page, just as with error messages.

Features

The following statistical information is logged:

- Mean absolute percentage error (MAPE)
- Squared root from the mean squared error (RMSE)
- Standard deviation of residuals
- R squared (r^2) as the measure of how well the model fits the data
- Number of outliers (where outlier correction is active)
- Forecast model (with automatic model selection)
- Smoothing factors (where optimization is active)

Activities

Choose this option in order to log statistical information about the forecast result.

4.2.1.2.6.11.8 Ignoring Initial Zeros

Use

Use this function to determine whether a string of zeros at the beginning of a selected period in the past are to be taken into consideration for the forecast.

> You want to calculate forecast values for your product. The past period covers the last 24 months. However, the time series of products that have only existed for a few months begins with a lengthy sequence of zeros. You do not want the forecast values to be influenced by these zeros so you select the option *Ignore Initial Zeros*.

Activities

Choose this option if initial leading zeros are to be excluded from the forecast calculation.

4.2.1.2.6.11.9 Model Initialization

The forecast model is initialized automatically. To do this the system needs a particular minimum number of historic values for the particular model. This is presented in the following table:

Minimum number of historic values per model

Model	Number of historic values
Average	1
Moving average	Order of moving average
Weighted moving average	Order of moving average
Linear regression	2
Constant	1
Trend	3
Seasonal	1 seasonal length
Seasonal trend (mult./add.)	2 * seasonal length

For the exponential smoothing algorithm, starting values are required for the base value (level), the trend value (growth rate), and the seasonal indices. The calculation of these starting values is called *model initialization*. This initialization is performed heuristically and depends on the specific model:

- **Constant model** (simple exponential smoothing)

  The starting value for the base value is calculated as the medial average of the first five values (that is, the average of the values after removing the minimum and maximum value).

- **Linear model** (Holt's exponential smoothing)

The starting values for the base value and trend value are determined by the regression line for the first eight values with the *least squares method* (Ordinary Least Squares [*OLS*]).

- **Trend-seasonal model** (Holt-Winter's exponential smoothing)

  The **trend component** is estimated by calculating the centered moving average of order *s* for the first four seasons. *s* denotes the seasonal length.

  If, for example, s = 12 for monthly observation values, the trend component is estimated for the first four seasons, the first 48 months.

  The system can also estimate a trend component if less data is available. However, at least two seasons must be provided.

  The centered moving average is a time series T and represents the estimation of the trend component. Starting values for the base value and trend value are calculated using the regression line (OLS) by time series T.

  The **seasonal component**, that is the seasonal indices, is estimated in the following ways:

  - For the *multiplicative* seasonal component the average of the local indices is calculated: S(t) = Y (t) / T (t).

  - For the *additive* seasonal component the difference between the local indices is calculated: S(t) = Y (t) / T (t).

  Y(t) denotes the observed value and T(t) denotes the value of the trend component for that time.

- **Season model** (Winter's exponential smoothing)

  This model corresponds to the trend-seasonal model but without the estimation of the trend component.

4.2.1.2.6.11.10 orecast (Old)

Use

A forecast is a projection of existing data using statistical methods. You use forecast functions to draw conclusions from the historical development of key figures and to predict their likely future development.

Prerequisites

There must be data available that can serve as reference data for the forecast. In addition, the planning level, in whose context you create a forecast function, must contain at least one characteristic with time reference (for example, fiscal year).

Features

Based on existing plan or actual data, new plan data is generated for a certain time period. Various forecast strategies or models are supported and you determine these when you create a forecast function (you can make changes later if necessary). The available forecast strategies include the following:

- Exponential Smoothing (with and without Test for Trend and Seasonality) [External]
- Moving Average [External]
- Constant Model [External]
- Trend Model [External]
- Test for Trend and Seasonality

To determine the data on which the forecast is to be based, specify the characteristics you want as reference data. The time characteristic you specified, which is required for each forecast function, is automatically included in the list of reference data.

You can determine that the forecast is only carried out for those data records that have a specific characteristic combination. You define these combinations by including the relevant characteristics as condition fields for the function. Apart from characteristics, you can also use the generic entry "key figure name" so that the function is carried out for certain key figures only. When you create a parameter group for the function, a column appears for each selected characteristic and you can enter the values here.

> The time characteristic that you choose for a forecast function cannot be used at the same time in the same function as a characteristic for defining conditions. Therefore, the system automatically assigns this characteristic to the reference data and removes it from the list of characteristics for conditions.

See also:

Model Selection [Page 228]

Forecast Model [Page 234]

Forecast Parameters [Page 235]

Forecast Formulas [Page 236]

4.2.1.2.6.11.10.1 Model Selection

Use

Before you run a forecast, you must specify which model the system should use to calculate the forecast values. Your selection will depend on your knowledge of the data pattern to-date.

Prerequisites

Historical values are available and you can make conclusions from these regarding the most appropriate forecast model.

Features

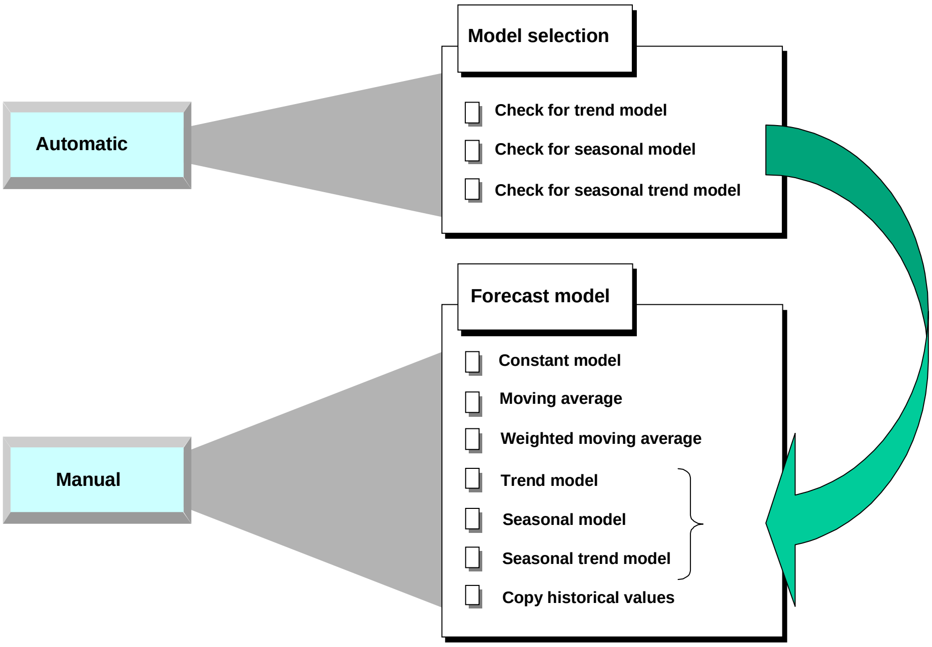
Model selection alternatives are as follows:

- manual model selection
  You determine the forecast according to your preferred model. This is suitable when you have a clear idea of the future development of the key figures although this

development will have a different pattern to their development in the past. In this case, the future data cannot be derived mathematically from the historical data and you must let the system know how to proceed.

- automatic model selection
  The pattern to-date enables you to determine the presence of seasonality or another trend and, therefore, you run a test to find the closest development pattern.

The following graphic shows the different approaches of automatic model selection and manual determination of a forecast model:



See also:

Automatic Model Selection [Page 231]

Manual Model Selection [Page 229]

Model Initialization [Page 233]

4.2.1.2.6.11.10.1.1 Manual Model Selection

Use

You make a manual model selection in order to determine a specific forecast strategy. This is appropriate if the pattern generated with the automatic model does not lead to the desired result.

Prerequisites

If you want to select a model manually, you must first analyze the historical data to determine whether a distinct pattern or trend exists. You then define your forecast model accordingly.

Features

Here, we describe how you can react to different patterns using the forecast's features:

Constant pattern

If your historical data represents a constant pattern, you can select either the constant model or the constant model with adaptation of the smoothing factors. In both cases, the forecast is carried out using first-order exponential smoothing. When adapting the smoothing parameters, the system calculates different parameter combinations and then selects the optimum parameter combination. The optimum parameter combination is the one which results in the lowest mean absolute deviation.

You have another two options if the historical pattern is constant; either the moving average model or the weighted moving average model.

In the weighted moving average model, you weight individual historical values so that the system does not give equal value to historical data when calculating the forecast values. By doing this, you can influence the calculation so that recent historical values play a greater role in the forecast than older ones—as is also the case with exponential smoothing.

Trend pattern

If your historical data represents a trend, you should select either the trend model or a second-order exponential smoothing model. In the trend model, the system calculates the forecast values using first-order exponential smoothing.

In the second-order exponential smoothing models, you can choose a model with or without parameter optimization.

Seasonal pattern

If your historical data represents a seasonal pattern, you can specify the seasonal model. The system calculates the forecast values for the seasonal model using first-order exponential smoothing.

Seasonal trend pattern

If your historical data represents a seasonal trend pattern, you can select a seasonal trend model. The system calculates the forecast values using first-order exponential smoothing.

Irregular pattern

If you cannot detect any of the above trends or patterns, and you still want the system to carry out a forecast, it is advisable to select either the moving average model or the weighted moving average model.

Forecast Models for Different Historical Patterns

Pattern	**Forecast model**
Constant	Constant model Constant model with smoothing factor adaptation Moving average model Weighted moving average model
Trend	Trend model (1st-order exponential smoothing)

	Exponential smoothing model (2nd-order, with and without parameter optimization)
Seasonal	Seasonal model (Winter's method)
Seasonal trend	Seasonal trend model (1st-order exponential smoothing model)
Irregular	No forecast Moving average model Weighted moving average model

See also:

Automatic Model Selection [Page 231]

Model Initialization [Page 233]

4.2.1.2.6.11.10.1.2 Automatic Model Selection

Use

If you do not want to specify a forecast model manually, you must instruct the system to make an automatic selection. With automatic selection, the system analyzes the historical data and then selects the most suitable model.

Prerequisites

The system requires a different number of historical values for different tests. For further information, see Model Initialization [Page 233].

Features

If you want to select a model manually, you must first analyze the historical data to determine whether a distinct pattern or trend exists. You then define your forecast model accordingly.

The following models are possible:

- Constant
- Trend
- Seasonal
- Seasonal trend

If the system cannot detect any regular pattern in the historical data, it automatically selects the constant model.

Model Selection Procedures

You choose between two procedures for automatic model selection:

Procedure 1

If you want the system to select the forecast model, you can choose between various statistical tests and test combinations which determine the model. The tests that are carried out depend on how much you know about the past data pattern.

Trend test

In the trend test, the system carries out a regression analysis on the historical values and finds out if there is any significant trend.

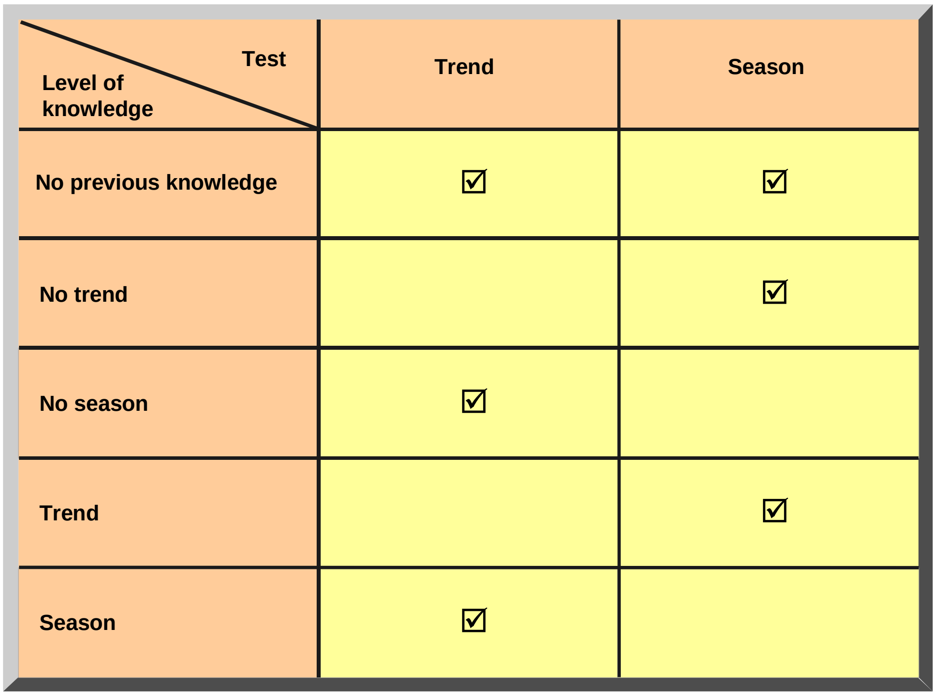
Seasonal test

In the seasonal test, the system ignores any possible trends in the historical values and then carries out an auto-correlation test.

Pattern	**Test**
No information	Test for trend and seasonal patterns
No trend	Test for seasonal pattern
No season	Test for trend pattern
Trend	Test for seasonal pattern
Season	Test for trend pattern

If you know that a particular pattern exists or does not exist before the model is selected, you can have the system test the historical time series for this trend or seasonal pattern.

If you are unaware of the historical pattern, the system carries out a trend test and a seasonal test. The forecast model is determined on the basis of which test produces the more significant results.



Test Level of knowledge	Trend	Season
No previous knowledge	☑	☑
No trend		☑
No season	☑	
Trend		☑
Season	☑	

Procedure 2

The system calculates the models to be tested using various combinations for alpha, beta, and gamma. The smoothing factors are also varied between 0.1 and 0.5 in intervals of 0.1.

The system then chooses the model which has the lowest mean absolute deviation (MAD). Procedure 2 is more precise than procedure 1, but takes much longer.

To work with procedure 2, choose the forecast strategy *Automatic Model Selection with Process 2* when you create a forecast function.

See also:

Manual Model Selection [Page 229]

4.2.1.2.6.11.10.1.3 Model Initialization

Use

Model initialization is the process of determining the necessary model parameters—such as the basic value, the trend value, and the seasonal indices —for the selected forecast model. Initialization takes place each time a planning time series is forecasted.

Features

The following table shows you which model parameters are necessary for each forecast model.

Model	**Model parameters**
Constant model	Basic value
Trend model	Basic value, trend value
Seasonal model	Basic value, seasonal indices
Seasonal trend model	Basic value, trend value, seasonal indices

As a general rule, the forecast model is initialized automatically. In order to do this, the system requires a certain number of historical values. This number depends on the forecast model, as shown in the following table.

Model	**Number of historical values**
Constant model	1
Trend model	3
Seasonal model	1 season
Seasonal trend model	1 season + 3
2nd-order exponential smoothing	3
Moving average	1
Weighted moving average	1

The system determines the basic value on the basis of the average and determines the trend using the regression analysis. The seasonal indices are given by the actual historical value divided by the basic value adjusted for the trend value.

These calculation methods are used for the constant, trend, seasonal, and seasonal trend models, depending on which parameters are to be determined.

A regression analysis is carried out for the second-order exponential smoothing model.

For the moving average and weighted moving average models, the system calculates an average value.

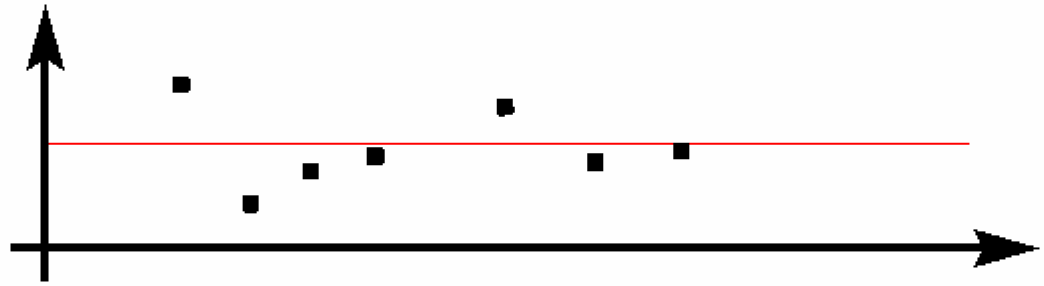
See also:

Forecast Parameters [Page 235]

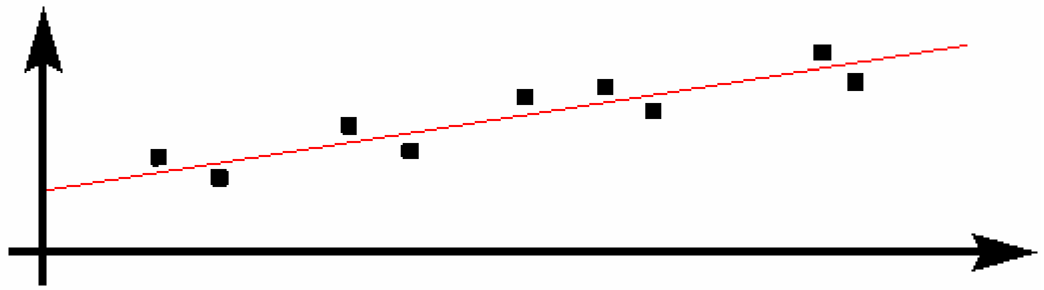
4.2.1.2.6.11.10.2 Forecast Models

When analyzing a time series various patterns can be recognized. The following different forecast models are derived from these patterns:

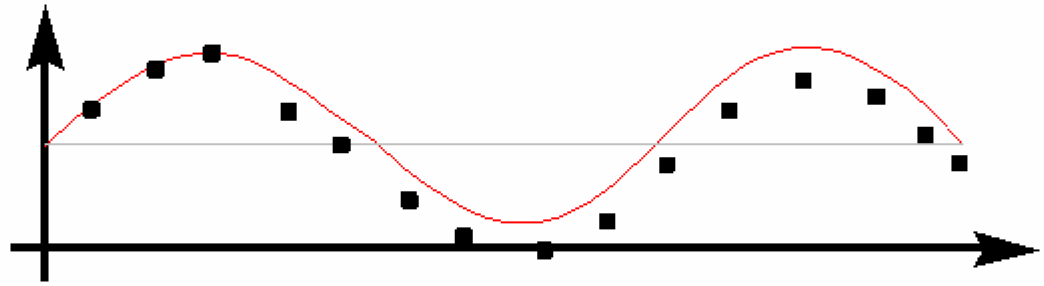
- Constant – There is a constant time series gradient when the time series varies statistically around a mean value:



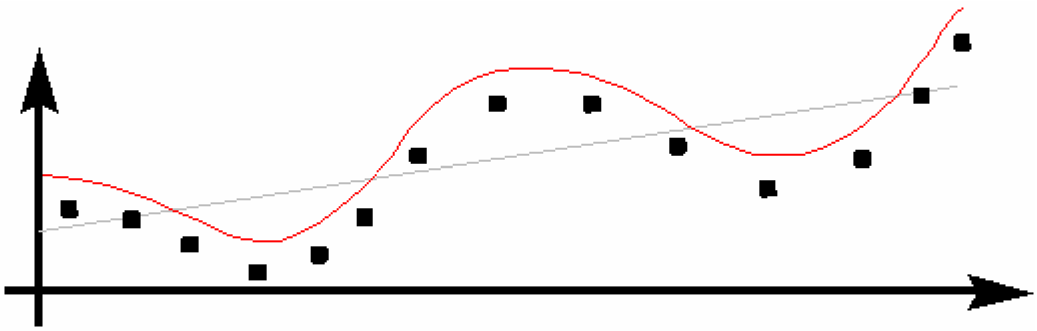
- Trend – With a time series gradient of type trend the time series value falls or rises constantly over a long period of time with only occasional deviations:



- Season – With a seasonal time series gradient, recurring variances from a base value occur at periodic intervals:



- Trend-seasonal – With a trend-seasonal time series gradient, seasonal variances occur from a continually increasing mean:



- Copying actual data (no forecast is executed) – copies the historic data updated from the operational application. You can then edit this as required.
- Irregular – No pattern can be detected in a series of consumption values.

4.2.1.2.6.11.10.3 Forecast Parameters

Maintenance of the following parameters is optional or mandatory, depending on which model you select. The following table shows which parameters are required for which model.

Forecast parameters Dependent on the Forecast Model

Specified model/model to be tested	Mandatory parameters
Constant models:	
First-order exponential smoothing model	Alpha factor
First-order exponential smoothing model with optimization of the smoothing factors	(none)
Moving average model	Number of historical values
Weighted moving average model	Weighting group
Trend models:	
First-order exponential smoothing model	Alpha and beta factors
Second-order exponential smoothing model	Alpha factor
Second-order exponential smoothing model with optimization of the smoothing factors	
Seasonal model: Winters' method	Alpha and gamma factors Periods per season
Seasonal trend model: First-order exponential smoothing model	Alpha, beta, and gamma factors

	Periods per season

Weighting Group

You only have to maintain this field if you selected the forecast model "weighted moving average". This key specifies how many historical values are included in the forecast and how these values are weighted in the forecast calculation.

Periods per Seasonal Cycle

You only have to specify the number of periods per season if you have selected the seasonal model or if you want the system to carry out a seasonal test.

The system uses the following factors for exponential smoothing, depending on the model. Thus, for example, only the alpha and the delta factors are required for the constant model, whereas all the smoothing factors are required for the seasonal trend model.

Alpha Factor

The system uses the alpha factor to smooth the basic value. If you do not specify an alpha factor, the system will automatically use the predefined alpha factor 0.2.

Beta Factor

The system uses the beta factor to smooth the trend value. If you do not specify a beta factor, the system will automatically use the predefined beta factor 0.1.

Gamma Factor

The system uses the gamma factor to smooth the seasonal index. If you do not specify a gamma factor, the system will automatically use the predefined gamma factor 0.3.

Delta Factor

The system uses the delta factor 0.3 to smooth the absolute mean.

See also:

Model Initialization [Page 233]

4.2.1.2.6.11.10.4 Forecast Formulas

The statistical forecast is based on different formulas:

- Formulas for forecast models
- Formulas that are used to evaluate the forecast results

Formulas for Forecast Models

Moving Average Model

This model is used to exclude coincidental irregularities in the time series pattern. The average of the n last time series values is calculated. You can calculate the average from n values at any time according to formula (1).

Formula for the Moving Average

$$(1)\quad M(t) = \frac{V(t) + V(t-1) + \ldots + V(t-n+1)}{n}$$

$$(2)\quad M(t+1) = M(t) + \frac{V(t+1) - V(t-n+1)}{n}$$

Thus, the new average is calculated from the previous average value and the current value weighted with 1/n, minus the oldest value weighted with 1/n.

This procedure is only suitable for time series that are constant, that is, for time series with no trend-like or season-like patterns. Since all historical data is equally weighted with the factor 1/n, it takes precisely n periods for the forecast to adapt to a possible level change.

Weighted Moving Average Model

You can improve the moving average model by introducing weighting factors for the individual historical values. In the weighted moving average model, every historical value is weighted with the factor R. The sum of the weighting factors is 1 (see formulas (3) and (4) below).

Formula for the Weighted Moving Average

$$(3)\quad M(t) = \frac{V(t)}{n} + \frac{V(t-1)}{n} + \ldots + \frac{V(t-n+1)}{n}$$

$$(4)\quad M(t) = R(t) * V(t) + R(t-1) * V(t-1) + \ldots + R(t-n+1) * V(t-1)$$

M	**=**	**Average**
V	**=**	**Consumption value**
R	**=**	**Weighting factor**

If the time series to be forecasted contains trend-like variations, you will achieve better results by using the weighted moving average model rather than the moving average model. The weighted moving average model weighs recent data more heavily than older data when determining the average, provided you have selected the weighting factors accordingly. Therefore, the system is able to react more quickly to a change in level.

The accuracy of this model depends largely on your choice of weighting factors. If the time series pattern changes, you must also adapt the weighting factors.

First-Order Exponential Smoothing Model

The principles on which this model are based are:

- The older the time series values, the less important they become for the calculation of the forecast.
- The present forecast error is taken into account in subsequent forecasts.

Constant Model

The exponential smoothing constant model can be derived from the above two considerations (see formula (5) below). In this case, the formula is used to calculate the basic value. A simple transformation produces the basic formula for exponential smoothing (see formula (6) below).

Formulas for Exponential Smoothing

Determining the Basic Value

$$(5) \quad G(t) = G(t-1) + \alpha\,(V(t) - G(t-1))$$

$$(6) \quad G(t) = \alpha V(t) + (1 - \alpha)\,G(t-1)$$

$G(t)$ = Current basic value for current period (t)

$G(t-1)$ = Previous basic value from previous period

$V(t)$ = Actual requirement for current period (t)

α = Smoothing factor for the basic value

To determine the forecast value, all you need is the preceding forecast value, the last historical value, and the "alpha" smoothing factor. This smoothing factor weights the more recent historical values more than the less recent ones, so they have a greater influence on the forecast.

How quickly the forecast reacts to a change in pattern depends on the smoothing factor. If you choose **0** for alpha, the new average will be equal to the old one. In this case, the basic value calculated previously remains; that is, the forecast does not react to current data. If you choose **1** for the alpha value, the new average will equal the last value in the time series.

The most common values for alpha lie, therefore, between **0.1** and **0.5**. For example, an alpha value of **0.5** weights historical values as follows:

1st historical value: 50%

2nd historical value: 25%

3rd historical value: 12.5%

4th historical value: 6.25%

The weightings of historical data can be changed by a single parameter. Therefore, it is relatively easy to respond to changes in the time series.

The constant model of first-order exponential smoothing derived above can be applied to time series that do not have trend-like patterns or seasonal variations.

General Formula for First-Order Exponential Smoothing

Using the basic formula derived above (6), the general formula for first-order exponential smoothing (7) is determined by taking both trend and seasonal variations into account. Here, the basic value, the trend value, and the seasonal index are calculated as shown in formulas (8) - (10).

Formulas for First-Order Exponential Smoothing

Forecast value for next period (t+i)

$$(7)\quad P(t+i) = (G(t) + i * T(t)) * S(t-L+i)$$

where:

Basic value:

$$(8)\quad G(t) = G(t-1) + T(t-1) + \alpha \left[\frac{V(t)}{S(t)} - G(t-1) - T(t-1) \right]$$

Trend value:

$$(9)\quad T(t) = T(t-1) + \beta \left[G(t) - (G(t-1) - T(t-1)) \right]$$

Seasonal index:

$$(10)\quad S(t+L) = S(t) + \gamma \left[\frac{V(t)}{G(t)} - S(t) \right]$$

For the constant model

$T(t) = 0, \beta = 0, S(t) = 1.0, \gamma = \text{Gamma} = 0$

For the trend model

$S(t) = 1.0, \gamma = \text{Gamma} = 0$

For the seasonal model

$T_{(t)} = 0, \beta = 0$

P(t+i) = The forecast calculated for the period (t+i) in the current period (t)

i = Forecast horizon

G(t) = The current basic value for the current period (t)

G(t-1) = The previous basic value from the previous period

L = Period length (frequently 12)

$V(t)$ = Actual requirement for the current period (t)

$T(t)$ = The current trend value calculated from the current period

$T(t-1)$ = Previous trend value from the previous period

$S(t)$ = Seasonal index for the period (t)

$S(t-L)$ = Previous seasonal index for the period (t)

α = Smoothing factor for the basic value 'G', $0 < \alpha < 1$

β = Smoothing factor for the trend value 'T', $0 < \beta < 1$

γ = Smoothing factor for the seasonal indices 'S', $0 < \gamma < 1$

Second-Order Exponential Smoothing Model

If, over several periods, a time series shows a change in the average value which corresponds to the trend model, the forecast values always lag behind the actual values by one or several periods in the first-order exponential smoothing procedure. You can achieve a more efficient adjustment of the forecast to the actual values pattern by using second-order exponential smoothing.

The second-order exponential smoothing model is based on a linear trend and consists of two equations (see formula (11)). The first equation corresponds to that of first-order exponential smoothing except for the bracketed indices. In the second equation, the values calculated in the first equation are used as initial values and are smoothed again.

Formulas for Second-Order Exponential Smoothing

Seasonal index:

$$(11) \quad \overset{(1)}{G}(t) = \alpha\, V(t-1) + (1-\alpha)\, \overset{(1)}{G}(t-1)$$

$$\overset{(2)}{G}(t) = \alpha\, \overset{(1)}{G}(t) + (1-\alpha)\, \overset{(2)}{G}(t-1)$$

$G^{(1)}$ = Simply-smoothed basic value

$G^{(2)}$ = Doubly-smoothed basic value

V = Historical value

α = Smoothing factor

Forecast Evaluation Criteria

Every forecast should provide some kind of basis for a decision. The system calculates the following parameters for evaluating a forecast's quality:

- Error total
- Mean absolute deviation (MAD)
- Tracking signal
- Theil coefficient

The system calculates the parameter named for evaluating a forecast's quality when executing a forecast with automatic model selection. The values calculated for the parameter are only used for internal processing and can be neither displayed nor changed.

Error Total

$$(12) \qquad ET = \sum_{t-1}^{n} \left[V(t) - P(t) \right]$$

Mean Absolute Deviation for Forecast Initialization

$$(13) \qquad MAD = \frac{1}{n} \sum_{t-1}^{n} \left| P(t) - V(t) \right|$$

$$n = \text{Number of periods for initialization}$$

Mean Absolute Deviation for Ex-Post Forecast

$$(14) \qquad MAD(t) = (1 - \delta) * MAD(t-1) + \delta * \left| V(t) - P(t) \right|$$

Tracking Signal

$$(15) \qquad TS = \left| \frac{ET}{MAD} \right|$$

Theil Coefficient

$$(16) \qquad U = \frac{\underset{(t=1)}{\overset{(T)}{S}} (P_{(t)} - V_{(t)})^2}{\underset{(t=1)}{\overset{(T)}{S}} (V_{(t)} - V_{(t-1)})^2}$$

4.2.1.2.6.11.11 Forecast Exit

Purpose

The forecast function of type exit in BW-BPS makes the same functionality available to you as the standard functions for forecast calculation newly implemented in BW-BPS (see Forecasting [Page 216]).

The forecast exit is available as of SEM 3.1B SP10 or SEM 3.2 SP6.

See also:

Exit Function for Forecasting [Page 243]

Forecast Parameters [Page 244]

Exit Function for the Log Display [Page 250]

4.2.1.2.6.11.11.1 Exit Function for Forecasting

Use

The forecast exit supports a range of **methods** that can be categorized as follows:

- Moving average (simple total average, moving average of specific order, weighted moving average)
- Exponential smoothing (simple, linear, seasonal, trend-seasonal exponential smoothing)
- Linear regression

You select a method in which to determine a forecast strategy.

There are more forecast strategies (see Obligatory Forecast Parameters [Page 245] under forecast strategy 11FCSTR).

Features

The forecast exit forecast strategies also offer the following functionality:

- Automatic model selection [Page 221]
- Outlier correction [Page 223]
- Trend dampening [Page 224]

  Parameter 17TDAMP (*trend dampening factor*) controls trend dampening (see Optional Forecast Parameters [Page 247]).
- For exponential smoothing:
  - Optimization of smoothing factors for exponential smoothing [Page 222]

    You can select a particular error measure using the *optimization variable* parameter (22OPTVAR). See Optional Forecast Parameters [Page 247].
  - Model initialization [Page 226]

Activities

1. Create a planning function of type `Exit` and enter the names of the following function modules:

   Function modules for exit function

Field	Technical name
Function module	UPF_FORECAST_EXIT_EXECUTE

Function module for init.	UPF_FORECAST_EXIT_INIT

2. Determine the parameters. You can find more information on parameters under Forecast Parameters [Page 244], Obligatory Forecast Parameters [Page 245] and Optional Forecast Parameters [Page 247].

   You can find more information on creating an exit function under Exit Function [Page 257].

See also:

Forecast Parameters [Page 244]

Exit Function for the Log Display [Page 250]

4.2.1.2.6.11.11.2 Forecast Parameters

In general the parameter fields of a planning function control how the function handles data. In the case of forecasting, a large number of parameters are available.

A forecast strategy (11FCSTR) and any related parameters, and a time variable (TMVAR) that defines which periods of the past are to be used for the observed values, are obligatory parameters.

All other parameters are either always optional, or are only obligatory when used in connection with particular forecast strategies. You can leave out any optional parameters that you do not need.

The following table provides an overview of all available parameters and data elements. (The initial numbers are used to display the parameters in a suitable order.)

Parameters and data elements

Parameter	**Data element**	**Description**
11FCSTR	UPF_Y_FC_STR	Forecast strategy (ID)
12PERIO	PERIO	Periods per season
13OUTL	UPF_Y_OUTLIER_CORRECTION	Outlier correction
14SIGFAC	UPF_Y_SIGMA_FACTOR	Sigma factor for outlier correction
15MAORD	UPF_Y_MA_ORDER	Order of moving average (length of interval)
16GEWGR	GEWGR	Weighting group
17TDAMP	UPF_Y_TREND_DAMPENING	Trend dampening factor
18ALPHA	UPF_E_ALPHA	Smoothing factor *alpha* for the base value
19BETA	UPF_E_BETA	Smoothing factor *beta* for the trend value
20GAMMA	UPF_E_GAMMA	Smoothing factor *gamma* for the seasonal component

21OPT	UPF_Y_FC_OPT	Optimization of smoothing parameters
22OPTVAR	UPF_Y_FC_OPT_VARIABLE	Optimization variable (for example, MSE, MAPE)
23ALPH_F	UPF_Y_ALPHA_FROM	Lower limit for *alpha* optimization
24ALPH_T	UPF_Y_ALPHA_TO	Upper limit for *alpha* optimization
25ALPH_S	UPF_Y_ALPHA_STEPSIZE	Step size for *alpha* optimization
26BETA_F	UPF_Y_BETA_FROM	Lower limit for *beta* optimization
27BETA_T	UPF_Y_BETA_TO	Upper limit for *beta* optimization
28BETA_S	UPF_Y_BETA_STEPSIZE	Step size for *beta* optimization
29GAMM_F	UPF_Y_GAMMA_FROM	Lower limit for *gamma* optimization
30GAMM_T	UPF_Y_GAMMA_TO	Upper limit for *gamma* optimization
31GAMM_S	UPF_Y_GAMMA_STEPSIZE	Step size for *gamma* optimization
TMVAR	UPC_Y_VARIABLE	Time variable for defining the past periods used for the observed data
KYFNM<nn>	UPC_Y_KYFNM	Forecast key figure, for example, z.B. KYFNM01 = 0AMOUNT
<IOBJNM>	UPC_Y_CHAVLINT<better appropriate data element for IO value>	Reference characteristic value, for example, 0VERSION = "ACTUAL"
REFRATIO	UPC_Y_KYFNM	Reference key figure (IO name)
CHANM<nn>	UPC_Y_CHANM	Name of characteristic (for log only), for example, CHANM01 = 0MATERIAL
DISREG	UPF_Y_DISREG	Skip zero values

For more information on obligatory parameters, see Obligatory Forecast Parameters [Page 245]; for more information on parameters that are either always optional, or that are obligatory only when used in connection with a forecast strategy, see Optional Forecast Parameters [Page 247].

See also:

Exit Function for Forecasting [Page 243]

4.2.1.2.6.11.11.2.1 Obligatory Forecast Parameters

The following parameters are always required:

Forecast strategy 11FCSTR

This forecast strategy determines how forecast values will be calculated. The following table provides an overview of the strategies available and which parameters have to be set in relation to which strategies.

Forecast strategies and dependent parameters

ID	Description	Dependent parameter
11	Average	
12	Moving average	Order of moving average 15MAORD
13	Weighted moving average	Weighting group 16GEWGR
21	Simple exponential smoothing (constant model)	Smoothing factor alpha 18ALPHA **or** Optimization of smoothing parameters 21OPT and optimization variable 22OPTVAR with, where necessary, search space limits 23ALPHA_F, 24ALPHA_T and initial step sizes for optimization 25ALPH_S.
22	Linear exponential smoothing (trend model)	Trend dampening factor 17TDAMP; Smoothing factors alpha 18ALPHA, beta 19BETA **or** Optimization of smoothing parameters 21OPT and optimization variable 22OPTVAR with, where necessary, search space limits 23ALPHA_F, 24ALPHA_T, 26BETA_F, 27BETA_T, and initial step sizes for optimization 25ALPH_S, 28BETA_S.
23	Seasonal exponential smoothing	Periods per season 12PERIO; Smoothing factor alpha 18ALPHA, gamma 20GAMMA **or** Optimization of smoothing parameters 21OPT and optimization variable 22OPTVAR with, where necessary, search space limits 23ALPHA_F, 24ALPHA_T, 29GAMM_F, 30GAMM_T , and initial step sizes for optimization 25ALPH_S, 31GAMM_S .
24	Trend-seasonal exponential smoothing	Periods per season 12PERIO; Trend dampening factor 17TDAMP; Smoothing factor alpha 18ALPHA, beta 19BETA, gamma 20GAMMA **or** Optimization of smoothing parameters 21OPT and optimization variable 22OPTVAR with, where necessary, search space limits 23ALPHA_F, 24ALPHA_T, 26BETA_F, 27BETA_T, 29GAMM_F, 30GAMM_T, and initial step sizes for optimization 25ALPH_S, 28BETA_S, 31GAMM_S.
31	Simple linear regression	Trend dampening, if necessary

41	Automatic model selection	Periods per season 12PERIO, if necessary; Trend dampening factor 17TDAMP

The parameters for outlier correction (*outlier correction* 13OUTL and *Sigma factor* 14SIGFAC) can be combined with any strategy.

Time variable TMVAR

Define a variable for the time characteristic (e.g. 0FISCPER) and specify which periods the past periods should be, and thereby, what the space of time for your observed values is. The observed values are the reference data.

See also:

Forecast Parameters [Page 244]

Optional Forecast Parameters [Page 247]

4.2.1.2.6.11.11.2.2 Optional Forecast Parameters

Periods per season 12PERIO (obligatory for seasonal models)

Number of periods in a season, for example, 12PERIO = 12 if your data is monthly observed values.

Outlier correction 13OUTL

Enter “X“ if you want the system to correct outlier values. For more information on outlier correction, see Outlier Correction [Page 223].

Sigma factor for outlier correction 14SIGFAC

If you have activated outlier correction using parameter 13OUTL, you can determine the deviation value as of which outlier correction is to be performed. Parameter 14SIGFAC is optional. 2.0 is used as the default value.

Order of moving average 15MAORD

For forecast strategy 12 (moving average) it is obligatory to enter an order for the moving average.

If no value is specified or if the value is less than 2, forecast strategy 11 (simple average) is applied automatically.

Weighting group 16GEWGR

For forecast strategy 13 (weighted moving average) it is obligatory to enter a weighting group.

Trend dampening factor 17TDAMP

You can determine a trend dampening factor to dampen the trend for forecast values. For more information on trend dampening, see Trend Dampening [Page 224].

Smoothing factor alpha 18ALPHA, beta 19BETA, gamma 20GAMMA

These parameters are used for exponential smoothing. These parameters are ignored if you have selected optimization (estimation of smoothing factors) or automatic model selection.

Alpha is used for all exponential smoothing models, beta only for those containing a trend component, and gamma only for those containing a seasonal component.

These parameters are optional. All smoothing factors have a default value of 0.3.

Optimization of smoothing factors 21OPT

Enter “X“ if you want the system to estimate smoothing factors. For more information on estimating smoothing factors, see Optimizing Smoothing Factors for Exponential Smoothing [Page 222].

Optimization variable 22OPTVAR

The optimization variable defines the error measure to be used for optimizing smoothing factors. The following values are possible:

1. MSE (mean squared error)
2. MAE (mean absolute error)
3. MAPE (mean absolute percentage error)
4. MAD_S (smoothed absolute deviation with exponentially decreasing weights for the residuals).

Error is understood as the difference between an observed value and a forecast value.

This parameter is optional. The default value is MSE.

Search space limits

To restrict the search space for alpha, beta, or gamma, you can set the appropriate parameters: 23ALPHA_F and 24ALPHA_T, 26BETA_F and 27BETA_T, 29GAMM_F and 30GAMM_T. The parameters that end with “F” (from) specify the lower limits. The parameters that end with “T” (to) specify the upper limits of the search space.

These parameters are optional. The default value for lower limits is 0.0. The default value for upper limits is 1.0.

Initial step size for optimization 25ALPH_S, 28BETA_S und 31GAMM_S

You can set initial step sizes for each smoothing factor in the optimization procedure. The search algorithm uses these step sizes in the first optimization phase in which a grid search is performed.

Therefore small step sizes mean better starting values for the direct search, but also mean increasing calculation costs. This is especially critical for models with trend and seasonal components, where all smoothing factors (alpha, beta, and gamma) are required.

These parameters are optional. The default value is 0.1 in each case.

Forecast key figures KYFNM<nn>

You can specify key figures for which you want to calculate forecast values. This parameter has to be set if any key figures included in your planning level are to be excluded from the forecast. You do not need to set this parameter if forecasting is desired for all key figures.

Enter KYFNM01 = 0AMOUNT to calculate forecast values for a key figure *amount*. You can enter additional forecast values, for example, KYFNM02 = 0QUANTITY for a key figure *quantity*.

Reference characteristic values

You can define characteristic values for the observed values (past periods).

Your planning data is differentiated from the observed values by means of a characteristic *version* 0VERSION. You want to calculate forecast values for planning on the basis of your observed values. Define 0VERSION as a parameter and enter “ACTUAL” or something similar.

As parameter names are limited to eight characters, the characteristic name may have to be shortened. For this to work, the first eight characters of the characteristic name have to be unique.

Reference key figure REFRATIO

If the observed values are to be taken from another key figure, specify the name of this key figure in the parameter REFRATIO.

The forecast variable is key figure 0KEYF1. The observed values are to be taken from key figure 0KEYF2. You have to set REFRATIO = “0KEYF2” to define this reference.

Characteristic name for the log display

In order to be able to display characteristics in the log display that describe or identify the objects for which the forecast is performed, you can enter the appropriate parameters for characteristic names (CHANM<nn>).

You want to create forecast data for every product in a region that is identified by the characteristics 0REGION and 0PRODUCT. Create parameters CHANM01 and CHANM02 and assign the values of characteristics “0REGION” and “0PRODUCT” to them.

Skip zero values DISREG

Enter “X“ if you want the system to disregard zero values in the observed values.

4.2.1.2.6.11.11.3 Exit Function for The Log Display

Use

The *log display* function is a planning function of type ”exit“. It displays error messages and statistical information for the forecasting run. The log information is displayed in a dialog box. You can also save the log information in the application log (see below, under the “activities” section, step 2 for parameter SAVELOG).

Integration

This function has to be executed immediately after executing the forecast function because the log information is only stored in the main memory temporarily. (This is not the case if you save the log information in the application log).

Features

The log display provides the following information:

- Determination coefficient R squared (r^2) as the measure of how well the model fits the data
- Standard deviation of errors
- MAPE (*Mean Absolute Percentage Error*, see Optional Forecast Parameters [Page 247] under optimization variable 22OPTVAR)
- RMSE (*Square Root of the Mean Squared Error* MSE)
- Selected model (when using the automatic model selection)
- Smoothing factors alpha, beta, gamma (entered values or estimated values)
- Trend dampening factor (if specified)
- Number of outliers (determined in outlier correction)
- Sigma factor (if used for outlier correction)

Activities

1. Create a planning function of type “exit” and enter the names of the following function modules:

Function modules for exit function

Field	Technical name
Function module	UPF_FORECAST_LOG_DISPLAY
Function module for init.	UPF_FORECAST_LOG_DISPLAY_INIT

2. Determine the parameter.

Parameter and data element

Parameter	Data element	Description
SAVELOG	(Character field of length 1)	Saving log information.

If you want the system to save the log information persistently instead of displaying it immediately, set SAVELOG to ‘X’. The information will then be stored in the application log and will not be displayed in the dialog box.

You can use SAVELOG in batch mode or in planning sequences.

This parameter is optional. The default value is ’ ’. As standard, the log information is displayed in a dialog box.

You have to specify an arbitrary data element of type ‘char’, with a minimum length of one character.

You can find more information on creating an exit function under Exit Function [Page 257].

See also:

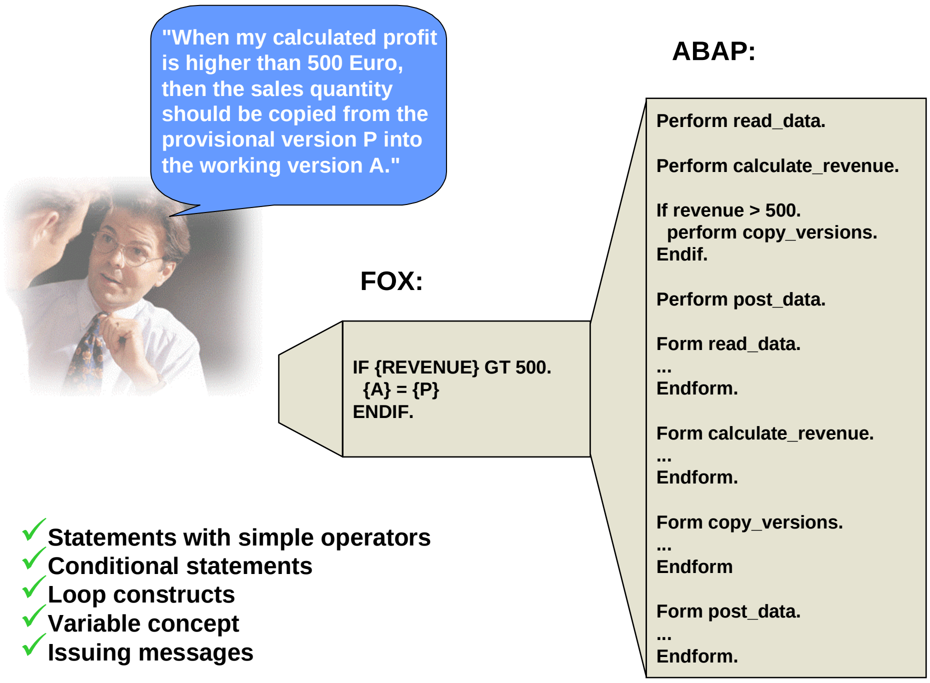
Exit Function for Forecasting [Page 243]

4.2.1.2.6.12 Formula

Use

You can define formulas which define how the transaction data is to be processed in order to generate plan data. Formula functions enable you to use extended mathematical functions to calculate plan data.

In addition to different calculation functions, which you can use for value assignment in formulas, there is also the possibility to model complex flow structures with the formula language FOX. In this way you get closer to the possibilities that you have when using exit functions. However, the complexity of the programming is considerably less for formulas than for ABAP statements which you have to program when using user exits. The following graphic illustrates this difference by comparing the solution of a business planning task with the methods of FOX and of ABAP:



"When my calculated profit is higher than 500 Euro, then the sales quantity should be copied from the provisional version P into the working version A."

FOX:

```
IF {REVENUE} GT 500.
  {A} = {P}
ENDIF.
```

ABAP:

```
Perform read_data.

Perform calculate_revenue.

If revenue > 500.
  perform copy_versions.
Endif.

Perform post_data.

Form read_data.
...
Endform.

Form calculate_revenue.
...
Endform.

Form copy_versions.
...
Endform

Form post_data.
...
Endform.
```

✓Statements with simple operators
✓Conditional statements
✓Loop constructs
✓Variable concept
✓Issuing messages

Integration

If, in the source code of a formula, you create connections between data records (for example, “increase planned profit for region EMEA by 10%, when the profit made in region Asia exceeds 5 million Dollars; otherwise by 5%”), you need to know about data transfer to planning functions [Page 203] in BW-BPS.

Features

When executing a formula function, the values of key figures in a planning level are always changed.

You have to specify those fields that are to be changed during the calculation as **fields to be changed**. These fields are for planning level characteristics and the *Key Figure Name* field. Field selection allows you to define whether the transformation described by the formula should take place for all key figures in the same way, or only for certain key figures.

The following different types of formula calculation exist:

- For the fields to be changed, select characteristics only and not the generic entry *key figure name*. The transformation is performed for all the key figures of a planning level. It is not necessary to formulate a separate equation for all key figures.

  If you select the characteristic *fiscal year*, the following formula, for example, can be formed: **`{ 2004 } = { 2003 } * 1.1.`**

  The calculation is performed for all key figures for the year 2004.

  Remarks on syntax: If characteristic values appear in operands, operands have to be enclosed in curly brackets { and } so as to avoid operands being confused with constants.

For the formula calculation, the values of the characteristic that you are using for addressing are not restricted to the values in the package.

This is how formula `Plan = 1.15 * actual`, for example, works even if the package is restricted to data of type *Plan*, data of type *Actual* is read automatically too.

If you select the field *Key Figure Name* for the conditions, you can restrict calculation to key figures of your own choice. The key figures that you explicitly assign a value to, in the definition of a parameter group in the formula editor, are then transformed.

- Include the generic entry *Key Figure Name* and one or more characteristics in the fields to be changed. In this case for every operand in the formula editor, you specify a combination of one value for every characteristic included in the list and if applicable the name of the key figure to be changed.

  If you select the field *Key Figure Name* as the field to be changed and any characteristic, for example, fiscal year, the following formula is, for example, formed: `{ REVENUE, 2004 } = { PRICE, 2003 } * { SQTY, 2003 } * 1.1.`

  The key figure is only calculated for the year 2004. The data for 2003 that is consulted for the calculation does not have to be in the package or level.

- Select the generic entry *Key Figure Name* as the field to be changed for the formula function and select particular characteristics as the condition fields of a revaluation function. The transformation is only performed for data records for which the affected characteristic value combination exists. When creating a parameter group for the function, a separate formula editor is displayed for every one of these characteristics. You can enter the formula for the respective characteristic value combination in this formula editor.

  If you only select the *Key Figure Name* as the field to be changed, the following formula, for example, is formed:

  ```
  REVENUE = PRICE * SQTY
  ```

  The formula is calculated for each individual record that meets the conditions.

You can find more information on how the selection of characteristics and key figures affect operand syntax in the formula editor under Formula Operands (Dependent on Key Figure and Characteristic Selection) [Page 255].

Mathematical Operators and Functions

You can use the following mathematical functions and operators to calculate a key figure:

- Single-figure operators

  +, **–** (as debit/credit signs)

- Two-figure operators

+, –, *, /, DIV (division in whole numbers), MOD (modulo division), ** (multiplying)

- Investment mathematical functions

  DECL (straight-line depreciation), DECD (declining-balance method of depreciation), DISC (discounting), PERP (perpetual bond)

- Other functions

  ABS, CEIL, FLOOR, FRAC, TRUNC, SIGN, MAX, MIN, SQRT, LOG, LOG10, EXP, COS, SIN,TAN, COSH, TANH, ASIN, ACOS, ATAN

You find further information (in particular about the number and meaning of function parameters) in the F1 help.

Processing Several Formulas in a Formula Function

All functions return a simple score (and no time series).

However in special cases, for example with depreciation functions, you want to observe the development of the net book value in chronological order. You can attain this by including both the generic entry *Key Figure Name* as well as a time-bearing characteristic, for example fiscal year, in the list of fields to be changed. The structure of the operands arises from this **{ <Key figure name>, FYEAR }**.

You can process a number of formulas in one step using your own formula function by entering the required formulas in succession in the formula editor.

For a complete straight-line depreciation of a acquisition worth 10,000 € with a yearly depreciation rate of 25% (corresponding to a duration of depreciation of 4 years), beginning in the year 2000, you enter the following formula in the formula editor:

```
{ DEPREC, 2003 } = DECL ( 10000, 0, 25, 1 )
{ DEPREC, 2004 } = DECL ( 10000, 0, 25, 2 )
{ DEPREC, 2005 } = DECL ( 10000, 0, 25, 3 )
{ DEPREC, 2006 } = DECL ( 10000, 0, 25, 4 )
```

As a result, every formula delivers the net book value of the acquisition after the specified number of years.

You can carry out calculations for numerous key figures simultaneously within one parameter group in the formula editor. Here interim results from prior formula steps within a parameter group are referred to for further calculations in this parameter group.

The following examples show how you can use key figures in the formula editor in different ways:

```
REVENUE = PRICE * SALES QUANTITY
ROCE = NOPAT / CAPEMPL
```

The key figures do not overlap – every key figure appears exactly once in the quantity of all formulas in the formula editor.

```
REVENUE = REVENUE * 1.1
ROCE = NOPAT / CAPEMPL
```

The key figure **REVENUE** appears on both sides of the assignment operator.

```
PRICE = PRODUCTION COSTS * 1.2
REVENUE = PRICE * SALES QUANTITY
```

The key figure **PRICE** appears in different formulas on different sides of the assignment operator; **PRICE** goes into the second formula with the value that resulted as the interim result from the calculation in the first formula. The following sample values should make this clearer for the different execution periods.

Key figure	Before execution	After 1st formula	After 2nd formula
PRICE	7	9,6	9,6
PRODUCTION COSTS	8	8	8
REVENUE	0	0	960
SALES QUANTITY	100	100	100

Reusing Parameter Groups

The parameter groups for a formula function can be quite comprehensive and execute complex operations. BW-BPS offers you the possibility to transfer the formula code of any other parameter group into a parameter group created by you. All parameter groups of formula functions existing in the system are available for this.

To transfer the code of another parameter group, proceed as follows:

1. Create a new parameter group for a formula function.
2. Choose *Copy Formula*.

   The system displays all existing parameter groups for formula functions in a list.
3. Select the parameter group which you want to transfer and confirm your selection.

   The system inserts the formula statements of the parameter group you selected into the formula editor of the new parameter group.

See also:

Exit Function [Page 257]

Data Transfer to Planning Functions [Page 203]

Planning Functions [Page 196]

You can find more information on using the formula enhancements **FOX** (FOrmula eXtension) in the F1 help for a planning function of type *Formula*.

4.2.1.2.6.12.1 Formula Operands (in Dependency on Key Figure and Characteristic Value)

Definition

The operands that you use as a formula element within a parameter group in the formula editor are either concerned with constants (for example, a surcharge value that is added to the value of the key figure) or with expressions that result from the name of a key figure, the value of a characteristic (or several characteristics), or a combination. An operand is, therefore, to be understood as a tuple of the available characteristics and key figures that you assign a value to with the formula.

Use

The syntactically correct form of an operand is directly dependent on which elements that you have included in the list of fields to be changed. Because you edit this list at planning function level, the resultant syntactic requirements for operands apply for all parameter groups that you create for a given formula function.

The following table shows the relationships:

Fields to Be Changed	Operand Form	Example	Explanation
only key figure	<key figure name>	REVENUE = PRICE * SQTY	a)
only characteristic	<characteristic value>	2004 = 2003 * 1.1	b)
several characteristics	{ <characteristic value>, <characteristic value>, ... }	{ 2004, EMEA } = { 2003, EMEA } * 1.1	c)
key figure and several characteristics	{ <key figure name>, <characteristic value>, <characteristic value>, ... }	{ REVENUE, 2004, EMEA } = { REVENUE, 2003, EMEA } * 1.1	d)

Explanations:

a) If you only include the generic key figure name entry in the list of fields to be changed, you can use constants as well as all the key figures in the planning level as operands in the formula editor.

b) If you only include a characteristic (and no key figures) in the list of fields to be changed, you can use constants as well as all the valid values of the selected characteristic as operands in the formula editor. The revision of key figure values carried out by the formula has the same effect on all key figures that are contained in the planning level (in the example given, all key figure values for the fiscal year 2004 are increased by 10% compared with the values from the previous year).

c) If you include several characteristic (but no key figures) in the list of fields to be changed, you can use constants and all of the combinations of valid values for the selected characteristic as operands in the formula editor. Since every value combination within the formula has the character of an individual operand, the values of every combination must be grouped together in braces and separated from each other by commas. As with b), the revision has the same effect on all key figures of the planning level (in the example given, all key figure values for the fiscal year 2004 and the region “Europe, Middle East, Africa” (EMEA) are increased by 10% compared with the values from the previous year).

d) If in addition to the case described in c) you also include key figures in the formula, you can use constants and all of the combinations of valid values for the selected characteristic, as well as the name of a key figure as operands in the formula editor. Since every value combination of characteristic values and key figure names within the formula has the character of an individual operand, the values of every combination and the key figure name must be grouped together in braces and separated from each other by commas.

Note that a key figure that you want to use as an operand must be inserted into the formula with its technical **name**, whereas for the characteristic you insert a **value** of the corresponding characteristic.

For the assignment of decimal fractions to an operand, use a point "." for the decimal hyphen. The comma "**,**" is reserved for separating the elements of an operand.

So that it is easier to enter syntactically correct operands, the system shows which operand elements are to be transferred and in which order this must happen, above the formula editor.

You can check your formulas for their accuracy with the help of *Syntax Checks* .

You can obtain more information about the mathematical operands that are available by positioning the text cursor anywhere in the formula editor and using the F1 help.

Likewise, you can obtain input help for entering an operand from anywhere in the formula editor by using the F4 function key. Permissible values are offered here for every operand element. When you confirm, the system inserts the syntactically correct operands into the formula rows.

See also:

Formula [Page 251]

4.2.1.2.6.13 Exit Function

Use

You can define your own planning functions of the type *Exit Function* to perform specific planning tasks that you cannot solve with any off the planning functions offered by BW-BPS. You will need to provide function modules to modify the transaction data. Exit functions offer you extensive control over every detail for calculating plan data, however they also require the most work as you must write your own ABAP programs.

BW-BPS exit functions are similar to the SAP customer exits [External] as they offer a defined interface to customer-specific enhancements. However they differ from the customer exits in the extent that the execution of the exit functions takes place within the planning environment (and not in a special transaction). In addition you can set up as many exit functions or function modules as you want, whereas BW-BPS only provides one branch to a customer exit.

Prerequisites

If you want to set up exit functions you require thorough knowledge of the ABAP programming language and the interface of the function modules that you call up using the exit function. You will find more information in the F1 help. You can call this when creating and editing an exit function for the field *Function Module.*

Features

The functionality of an exit function is only limited by the possibilities of the ABAP programming language. How well the function modules actually perform depends on the program logic you define.

SAP delivers sample function modules in the function group **UPFX** (development class **UPF**) in order to give you an impression about the possibilities of planning with exit functions. You can use this function module as reference material or as the basis for programming your own function module. Among other things, they contain valuable hints on a generic approach to a solution for dealing with the client-specific, technical names of the automatically generated planning objects. Problems can arise after a transport from Customizing into the production clients when a static approach is used.

When creating an exit function, you enter the technical name of the function module that should be executed when calling up the exit function in the field *Function Module*. Additionally, you can specify the name of a further function module that you wish to use for initialization purposes.

Using a function module for initialization is useful if you not only want to change data records with the exit function, but also want to create new ones. Initialization then makes sure that the key of the new data records to be created is transferred to the main function module via a parameter designated for this. This ensures that this function module is processed in any case, even if no data records exist yet.

Selection of characteristics

You can select fields to be changed from the list of characteristics that are contained in the planning level. This allows you to define which characteristics you can access within your function module in order to execute certain selections. Independent of what you select here, you can always access all the key figures that are contained in the planning level.

If you create references between data records (for example, "increase planned revenue for the region EMEA by 10% when the revenue of the Asia region exceeds 5 million dollars; otherwise by 5%") in the source code of function modules that are processed when the exit function is called up, you need information on how BW-BPS executes the data transfer to planning functions [Page 203] internally. Make sure that you can actually access the necessary data records at the time of execution.

Essentially you can change the values of key figures with an exit function but not those of characteristics. Nevertheless, if you require such functions (for example for reposting), you can achieve this by first copying the data record to be changed into a local work area, making the necessary changes, saving the changed data record, and deleting the original data record.

Definition of parameters

You can also create any number of parameter fields. You evaluate both the fields to be changed as well as the parameters within your function modules by creating suitable import parameters for the function module. You use these import parameters to transfer the fields to be changed and the parameters to the function module.

A further special feature applies in the case of parameter fields: You create a parameter field with a user-defined **name**. You supply the fields with the values in a parameter group to the exit function. These values are transferred to the assigned function module when executing the function. To enter values for the parameter fields, it is also necessary that you specify a **data element** for every parameter field of the exit function. This data element is required to make validations and input help available for the definition of parameter values within the parameter group to the exit function. For the input help for parameter values, the name of a table and a field that it contains can be specified.

Combination of exit functions with other functions

You can include exit functions in planning sequences [Page 267] in the same way as all other types of functions. In this way you have the option, for a particular planning task, of combining the delivered, user-defined functions with the function modules that you execute when calling up exit functions.

See also:

Formula [Page 251]

Data Transfer to Planning Functions [Page 203]

Planning Functions [Page 196]

4.2.1.2.6.14 Currency Translation

Use

You use this function to translate key figure values that represent sums of money into another currency. You require this function if you want to plan key figures, for example sales or revenue, for different companies of your enterprise that balance in different local currency. Translation into a unified currency (for example the group currency [External]) then means that the plan figures of the different companies can be compared with each other.

> You can find more information on currency translation in SAP BW under Currency Translation [External].

Prerequisites

In order to use this function, the necessary currencies, exchange rates, and currency translation keys must be maintained in your system.

Features

In the definition of a function for currency translation you define, in the list of **fields for conditions**, those characteristics according to which you want to break down the different currency translations that the function should carry out. You would normally select the characteristic *country* for this.

The function for currency translation always translates from the currency in which a key figure is currently conducted into the target currency that is determined by you. The function automatically determines all financial key figures from the planning level and makes them available for currency translation in the **parameter groups**. For the characteristic value combinations that result from the selected condition fields, you enter the type of currency translation that you have chosen. Here you have the following possibilities:

- *Key figure*
  You enter both the initial key figure (*from key figure*) that you want to translate, and the target key figure (*to key figure*), in which the translated value should be stored.

  The **value** of the initial key figure is not changed. However, if you specify the same key figure for both fields, the original values of the key figure will be overwritten by those of the target currency.

  If, within one parameter group, you translate different initial key figures into the same target key figure, then the values in the target key figure are added up.

  The **currency** of the initial key figure can always be determined: The currency is always determined automatically and does not have to be entered. Either the key figure has a fixed currency or the currency comes from a characteristic. If the target key figure does not have a fixed currency you can determine this (see *exchange type*, *target currency*, and *target currency characteristic*).

- *Exchange rate type*
  Here you enter which exchange rate (in relation to the date of translation) should be used for the currency translation, for example, average rate, current exchange rate, historical exchange rate.

- *Target currency*
  Here you determine the currency in which the key figure values are to be translated. Under the currencies you select the one which is available in your system directly.

- *Characteristic target currency*
  Here you can select a characteristic that is contained in the currency-bearing planning level, through which the target currency is determined. The data type of the characteristic is not user-defined but has to be able to incorporate a special currency key (Data type `CUKY` in the Data Dictionary).

  The key figure *sales* as well as the characteristics *country*, *company* and *group currency* are contained in a planning level. Through a compound characteristic with the characteristic *company,* the characteristic *local currency* is also contained. You restrict the group currency to US Dollar and the local currency to German Mark. In a planning package that is restricted to the country Germany, you enter sales figures that are posted in local currency (that is German Mark).

  Subsequently you want to translate the sales figures of all companies into the group currency US Dollar. For this you set up a parameter group for the currency translation function and define that the target currency should be determined using the characteristic group currency.

- *Translation date, current date, time base*
  Decide on one of the three possible ways to determine the exchange rate date that the currency translation should be based on:

  - *Translation date*: You set a certain date as fixed.

- *Current date*: The translation takes place at the exchange rate that is valid at the time of the translation function.
- *Time base*: You select one of the time bases that are offered by the system, this should be used in conjunction with one of the time-bearing characteristics for calculating the exchange rate date that is contained in the planning level.

  Which time base you can set up depends on which time-bearing characteristics you have included in the planning level:

  - In the case of the characteristic *fiscal year/period* you can choose between the time bases “start period” and “end period”.
  - In the case of the characteristic pair *period* and *fiscal year* you can choose between the time bases “start fiscal year” and “end fiscal year”.

  The three ways of determining the exchange rate date are mutually exclusive. If you make entries for more than one field, then the system rejects your entries.

See also:

Currency Translation: Notes on Modeling [Page 261]

Unit Conversion [Page 264]

Planning Functions [Page 196]

4.2.1.2.6.14.1 Currency Translation: Notes on Modeling

When you choose a planning function of type *Currency Translation* you are, to a large extent, free to include the type and number of characteristics and key figures that you want in currency translation. Some special features can occur here in relation to the SAP BW data model that we will explain in this section.

The SAP BW data model manages currency fields such as characteristics. As a result these fields are part of the logical key of a data record. The currency keys are in the currency fields for one or several key figures. As a result, you determine a part of the semantics and the formatting attributes (decimal places) for these key figures. Therefore there may be transaction data records that only differ in the currency fields.

The key figures can then, for example, on the one hand to be interpreted as values in the global company currency, on the other hand as values in group currency.

You can derive which currency is used in particular cases from characteristics such as *company*, *group*, *company code* or *profit center*. A currency key is assigned to these characteristics as an attribute.

Scenario

In group-wide planning scenarios, plan data is mostly entered in different currencies for subsidiaries. It is planned in the respective company currency decentrally. This regional data is transferred to the group headquarters, converted into the group currency and revised there.

It is often not sufficient when the central partner only works with plan data in group currency. On the contrary, as a rule access to the output value in company currency must be possible, so that the decentralized planner can respond to queries quickly and simply. For the following observations it is also assumed that the central planner must also still plan a quantity in addition to the key figures for amounts.

Solution Approaches

We present two different approaches below in order to model the planning scenario described with BW-BPS:

- ...

a. Only one key figure is used for amounts (*amount*). If required it is translated into group currency.

b. Two key figures (*company amount*, *group amount*) and two different characteristics (*company currency*, *group currency*) are used for the currency key.

The following tables show the transaction data records: There are two data records in each case. The second result data record results from a currency translation from the first record.

Case a)

Company	Company currency	Unit of measurement	Amount	Quantity
913	USD	ST	100,00	50,000
913	EUR	ST	96,00	

Case b)

Company	Company currency	Group	Group currency	Unit of measure ment	Amount company	Amount group	Quantity
913	USD			ST	100,00		50,00
913	USD		EUR	ST		96,00	

The following stands out in the results records:

- In both cases the *Unit of Measurement* field is filled, although the quantity is 0 in the results record.
- In case b) the *Group* field is not filled. Here two data records are also formed by the currency translation although there would actually be space in the first data record to include the amount.

Working with two different data records is unfavorable on one hand because it affects performance and on the other because a new currency translation can change the result.

The following tables show this with the data records in case a) after a second execution of the currency translation:

Case a) data records before a new currency translation

Company	Company currency	Unit of measurement	Amount	Quantity
913	USD	ST	100,00	50,000

913	EUR	ST	96,00	

Case a) data records after a new currency translation

Company	Company currency	Unit of measurement	Amount	Quantity
913	USD	ST	100,00	50,000
913	EUR	ST	192,00	

Both initial data records are converted into euro and the result is added up: 100 USD = 96 EUR + 96 EUR = 192 EUR. It follows from this that in case a) before a currency translation the target data must be consistently deleted.

Define a planning sequence [Page 267] that contains a delete function at the start. The planning sequence contains the function for currency translation as a second function. The delete function deletes the records that result from translation. The trick is to find those records that have resulted from the currency translation. As this is difficult, SAP recommends (as in case b) that you define currency translation between different key figures.

You can avoid the occurrence of two data records in case b) by defining characteristic relationships [Page 110]. If the first data record is entered, the system automatically adds the values for the *Group* and *Group Currency* fields to the characteristic relationship defined.

The following table shows the data record before and after unit conversion:

Case b) data record before currency translation with characteristic relationship

Company	Company currency	Group	Group currency	Unit of measurement	Amount company	Amount group	Quantity
913	USD	9	EUR	ST	100,00		50,00

Case b) data record after currency translation with characteristic relationship

Company	Company currency	Group	Group currency	Unit of measure ment	Amount company	Amount group	Quantity
913	USD	9	EUR	ST	100,00	96,00	50,00

This result is already closer to the expectations since here only **one** transaction data record is formed. As the fields are already correctly derived you do not need to execute a delete function before a new unit conversion.

In our example, we have also planned a quantity with a unit of measurement. If in the group in a planning session, you only plan the amount in group currency a transaction data record is formed in which the *Unit of Measurement* field is initial. For that reason, with planning functions with fields to be changed (for example, formulas) or those which work with reference data (for example, distribute with reference data), you must make sure that objects exist in the system which have logically different keys and if necessary behave differently during the selection.

See also:

Currency Translation [Page 259]

4.2.1.2.6.15 Unit Conversion

Use

You use this function to convert key figure values, that represent quantities, into another unit. Conversion into a common unit means that the balances of the different companies can be compared with each other.

> You require this function if you want to plan key figures such as, for example, stock for different companies of your enterprise that use your quantities in units common to different countries or industries. So, for example, fuel stocks can be executed in liters or barrels, weights can be specified in kilograms or Anglo-Saxon pounds and machine runtimes in seconds or minutes.

Prerequisites

In order to use this function, the necessary units and conversion keys must be maintained in your system. You make these settings in the SAP Implementation Guide (IMG) under *General Settings* → *Check Units of Measurement*. You can find more information on the general approach for the management of units of measurement in the SAP System under Concept [External].

Features

In the definition of a function for unit conversion, in the list of **fields for conditions** you define the characteristics according to which you want to break down the different unit conversions that the function should carry out. For example, you can make a distinction according to the characteristics *country* or *industry sector*.

The function for unit conversion always calculates from the unit in which the key figure is currently conducted into the target unit that is determined by you. During the conversion, it is rounded up or down to the nearest whole number. The function automatically determines all quantity key figures from the planning level and makes them available for unit conversion in the **parameter groups**. For the characteristic value combinations that result from the selected condition fields, you enter the type of unit conversion that you have chosen. Here you have the following possibilities:

- *Key figure*
  You enter both the initial key figure (*from key figure*) that you want to translate, and the target key figure (*to key figure*), in which the translated value should be stored.

  The **value** of the initial key figure is not changed. However, if you specify the same key figure for both fields, the original values of the key figure will be overwritten by those of the target unit.

  > If, within one parameter group, you translate different initial key figures into the same target key figure, then the values in the target key figure are added up.

  You can always define the **unit** of the initial key figure. Either the key figure has a fixed unit or the unit comes from a characteristic. If the key figure does not have a fixed unit you can determine this (see *exchange type* and *target unit characteristic*).

- *Conversion unit*
  Here you enter which unit the key figure should be converted into.

- *Characteristic target unit*
  Here you can select a characteristic that is contained in the unit-bearing planning level, through which the target unit is determined. The data type of the characteristic is not user-defined but has to be able to incorporate units (data type `CUKY` in the Data Dictionary).

  Defining the target unit with a *translation unit* or with the *target unit characteristic* is only allowed if a certain unit for the key figure is not already predetermined in the definition of the target key figure.

  You can only define the target unit with **one** of the methods that are described. If you predetermine a *translation unit* as well as a unit-bearing characteristic, the system rejects this entry.

  The target value is overwritten during the conversion.

See also:

Unit Conversion: Notes on Modeling [Page 265]

Currency Translation [Page 259]

Planning Functions [Page 196]

4.2.1.2.6.15.1 Unit Conversion: Notes on Modeling

When you choose a planning function of type *unit conversion* you are, to a large extent, free regarding the type and number of characteristics and key figures that you want to include in unit conversion. Some special features can occur here in relation to the SAP BW data model that we will explain in this section.

The SAP BW data model manages unit fields such as characteristics. As a result these fields are part of the logical key of a data record. The keys of units for one or several key figures are in the unit fields. As a result, you determine a part of the semantics and the formatting attributes (decimal places) for these key figures. Therefore there may be transaction data records that only differ in the unit fields.

The key figures can then, for example, be interpreted as the value in the local unit of measurement of a subsidiary in one case, and in the other case, as values in the valid unit standardized for the group.

You can derive which unit is used in particular cases from characteristics such as *company*, *company code* or *material*. A unit key is assigned to these characteristics as an attribute.

Scenario

In group-wide planning scenarios, plan data is mostly entered in different currencies for subsidiaries. However, decentralized it is planned in the local unit. This regional data is transferred to the group headquarters, converted into the group unit and revised there.

It is often not sufficient when the central partner only works with plan data in the group unit. On the contrary, as a rule access to the output value in the local unit must be possible, so that the decentralized planner can respond to queries quickly and simply. For the following observations it is also assumed that the central planner must also still plan an amount in addition to the key figures for quantities.

Solution Approaches

We present two different approaches below in order to model the planning scenario described with BW-BPS: As an example the quantities of a liquid are used which are entered at company level in canisters and at group level in pallets. It is based on the assumption that 100 canisters fit into 1 pallet. The approaches differ as follows:

- ...

c. Only one key figure is used for quantities (*quantitiy*). If required it is converted into group unit.

d. Two key figures (*company amount*, *group amount*) and two different characteristics (*company currency*, *group currency*) are used for the currency key.

The following tables show the transaction data records: There are two data records in each case. The second result data record results from a unit conversion from the first record.

Case a)

Company	Group currency	Company unit	Amount	Quantity
913	USD	Canister	50	6000
913	USD	Pallet		60

Case b)

Company	Company unit	Group	Group unit	Group currency	Amount	Quantity group	Quantity company
913	Canister			USD	50		6000
913	Canister		Pallet	USD		60	

The following stands out in the results records:

- In both cases the *unit of measurement* field is filled, although the quantity is 0 in the results record.
- In case b) the *group* field is not filled. Here two data records are also formed by the unit conversion although there would actually be space in the first data record to include the new calculated (group) quantity.

Working with two different data records is unfavorable on one hand because it affects performance and on the other because a new unit conversion can change the result.

The following tables show this with the data records in case a) after a second execution of the unit conversion:

Case a) Data records before a new unit conversion

Company	Company currency	Unit of measurement	Amount	Quantity
913	USD	Canister	50	6000
913	USD	Pallet		60

Case a) Data records after a new unit conversion

Company	Company currency	Unit of measurement	Amount	Quantity
913	USD	Canister	50	6000
913	USD	Pallet		120

Both initial data records are converted into pallets and the result is added up: 6000 canisters = 60 pallets + 60 pallets = 120 pallets. It follows from this that in the case a) the target data must be consistently deleted before unit conversion.

You can avoid the occurrence of two data records in case b) by defining characteristic relationships [Page 110]. If the first data record is entered, the system automatically adds the values for the *group* and *group unit* fields to the characteristic relationship defined.

The following table shows the data record before and after unit conversion:

Case b) Data record before unit conversion with characteristic relationship

Company	Company unit	Group	Group unit	Group currency	Amount	Quantity group	Quantity company
913	Canister	9	Pallet	USD	50		6000

Case b) Data record after currency translation with characteristic relationship

Company	Company unit	Group	Group unit	Group currency	Amount	Quantity group	Quantity company
913	Canister	9	Pallet	USD	50	60	6000

This result is already closer to the expectations since here only **one** transaction data record is formed. As the fields are already correctly derived you do not need to execute a delete function before a new unit conversion.

In our example, we have also planned an amount with a currency. If, in the group in a planning session, you only plan the quantity in group unit, a transaction data record is formed in which the unit field is initial. For that reason, with planning functions with fields to be changed (for example, formulas) or those which work with reference data (for example, distribute with reference data), you must make sure that objects exist in the system which have logically different keys and if necessary behave differently during the selection.

See also:

Unit Conversion [Page 264]

4.2.1.2.6.16 Planning Sequences

Use

A planning sequence is a list of planning functions and parameter groups that are processed in the order you have previously determined. You use this function to automate the sequential processing of a user-defined number of planning functions that you have defined. This is useful if you want to regularly carry out certain complex operations on your data. You can combine multiple function call-ups in a planning sequence and these are then processed as one single step.

The benefit of planning sequences becomes greater the more processing steps are required to complete a planning task. The unique definition of a planning sequence ensures that all required steps are carried out and that the correct sequence is observed.

> At the start of each quarter, you wish to plan the sales in the 'Pharmaceuticals' division for a group of subsidiaries. Here, various planning functions (such as copy, revaluate, and distribute) each supply partial results that are then used by the next function. You can combine this chain of functions in a single operation.

Integration

Planning sequences can be either of two types: local and global:

- You create a **local** planning sequence in the context of a planning area. In this way you combine a selection of functions, which you have defined, that belong to this planning level. You are not able to access functions that you have created in the context of other planning levels.
- You create a **global** planning sequence without reference to any specific planning area (*Goto → Global Planning Sequences*). Here, you include functions from multiple planning areas. You can also choose to have global planning sequences processed in the background. You can find out about the status and result of the sequences processed in the background with the functions *Tools → Log and Tools → Job Overview* In addition there is also the possibility to schedule global planning sequences in time with the tools of the Computing Center Management System.

> Manual planning cannot be assigned to a planning sequence as this function is interactive and requires user input. This is not compatible with the concept of processing planning steps automatically.

Prerequisites

In order to be able to define a planning sequence, you must have already defined functions in the same planning level and have created parameter groups for these functions. You must also ensure that all the functions and parameter groups combined in the sequence are compatible with the characteristic value restrictions that you made in the planning level and planning package. In the case of a global planning sequence, these prerequisites apply to the functions contained in the sequence, with reference in each case, to the planning level to which the functions belong.

> The success of sequential processing of plan data depends greatly on the make-up of the functions that are to be carried out. This can lead to surprising effects: It may happen that some functions in a sequence only work if certain characteristic values appear in the data being supplied. If this is not the case, some subsequent functions may not be carried out.

Features

For each processing step, you determine the desired planning function with an appropriate parameter group. There are no restrictions here – you can, for example, include the same combination of function and parameter group several times within one sequence. The processing of a sequence starts with the data status as it is in the BW-BPS data basis. Each further step receives the data as it is passed on from the previous step.

> If a function in the planning sequence cannot be carried out successfully, the sequence is terminated at this point and the system issues an appropriate error message. In this case the changes that were made in previous steps are retained. In a situation of this kind, record exactly which steps have been performed successfully and which have not. After you have corrected the problems you can restart at the correct place and process the steps that are incorrect.

Scheduling Global Planning Sequences in Time

You can schedule global planning sequences in time with the Computing Center Management System [External] tools. This opens up various advantages:

- The execution of planning sequences that process large quantities of data and therefore overload the system can be scheduled to run at times when the load on the system is less. You can, for example, also use these functions to schedule extensive planning sequences, so that they do not overlap with the regular data update of SAP BW from the operational systems
- You can group together periodic recurring planning tasks in a global planning sequence and schedule them accordingly. For example you could automatically schedule a regular sales forecast on the basis of updated actual data at the start of every period

To schedule a global planning sequence in time, proceed as follows:

1. In the ABAP Editor execute the report **UPC_BUNDLE_EXECUTE** that controls the processing of global planning sequences
2. In the selection screen of the report, select the global planning sequence that you want to process.
3. Save the current settings for the report as a variant.
4. Using *System* → *Services* → *Jobs* → *Define Job*, define a job with which you want to execute the report **UPC_BUNDLE_EXECUTE** with the variant already created at the desired point in time.

See also:

Planning Functions [Page 196]

4.2.1.2.7 Planning Folder

Definition

A planning folder is used to put together business-related planning levels, planning packages, planning layouts, and planning functions within a planning area that was defined before. The objects put together in this way are presented in a special user interface when executing the planning folder, and give the users the impression of a contained planning application, which is easy to operate.

Use

Planning folders make navigation and overview easier within a planning session when you want to plan with a limited number of layouts, packages and functions which are related from a point of view of content. This connection is established by common selection characteristics in all included packages, as well as by planning functions that process the data of an input layout to results, that are displayed again in output layouts.

For example, year-end planning normally refers to a balancing unit, a planning time frame, and a plan version (other versions or actual data are only required as reference data). The items of the year-end closing are planned in various differentiations within this global definition criteria: Sales revenue and sales costs according to market segments, overhead costs according to areas of responsibility, investments according to projects or investment programs. Planning functions derive other items from items, which were planned manually or were transferred from other subplans, and generate the plan closing on a summarized level. Numerous planning levels, planning packages, planning layouts, planning functions and parameter groups are required in such a scenario, which are, however, all part of a contained model. Visualization using two related layouts (data entry, and output or result layout) makes it easier to follow the effects of plan changes on the company's target figures.

Structure

The planning objects that you put together in a planning folder come from the quantity of objects that you create in the planning environment [Page 90]. With planning folders, you create another view of the quantity of previously created planning objects.

Planning folders are structured as follows:

Layouts

- *Input Layouts*: Here you insert layouts of manual planning that should be ready for input when executing the planning folder.
- *Output Layouts*: Here you insert layouts of manual planning that are not ready for input when executing the planning folder, but should only be used to display values. Alternatively, for the tabular display of values, you can determine that the data is prepared graphically for an output layout.

Functions

- You can determine individual planning functions [Page 196] or parameter groups for input and output layouts that are to be available when executing the folder to change data.
- In addition to input and output layouts, you can include layout-independent *Global Planning Functions* in the planning folder. These functions are constantly available when executing a planning folder, independent of which input and output layouts are active.
- If you include global planning sequences [Page 267] in a planning folder, you can also transfer data from other planning areas into the planning area that the folder is assigned to.

When you execute the folder the system offers, by default, a pushbutton for each function with which you can execute the corresponding function manually.

Alternatively you can determine that a function is to be executed automatically, that is, dependent on particular system events being entered. You can find further information under:

- Automatic Execution of Planning Functions [Page 279]
- Automatic Execution of Global Functions [Page 282]

Variables

You can include variables [Page 102] in the planning folder for both input and output layouts. This means that the variables with their current value are displayed at runtime, so that you can recognize at a glance for which characteristic value combination the data displayed is valid. Including variables in a planning folder can even be absolutely necessary when variables are contained in the layouts used for which further restrictions must be made at runtime.

Documents

Documents either form part of a layout or are included in the planning folder as a list of documents (see Documents in Planning Folders [Page 283]).

Integration

To edit or execute planning folders, choose *Business Planning and Simulation* → *Planning Folders* → *Customizing* (transaction code UPSPM) or *Execute* (transaction code UPSPL) from the *SAP Easy Access Menu* screen.

When processing planning folders, the planning objects are always offered which are contained in the planning profile [Page 123] set up for you in the planning environment. To have access to all available planning objects, you must first choose *Edit* → *General Overview* in the planning environment.

See also:

Editing a Planning Folder [Page 271]

Editing a Planning Folder: Details [Page 274]

4.2.1.2.7.1 Editing a Planning Folder

Use

To make settings that affect a planning folder as a whole, use the functions described here. The section Editing a Planning Folder: Details [Page 274] shows you how to proceed to include planning objects in a planning folder and to edit in it.

Integration

The functions described here partly use general services of the SAP System, for example, transport and translation functions.

Features

Generating a Web interface for a planning folder

You can generate a Web interface [External] that is used as an alternative representation of the objects contained in the folder for a planning folder that you have declared as Web-enabled on creation. You then generate an executable Web application in the Web Interface Builder [Page 285] that is based on the Web interface generated in this way.

Proceed as follows to generate a Web application from a planning folder:

1. Choose *Goto* → *Generate Web Interface*.

   The *Generate Web Interface from Planning Folder* screen appears.

2. Adopt or modify the technical name that the system proposes for the Web interface.

3. Adopt or modify the package [External] that the system proposes for the Web interface.

   The system always proposes the reserved name **`$TMP`** as the package. The Web interface to be generated is only available in the current BW system if you retain this setting. If you also want to use the Web interface in other systems then you must enter the name of an existing package here. This means that the system creates a transport request for the interface. You can import this into the desired target system.

4. Choose *Execute* to generate the Web interface.

Also take note of the *Note on Export* that you can look at in the system using the pushbutton with the same name.

Displaying a legend

To provide an overview of which symbols are used in the functions for editing planning folders, choose *Display Legend*.

Standard planning folder

You can select one planning folder from the quantity available in the system and mark it as your personal standard planning folder using *Planning Folder* → *Set as Standard*. This means that the folder marked in this way is already preselected in the list of available planning folders. In addition, such a folder can be automatically executed using a special transaction. You remove the selection with *Edit* → *Reset Standard Folder*.

Copying a planning folder

You can copy a planning folder using *Copy Planning Function*. Here, all objects contained in the folder to be copied, are also included in the new planning folder. You can also use these functions in particular to change one of the unchangeable basic settings of a folder in an indirect way. By copying one of the folders you can, for example, derive a non Web-enabled folder from a Web-enabled folder and the other way around.

Deleting a planning folder

If you no longer need a planning folder you can delete it using 🗑 *Delete*. The planning objects combined in the folder are of course not deleted. The system first only sees the deletion as a deletion flag. The folder still remains visible in the list and is indicated with a deletion icon 🗑. Only when you leave the folder overview does the system ask whether you actually want to delete the indicated folder.

Searching for a planning folder

If you have created lots of planning folders it can be difficult to find a specific folder. In this case you use the search function 🔍. The system searches for the text that you enter as a search term both in the technical name and in the description of the planning folder, and marks the first folder in which a match is found.

Structuring planning folders

You can group together a quantity of planning folders that belong together in terms of their content in the list of planning folders for a folder. First create the folder. Here you proceed as when creating a new planning folder and choose the entry *Folder (Not Executable)* for *Attributes when Executing*. Then you can assign existing planning folders to the folder using drag and drop and create new planning folders (or also other folders) in it.

Planning folder with variant

You can create variants for a planning folder. In this way, you have the option of defining the execution of certain planning folders as independent menu options in the SAP Workplace. To assign a variant to a planning folder, choose *Goto* → *Create variant*.

Translating a planning folder

If you want to use a planning folder in different regional organizations in your company then it is necessary that the interface texts contained in the folder are translated into the respective language of the country. Select the required folder and choose *Extras* → *Translate Folder*. You get to the *Translation of Planning Folders* screen. The system presets the required folder. When you have entered the desired target language you get to the SAP System translation function where you can translate the texts.

On the *Translation of Planning Folders* screen you can also choose the option *All Folders of the Planning Application* or select a set of planning folders using the option *Planning Folder* with ⇨ *Multiple Selection*.

Transporting a planning folder

In order to use a planning folder in other systems or clients, you must transport it there first. Select the required folder and choose *Extras* → *Transport Folder*. After you have entered or created a Customizing transport request, the system adds all necessary objects to the request.

4.2.1.2.7.2 Editing a Planning Folder: Details

Use

You edit planning folders in order to put together planning objects that should be available for users when executing the planning folder. With a suitable selection of planning objects you can ensure that the users are led to the processing of a particular planning task in a target-oriented way, without having to worry about the extensive setting possibilities of the planning environment [Page 90].

Integration

When processing planning folders, the planning objects are offered that are contained in the planning profile [Page 123] set up for you in the planning environment. To have access to all available planning objects, you must first choose *Edit* → *General Overview* in the planning environment.

Prerequisites

You can only include those planning objects in planning folders that have already been created in the planning environment.

Features

Assigning planning layouts

In a planning folder you make one or more layouts of manual planning available with which the data can be displayed, and if required, also changed manually. If you assign more than one planning layout in the areas *Input Layouts* or *Output Layouts*, you can switch between the different layouts when executing the planning folder by using pushbuttons. These are automatically generated by the system at runtime.

> In planning folders that simultaneously display two planning layouts; you can only use such layouts for whose header area a unique selection was set in the planning environment. To fulfill this requirement, you can define all the characteristics contained in the header area of a layout to a single value in the planning package.
>
> If such a definition means that the layouts included in the folders become too inflexible, then you can solve this problem with the use of variables [Page 102]: In the planning level, choose a variable as the selection for a characteristic of the layout header area. You also add this variable to your planning folder. When executing the planning folder the system offers you the variable to set different selections for the data to be displayed in the layout.

By default, the input and output areas are arranged horizontally when a planning folder is executed: Input areas are displayed in the top part of the screen with output layouts below. In folders that are not Web-enabled, layouts can also be arranged vertically: Choose *Planning Objects* → *Planning Folder Attributes*. The *Technical Detail Settings* dialog box appears. If you have set the *Arrange Layouts Vertically* indicator the input layout is displayed on the left-hand side of the screen and the output layout on the right when you execute the folder.

Assigning planning functions

You can assign planning functions (more precisely: parameter groups of planning functions) to each planning layout which you include in a folder. The assignment means that at runtime the functions are only offered or executed when the layout that you have assigned to the functions (layout-dependent functions), is active. For more information, see Automatic Execution of Layout-Dependent Planning Functions [Page 279].

Variables in planning folders

If, in the planning area that you have assigned to a planning folder, variables of the type *Fixed Value* are defined and used in all planning levels that you include in the planning folder, the system can offer you these variables when executing the planning folder for the entry of a value. In this way, you can present different scenarios with minimal effort, or change the focus of displayed data. To use this function, you add the desired variables to the folder.

> To ensure that all those variables that are used in the layouts of the folder are contained in the folder, choose *Goto* → *Global Settings*. The system then lets you add missing variables to the folder.

Display formats for output layouts

For the planning layouts that you transfer in the area of output layouts of a folder, you can choose whether the date contained in the layout should be presented as a table, graphic or a combination of both by using *Output Characteristics* (or using *Output Characteristics* in the context menu of the layout).

> The graphical formatting of key figures contained in the output layout of a planning folder is independent of the graphics that you include directly in a planning layout in the context of the planning environment. Therefore, changes to graphics in the planning environment do **not** have an effect on the graphics displayed in the planning folder.

Editing planning objects directly

For every planning object that you include in a folder, the system offers you the option of going to the planning environment and making the changes to the object directly. To do this, choose *Edit Object*.

> You can also use this function to create additional objects, which you still require in a planning folder, in the planning environment.

Further functions

You can add global planning functions to the folder. These are offered to you independently of the active layout. Possible application cases for this are central preparatory tasks, for example, deleting existing data and its replacement by specific output values.

> The system conducts a log over all steps, which you execute when processing planning folders. With the help of this log you can undo or repeat every individual processing step.

Searching for planning objects and folder elements

With the search function *Search*, you can search for planning objects or folder elements in the folder that you are editing, as long as you know the name (or part of it). This function is especially useful when finding planning objects because the number of these objects can quickly become quite large. The search function searches for the string you entered in all texts which are displayed in the list of the processing screen.

> Use the search function from the application toolbar to search for character strings in the folder. Use the search function from the context menu of the hierarchical display of planning objects to search for character strings in the planning objects.

Activities

1. To create or edit a new **planning folder**, choose *Business Planning and Simulation* → *Planning Folders* → *Customizing* from the *SAP Easy Access* screen.
2. Select the planning folder that you want to edit and choose *Edit Planning Folder*. The *Planning Folder* screen appears.
3. Select a planning level for which manual planning layouts exist.
4. Select a package in this planning level whose data you want to display or edit with a layout.
5. Select the **planning layout** that you want to use and drag it, while keeping the mouse button pressed down, into the right screen area to one of the nodes *Input Layouts* or *Output Layouts*.

   > You can also select the desired layout first and then drag the planning package to be processed into the right screen area. Or, select the planning package, planning layout, and the nodes in the right screen area, below which the layout should appear, and choose *Add Object*.

6. Enter a descriptive text and a technical name for the layout.

   > The system automatically generates a default value for the technical name. With the descriptive text you should note that the text that you enter here is used as a label for the automatically generated pushbuttons when executing the planning folder. Therefore the text should be as short as possible.

7. Select a parameter group of a **planning function** that you want to offer within the planning application, and drag it, keeping the mouse button pressed down, to one of the planning layouts that you included in the folder.
8. If you need **global planning functions**, first choose the parameter group and package, then *Planning Objects* → *Create* In the *Create Planning Object* dialog box, enter the name of the object, a text for the quick info and where necessary, a new technical name.
9. If you require **global planning sequences**, choose *Planning Objects* → *Spec. Functions / Global Sequences*, and transfer the required sequence from the list of planning sequences. In the *Create Planning Object* dialog box, enter the name of the object, a text for the quick info and where necessary, a new technical name.

10. Add all further required objects to the planning folder, and save the planning folder.

    If the functions that you offer in a planning folder should be (or have to be) executed in a certain sequence, add the function keys in the required order so that the user is lead through the application as intuitively as possible.

    But note that there is no possibility in planning folders to technically force a specific operating sequence. However, in case this is necessary for your purposes, create a planning sequence which processes the different single functions in the sequence intended, and include this sequence as a function in your planning folder.

11. Where necessary, determine the **time of automatic execution** for the layout-specific or global functions. You can find more information under Automatic Execution of Layout-Dependent Planning Functions [Page 279] andAutomatic Execution of Global Functions [Page 282].

4.2.1.2.7.2.1 Planning Folder: Portfolio Graphic

Use

You use portfolio graphics in a planning folder in order to visualize the data of a planning layout for which a presentation in several dimensions makes sense or is necessary. In contrast to other graphic types, a portfolio graphic presents the data points so that a data point represents a certain characteristic combination which is shown dependent on two or more key figures. This type of graphic is often used for decision support.

> You are interested in the sales that your article has made in the last five fiscal years compared to the sales costs spent on it. In the planning layout it is based on, you include the characteristics *article* and *fiscal year* in the lead column. You transfer the key figures *sales* and *sales costs* into the data columns. The graphic then shows how high the sales and sales costs were for every combination of article and fiscal year with the position of the appropriate data point. If in addition, you insert the profit-sales ratio in a third data column, its values are presented by the size of the data points.

There is another difference between portfolio graphics and other types of graphics regarding the types of planning folders that are supported: While you can only assign the other types of graphics to the output layouts of two-piece folders, portfolio graphics can also use one-piece planning folders of the type *Folder Without Separate Output Area, Not Web-Enabled*. If you use this option, the system can display the graphic over the entire screen area.

Integration

The portfolio graphic is a special case of a graphic output layout of a planning folder. You use it the same as other types of graphics but have considerably more setting possibilities.

Prerequisites

The SAP Internet Graphics Server (release 6.25 or higher) must be installed on your system landscape in order to define and display portfolio graphics. The connection between the SAP

Internet Graphics Server and the system in which you operate planning must take place using an RFC destination of the type TCP/IP with the predefined name **`IGS_RFC_DEST`**.

The values of the characteristics in the header area of the planning layout, which you want to visualize with a portfolio graphic, must be restricted to single values.

Features

Standardized graphic generation

If you do not make any other settings, the system proceeds as follows when converting layout data into a portfolio graphic:

- A data point is generated for every row of the planning layout. The value of the key figure from the first data column determines the X position of the data point, the value of the key figure from the second data column determines the Y position.
- The headings of the first two data columns are transferred as the heading of the X and Y axis of the diagram.
- If the layout contains other data columns, the values contained in them are converted as follows (the *Fixed Size* column refers to the state of the key figure with the same name):

Number of Data Columns	Fixed Size	Graphic Conversion
2	n/a	All data points have the same diameter.
3	no	The value of the third data column is represented by the diameter of the data points.
3	yes	The value of the third data column is ignored.
4	no	The values of the fourth data column have no visible effect.
4	yes	The values of the third and fourth data column are used to present trend arrows, provided that column 1 and 3 and 2 and 4 contain the same key figures; otherwise a pie chart is generated the segments of which are in a ratio the same as the values in column 3 and 4.
5	no	The values of the fourth and fifth data column are used to present trend arrows, provided that column 1 and 4 and 2 and 5 contain the same key figures; otherwise a pie chart is generated the segments of which are in a ratio the same as the values in column 4 and 5.
5	yes	The values of the third and fourth data column are used to present trend arrows, provided that column 1 and 3 and 2 and 4 contain the same key figures; otherwise a pie chart is generated the segments of which are in a ratio the same as the values from column 3, 4, and 5.

- All rows of the layout, the characteristics of which only differ in one value in the lead column, are grouped together in a series marked in color. The value of the free characteristic is transferred as the heading of the data points. The other characteristic values are transferred as an entry in the legend.
- The descriptions of the characteristics of the planning layout that are not hidden are transferred as the diagram title.

User-defined graphic generation

It is possible to adopt the system behavior described above to your requirements:

You can influence the way in which the data series are formed from the characteristics in the lead column. The system offers you a list of different variants to choose from that are determined dynamically from the attributes of the planning layout.

For layouts with three data columns you can set up what should happen with the rows that contain the value 0 in the third data column. A data point generated from such a row can be displayed or hidden, and you can decide whether to take account of corresponding rows for the generation of the data series.

You use the *Fixed Size* indicator to determine that an existing third data column is not used for the diameter of the data points, but is used instead for the generation of trend arrows or pie charts together with the subsequent data columns.

XML Customizing

The system generates the portfolio graphics with the help of two XML files. One file contains the data points and series of the graphic, while the other file contains control information to prepare the graphic. You can define templates in the system for these files in order to influence the generation of graphics. You can find more information on this directly in the system by choosing *Other Settings* in the *Graphic Settings* dialog box for a layout and then calling up the F1 help.

Activities

1. Choose *Business Planning and Simulation → Planning Folders → Customizing* from the *SAP Easy Access Menu* screen. You reach the *Planning Folders* screen with the list of all planning folders available in the system.
2. Select the planning folder you want to edit by double-clicking on it. The *Change Planning Folder* screen appears.
3. Select an output layout of the planning folder and choose 📊 *Output Characteristics.* The system displays the *Graphic Settings* dialog box.
4. Choose *Output as Graphic* and *Portfolio Graphics* as the *Output Type*.
5. Decide how you want to proceed:

- Confirm your entries in order to generate the portfolio graphic in accordance with the system settings.
- Choose *Other Settings* and make the desired settings in order to design the presentation of the portfolio graphic according to your requirements.

4.2.1.2.7.3 Automatic Execution of Layout-Dependent Planning Functions

Use

When editing a planning folder you are able to assign one or more planning functions to each planning layout for manual or automatic execution.

At runtime these planning functions are offered or executed automatically when the layout to which you have assigned the functions is active.

You can only determine the automatic execution of planning functions for functions of this kind that belong to a planning layout that is either used in a one-piece planning folder or as an input layout in a two-piece folder.

Features

Assigning planning functions to a planning layout

To assign a planning function to a layout, select the parameter groups for the planning function and the desired planning package. In this way you let the system know for which data the planning function is to be executed.

You are not restricted to using functions and packages from the same planning level to which the layout belongs.

For example, you can execute a function that calculates the planned revenue for all affiliated groups, while in the layout only the values are displayed for a specific company.

Determining the execution point for automatic execution of planning functions

With the icon you can determine that a function is not offered as a pushbutton, but instead is executed automatically either **before** or **after** displaying a planning layout. In this way, you can realize, for example, initialization or updates.

To use the possibility of automatic execution, select the assigned function and choose ⚙ *Function Attributes*. The system displays the *Execution of Functions* dialog box. Here you can choose to either execute the function manually using a pushbutton, or to set up automatic execution before displaying or before changing the layout.

The time for the automatic execution of a function is oriented towards the screen flow logic as it is known in ABAP programming:

- *Execute function before layout display*:
  Corresponds to the processing time PBO [External]. The function is executed before the planning layout is displayed. If the function changes data which is displayed in the layout, the changes are immediately visible in the layout. On the other hand, the original status of data before executing the function is not visible.

  You assign a function to a planning layout. This function copies the actual data of the current fiscal year as plan data into the next fiscal year before displaying the layout. As a result, you get an orientation basis for the manual entry of exact plan values. However, in this case, you must make sure that the manually adjusted values are not overwritten each time the layout is opened.

- *Execute function before layout change*:
  Corresponds to the processing time PAI [External]. The function is executed when you exit the planning layout. If the function changes data which is displayed in the layout, the changes are not immediately visible after opening the layout.

Case 1: You create a planning layout by entering plan data in the local currency of your company. You assign a function to this planning layout which automatically translates the amounts entered into the group currency when exiting the layout. In this way, you make sure that the amounts planned in the local currencies represent all subsidiaries at any time at group level in a standard currency, and can be compared with each other.

Case 2: You use a planning layout for the approximate planning of sales figures of a product. Here you enter the planned quantities in a time period divided into quarters. However, these quarterly values should be broken down into periods in a later planning phase. As preparation for the period-specific sales planning, you assign a planning function to the layout, which divides the entered quarterly values by three, and posts the result into the three periods of every quarter. You can refine this procedure by first entering the sales goal briefly for the whole fiscal year. You assign a function to the layout which reproduces the seasonal sales fluctuations of the last fiscal years with the help of a distribution key, and distributes the plan data period-specific with the expected fluctuations to the fiscal year to be planned.

Both execution times (*before layout display* and *before changing the layout*) are triggered by the activities listed below.

- Changing a variable value
- Selecting another layout
- Executing a planning function
- Saving

Note that within planning folders you can assign functions to a layout that operate on completely different data than the layout itself. In such a case, executing the function will normally bring about no changes to the data displayed in the layout.

In addition to the procedure described here, BW-BPS also offers the possibility to assign a function for automatic execution directly in the layout definition to a planning layout. However, a function assigned in this way is not executed when the layout is opened within a planning folder.

Activities

1. Select the layout to which you want to assign the planning function.
2. Select the parameter group for the planning function as well as the desired planning package.
3. Choose *Add Object*.
4. Choose *Function Attributes* and make the necessary selection.

See also:

Editing a Planning Folder: Details [Page 274]

Automatic Execution of Global Functions [Page 282]

[Automatic Execution of Planning Functions [Page 145]]

4.2.1.2.7.4 Automatic Execution of Global Functions

Use

When you edit a planning folder you are able to assign one or more global planning functions or planning sequences to manual or automatic execution under the *global planning functions* folder.

At the time of execution, these planning functions are offered or executed automatically, independently of the layout that is active.

Features

Assigning global planning functions or sequences

Various functions in the context menu of the *Global Planning Functions* folder allow you to create global planning functions or sequences as planning objects. For more information, see the section “activities”.

Determining the execution point for automatic execution of planning functions

The following options are available:

- *Function as pushbutton*: The function can be executed using an appropriate pushbutton.
- *Execute function at folder start*: The function is executed automatically before the planning folder is started and, in particular, before variable values can be entered. If an error occurs during execution, the execution of the folder is terminated.
- *Execute function when saving*: The function is automatically executed each time before the changed plan data is saved. If an error occurs, the data is not saved.

Activities

1. Create the required global planning functions or sequences as planning objects.
2. Choose *Function Properties*. The dialog box *Execution Time of Functions* appears.
3. Make the required choice. According to the time of execution that you have chosen, the system highlights the planning object with a particular symbol.

See also:

Editing a Planning Folder: Details [Page 274]

Automatic Execution of Planning Functions [Page 279]

4.2.1.2.7.5 Documents in Planning Folders

Use

You use documents in planning folders to enable access to qualitative information that is related by content to the characteristics and key figures of a planning package. For more information on the general concept of the documents function in BW-BPS, see Documents [Page 191].

Integration

The documents which you display, create, or edit in the context of a planning folder are the same documents which you also edit in the planning environment with the documents function.

Features

You can use documents in a planning folder in different ways:

- In a displayed planning layout as a comment to a cell.
  With this variant, no special activities are required in the Customizing of the planning folder. The only decisive factor is whether a planning layout used in the planning folder contains documents and whether the document display has been activated for the layout in the Layout Builder.
- As a list of documents which are assigned to the key figures and characteristic values of a planning package.
  With this variant, you transfer the *Documents* (**0-DOC**) function node in the Customizing of the planning folder, which was created in the planning environment for a planning level, into the planning folder. When executing the planning folder, the system offers the available documents in a list for display or editing.

You can display, edit, and delete existing documents, and you can create new documents.

Activities

To offer documents in a document list in a planning folder, proceed as follows:

1. Select the desired folder in the overview of the planning folders, and choose *Planning Folder → Change Planning Folder*. The detail screen to edit the planning folder appears.
2. Select planning area, level, and package for which documents exist, and for which the documents function was created in the planning environment, by double-clicking on them.
3. From the list of planning functions, choose the *Documents* (**0-DOC**) entry and insert it in the planning folder in one of the areas *Input layouts* or *Output layouts*, for example, using Drag&Drop.

   Although documents in BW-BPS have the status of a function, within planning folders they are handled similarly to layouts. This means you can neither insert the *Documents* planning function in the *Global Planning Function* area of a planning folder, nor assign it to a planning layout as a local function. One of the reasons for this is that both listing and displaying documents in the application screen requires just as much space as displaying a planning layout. Therefore, a juxtaposition of planning layout and documents is not useful.

4. Define a name for the document list.
5. If necessary, repeat the steps named for the documents function in connection with further planning areas and levels.
6. Save your changes.

In the executed planning folder, you can select the document list in the same way as a planning layout. The system offers you pushbuttons for the different editing possibilities for the documents.

4.2.1.2.7.6 Executing a Planning Folder

Use

You execute a planning folder in order to use the planning layouts and functions that are connected in terms of content (and therefore contained by the folder) in the sense of a business planning application.

Features

Display in SAP GUI or in Web browser

You can decide whether you want to execute a planning folder in the SAP GUI or in a Web browser (the prerequisite for executing it in the Web browser is that you have selected a Web-enabled variant when creating the folder).

To execute a planning folder in the Web browser you must first generate a Web interface for the folder. You can then convert this into an executable Web application using the BW-BPS Web Interface Builder [Page 285]. For more information, see Editing Planning Folders [Page 271].

If you have created a planning folder that is **not** Web-enabled you can generate a Web-enabled folder from this later by copying the folder. Here you can specify that it should be Web-enabled in the general folder attributes of the new folder. All other attributes of the original folder are transferred unchanged to the new folder when copying.

Switching between different planning layouts

You can switch between the different layouts assigned to a planning folder. If more than one layout has been assigned to the input or output area of the folder you can do this by choosing one of the pushbuttons in the executed folder. The system automatically generates these for every assigned layout.

Data that you enter or change in the context of planning folders is subject to the general BW-BPS saving concept. All changes are stored internally on a temporary basis and also remain between layouts during the switch. Therefore, you only need to save the data when exiting the planning folder.

Controlling data changes

If in Customizing you have defined a corresponding connection regarding content between input and output layouts, you can change the data in the input layout and check the effects directly in the output layout.

For example, you can enter or change primary costs at cost center level in the input layout, and check the effects on the aggregated costs for the entire controlling area in the output layout.

Setting variables

As with the planning environment, you can set user-specific values for all variables (not just those in the particular planning folder) when you execute. Choose *Goto* → *Set variables*. The *Set Variables* dialog box appears.

Showing or hiding variables and planning functions

The following functions serve to increase the available size of the planning folder layout.

If you have included **variables** in a planning folder then the system displays these variables with their current value in the screen area.

- You can hide the display of variables using *Close Variables*.
- You can show them again using *Display Variables*.

You also have the opportunity to hide variable values using , and show them again using .

If you have included **planning functions** in the planning folder, the system displays pushbuttons with the symbol for the appropriate functions.

- You can stop functions being displayed using *Close Functions*.
- You can show them again using *Display Functions*.

Searching for planning folders

You use the search function to search for a specific folder if you know its name or a part of the name. This function is very helpful when working with folders that were not created in the context of a planning application because the number of folders can quickly become quite large. The search function searches for the character string you entered in all texts displayed in the list of folders. If you have shown the technical ID then the search extends to the name of the planning folder, the technical name, and for Web-enabled folders to the URL displayed in the list.

4.2.1.2.8 Web Interface Builder

Use

You can create Web-enabled planning applications in the form of Business Server Page [External] applications (BSP applications). You can execute this type of planning application in the Web browser. You do not require additional software installed on client work centers; you just need an HTTP connection to an application server.

Web-enabled planning applications are a useful alternative to applications that are executed in the SAP GUI since applications created with the Web Interface Builder always contain exactly those functions that are necessary for a specific planning task.

In principle, you can also create BSP applications - as generated by the Web Interface Builder - in the ABAP Workbench. However, the Web Interface Builder allows you to create Web applications much more easily and quickly due to its predefined components. If necessary, you can revise the applications created in the Web Interface Builder in the ABAP Workbench.

The Web-enabled planning applications are described by Web interfaces. On the basis of the Web interfaces, the Web Interface Builder generates ABAP, HTML, and JavaScript code for an executable BSP application.

A Web interface encapsulates the elements of a Web-enabled planning application in the form of components that have a name and other attributes.

Planning layouts, pushbuttons for executing functions and selectors for characteristic values are all examples of components. Further components such as page, link, text or standard pushbutton for saving and exiting are additional components that are required for executable BSP applications.

The values of properties can be fixed values (constants) or references to the properties of other components.

For example, the “planning area” property of a package selector contains the technical name of a planning area. In comparison, the “package” attribute of a layout contained in the Web interface can refer to the technical name of the package selector.

Integration

The planning applications created with the Web Interface Builder are based on the planning objects (planning area, planning levels, planning functions and so on) that you edit in the BW-BPS planning environment [Page 90].

For example, you have to have created a planning layout for manual planning in the planning environment before you can reference the layout and set parameters using an appropriate component in the Web Interface Builder.

You can use the Status and Tracking System [Page 310] to bundle planning applications that have been created with the Web Interface Builder into enterprise-wide planning projects and monitor their progress.

You can generate a Web interface on the basis of a planning folder [Page 269] and process this further in the Web Interface Builder.

Prerequisites

In order to create planning applications with the Web Interface Builder, planning objects must already have been created in the planning environment:

- For a simple planning application that is only used for manual data entry you require at least one planning area, one planning level, one planning package as well as a manual planning layout.
- For more complex planning applications, you require planning functions and parameter groups as well.

Furthermore, the necessary authorization must be entered in your authorization profile. For more information, see Authorizations [External].

Features

You create planning applications with the Web Interface Builder and execute them in the Web browser.

Working with the Web Interface Builder includes two basic steps:

1. Compiling the **Web interface** from components that represent BW-BPS planning objects.
2. Generating the executable **Web application** on the basis of the Web interface you created.

You can choose whether you want to create a Web interface manually or with the help of the integrated wizard. The wizard leads you through the creation process in several steps and helps you to create an application in as little time as possible. You can edit the Web interfaces created with the wizard manually.

An individual planning application can access various planning areas. It can span one or more HTML pages. For applications that contain several pages, a predefined navigation button is available. You use this to switch between pages when you execute the application.

Enhancement for Web Interfaces

When you create a Web interface using the BW-BPS Web Interface Builder, the system generates a BPS page in accordance with the specified Customizing. BSP event handlers are implemented in ABAP OO class CL_UPWB_BSP_APPL. You can change the standard behavior and add new functions by deriving your own sub-classes and redefining their methods (see Enhancement for Web Interfaces [Page 309]).

Goto Menu

You can display a list of the available Web interfaces, along with information on the change history, by choosing *Goto* → *Interface Overview*.

You can choose between different variants for splitting the screen into different work areas by choosing *Goto* → *Settings*.

Finally, the Web Interface Builder allows you to branch directly to the areas of the SAP system where the relevant objects are managed when you are creating and editing Web interfaces:

- **BW-BPS planning environment** (*Goto* → *Business Planning and Simulation*)
  Here you create and edit the objects that you require to access BW-BPS data: planning areas, levels, packages, layouts.

- **ABAP Workbench** (*Goto* → *ABAP Workbench*)
  Here you can manually revise/rework the BSP application generated by the Web Interface Builder for a Web interface.

- **Class Builder** (*Goto* → *Class Builder*)
  In the Class Builder of the ABAP Workbench you can revise the automatically generated class or edit a class that you have created yourself and assigned to a Web interface.

- **MIME Repository** (*Goto* → *MIME Objects*)
  For each BSP application that you generate from a Web interface, the system automatically generates a folder with the same name in the ABAP Workbench MIME

Repository [External]. Here you can edit the multimedia-based objects (like graphics) that belong to a Web interface.

You can use cascading style sheets (CSS) to modify the appearance of the planning applications created with the Web Interface Builder to suit your own requirements. To do this, you import the centrally delivered style sheet into the folder automatically generated for your application in the MIME Repository as a MIME object and make the required modifications. The style sheet contains all the style classes that the Web Interface Builder uses.

- **HTTP service maintenance** (*Goto → HTTP Service Maintenance*)
  In this transaction you make security-relevant settings for your Web interface. For example, you can ensure that a trusted transfer protocol is used or activate and deactivate a Web service.
- **Online text repository** (*Goto → OTR*)
  In this transaction you maintain the UI texts for your Web interface that are relevant for translation (for example, pushbutton labels, static texts). In the appropriate attribute of the component, you enter a reference to the corresponding OTR text element instead of a plain text.
- **Translation of OTR text elements** (*Goto → OTR → OTR Translation*)
  In this transaction you translate the OTR text elements created for a Web interface into other languages.
- **Translation** (*Goto → Translation*)
  In this transaction you translate the short texts for the Web interface descriptions. This is only relevant to the input help texts in transactions BPS_WB and BPS_WIF0.

4.2.1.2.8.1 Creating and Editing Web Interfaces

Use

You create and edit Web interfaces in order to generate a BSP application that is based on the Web interface. The BSP application accesses the planning objects using a Web browser.

Prerequisites

To create a functional BSP application, you must already have created planning objects in the planning environment.

SAP delivers planning applications based on Web interfaces that have been created with the Web Interface Builder. To avoid conflicts with the delivered Web interfaces, create your Web interfaces in the **namespace** that is reserved for customers. This begins with **Y** or **Z**.

Note that the Web interface development objects delivered by SAP are part of the packages [External] delivered by SAP. We recommend that you do not change these objects as this represents a modification [External] and has organizational and technical consequences. However, if you do want to make changes, copy the delivered objects and then change the copies.

Procedure

Use the Wizard to Create a Web Interface

The Web Interface Builder has a wizard that supports you in creating a new Web interface:

1. On the *SAP Easy Access Menu* screen, choose *Business Planning and Simulation* → *Web Interface Builder* → *Customizing*. The initial screen of the Web Interface Builder appears.
2. Choose (*Web Interface* → *Create with Wizard)*. The start page of the wizard appears.
3. In the following steps of the wizard, you put together the elements that you require:
   a. *Start*
   b. *Specify Name for Web Interface*
   c. *Select Planning Areas*
   d. *Select Planning Levels*
   e. *Create Individual BSP Pages*
   f. *Complete*

   For each step, the wizard offers information about the possible settings. In the first step, you only have the option of switching to the next step. However, when you have carried out more than one step you can switch between the steps that you have already performed in any order. This allows you to change the settings you made for an earlier step later in the process.
4. When you have made all the settings and have arrived at the last step, choose *Complete*. The system creates a new Web interface in accordance with your settings.
5. To save the Web interface, choose (*Web Interface* → *Save)*.

   A Web interface that was created using the wizard is no different to a manually created Web interface. Therefore, the subsequent processing of a Web interface created using the wizard is the same as the subsequent processing of a manually created Web interface.

Create a Web Interface Manually

1. On the *SAP Easy Access Menu* screen, choose *Business Planning and Simulation* → *Web Interface Builder* → *Customizing*. The initial screen of the Web Interface Builder appears.
2. Choose (*Web Interface* → *Create)*.

   The system displays a dialog box in which you can determine basic properties of the Web interface. Depending on what you enter here, the system automatically creates certain elements for you for the new Web interface.
3. In the *Number of Pages* field, enter whether the generated application should consist of one or more HTML pages.
4. Select which of the available *Standard Elements* you want the system to create on each page of the application.

You can manually correct the selection that you make in the two last steps when you create a Web interface by adding further elements or removing elements that are not required.

5. Edit the new Web interface as described in the following *Edit a Web Interface* section.

Edit a Web Interface

1. On the *SAP Easy Access Menu* screen, choose *Business Planning and Simulation → Web Interface Builder → Customizing*. The initial screen of the Web Interface Builder appears.
2. Choose (*Web Interface → Open)*.
3. From the list of available Web interfaces, select the one you require. The system displays the hierarchical element structure of the Web interface.
4. Edit the components of the Web interface (see Components for Web Interfaces [Page 298]):
   - To add a new component to the Web interface, choose:
     - *Create Page* in the context menu of an element of type “application”. A dialog box appears where you can enter the name of the page.
     - *Create Subcomponents* in the context menu of an element of type “page” or “container”. A dialog box appears with an overview of the subcomponents available in the system. Select the required element type.
   - To edit the attributes of an existing component, double-click on its name in the element structure or choose *Change Attributes* in the context menu of the component.

   The system displays the attributes of the component in the attribute editor.

   The different attributes of the components are documented in the system. For more detailed information about a specific attribute, choose the appropriate attribute in the attribute editor and then use F1.
5. When you have created all required elements and have set their attributes, save the Web interface.
6. Choose (*Edit → Check Consistency)*. If applicable, the system highlights problems such as, for example, incorrect references between elements of the Web interface.
7. Choose (*Edit → Generate)*. The system generates the objects specified in the *Result* section below.
8. Choose *Edit → Display Preview* or *Display in External Browser* to test the application.

   If you are in the test phase when you create or change a Web interface and you want to check your results as quickly as possible, you can use the quick preview in the Web interface instead of the last steps two that are mentioned. To do this, choose *Goto → Settings*. Under *Preview* in the *Settings* dialog box, choose the *Quick Preview w/o Generating* option. This means that you can display the Web interface in the preview window without having to generate the BSP application first (which can be time-consuming).

Result

As a result of the subsequent generation in the Web Interface Builder, the system has generated the following objects:

- For each element of type “page” that you have created in the Web interface, the system generates a BSP page which contains the elements required for the Web interface. This page is stored in the system. At the runtime of the application, the SAP Web Application Server generates a HTML-format page from every requested BSP page. This is displayed in the browser.
- In addition to that, additional pages are generated for special purposes (for example, a page which is displayed when exiting the Web application).

When using the SEM-BPS design (see Design Templates for Web Interfaces [Page 299]): If you have specified in the properties of the Web interface that you want to use a customer-designed class, the system generates this class. The generated BSP pages then use the processing logic implemented in this class (and not the standard class of the Web Interface Builder).

See also:

SAP Web AS Architecture [External]

Creation of Web Applications with Business Server Pages [External]

4.2.1.2.8.2 Integration of BEx Web Applications

A Web interface consists of a more or less complete set of elements and information for executing a planning task. If you want to display the plan data in the context of reporting or display additional details, you can integrate BEx Web applications into a Web interface.

⚠

The Web Interface Builder does not support the integration of a BEx Web application as one of its own generic components. You can develop integration yourself, using the Web Interface Builder. The rest of this document describes various examples of customer scenarios. You can adapt these examples to your own specific requirements accordingly. SAP does not offer support for source code developed by customers.

Define your query so that it has a navigational state that is useful for planning. This is called every time you refresh the Web Interface Builder. This means that each time the Web interface is refreshed, the navigational state of the query is reset to the start state.

The following examples show:

- How you add a BEx Web application to a Web interface.

First you have to add an IFRAME definition to the Web interface. The source attributes of the IFRAME definition have to contain a fixed URL.

For information on the URL of a BEx Web application, see Creating an IFRAME for BEx Web Applications [Page 292].

- How you add a BEx Web application to a Web interface where the filter values of the BEx Web application are synchronized with the Web interface.

  First you have to specify how the BEx Web application uses JavaScript to get the current selection from a BW-BPS variable selector. You then embed an IFRAME definition in the Web interface and a JavaScript source code that passes the variable values from the BPS application to the BEx Web application.

  - Accessing the generated HTML of the Web interface is problematic because this differs according to the appearance of the selector; the values have to be determined in different ways. This is not possible for variables that are not visible because the corresponding HTML does not exist.

    The filter value is determined by the BW-BPS variable value selector. See Filter Values from BW-BPS Variables [Page 293].

  - One solution is to add script statements for the Business Server Pages to the user-defined HTML section of the Web interface. At runtime, statement `<%=descr(VarvlSel_Product/value)%>` is replaced by the value of the variable selector `VarVlSel_Product`, independently of the appearance of the variable selector.

    The filter value is determined by a planning layout using a row selection. See Filter Values from Row Selection in Planning Layouts [Page 296]. This example builds on the previous example.

4.2.1.2.8.2.1 Creating IFRAMEs for BEx Web Applications

9. You are in the Web Interface Builder. Open the Web interface that you want to modify.
10. To add JavaScript and HTML source code to the required Web interface, create a component of type *Text*.

    a. In the context menu of an element of type “page” or “container”, choose *Add Subcomponents*. A dialog box appears in which an overview is provided of the subcomponents available in the system.

    b. Choose the element type *HTML Text, Image*.

    c. Enter a name for the component.

    d. Set the value of the *HTML* property to *True*.

    e. Use input help for the *Text* property and choose the *Edit Long Text* option.

    f. The *Edit Text* dialog box appears. Here you can enter the HTML fragments given in the examples below.

11. Move the text components in the Web interface so that they are where you want them to appear in the BEx Web application.
12. Copy the source code and replace the name `MYWEBAPPL` with the name of your own BEx Web application.

HTML source code for integrating a Web application

```
<iframe
    id="BWReport01"
    name="BWQUERY"
    src="/sap/bw/BEx?cmd=ldoc&no_session_cookie=X&TEMPLATE_ID=MYWEBAPPL"
    width="800"
    height="600">
    Frames are not supported.
</iframe>
```

13. Choose *Generate*. Save and generate the planning application.

4.2.1.2.8.2.2 Filter Values from BW-BPS Variables

Prerequisites

- A BW-BPS variable is defined for the InfoObject that is used for filtering. Note that if this InfoObject is a compound InfoObject, the higher-level value also has to be added to the filter.

  Fiscal year (0FISCALYEAR) is compounded with fiscal variant (0FISCVARN). If the year is 2004 and the variant is Q4, the filter value must be Q4/2004.

- The key of the characteristic must not contain special characters or spaces; it must only contain letter and numbers.
- The example only works with Microsoft Internet Explorer 6.0.

Procedure

Example 1

1. Create a text component (see Creating IFRAMEs for BEx Web Applications [Page 292]) and move it in the Web interface so that it is where you want it to be displayed in the BEx Web application.

   In this example, the name of the text component is `BWTemplate_with_filter`.

2. In the component tree above the text components, create a component of type *Selector Variable Value* and define the properties for the layout and the format.

In this example, the name of the variable selector is `VarvlSel_Product`.

3. In the properties of the variable selector, change the name of the planning area, the variable and the format, where this is possible. Depending on the design, this option may not be available.

   The format has to correspond to one of the following models:

   - Text (key)
   - (Key) text
   - Key

   Text alone is not supported since the proposed solution requires the key information in order to set the filter.

   Other properties, for example: *Appearance* dropdown, *Number of Fixed Rows* 1, *Result of Selection* true, *Formatting* 0BPS_PROD-VALUE.

4. Copy the source code and specify the required values in accordance with your requirements.

   BEx Web application `MYWEBAPPL`

   InfoObject `0BPS_PRODU`

   Variable selector `VarvlSel_Product`

   If you are using a compound InfoObject, change the *comp* variable accordingly.

JavaScript source code for determining the filter value for the BEx Web application from a BW-BPS variable

```
<iframe id="BWReport03" name="BWQUERY" src="/sap/bw/mime/bex/icons/pixel.gif" width=

<script language="JavaScript" type="text/javascript">
  var frame  = document.getElementById( 'BWReport03' );
  var url    = '/sap/bw/BEx?cmd=ldoc&no_session_cookie=X&TEMPLATE_ID=';
  var filter = '&FILTER_IOBJNM=0BPS_PRODU&FILTER_COLLAPS=&FILTER_VALUE_EXT=';

// Example using fiscal year 2004 and variant K4: K4/2004 => comp = ‘K4/’
  var comp   = '';
  var appl   = 'MYWEBAPPL';
  var varvl = '<%=descr(VarvlSel_Product/value)%>';
// Option: Adjust the '--SELECT--' to your local language
  if (varvl != "" && varvl != "-- SELECT --"){
// expected formatting of variable selector: Text(Key) or Key
    var convert = /\((.*)\)/;
    convert.exec(varvl);
// If there is value in brackets it is the key
    if ( RegExp.$1 != "") varvl = RegExp.$1;
// create URL
  url = url + appl + filter + comp + varvl;
// remove possible white spaces
  url = url.replace(/ /,"");
// set source attribute of iframe to new URL
  frame.src = url;
}</script>
```

5. Choose *Generate*. Save and generate the planning application. The filter value for BW-BPS products is taken from the BW-BPS variable selector.

Example 2

This example of source code shows how you determine more than one variable value. First follow steps 1 – 3 of example 1.

6. Copy the source code and specify the required values in accordance with your requirements.

   Variable selector: `VarvlSel_Product`, `VarvlSel_Customer`

   BEx Web application: `MYTEMPLATE`

   Variables: `PRODUCT`, `CUSTOMER`

   If you are using compound characteristics, insert the value of the compounding characteristic.

HTML and JavaScript source code for determining variable values for the BEx Web application from BW-BPS variables

```
<iframe
id ="myWebAppl"
src="/sap/bw/BEx?cmd=ldoc&TEMPLATE_ID=MYTEMPLATE" style="width:800;height:150">>
</iframe>

<script language="JavaScript" type="text/javascript">

function varValueConvert(dispValue){
/* Expected formatting of variable selector: Text(Key) or Key
   If there are more than one (...) expression the first is taken.
*/
  var keySection = dispValue.match(/\(.*?\)/);
  if (keySection){
    var keyValue = keySection[0].replace(/\(|\)/g,"");
    alert(dispValue + ' : ' + keySection + " : " +keyValue);
    return keyValue;
   } else {
    alert(dispValue);
    return dispValue;
   }
}

var frame = document.getElementById( 'myWebAppl' );
var url = '/sap/bw/BEx?cmd=ldoc&TEMPLATE_ID=';
var queryVar1 = '&VAR_NAME_1=PRODUCT&VAR_VALUE_EXT_1=';
var queryVar2 = '&VAR_NAME_2=CUSTOMER&VAR_VALUE_EXT_2=';

var appl = 'MYTEMPLATE';
var bpsVarvl1 = '<%=descr(VarvlSel_Product/value)%>';
var bpsVarvl2 = '<%=descr(VarvlSel_Customer/value)%>';

bpsVarvl1 = varValueConvert(bpsVarvl1);
bpsVarvl2 = varValueConvert(bpsVarvl2);

// create URL
url = url + appl + queryVar1 + bpsVarvl1 + queryVar2 + bpsVarvl2 ;
// remove possible white spaces
  url = url.replace(/ /,"");
// alert('Generated URL: ' + url);
// set source attribute of iframe to new URL
frame.src = url;

</script>
```

7. Choose *Generate*. Save and generate the planning application.

4.2.1.2.8.2.3 Filter Values from Row Selection in Planning Layouts

Prerequisites

- You have implemented the example given in Filter Values from BW-BPS Variables [Page 293]

- You have defined a manual planning layout that is used in the lead columns of the InfoObject that you want to filter.
- You have defined a BW-BPS variable for the InfoObject. This is used for filtering. Note that if this object is a compound InfoObject, you have to add the value of the compounding InfoObject to the filter as well.

  Fiscal Year (0FISCALYEAR) is compounded with *Fiscal Year Variant* (0FISCVARN). If *Fiscal Year* has value “2004” and *Fiscal Year Variant* has value “Q4“, the filter value must be “Q4/2004“.

- The key of the characteristic must not contain special characters or spaces; it must only contain letter and numbers.
- The example only works with Microsoft Internet Explorer 6.0.

Procedure

1. Navigate to the page you created in the *Filter Values from BW-BPS Variables* example and in the context menu, select *Create Subcomponent*. A dialog box appears with an overview of the subcomponents available in the system.
2. Choose the *TabStrip Control* element type and transfer the proposed names.

   In this example, the name of the tab strip control is `TabStrip2`.

3. In the context menu, choose *Create Tab* to create two tabs and enter a name in the field of the *Label* property.

   In this example, the names of the tabs are:

   `TabPage2` for `Manual Planning`

   `TabPage3` for `BEx Web Application`

4. Move the variable selector that you created in the example *Filter Value from BW-BPS Variable* to `TabPage2`.
5. Move the text components for the BEx Web application that you created in the example *Filter Value from BW-BPS Variable* to `TabPage3`.
6. In the context menu for `TabPage2`, choose *Create Subcomponent* to create a component with element type *Planning Layout* for manual planning.

   In this example, the name of the layout component is `Layout3`.

7. In the properties of the layout component, specify the required planning area, planning level, package and layout and use the input help for the *Row/Cell Selection* field to select the line (row).
8. Navigate to the variable selector you created in the *Filter Values from BW-BPS Variables* example and in the properties in the *Source for Selection* field, specify the name you used when you created the layout component.

The view is automatically set to not visible.

9. Choose *Generate*. Save and generate the planning application.

The following steps describe how you test the Web interface.

10. Start the Web interface.
11. Navigate to the *BEx Web Application* tab page. The BEx Web application is empty since a valid filter value is not available.
12. Switch to the *Manual Planning* tab page and choose a row.
13. Switch to the *BEx Web Application* tab page. The filter now contains the value of the row selected in the navigation block.

4.2.1.2.8.3 Components for Web Interfaces

Definition

The different components in Web interfaces are used for the following purposes:

- To offer the end user of a Web application operating elements with which the application can be controlled.
- To establish a connection between the planning objects created in the planning environment and the Web interface.

Each component has a number of properties with whose values you can determine the behavior of a specific Web application.

Structure

The following table gives a brief overview of the components available and their usage. There is a more detailed description for every attribute in the attribute editor of the Web Interface Builder in the F1 help.

Type	Use
Application	Initial element of a Web interface. Contains central properties of the application such as your URL, the package used, and settings for the Web server.
Page	Represents an HTML page of the Web application. A Web application can contain one or several pages. Pages have attributes which influence the arrangement of the elements on the page, control their appearance, and determine the page context (stateless or stateful).
Send pushbutton	Sends the entries and settings which were made in the Web application to the server. Changed data is saved in the BW-BPS internal buffer, but not in the database.
Exit pushbutton	Closes the Web application and ends the connection to the Web server.
Function pushbutton	Triggers the execution of a planning function with a specific parameter group.

Navigate pushbutton	Enables the navigation between the pages of a Web application which has several pages.
Save pushbutton	Sends the entries and settings which were made in the Web application to the server. Changed data is saved persistently in the database.
Container	Invisible element to group lower-level elements.
Group	Visible element to group lower-level elements. Contains a heading as well as an invisible grid to arrange the lower-level elements in a table layout.
Tabstrip	Is used as a higher-level element of tabs, to which further components can be arranged.
Tab	Contains a number of other components. The individual tabs within a tabstrip component are independent of each other.
Planning layout	Table with editable cells which represents a planning layout [Page 128] of manual planning [Page 127]. Used to display and enter data.
Hyperlink	Text with assigned link target (URL)
Text	Static text.
Selectors: Function/layout, package, variable value	Assignment elements which establish the technical connection between the Web interface and planning objects. Either visible or invisible.
Chart	Enables the graphical output of plan data in the form of different diagram types (for example, bar chart, and pie chart).
Error output	Determines the position on HTML page where system error messages are issued. If there are no error messages, then this error output is invisible.

4.2.1.2.8.4 Design Templates for Web Interfaces

Use

The visual appearance of a Web interface is determined by the design template used by the Web Interface Builder. Exactly how this is produced depends on whether you use the new design or the old design.

Features

New Design (BW-BPS)

With the new design concept, the visual appearance of a Web interface is determined by the *Theme*. Each theme is defined by a MIME files record for rendering HTML (CSS files (*Cascading Style Sheet*), images, JavaScript files). These are stored in the MIME Repository.

For more information on the *Theme* concept, see the Business Server Pages documentation under Setting Style Sheets for HTMLB, XHTMLB and PHTMLB [External].

SAP delivers several predefined *SAP Themes*:

- `sap_standard`
- `sap_tradeshow` (default setting)
- `sap_chrome`
- `sap_highcont`
- `sap_hcb`

You can select one of these *SAP Themes* using the input help for the *Theme Path* field.

If you want to adapt the visual appearance to your own requirements, you can create your own *Theme* on the basis of the *SAP Theme*. You change the style classes in your *Theme* to suit your requirements.

The most simple way to change the visual appearance of a Web interface is as follows: If you are using the Web interface in SAP Enterprise Portal, call the *Theme Editor* by choosing *System Administration* → *Portal Display*. You have to adapt the component-specific styles. A Web interface uses the following styles: *Application*, *Button*, *Cursor*, *Label* and *Field*, *Link*, *Text*, *Group*, *List*, *Table*, *Tree*. The visual appearance of the planning layout is determined by the "BI Web Reports 3.0" styles.

If you are using a Web interface in SAP Enterprise Portal and have an absolute font size in particular classes of the BW style sheet 3.0, the combination of the BW style sheet 3.0 and the other component-specific styles may mean that the font size in the planning layout of your Web interface is very small.

Therefore you must **not** specify the font size in these classes of the BW style sheet 3.0, or you must only specify a **relative** font size (for the parent element font sizes).

Font sizes can have the following values: `larger, smaller,` *<number>*`%,` *<number>*`em`.

The definitions are based on the following classes of the BW style sheet 3.0:

SAPBEXTableGrid

SAPBEXCellspacing

SAPBEXchaText

SAPBEXaggItem

SAPBEXstdItem

SAPBEXHLevel0

SAPBEXHLevel1

SAPBEXHLevel2

SAPBEXHLevel3

SAPBEXaggData

SAPBEXstdDataOdd

SAPBEXstdData

SAPBEXDataErr

If you want to change the visual appearance of your Web interface but you are not using it in SAP Enterprise Portal, you have to create your own *Theme* manually in the BI system and modify it.

For more information and a simple example of this procedure, see Using Your Own Style Sheet (New Design) [Page 301].

Old Design (SEM-BPS: Old and New SEM-BPS Design)

With the old design concept, the visual appearance of a Web interface is defined by a predefined style sheet delivered by SAP. If you want to adapt this format template to your own requirements, you can create an additional CSS file and use that. You can define a format file of this type for each page of your Web interface.

If you have integrated your Web interface (based on the old design) into the portal and have changed the portal format template, you have to modify the format template of the Web interface separately.

For more information, see:

CSS Classes for Web Interfaces (Old Design) [Page 304]

Modifying CSS Classes for Web Interfaces (Old Design) [Page 308]

Activities

- When you create a Web interface, you are automatically using the new design.
- When you edit a Web interface that was created using the old design, you can continue to work with the old design or you can switch to the new design.

  In the *Old Design* property of the Web interface, choose value `false` if you want to use the new design.
- You **cannot** switch back to the old design from the new design.

4.2.1.2.8.4.1 Using Your Own Style Sheet (New Design)

Use

If you want to adapt the visual appearance of a Web interface to your own requirements, you can create your own *Theme* based on the *SAP Theme*.

We recommend that you create copies of the standard SAP style sheet first, make these known to the Web Interface Builder and then modify them to suit your own requirements.

Note that you **cannot** use this procedure if you are using the SAP Enterprise Portal style sheet. You have to edit this in the theme editor in the portal.

Prerequisites

You are using the new design (see Design Templates for Web Interfaces [Page 299]).

Activities

The following example assumes that you want to use the standard SAP style sheets used by the Web Interface Builder as the basis of your modifications. These always contain one *SAP Theme* that is used for formatting all components (except the planning layout) and one or more Web browser-specific *SAP BI Style Sheets* that are used for formatting the planning layout.

In this example, we use the following standard SAP style sheet as the template:

- *SAP Theme* `sap_standard`
- SAP BI style sheet `BWReports_ie6.css` which corresponds to this *SAP Theme*.

Copy SAP theme

1. In the ABAP Editor (SE38), call program BSP_UPDATE_MIMEREPOS and choose *Execute*.

   Make the following settings.

   a. Enter the MIME Repository path.

      In this example we use:

      `/SAP/PUBLIC/BC/UR/Design2002/themes/sap_standard`.

      When you specify the path, note that it is case sensitive.

   b. Choose the *Export to disc*. option.

2. Choose *Execute*. A dialog box appears in which you can select a target directory on your local PC.

   Note that this folder must be empty.

   The export begins after you confirm the target directory. The export may take a few minutes. Check the content of the target folder in order to monitor the progress.

3. Call program BSP_UPDATE_MIMEREPOS and choose *Execute*.

   Make the following settings.

   a. Enter the path of the target MIME Repository.

      In this example we use:

`/SAP/BC/BSP/SAP/PUBLIC/CUSTOMTHEME`.

b. Choose the *Import from disc*. option.

4. Choose *Execute*. A dialog box appears in which you can select a root directory on your local PC.

   The import begins after you confirm the root directory. The import may take a few minutes.

Copy SAP BI style sheet

5. In the object navigator (transaction SE80), open the *MIME Repository* and navigate to the appropriate SAP BI style sheet.

   In this example we use:

   `/SAP/BW/BEx/StyleSheets/BWReports_ie6.css`.

   In the context menu of the style sheet, choose *Upload/Download* → *Download*. A dialog box appears in which you can select a target directory on your local PC. The export begins after you confirm the target directory.

   Repeat this step for all Web browser types that you want to support. The last few characters in the name of the style sheet indicate the type.

   IE6 = Microsoft Internet Explorer 6

   N6X = Netscape Communicator 6

   A style sheet without a Web browser-specific ending is used by default for all other Web browsers.

6. In the object navigator (transaction SE80), open the *MIME Repository* and navigate to the directory for your customer-specific *Theme*.

   In this example, we created a directory `customtheme` under the following path:

   `/SAP/BC/BSP/SAP/public/customtheme`

   In the context menu of the directory, choose *Import MIME Objects*. A dialog box appears in which you can select a file. Choose the style sheet that you exported and modified locally. Choose *Open*. The import starts.

   Check that the name of the style sheet corresponds exactly to the following example: `BWReport_ie6.css`.

Including your own themes in the Web Interface Builder

7. In the Web Interface Builder, call your Web interface.

   In the context menu of the root node, choose *Change Properties*. In the *Theme Path* field, enter the path of the directory for the customer-defined changes to the *Theme* (`/SAP/BC/BSP/SAP/public/customtheme`, in this example).

8. Choose to generate the Web interface.

Test the result

9. Execute the Web interface. You should see the same visual appearance as before, but this time the system uses the copy.

10. You can check the result by displaying the source code and searching using the name of the CSS class. These should contain the path to your directory that contains the customer-defined changes to the *Theme* (`.../customtheme`, in this example).

    The input help for the *Theme Path* field does **not** show this path.

4.2.1.2.8.4.2 CSS Classes for Web Interfaces (Old Design)

Definition

Cascading style sheets (CSS) are used to format HTML pages and make the logical separation of content and external layout easier. Cascading style sheets contain classes, the attributes of which determine the format definition of HTML elements. SAP delivers predefined classes for formatting the Business Server Pages generated by the Web Interface Builder. These are described below.

Use

If you have the necessary experience with the relevant technologies (CSS and HTML), you can change the format definitions contained in the CSS classes and modify the design of the Business Server Pages generated by the Web Interface Builder to suit your requirements.

> You can format the pages so that the color and font correspond to your company's corporate design guidelines. Other modifications are also possible, for example, changed spacing and font sizes if you want to use the generated Web applications on monitors that have different resolutions.

The CSS classes for Web interfaces are contained in a predefined style sheet that is delivered as an object in the MIME Repository. You **cannot** change the predefined style sheet. But you can assign a copy of this predefined style sheet to a Web interface opened in the Web Interface Builder and then edit the style classes contained in the copy. For more information, see Modifying CSS Classes for Web Interfaces (Old Design) [Page 308].

Structure

The style sheet delivered by SAP contains the class definitions listed below and additional style strings of individual HTML tags.

The SEM-BPS design variants listed in the table are based on the old SEM-BPS design.

The different columns have the following significance:

- **HTML tag**: HTML tag, the appearance of which is modified by the specified CSS classes as compared to the HTML standard.
- **SAP class**: Central CSS class delivered by SAP with the Web Application Server. If a derived BPS class is not available, this class is used in the generated Web interface; if you want to change the style of your Web interface, you can modify this class.

- **Additional style strings**: Differences in format definitions between SAP and BPS classes.
- **Derived BPS class**: CSS class delivered specifically for implementation in the Web Interface Builder. This class is derived from the SAP class. If a derived BPS class is available, this class is used in the generated Web interface; if you want to change the style of your Web interface, you can modify this class.

HTML Tag	**SAP Class**	**Additional Style Strings**	**Derived BPS Class**	**Description**
General				
body	sapBdy			document body
textarea	sapTedWhl			document
a	sapBtnStd			pushbutton (Internet Explorer)
Tab pages				
td	sapTbsTabSel			active tab
td	sapTbsTabBlk			space between tabs
td	sapTbsTab			inactive tab
a		text-decoration:none	bpsTbsLnk	hyperlink on inactive tab
table	sapTbsWhl			table (1x1) for selected element
Texts, hyperlinks				
span, a	sapTxtStd, sapTxtH1, sapTxtH2, sapTxtH3,			text (span), hyperlink (a)
span, a	sapTxtEmp, sapTxtLbl, sapTxtLblSml, sapTxtLeg, sapTxtRef			text (span), hyperlink (a)
span	sapLnk			hyperlink (inside of **<a>**)
Group fields				
table	sapGrcWhl			outermost table (1x1)
table	sapGrpWhlBox			inner table (2 rows: header area, data area)
table	sapGrpHdr			header area (table, 1 row)
td	sapGrpTtlBox			table cell in header area
div	sapGrpBdyBox			group body

Input layout (old SEM-BPS design)				
table	sapTbvStd		bpsLay	table for input layout
td	sapTbvHdrAlt		bpsLayHdrAlt	header area cells in lead columns, cells for foot notes, empty layout, layout error
td	sapTbvHdrAlt	text-align: right	bpsLayHdrAltKfg	header area cells in data columns
td	sapTbvHdrStd		bpsLayHdrStd	header area cells (output layouts)
td	sapTbvCellStd	text-align: right	bpsLayCellStd	data area cells (incl. totals), header area cells (input layouts)
Input layout (new SEM-BPS design)				
td			bpsLay2chaText	header area cells in the lead columns
td			bpsLay2stdItemCol	header area cells in data columns
td			bpsLay2aggItemCol	header area cells in data columns, totals columns
td			bpsLay2stdItem	lead column cells (also in new rows): standard header area cells
td			bpsLay2aggItem	lead column cells (also in new rows): totals cells
td			bpsLay2HLevel0, bpsLay2HLevel1, bpsLay2HLevel2, bpsLay2HLevel3	lead column cells (also in new rows): hierarchy cells
td			bpsLay2stdData	data area cells (also in new rows): standard cells
td			bpsLay2stdDataOdd	data area cells (also in new rows): color change cells
td			bpsLay2aggData	data area cells (also in new rows): totals cells
input	sapEdfTxtEnbl	text-align: right	bpsLayEdfKfg	input of key figure values in data cells
input	sapEdfTxtEnbl	background-color: #FFD8D8; text-align: right	bpsLayEdfKfgErr	input error with key figure values
input	sapEdfTxtEnbl		bpsLayEdfChvl	input of characteristic

				values in lead column
input	sapEdfTxtEnbl	background-color: #FFD8D8	bpsLayEdfChvlErr	input error with characteristic values
select	sapDdlWhl		bpsLayDdl	characteristic value selector
span	sapDdlTxt		bpsLayDdlTxt	row in dropdown box
select	sapDdlWhl	background-color: #FFD8D8	bpsLayDdlErr	input error in characteristic value selector
Selectors				
select	sapDdlWhl		bpsSelDdl	selector
span	sapDdlTxt		bpsSelDdlTxt	row in dropdown box
select	sapDdlWhl	background-color: #FFD8D8	bpsSelDdlErr	input error in characteristic value selector
input	sapEdfTxtEnbl	text-align: right	bpsSelEdf	input field for selector
input	sapEdfTxtEnbl	background-color: #FFD8D8; text-align: right	bpsSelEdfErr	input error in selector
table			bspSelGrp	table to include radio buttons
td			bspSelGrpTxt	text for radio buttons
Selector for hierarchies				
table	sapTreWhl			general hierarchy
div	sapTreNl1, sapTreNl2, sapTreNl3, sapTreNl4			hierarchy levels
div			bpsTreSel	selected entry
span	sapTreExpNc			nodes (no symbol; not used)
span	sapTreExpOp			nodes (symbol "expanded"; not used)
span	sapTreExpClo			nodes (symbol "collapsed"; not used)
span			bpsTreOpt	text of a hierarchy entry

Note the following when modifying the classes:

- You can switch individual customer-defined classes on or off by including or excluding the appropriate classes in a comment in the style sheet.
- For elements that have a derived BPS class, you should make any modifications in this derived class (and not in the higher-level SAP class).

- For more information about using these classes, see the HTML source code of the Web page.
- You cannot change the CSS class that the Web interface uses without modifying the generated BSP application. To modify the style, use the standard classes mentioned above.

4.2.1.2.8.4.3 Modifying CSS Classes for Web Interfaces (Old Design)

Use

If you have the necessary experience with the relevant technologies (CSS and HTML), you can modify the CSS classes to adapt the design of the Business Server Pages generated by the Web Interface Builder to suit your requirements.

Prerequisites

An application for editing cascading style sheets is registered on your client PC (file name extension **.CSS**). If you do not have a special editor, a simple text editor is also suitable.

Procedure

Edit CSS Classes

1. In the Web Interface Builder, choose (*Goto* → *MIME Objects*). The *MIME Repository Browser* dialog box appears. The MIME objects that have already been assigned to the Web interface are listed here.
2. Choose *Import CSS Template*. The system generates a copy of the predefined style sheet (`custom.css`) and adds an entry to the list of MIME objects.
3. Select the new list entry and choose *Edit*. The system starts the application that is registered on the client PC to edit cascading style sheets.
4. Only comment out the classes that you want to modify. Make the required changes and save. The system automatically transfers your changes as a new active version of the style sheet.

   For more information, see CSS Classes for Web Interfaces (Old Design) [Page 304].

Reuse Changed CSS Classes

Changes made to the delivered CSS classes only affect the Web interface that you are currently editing. Other Web interfaces are not affected by the changes. However, you can assign a changed style sheet to other Web interfaces so that the changes you make centrally are applied to all assigned Web interfaces. To do this, proceed as follows:

1. In the Web Interface Builder, open a Web interface for editing to which you want to assign a style sheet of another Web interface (master interface).

2. If you want to display the list of properties for the page and their values, double-click on an element of type page.
3. For the attribute *Additional CSS File* in the *Value of an Attribute* column, choose the input help (F4). The *MIME Repository Browser* dialog box appears. The MIME objects that have already been assigned to the Web interface are listed here.
4. Choose *Other Folder*. The system lists the Web interfaces that are available in the system.
5. Select the master interface with the style sheet that you want to assign to the current Web interface. The system displays the MIME objects of the Web interface you entered in the *MIME Repository Browser*.
6. Select the list entry that represents the style sheet of the master interface and confirm. The system assigns the style sheet of the master interface to the current Web interface.
7. Repeat the last steps for each page of the Web interface.

HTML style sheets can only be assigned to one individual page and not a Web interface as a whole. If you want to modify the layout of a planning application that includes several HTML pages, you have to assign the required style sheet to each page of the Web interface individually.

Example

To change the appearance of a tab page, you have to modify the following CSS classes:

- **`sapTbsTabSel`**
- **`sapTbsTabBlk`**
- **`sapTbsTab`**
- **`sapTbsLnk`**
- **`sapTbsWhl`**

In order to apply the changes, you have to search for the appropriate classes in the style sheet, activate them by removing the comment rows and then set the required attribute values.

Result

The planning application that you generate on the basis of the current Web interface (using the Web Interface Builder) uses the format definitions of the style sheets you have assigned and modified for the graphical formatting of the page elements.

4.2.1.2.8.5 Enhancements for Web Interfaces

Use

When you create a Web interface using the Web Interface Builder, the system generates a BSP page in accordance with the settings in Customizing. The BSP event handlers for the *OnInitialisation* and *OnInputprocessing* events are implemented in ABAP class CL_UPWB_BSP_APPL. You can modify the standard system behavior or add new functions.

Prerequisites

You have derived a customer-defined subclass from class CL_UPWB_BSP_APPL.

Features

You can modify the following methods of class CL_UPWB_BSP_APPL.

Methods of class CL_UPWB_BSP_APPL that can be redefined

Technical Name	Description
LOAD	This method allows you to add tasks that are executed once only when the initial execution is performed. This method allows you to access all the components related to a Web interface. Example: Setting a layout in display mode.
DISPATCH	This method forwards the individual events for a component. It is called at least once for each component related to a Web interface. It is called once for each task of the higher-level processes (initialization or input processing, for example).
INITIALIZATION	This method is called for the *Initialization* event of a BSP page.
INPUTPROCESSING	This method is called for the *Process After Input* (PAI) event of a BSP page.
FILTER_MESSAGES	This method allows you to filter or change messages. Example: Displaying either warnings or error messages only.

4.2.1.2.9 Status and Tracking System

Use

The BW-BPS Status and Tracking System helps you to monitor the progress of the different planning tasks within your company. The system keeps a record of the iterations which run through a specific planning task in practice (requirement notification bottom-up, resource assignment top-down) and brings together the planning objects created in the planning environment with the employees who use these objects to process the actual planning tasks.

In the Status and Tracking System, the planning process is divided into:

- *Subplan*:
  A specific business subarea of business planning, for example profit planning, balance sheet planning, cost center planning.
- *Planning session*:
  A single program run of the iterative planning process where every iteration aims to

better adjust the deviating plan values (for example, requirement notification vs. resource assignment, sales key figures vs. sales targets). Planning sessions are used in the Status and Tracking System to version the different planning cycles so that the approximation process is documented in the system and can be understood. The status of a particular subtask can change several times within a planning session (for example new, for approval, rejected, for approval, approved). A planning session is completed when all the subtasks contained in it have the status “approved”.

- *Organizational hierarchy*:
  Part of the hierarchical organizational structure of your company which describes the employees involved in the planning process and how they relate to each other within the business. Each hierarchy node represents an area of responsibility for a specific subplan within the planning process. The hierarchy also serves as a guideline for the approval process within a subplan and for performing status-dependent notification management.

The Status and Tracking System is a Web-based application which is executed exclusively in a Web browser. This means that you can execute your tasks, supported by the system, wherever you are without requiring special local installations on your PC. You perform Customizing for subplans and the associated planning sessions in the BW system.

Integration

The Status and Tracking System acts as an organizational layer above the planning objects (planning areas, planning levels, planning functions) created in the BW-BPS planning environment. The planning environment makes available the basic planning elements that you require while the Status and Tracking System answers the question of how these objects are used and what their status is.

You can branch directly to any of the planning applications specified in Customizing (as long as it is a Web application) from the hierarchical levels of a subplan. For example, this is possible with planning applications created using the BW-BPS Web Interface Builder [Page 285].

You can use the BW-BPS Status and Tracking System together with BI Integrated Planning. For more information see Integrating BI Integrated Planning [Page 341].

Prerequisites

- You have created planning objects in the planning environment [Page 90] that you arrange together from an organizational perspective using the Status and Tracking System.
- You have activated the Web services used by the Status and Tracking System. For more information, see Maintaining Web Services for STS [Page 314].
- The status administration of the Status and Tracking System contains an automatic notification system which is controlled by the current status of a subplan. Notification is by e-mail. Therefore you have to maintain the e-mail addresses of the persons to be notified in the user data in the system.
- Before you can work with the Status and Tracking System the necessary notifications have to be entered in your authorization profile. For more general information, see Authorizations [External]. For more specific information, see Authorization Objects in the STS [Page 313].

Features

You can define subplans and planning sessions freely. Each subplan is assigned to one planning area. The planning area contains the relevant data for the subplan. You can assign several subplans to one planning area. This allows you to achieve the general goal of integrated planning more easily than the option of using a large number of planning areas which contain a very specialized set of key figures and characteristics.

Within a subplan and planning session that is to be completed within it you enter the following data for each unit relevant to your organizational structure [External] (along a cost center hierarchy that you have determined which is available in SAP Business Warehouse, for example):

- Person responsible:
  A person who is responsible for planning within the respective organizational unit (for example, a cost center manager). If the required data is maintained in the system and the corresponding Business Add-In [External] designed to read this data is implemented, the Status and Tracking System can determine the person responsible automatically. For more information, see Determining a Hierarchy for the Subplan [Page 322] and Authorization Objects in the STS [Page 313].
- Completion time:
  The time by which planning in the corresponding planning session must be completed and reported to the next higher-level unit.
- Planning transaction:
  URL of the Web planning application that is executed by the respective organizational unit within the planning session.
- Control report:
  URL of a Web report [External] stored in the intranet that allows you to check the current planning status.

You also maintain a record of header characteristics for each planning session within a subplan. The values that you select for the header characteristics determine the limits of the dataspace within which the respective planning session is planned.

You use the header characteristics to restrict a planning session to a certain version, time frame (for example, 001.2002 to 006.2002), or subquantity of the article groups to be planned (for example, dairy products, desserts). As soon as the planning session is completed, the Status and Tracking System locks the plan data entered for the selection of header characteristics against further changes.

Once configured, you make the Status and Tracking System available to the person responsible for planning. When the status of a subtask is confirmed, the system automatically generates an e-mail to the persons responsible for the next higher-level unit. If the reported plan data is rejected, an appropriate e-mail is also sent automatically with the request for revision to the lower-level unit. You can change the notification texts SAP deliver or replace them with your own texts. See Changing Notification Texts [Page 329].

See also:

Customizing the Status and Tracking System [Page 316]

Overview: Planning with the Status and Tracking System [Page 334]

4.2.1.2.9.1 Authorization Objects in the STS

Use

In the Status and Tracking System, three authorization objects are assigned to the following user groups:

Authorization Objects	**User Groups**
R_STS_PT	Authorization for the planning session and subplan for the person responsible or their substitutes.
R_STS_SUP	Authorization for special access to the Status and Tracking System for planning coordinators.
R_STS_CUST	Execution of Customizing for the Status and Tracking System for technical administrators.

Features

Technical Administrator

The technical administrator is responsible for executing Customizing for the STS.

> You want to perform planning for a cost center. First the technical administrator creates a cost center hierarchy in the BI system and a subplan in the Status and Tracking System. He or she imports this hierarchy into the STS for the subplan.
>
> The technical administrator can then assign a person responsible to each cost center in the cost center hierarchy. The person responsible can also be represented by a substitute.

For more information, see Customizing the Status and Tracking System [Page 316].

Person Responsible

Persons responsible are assigned to an area of responsibility (hierarchy node) within a subplan. They are either assigned to their own area of responsibility directly or are assigned as a substitute. They use the STS to:

- Jump to their own planning application
- Set a status for their own planning
- Monitor and set statuses for any people who report to them.

> The person responsible for node cost center 4711 of the “cost center planning“ subplan logs on to planning session 1 in the STS. This allows him/her to obtain the detailed view for the cost center.

For more information, see Organization of Planning Tasks [Page 336].

Planning Coordinator

Planning coordinators monitor the whole planning process for at least one subplan and therefore possess special access rights in the STS.

> For monitoring the subplan “cost center planning“ in planning session 1, the planning coordinator has a subplan and planning session-related monitor at his or her disposal. As the initial view, this monitor shows an overview of the entire cost center hierarchy with detailed information. The planning coordinator also has the option to change to a table view. Various filters are available in this view. In both views the planning coordinator has the ability to branch to a detailed view of an individual cost center (area of responsibility).

For more information, see Organization of Planning Tasks [Page 336].

See also:

Planning with the Status and Tracking System [Page 334].

4.2.1.2.9.2 Maintaining Web Services for STS

Use

SAP delivers the Web services [External] that are needed to use the Status and Tracking System in an inactive state. The reason for this is that in an unprotected system and network infrastructure, third parties could gain unauthorized access to your system through the internet. Before starting up the Status and Tracking System it is therefore necessary to activate the relevant Web services.

The URL of the Status and Tracking System, which is sent in a start e-mail, opens the Status and Tracking System. By default this is opened in the log-on language of the administrator that sent the start e-mail. You can modify the relevant service so that the user can select a log-on language that is not the same as the administrator language. You can also change the initial passwords for the first log-on by modifying the services.

Prerequisites

Make sure that your company’s system and network infrastructure is adequately protected to prevent unauthorized access to your system. This includes measures such as firewalls, supporting secure protocols (HTTPS), secure passwords for logon to network and to the system, and restrictive authorization profile.

Procedure

Activating Web Services

Activate the Web services used by the Status and Tracking System. Proceed as follows:

1. In the SAP menu, choose *Tools* → *System Administration* → *Administration* → *Network* → *HTTP Service Hierarchy Maintenance (Transaction SICF)*. The *Maintain Service* screen appears.
2. Choose *Find*.

3. Enter **`tunguska`** as the search term and confirm. The system expands the HTTP service tree and displays the entry *Status and Tracking System*.
4. In the context menu of the entry, choose *Activate Service* and confirm the subsequent system message.
5. Repeat this step for the service **`tunguska_detail`** (*Detail Screen: Status and Tracking System*).

Modifying Web Services

You are able to modify the Web services used by the Status and Tracking System. Proceed as follows:

1. Call the dialog box for changing the required service by double-clicking on the service.
2. Choose ✎ *Change.*
3. Make the desired changes on the relevant tab pages. Take note of the following:

   Tab Page: *Service Data*

   If, instead of the standard selection, you select SSL (*Secure Socket Layer*) under *Security Requirements*, the SSL service has to be active on the Web Application Server.

   Tab Page: *Error Pages* → Tab Page: *Error Pages*

   If, instead of the standard *Explicit Answer Page* selection you select the *Redirect to URL* option, the Single Sign On Service (SSO) has to be activated on the Web Application Server.

   If you have chosen *Redirect to URL*, the following URL is displayed by default as changeable in the input field under the *Redirect* radio button:

```
/sap/public/bsp/sap/system/login.htm?sap-
url=<%=PATHTRANS%>&BspLanguage=EN&BspLanguageVisible=X&BspChange
PasswordVisible=X&BspClient=000&BspClientVisible=X&BSPAccessibil
ityVisible=X
```

The following table explains the components of the URL and the additional parameters in detail:

URL Component	What You Need to Know
`/sap/public/bsp/sap/system/login.htm?sap-url=<%=PATHTRANS%>`	URL for redirect. This part should remain unchanged.
`&BspLanguage=EN`	By default, English is used as the log-on language. (However this is not displayed).
`&BspLanguageVisible=X`	An additional box is displayed when you log-on that allows you to select the language. This, together with the `&BspLanguage=EN` option, means that English is predefined and another language can be chosen using the selection box.
`&BspChangePasswordVisible=X`	A switch is also displayed that allows users to log-on and change their passwords.
`&BspClient=000`	Client 000 is used by default. (However this is not displayed).

&BspClientVisible=X	An input field for the client is also displayed. This, together with the &BspClient=000 option, means that client 000 is the preselection in this field.
&BSPAccessibilityVisible=X	

As the administrator, you can configure the system according to your own guidelines by using the parameters.

For more information about configuration options, see SAP Note 498936 or System Log On [External].

4.2.1.2.9.3 Customizing the Status and Tracking System

Use

You use this function to create or edit the basic elements with which the Status and Tracking System works. As a result of this, you adjust the Status and Tracking System to the conditions in your company.

SAP delivers predefined elements for some element types (for example, status values) that are integrated in the internal function flow. You create the required elements for other element types yourself since this requires knowledge of the specific conditions and responsibilities in your company.

Prerequisites

The planning area of the characteristics and key figures that you access in the Status and Tracking System must be a basic planning area. You cannot use multi-planning areas [External] in the Status and Tracking System.

In order to assign a hierarchy to a subplan, you must have created a hierarchy on the desired characteristic in the SAP Business Information Warehouse.

Functions

You define the subplans you require (for example, “sales planning EMEA 2002”, “budget planning marketing 2002”) and the different planning sessions (for example, “requisition note (bottom-up)”, “resource distribution (top-down)”).

Subsequently, you assign the desired planning objects (planning area, hierarchy-bearing characteristic) and the subplans you defined to each other. The following applies for the relationship between planning areas and subplans:

- A subplan is always assigned to exactly one planning area.
- Several subplans can be assigned to one planning area.

These rules implement two general modeling principles for working with BW-BPS.

- A planning area should always serve as the solution to a clearly distinguished business task as it is also described by a subplan.
- The fact that a planning area can be referenced by several subplans takes the goal of plan integration into account. For example, within profit planning the sales plan figures

gained also flow into the income statement planning. For this reason, it is suitable to combine both subplans at least partly in the same planning area.

The assignment of a hierarchy-bearing characteristic that contains the master data of the relevant organizational structure of your company is a central setting in the Customizing of the Status and Tracking System. As a result of this, you transfer the hierarchical structure of a part of your organizational structure (for example, by a hierarchy that is defined for the characteristic cost center) to the planning process within a subplan. This means the subplan is split by the assignment of a hierarchy into as many individual tasks as the hierarchy contains nodes.

See also:

Executing Customizing [Page 317]

Adjusting Notification Texts [Page 329]

4.2.1.2.9.3.1 Executing Customizing

Use

You execute the Customizing of the Status and Tracking System in order to create subplans and planning sessions, to schedule them chronologically, and to make an assignment between individual planning tasks and the employees responsible for them.

Procedure

The different settings which you make in Customizing, can be divided into two areas:

- General settings
  Here you create basic elements that you require to control the planning process with the Status and Tracking System (for example subplans, planning sessions). In addition, you assign a hierarchy to the subplans.

  Executing the general settings is a prerequisite of being able to execute subsequent settings for the planning session.

- Settings for the planning session
  Here you make settings that refer to a certain planning session within a subplan. In particular, you assign your employees to planning tasks to be fulfilled within the planning session.

You make all the settings described below in the initial screen for Customizing of the Status and Tracking System. To do this, choose *Business Planning and Simulation* → *Status and Tracking System* → *Customizing* from the *SAP Easy Access Menu* screen.

General Settings

Subplans

1. Select the option *Define Subplan* and choose *Execute*.

   You get to the view maintenance for subplans.

2. Enter the names of the subplans that are relevant for your company.

Planning Sessions

1. Select the option *Define Planning Session* and choose *Execute*.

   You get to the view maintenance for planning sessions.

2. Enter the names of all required planning sessions.

   Here you enter all names for planning session in all subplans relevant for your company. You establish the connection between subplan and the planning session intended in it subsequently in the setting for the planning session.

Hierarchies

1. Select the option *Determine Hierarchy for Subplan* and choose *Execute*.

2. Make the desired settings. For more information, see Determining a Hierarchy for the Subplan [Page 322].

Settings for Planning Session

1. Choose the desired combination of *Subplan* and *Planning Session* for which you want to execute the subsequent settings.

2. Select the option *Determine Attributes for Plng Session* and choose *Execute*.

   The system displays a dialog box to set up the attributes. You can make the following settings here:

   - *Determine Planning Direction*
     Determines the direction in which the assigned organization structure is run during the planning session: top down or bottom up. How the e-mails are sent, and which status is available for the subtasks within the planning session depends on this setting (for example, the status “approved” is only available in bottom-up planning). Note that with top-down planning the start e-mail is sent to the person responsible at the uppermost level of the hierarchy. With bottom-up planning, the start e-mail is sent to all responsible persons who have been maintained for the subplan hierarchy.

   - *Determine Planning Sequence*
     Determines whether it is possible to assign a global planning sequence [Page 267] to status switches. This is then automatically executed when an appropriate status switch occurs. For example, you could assign a sequence to the status switch from “for approval” to “approved”, which copies the approved data into another version. You can choose whether the global planning sequence is to be executed synchronously or as a background job when a status switch occurs.

     💡

     Only the global planning sequences existing in the system are available as a function which should be executed on a status switch.

   - *Automatic E-Mail Dispatch Active*
     Determines whether the automatic e-mail notification system should be used when status switches occur or not.

   - *Use Time*
     Determines whether the time of completion is defined with a time and time zone as well as a date. See Determining Date, Person Responsible, Layouts [Page 324] .

Taking account of the time zone only has an effect on monitoring the completion times. Time data that the system logs during the planning session (for example, time stamp for status switch or comments) are always saved in system time.

    - *Use Cascading Status Switch*

      By cascading status switch we mean a status switch that is assigned to a particular level of a hierarchy and is passed down the hierarchy from that level. The starting point can be any characteristic node that has further characteristic nodes arranged under it. The cascading status switch only affects the lower-level nodes and does not change the initial node. In contrast to a normal status switch, the status of the lower-level nodes only is set with a cascading status switch. No global planning sequences are executed and no e-mails are sent.

    - *Enter Text Element*

      A text element can be created with additional information on the planning session of a subplan. See Maintaining Text with Additional Information [Page 333].

  - *Enter Theme Root*
  - You can use different stylesheets in each subplan and planning session. For this purpose you enter the desired theme root here. SAP makes different, predefined theme roots available, for example, SAP_STANDARD, SAP_TRADESHOW or SAP_CHROME. If you do not make an entry here, the theme root set as the standard theme root in the system is used. You can find additional information on using your own stylesheets under Business Server Pages [External].

3. Select the option *Initialize/Reset Planning Session* and choose *Execute*.

   The system sets the status of all planning tasks in the planning session to “New”. In addition, the following objects are deleted (provided that they exist):

   - Comments on planning tasks
   - Write locks for already entered and approved plan data
   - Editing or status history on planning tasks

4. Select the option *Define Header Chars for Planning Session* and choose *Execute*.

   The system displays a dialog box in which you can select characteristics. You have already determined the planning area and a particular InfoCube by assigning a hierarchy to the selected subplan. The dialog box allows you to select those characteristic values of this InfoCube for which plan data is to be entered in the chosen planning session.

   The selection that you set for the header characteristics is used by the system to delimit the dataspace which is locked against changes after the completion of a planning session. If possible, you should only make a restriction to central characteristics, typically to the version- and time-bearing characteristics.

5. Select the option *Determine Date, Person Resp., Layouts* and choose *Execute*.

   The system displays a maintenance screen in which you can make numerous settings. For more information, see Determining Date, Person Responsible, Layouts [Page 324].

6. Select the option *Test Call with Display of URL* and choose *Execute*.

The system displays a dialog box in which the URL [External] that refers to the monitor of the Status and Tracking System is displayed. Choose *Test Call* to start the Web browser with the URL specified.

7. Select the option *Send Start E-mail* and choose *Execute*.

   The system generates an e-mail in which all employees involved in the planning session are informed that the planning session was started. You have the option to send this e-mail as a background job and schedule the start to take place immediately. In addition, the notification texts [Page 329] delivered by SAP contain hyperlinks to the direct start of data entry or to the start of the Status and Tracking System. You can change the predefined text of the e-mail to suit your requirements. For more information, see Adjusting Notification Texts [Page 329].

   The system automatically sends the e-mail to the persons responsible for planning. The e-mail addresses are used which you have maintained in step 5. Depending on whether it is a bottom-up or top-down planning session, the system proceeds differently:

   - Bottom-up: The start e-mail is immediately sent to all persons responsible for planning.
   - Top-down: The start e-mail is only sent to the person who is entered as the person responsible at the highest level of the assigned organizational hierarchy. The persons responsible for the next lowest hierarchy level are only informed when there is a status switch to *Completed & Released*.

Reminder Function

In order to make sure that a planning session is completed at the planned deadline, you can schedule a report which checks the status of subtasks which belong to the planning session at the time of execution. You can use this to search for planning tasks that do not have the status *Approved*. If such planning tasks are found then the system automatically sends an e-mail to the persons responsible to remind them about completing their tasks. You can change the predefined text of the e-mail to suit your requirements. For more information, see Adjusting Notification Texts [Page 329].

1. In the Customizing transaction of the Status and Tracking System, choose *Goto? Administration ? Reminder function* (or *Shift* + *F9*).
2. Or proceed as follows:
   a. From the *SAP Easy Access* screen, choose *Tools* → *ABAP Workbench* → *Development* → *ABAP Editor* (transaction **SE38**).
   b. Enter **UPS_STS_REMINDER** as the report.
   c. Choose *Execute* (F8).
3. Enter the desired criteria in the selection screen.
4. Choose *Goto* → *Variants* → *Save As Variant*.
5. Enter a name for the variant and save the variant.
6. Using *System* → *Services* → *Jobs* → *Define Job*, define a job with which you want to execute the report **UPS_STS_REMINDER** with the variant already created at the desired point in time.

   SAP recommends that you schedule the report so that it runs some days before the due date you defined for a planning session. This way, you give the persons responsible opportunity to complete the planning tasks not yet executed on time. For more information on scheduling and background processing of jobs, see Computing Center Management System [External] or Background Processing: Concepts and Features [External].

As an alternative to scheduling a report, you also have the option to execute the reminder function directly. To do this, choose *Execute* (F8) from the selection screen instead of following steps 3 to 5 listed above.

Copying Settings to Planning Sessions

You can use the settings which you have made for planning sessions to create further planning sessions. In this way, the Customizing effort of the Status and Tracking System can be considerably reduced.

To facilitate the reusability of Customizing, SAP delivers an auxillary program. You can start this in the Customizing transaction of the Status and Tracking System using *Goto? Administration ? Copy* (or *shift + F6*), or in the ABAP Workbench (transaction **SE38**):

- **UPS_STS_COPY_CUSTOMIZING**: Copying Customizing. You can find additional information by clicking on the info button.

When executing this program, you enter the subplan and the planning session whose settings you want to transfer, as well as the subplan and the planning session into which you want to copy the settings. Customizing that was already defined for the new planning session is deleted by this.

Deleting Subplans and Planning Sessions

You can either delete complete subplans or individual planning sessions.

To make it easier to delete subplans and planning sessions, SAP delivers an auxillary program. You can start this in the Customizing transaction of the Status and Tracking System using *Goto? Administration ? Delete* (or *shift + F2*), or in the ABAP Workbench (transaction **SE38**):

- **UPS_DELETE**: Deleting subplans and planning sessions. You can find additional information by clicking on the info button.

When you execute this program you either specify the subplan that is to be completely deleted, or the subplan and planning session that is to be completely deleted.

Transporting Subplans and Planning Sessions

You are able to transport subplans with the desired planning sessions.

To make it easier to transport subplans, SAP delivers an auxillary program. You can start this in the Customizing transaction of the Status and Tracking System using *Goto? Administration ? Transport* (or *shift + F8*), or in the ABAP Workbench (transaction **SE38**):

- **UPS_TRANSPORT_STS**: Create transport request for Customizing subplans/planning session. You can find additional information by clicking on the info button.

When you execute this program you specify a subplan and at least one planning sesion that is to be transported and choose a transport request.

Result

You have set up the subplans and planning sessions for your business planning and started a planning session.

4.2.1.2.9.3.1.1 Determining a Hierarchy for the Subplan

Use

You establish a connection between process-oriented objects of the Status and Tracking System and data-oriented objects of the planning environment by assigning a planning area, a characteristic, and one of the available hierarchies to the subplan. If necessary, you can choose different versions or validity periods for a hierarchy.

The assignment of a hierarchy-bearing characteristic that contains the master data of the relevant organizational structure of your company is a central setting in the Customizing of the Status and Tracking System. As a result of this, you transfer the hierarchical structure of a part of your organizational structure (for example, by a hierarchy that is defined for the characteristic cost center) to the planning process within a subplan. This means the subplan is split by the assignment of a hierarchy into as many individual tasks as the hierarchy contains nodes. Since the hierarchy is assigned directly to the subplan, it is valid for all the planning sessions that belong to the subplan.

> Even if a hierarchy for a characteristic is normally always used as a hierarchy that represents the organizational structure of your company (or a part of it), such a choice is not imperative. You can also a hierarchy for a characteristic that represents, for example, article groups or regions if the responsibilities in your company are oriented towards such categories.

Prerequisites

You can use BW hierarchies as hierarchies that are defined for the characteristics of the planning area that you want to use when editing the subplan. To determine a hierarchy, at least one of the characteristics in the planning area that you assign to the subplan must have a hierarchy. A hierarchy that you want to assign to a subplan must have the following attributes:

- The planning area must be a basic planning area You cannot use multi-planning areas [External] in the Status and Tracking System.
- The hierarchy should contain an **individual** top node at the uppermost level. The top node must be a text mode. This serves as the heading of the hierarchy.

> This prerequisite is not fulfilled for hierarchies that you have extracted from an SAP System into SAP BW. But you can insert a text node as a top node for such an extracted hierarchy subsequently. You use the SAP BW hierarchy maintenance for this.

- Only one individual characteristic node can be assigned directly under the text node. Additional characteristic nodes can be attached and additional levels inserted under this individual characteristic node. Text nodes are not permitted at these lower levels.
- Each characteristic node can only be used once in the hierarchy.
- Interval nodes are not allowed.

The hierarchy-bearing characteristic must be contained in the planning area that you assign to the subplan. But it is not necessary that this characteristic is also used in the planning levels.

Procedure

1. Choose *Business Planning and Simulation* → *Status and Tracking System* → *Customizing* from the *SAP Easy Access Menu* screen.

   The system displays the initial screen for Customizing the Status and Tracking System.
2. Select the option *Determine Hierarchy for Subplan* and choose *Execute*.

   The system shows the screen for hierarchy settings.
3. Enter which *Planning Area* you want to use.
4. Enter the *Characteristic* the hierarchy of which you want to assign to the subplan.
5. Enter the *Hierarchy* required for the characteristic.
6. Enter the date up to which the hierarchy you entered is valid.

   If the hierarchy you entered is not time-dependent, you must still enter the validity end date predefined by the system in this case 31.12.9999.
7. Under *Name Resolve*, enter one of the available implementations for the Business Add-In [External] contained in the Status and Tracking System for name resolve.

   The implementations offered here contain functions with which you can determine the assignments available in the system between characteristic values of the hierarchy-bearing characteristic and the system users that are assigned to these characteristic values. A conceivable implementation could, for example, determine the user who manages the cost center for every cost center. If you want to use one of the implementations offered then you no longer need to manually maintain the assignment of responsible persons in one of the later steps of Customizing. Implementations are available for the following characteristics of the SAP Business Information Warehouse:

   - Organizational Unit (0ORGUNIT) [External]
   - Funds Center (0FUNDS_CTR) [External]
8. Enter the *Subplan* to which you want to assign the settings made in the previous steps.

You can save the current settings permanently by creating a variant. To do this choose *Goto* → *Variants* → *Save as Variant* and assign a suitable name to the variant.

Result

You have created the prerequisites in order to make the settings valid for every organizational unit on completion date, responsible person and layout for the detailed arrangement of the different combinations of subplan and planning session. You can make the settings that you have made here visible on the initial screen of Customizing for the Status and Tracking System by selecting the appropriate subplan from the dropdown list, and choosing the pushbutton *Display Assignment*.

See also:

Determining Date, Person Responsible, Layouts [Page 324]

4.2.1.2.9.3.1.2 Determining Date, Person Responsible, Layouts

Use

You determine the date, the planning applications and control reports to be used for a subplan and planning session combination in order to control and monitor the planning process with the help of the Status and Tracking System. On the other hand, settings for the person responsible may only be made once, when the first planning session is determined. These settings affect all planning sessions within the selected subplan and must not be changed once they have been made. If you do make changes to the person responsible settings in subsequent planning sessions, these changes will affect all planning sessions (even those from before).

The following settings can be made for each characteristic node and affect the planning process in the following way:

- *Person responsible*: The persons responsible for a subtask within the planning session are either determined by an automatic analysis of the persons responsible for organizational units maintained in the system, or you must assign the subtasks manually. The e-mail address, which you also enter here, is used in the Status and Tracking System to inform the persons responsible about the start of a planning session or about changes in status, or to remind them of the approaching completion date of their planning task with the help of the integrated notification system.
- Time of completion: For each individual subtask within the planning session you specify the point at which it will be completed. The persons responsible for the subtasks see this when they call up the task assigned to them, and can then orient their schedule based on this. Furthermore, you can also activate an option in the Customizing [Page 316] of a planning session in a subplan which allows you to enter the exact time at which the task will be completed.
- Layouts: Here you are able to determine the URL of the planning application and the URL of the control report.

Prerequisites

You have assigned a hierarchy to the subplan. For more information, see Determining a Hierarchy for the Subplan [Page 322]. In addition, Web interfaces must be available which you assign to the subtasks within the planning session.

Procedure

1. Choose *Business Planning and Simulation* → *Status and Tracking System* → *Customizing* from the *SAP Easy Access Menu* screen.

   The system displays the initial screen for Customizing the Status and Tracking System.

2. Choose a subplan and a planning session.

3. Select the option *Determine Date, Person Resp., Layouts* and choose *Execute*.

   The system shows the screen with the hierarchy assigned to the subplan and the appropriate fields for the settings.

4. Choose *Expand Subtree* to expand the hierarchy in the left screen area.

5. Double-click on a node whose settings you want to edit.

6. Enter the system user name of the person responsible in the *Responsible Person* field and choose *Enter*.

   The system determines the name and the e-mail address of the person responsible from the system data (if available), and adds the entries into the fields *Name* and *E-Mail*. If you want to send e-mails you have to activate the appropriate option in the attribute settings for the planning session and ensure that all persons responsible have maintained their e-mail addresses in their user master data.

   Note that addresses entered in the Status and Tracking System are understood to be destination addresses (you can enter more than one e-mail address. Separate them using “;“). However, the system uses addresses in the user master data as sender addresses.

7. Enter the planned *Completion Date* for the subtask. If you are using a completion time, you can also specify a time zone which is valid for these nodes. If you do not specify a time zone the time zone for the person responsible for the corresponding nodes is used, as long as this has been maintained in their user master data. If a time zone has not been maintained in the user master data, the last time zone used for these nodes and this planning session is used, as long as this information is available. If no information on the time zone is available, the time zone for the system is used automatically. Following this logic, you are also able use the desired completion time for all lower-level units of the nodes you have just addressed.

8. Decide whether the completion date entered should also apply to the entries which are arranged below the node in the hierarchy that you are currently editing. Choose *Pass On to Lower-Level Units/Execute* to transfer the completion date entered to the lower-level nodes.

   The term “lower-level unit” refers to the case accepted as standard Customizing that the hierarchy assigned to the subplan belongs to a characteristic whose values represent the organizational units of your company.

9. In the *URL of Planning Application* field enter the HTML address with which the planning layout intended for data entry is started in the Web browser. Here you set up which means the persons responsible should use for Web-based data entry and for data display within the planning session in order to edit their subtask. These means are based technically on the layouts [Page 128] of manual planning [Page 127]. Supported as means of entry and display are:

   - Web Interfaces [External] that have been created with the Web Interface Builder [Page 285]

- ▪ Manual planning with BW-BPS as an application in the mySAP Workplace (in this case you use the URL which is used in the role menu at the start of the planning environment)

💡

- ○ You can reduce the amount of Customizing considerably if you set up the layouts for plan data entry with the help of variables [Page 102] with the replacement type "user-specific values", so that every person responsible is offered the relevant value range for planning (for example, the regions, customers, or article groupsto be edited by the respective person responsible).

10. Specifically, proceed as follows:

    - Specify the URL of a Web interface, if you know it.
    - If not, you can use input help (F4). The system offers for selection those Web interfaces that you have created with the Web Interface Builder.

11. From the list, choose the required interface that you want to assign to the particular nodes for data entry.

    The system inserts the URL [External] into the input field.

    💡

    The system only inserts the application-specific part of the URL. The basic URL which is dependent on the current system environment at runtime is determined dynamically on execution. The application-specific part of the URL displayed here is placed first and transferred in this form to the Web browser.

12. Decide whether the data entry means specified should also be used for the remaining entries of the same hierarchy level. Choose *Copy to All Units of Same Level/Execute* to transfer the specified data entry means to this node.

13. Enter the HTML address in the field *URL of the Control Report* with which the desired Web Report [External] of the SAP Business Information Warehouse is started in the Web browser.

    Specifically, proceed as follows:

    - Specify the URL of a Web report, if you know it.
    - If not, you can use input help (F4). The system provides the reports that you created in the system for selection.

14. From the list, choose the report that you want to use as the control report for the respective nodes.

    - The system inserts the URL [External] into the input field.

    💡

    The selection of a Web report with the input help is only possible when the BW System it is based on has at least release 3.0A. The system only inserts the application-specific part of the URL. The basic URL which is dependent on the current system environment at runtime is determined dynamically on execution. The application-specific part of the URL displayed here is placed first and transferred in this form to the Web browser.

15. Decide whether the Web report specified should also be used for the remaining entries of the same hierarchy level. Choose *Copy to All Units of Same Level/Execute* to transfer the specified Web report to this node.

16. You can name one or more substitutes for the person responsible. Note that substitutes are used per person.
17. Save the settings.

Result

You have fulfilled all the prerequisites in order to start planning in the current combination of the subplan and planning session. You can test calling up the monitor in the initial screen of Customizing for the Status and Tracking System and send an e-mail to all assigned persons responsible to notify them of the start of the planning session.

See also:

Determining a Hierarchy for the Subplan [Page 322]

Executing Global Planning Sequences upon Status Switch [Page 327]

4.2.1.2.9.3.1.2.1 Executing Global Planning Sequence upon Status Switch

Use

In the context of a planning session and subplan, you can determine that when changing a planning task from one status to another, a global planning sequence you have assigned is automatically executed. Possible uses for this are, for example:

- You assign a global planning sequence to the status switch from “for approval” to “approved”. This then copies the approved data into a version intended for this purpose.
- You assign a global planning sequence to the status switch from “new” to “in processing”. This then initializes the plan data with the actual values of the prior fiscal year.

Integration

All those global planning sequences [Page 267] are available that you created in the BW-BPS planning environment [Page 90].

Prerequisites

A global planning sequence that is assigned to the status switch of a planning task is only executed when you have set the *Automatic Function for Status Switch* indicator for the planning session. You make this setting in the Customizing the Status and Tracking System [Page 316] under *Determine Attributes for Planning Session*.

Functions

For every planning task that you define in the context of planning session and subplan, you can determine individually whether a planning sequence should be executed with the status switch of a task. You can define a global planning sequence for every combination of start and end status.

If, in the course of a planning session, the execution of a global planning sequence is triggered by a status switch, the following points have to be observed, according to the specific attribute settings:

- If, under *Determine Attributes for Planning Session*, execution is chosen as a background job, the sequence is not executed directly. Instead, the Status and Tracking System generates a batch job that is scheduled with *Start Date* "execute immediately". This means that delays are possible between the status switch and the execution of the sequence. On the other hand, this procedure ensures that you can also continue working with the Status and Tracking System without any delay if processing the planning sequence takes up more time. You can check the results of the job execution in the job log of the job overview. To do this, choose *System* → *Services* → *Jobs* → *Job Overview.* The Status and Tracking System uses the following naming structure for automatically generated jobs:

  STS_BUNDLE_<Planning sequence>_<Subplan>_<Planning session>

  The technical name is used for the name components **`<Planning sequence>`**, **`<Subplan>`** and **`<Planning session>`**.

- If, under *Determining Attributes for Planning Sessions*, synchronous execution is selected and a global planning sequence determined for the appropriate nodes, you receive a log for the execution of this global planning session when there is a change to the status of these nodes in the monitor of the Status and Tracking System. If errors occur the status of the nodes is not changed.

The assigned planning sequence can contain any planning functions. The system does not check whether the functions contained in the sequence have any connection with the data which is to be edited in the context of the planning task. Therefore, you are responsible yourself for a sensible selection of suitable functions.

Activities

To execute planning sequences with the status switch of planning tasks, proceed as follows:

1. Choose *Business Planning and Simulation* → *Status and Tracking System* → *Customizing* from the *SAP Easy Access Menu* screen.
2. Set the desired combination of *Subplan* and *Planning Session*.
3. Select the option *Determine Date, Person Resp., Layouts* and choose *Execute*.
4. Choose *Expand Subtree* to expand the hierarchy in the left screen area.
5. Double-click on a node whose settings you want to edit.
6. Insert a new row in the group frame *Execute Planning Sequence upon Status Switch*.

   Enter the desired values in the *From Status*, *To Status* and *Sequence* columns. The system offers you the possible values using dropdown lists or input help.
7. Note that here, the value *Any* is a placeholder for all possible statuses.. If, for example, you choose *Any* under the *From Status* and also under the *To Status*, the global planning sequence determined is executed upon any potential status switch.

   The following rules apply for the assignment of global planning sequences to status switches:

   – At most, one global planning sequence can be assigned to every combination of start and end status.

   – The start and end status cannot be identical (exception: *Any*).

8. Repeat the steps specified if necessary for other nodes of the hierarchy or status switches.
9. Save your settings.

4.2.1.2.9.3.2 Adjusting Notification Texts

Use

You adjust the texts delivered by SAP that are used for the notification system in the Status and Tracking System, to the requirements which apply to your company. The system automatically sends the texts to the responsible person by e-mail when the status in the planning process changes. You can change the texts (or replace the delivered texts with your own text objects) in order to convey additional information. Such changes or information could be, for example:

- A brief overview of the planning process in your company so that the employees can assess how their respective subtask reacts to the whole subplan [External].
- Overview of the employees responsible for planning, possibly with entries on their availability (telephone extension, core working time).
- Contact person at IT helpdesk in case of technical problems
- Stylistic adjustment of texts to the usual forms of address in your company.

The system automatically determines how many e-mails are to be sent to which recipient on the basis of the type of status change and the entries on the hierarchy which is assigned to the subplan. Every individual e-mail is individualized at runtime with the help of text symbols, which are contained in the text. The Status and Tracking System supports the following text symbols:

Symbol	Meaning
&version&	current planning session
&subplan&	current subplan
&duedate&	current due date for email recipient’s area of responsibility
&stsurl&	link (URL) to the Status and Tracking System
&plaurl&	link to the planning layout for the email recipient’s area of responsibility
&layout&	link to the BW report for the email recipient’s area of responsibility
&sender&	name of the sender
&receiver&	name of the recipient
&nodetext_receiver&	description of the email recipient’s area of responsibility

	(nodes)
&nodetext_sender&	description of the area of responsibility (nodes) of the person sending the email

If the texts delivered by SAP already fulfill your requirements then you do not need to carry out the steps described below.

Prerequisites

The texts which the notification system uses for e-mails, are SAPscript texts with the document class “general text”. In order to edit or create new texts, you need the corresponding authorizations for the transaction **SE61**. Depending on which type of adjustments you want to make, you require additional authorizations:

- If you want to use your own texts instead of the texts delivered by SAP, then you must enter this in the assignment table **UPS_CUSTTEXT** (package **UPS**). To do this you need the authorization to create and change table entries.
- If you do not want to create your own texts, but want to change the texts delivered by SAP, you require the authorization to execute modifications [External].

The notification system uses the following delivered texts:

Technical Name	**Reason**	**Use**	**Upon Status Switch to ...**
UPS_SEND_APPROVAL	03	Request for approval of completed planning	For approval
UPS_SEND_RECALL	06	Further processing of a task, which was already ready for approval	In process
UPS_SEND_APPROVED	04	Confirmation of approval to planner	Approved
UPS_SEND_CLOSED_TD	09	Request for approval of a closed planning in a top-down planning session	Completed and released
UPS_SEND_REJECTED	05	Rejection of a desired approval	Back to revision
UPS_SEND_REMINDER	07	Request to complete open planning tasks	(manual)
UPS_START_RUNDE	01	Notification about the start of a new planning session to all persons responsible	New
UPS_START_RUNDE_TOPDOWN	08	New start of a top-down planning session	New

Procedure

Adjusting Texts

1. From the SAP Easy Access screen choose *Tools* → *ABAP Workbench* → *Utilities* → *Documentation*.

   The system shows the initial document maintenance screen.

2. Set "General Text" as the *Document Class* with the desired *language*.
3. Decide whether you want to modify a delivered text, or create your own text:

   ○ To create your own text, enter a technical identifier from the customer namespace (**Z***) as the *Name*, and choose *Create*.

   You can make the work easier by using the texts delivered by SAP as a template for your own texts. First enter one of the names mentioned in the section *Prerequisites* above as the *Name* and choose *Document → Copy*. Subsequently enter the name of the new text to be created. In this way, the new text contains all the relevant text symbols for the respective usage. The Status and Tracking System replaces these with current values at runtime.

   ○ In order to modify a delivered text, enter one of the names mentioned in the section *Prerequisites* above, and choose *Change*.

   The system starts the editor for text processing.

4. 4. Make the desired changes, save and activate the text.

   5. Note that, in notification texts with the technical names UPS_SEND_REMINDER, UPS_START_RUNDE and UPS_START_RUNDE_TOPDOWN, and in notification texts that you have created with the same semantic content, the following text symbols cannot be used:

5. **&nodetext_sender&**
6. **&sender&**
7. **&layout&**

Including Your Own Texts in the Status and Tracking System

Normally the Status and Tracking System uses the texts mentioned in the table above in the section *Prerequisites*. If you want to use texts you have created yourself, you must make these known to the system. To assign your own texts to the different statuses of the Status and Tracking System, proceed as follows:

1. From the SAP Easy Access screen, choose *Tools → ABAP Workbench → Overview → Data Browser*.
2. Enter **UPS_CUSTTEXT** as the *Table Name*.
3. Choose *Table → Create entries*.

   The system displays the list of available assignments.

4. Choose *Edit → New Entries*.
5. Enter a reason. As a guideline, you can use the assignments listed above under “prerequisites”.
6. In the *Docu-Obj.* column, enter the name of the text you created that you want to assign to the reason.
7. Save your changes.

   The system uses the notification texts in the language version that is maintained in the user data of the recipient at the time of execution. If the language entry is missing in the user data, or if a language is entered for which no suitable language version of the notification texts is available, then the system uses the English language version. The English language version should be created without fail in order to guarantee optimal system performance. If you execute changes, you should therefore make sure that the changed text is translated for all languages relevant in your company. Here you use the SAP System translation function (from the *SAP Easy Access* screen *Tools* → *ABAP Workbench* → *Utilities* → *Translation* → *Short and Long Texts*).

The standard texts made available by SAP are delivered in all the languages in which SAP BW is available.

Result

The Status and Tracking System will use the texts you created or changed for the automatic notification for future changes of status within a planning session.

Example

The following text shows how you can adjust the delivered text **UPS_START_RUNDE** to the requirements in your company. The changed passages are highlighted in red and underlined; The IDs enclosed in "&" are text symbols which are replaced by current values at runtime:

Dear **&receiver&**,

A new planning session was started. Start your planning for
Subplan: **&teilplan&**
Planning session: **&version&**.

We ask you to complete planning by **&duedate&**. Do not forget to transfer your completed planning to your superior for approval.

For questions on the planning process, contact Dr. Smith (extension: -398)

For technical questions, contact Mr. Adams from the IT Helpdesk (extensions: -653)

You find the link to your layouts under:
&plaurl&.

You find the Status and Tracking System under:
&stsurl&

See also:

Documentation Maintenance Transaction [External]

4.2.1.2.9.3.3 Maintaining Text with Additional Information

Use

Additional information on the planning session for a subplan can be maintained. This is displayed in the detail view in the monitor of the Status and Tracking System.

Prerequisites

Texts that contain additional information are SAPscript texts with the document class “general text“. In order to edit or create new texts, you need the corresponding authorizations for the transaction SE61.

Procedure

1. From the *SAP Easy Access* screen, choose *Tools ? ABAP Workbench* ? *Utilities ? Documentation*.

   The system shows the initial document maintenance screen.

2. Set "general text" as the *Document Class* and the desired language.

3. To create your own text, enter a technical identifier from the customer namespace (**`z*`**) as the *Name*, and choose *Create*.

   The system automatically starts the editor for text processing.

   If you want to use HTML and the graphic PC Editor is displayed, you have to switch to the classic editor. To do this, choose *Goto ? Switch Editor* from this screen. You can use HTML here but have to ignore the warning that is displayed when you activate.

4. Make the desired changes, save and activate the text.

   The system uses the notification texts in the language version that is maintained in the user data of the recipient at the time of execution. If the language entry is missing in the user data, or if a language is entered for which no suitable language version of the notification texts is available, then the system uses the English language version. The English language version should be created without fail in order to guarantee optimal system performance. If you execute changes, you should therefore make sure that the changed text is translated for all languages relevant in your company. Here you use the SAP System translation function (from the *SAP Easy Access* screen Tools → *ABAP Workbench* → *Utilities* → *Translation* → *Short and Long Texts*).

See also:

Documentation Maintenance Transaction [External]

4.2.1.2.9.4 Planning with the Status and Tracking System

Use

You execute planning with the Status and Tracking System in order to structure the planning process. The Status and Tracking System provides the following advantages:

- Possibility of iterative alignment of planning results within a subplan with the definition of different planning sessions
- Clear assignment of subtasks to the persons responsible within a planning session
- Execution of planning taking account of clear deadlines
- Regular planning workflow on the basis of a status administration system.

Prerequisites

You have completed Customizing for the Status and Tracking System for at least one subplan and planning session combination. For more information, see Customizing the Status and Tracking System [Page 316].

Features

The Status and Tracking System is used as an environment in which to both carry out planning and monitor the planning process. You start a planning session by using the *Send Start E-mail* function, either in Customizing of the Status and Tracking System or in the planning coordinator's overview monitor, to prompt the system to send an e-mail to all employees who are taking part in the planning session. If you choose the second option, the system automatically starts a job in the background to send the e-mail and executes this immediately. The system uses notification texts for this e-mail. These contain a link which immediately starts the data entry session you are assigned to in the Web browser.

Authorizations

Different initial screens are displayed for the person responsible and the planning coordinator according to the specific task or area of responsibility.

The person responsible sees the overview page for his or her areas of responsibility first. This is displayed as a table. You can jump to the corresponding detail pages from here using the *Detail* link. However, if the person responsible is only assigned to one area of responsibility, either directly or as a substitute, the corresponding detail page is displayed immediately (see below).

The planning coordinator sees the overview page first. Two views are available: The default hierarchical view and a tabular view that the planning coordinator can switch to. By using the *Detail* link, the planning coordinator can jump to the corresponding detail screen from either of these views.

Overview Page

Tabular View

For each area of responsibility the following information is displayed in the form of a table:

- Current status
- Lock status (🔒 locked/🔓 🔓 not locked)

- Person responsible
- Current processor (bottom-up planning session)
- Status of the completion date (🚩 critical/ 🏳 not critical)
- Completion time
- Comments column, from where you can jump to the comments page (🗨 comments available/ 🗨 not available)
- Detail column.

In the tabular view you can also filter the following fields:

- Area of responsibility
- Current status
- Lock status (🔒 locked/🔓 not locked)
- Person responsible
- Current processor (with a bottom-up planning session)
- Status of the completion date (🚩 critical/ 🏳 not critical)
- Completion time.

Hierarchical View

As well as the tabular view, the following symbols are used:

⛔ *Aggregated Status Not Approved* (with bottom-up planning sessions), or *Aggregated Status Not Completed and Released* (with top-down planning sessions). This means that at least one node under the node in question has not yet been given status *Approved* (bottom-up planning session) or *Completed and Released* (top-down planning session).

⛔ *Aggregated Status Not Approved and Comment(s) Beneath Current Node*. In addition to the information given above, comments have been maintained for at least one of the nodes under the node in question.

✓ *Aggregated Status Approved* (with bottom-up planning sessions), or *Aggregated Status Completed and Released* (with top-down planning sessions). This means that all nodes under the node in question have status *Approved*.

✓ *Aggregated Status Approved and Comment(s)* (with bottom-up planning sessions), or *Aggregated Status Completed and Released* (with top-down planning sessions). In addition to the information given above, comments have been maintained for at least one of the nodes under the node in question.

Detail Page

In the top left-hand area of the screen you see:

- Navigation link *My Area of Responsibility*.

Arranged underneath this are:

- Subplan
- Planning session
- Selected area of responsibility

- Current status of selected area of responsibility with lock status, link to status history, and buttons for status change
- Due date and time (critical/ not critical)
- Person responsible with system user and contact data
- Current processor with system user and contact data (with bottom-up planning sessions)
- *Send E-Mail to* person responsible

Additional information for the respective planning session, if maintained. See Maintaining Text with Additional Information [Page 333].

You will find a number of additional buttons below:

- *Log Off*
- *Open Planning*: Opens the planning application maintained for the selected area of responsibility
- *Open Report*: Opens the control report maintained for the selected area of responsibility
- *Maintain Comments*: Jumps to the comments page for the selected area of responsibility.

A hierarchically structured table is displayed in the bottom part of the screen. The information is structured in the same way as in the hierarchical view of the overview page. However it only refers to the lower-level areas of responsibility. By expanding the hierarchy nodes you can display overview information on those areas of responsibility that are arranged lower down in the hierarchy. From her you can navigate to the leaves of the subhierarchy.

If you have applied the appropriate settings in Customizing, the buttons for a cascading status switch appear under this table.

Comment Page

Comments can be entered for all areas of responsibility. You can also forward these comments to the areas of responsibility that lie directly below. There is a checkbox under the input field for comments for this purpose.

See also:

Planning Using the Status and Tracking System [Page 336]

Modifying Notification Texts [Page 329]

Authorization Objects in STS [Page 313]

Performing Customizing [Page 317]

4.2.1.2.9.4.1 Organization of Planning Tasks

Use

You use this function to execute planning in accordance with the definitions determined by the Status and Tracking System for the current planning session within the subplan to be edited.

Prerequisites

You have completed Customizing for the Status and Tracking System for at least one subplan and planning session combination. For more information, see Customizing the Status and Tracking System [Page 316].

Features

The Status and Tracking System is used as an environment in which you carry out planning and monitor the planning process. As a result, there are different procedures for working with the Status and Tracking System:

Procedure	Owner
Performing planning [Page 338]	People who are assigned to an area of responsibility either directly or as a substitute and who want to perform planning in that area.
Checking planning results [Page 339]	People who are directly responsible for areas in order to check the planning results of those areas of responsibility.
Monitoring the planning process [Page 340]	Planning coordinator

The Status and Tracking System presents the pages that belong to a planning task in an almost identical form, irrespective of whether you want to execute planning or monitor the process. The only difference is that the status you can set for a task depends on the hierarchy level that is assigned to you (whether you are the supervisor or person responsible for executing the particular planning task).

Cascading Status Switch

If you are assigned to a superior level in the organizational hierarchy as a person responsible for a task and you have applied the appropriate settings in Customizing, you can change the status of all tasks that are assigned to you and the employees you are responsible for with one single action. You can perform a cascading status switch of this type along the hierarchy for the following status switches with bottom-up planning sessions:

- From “new” to “in process”
- From “in process” to “for approval”
- From “for approval” to “in process”.

With top-down planning sessions you can perform a cascading status switch for the following status switches:

- “Completed” and “released”
- “In process”

If you have defined planning sequences in Customizing that are to be executed automatically upon status switch, the following rule applies with a cascading status switch: The system does not execute the planning sequences; it only sets the status for the areas of responsibility.

Specifying Times in the Monitor

Note that all times specified in the monitor (such as the time at which comments were entered) are given in the time zone that the logged-on user has maintained in user master data. If no time zone has been maintained, the time zone for the system is used.

An exception to this is completion time: If a time zone has not been maintained, the system first checks whether a time zone has been explicitly specified in Customizing for the particular area of responsibility. If this is the case, the system uses this time zone. If this is not the case, the system uses the time zone for the system.

4.2.1.2.9.4.2 Executing Planning

Procedure

1. Open the e-mail with which you were informed that the planning session was started and click on the link contained in it to start the Web browser.

   Depending on how your system environment is configured, a dialog box is displayed that requests you to enter your logon data. (Also see Maintaining Web Services for STS [Page 314].) This concerns the user names and password for the SAP System in which you operate BW-BPS.

2. The system displays the overview page of the planning session in the Web browser. This is personalized for you. Select the task that you want to edit.

   If only one task exists, the overview page is not displayed. Instead, you go straight to the detail page for processing your task.

3. Choose *Open Planning* to start the planning application for data entry that is assigned to your task. The planning application is started in a new browser window.

4. Execute planning and save your changes.

5. Switch to the browser window again in which the page to edit your task is displayed.

6. If a control report exists for your area of responsibility, choose *Open Report* to view the result of your planning. (Also see Determining Date, Person Responsible, Layouts [Page 324]) The control report is started in a new browser window.

7. Set the status of your task corresponding to the degree of processing achieved:

   - Choose *In Process* to signal to your superior that you have started to edit your task but have not completed it yet.

   - Depending on the planning direction, you choose *Send for Approval* (with a bottom-up planning session) to request that your superior checks the results of your planning and, where necessary, approves them. Choose *Completed and Released* (with a top-down planning session) to request that the persons responsible who are directly below you start their planning.

     💡

     With a bottom-up planning session: You can set a task back to status *In Process* even if you have already set the status *Send for Approval*. To do this, use the *Reset* button. As a result of this, you cancel the change lock for the data concerned at the same time. In addition, this status switch triggers a notification that is sent to the superior.

     💡

     With each non-cascading status switch, the system automatically:

     - saves the status and (depending on the status change)

     - sets or lifts the lock for the selected area of responsibility

- triggers the automatic notification system of the Status and Tracking System, if it has been activated. See: Executing Customizing [Page 317]. Notifications assigned to a status are sent by e-mail. See Adjusting Notification Texts [Page 329].
- starts the global planning sequence for this status switch synchronously or as a background job. This happens as long as global planning sequences have been activated (see Executing Customizing [Page 317]) and a global planning sequence has been assigned to this status switch and your area of responsibility (see Executing Global Planning Sequences upon Status Switch [Page 327]).

8. Where necessary, enter *Comments* for your area of responsibility and, if you are a superior, for the employees who report to you, in order to record any special features. Choose *Save* to save the comments.
9. Choose *Log Off* to close the Status and Tracking System.

It is important that you log off from the planning session explicitly since otherwise the connection to the SAP System application server is kept and the planning objects you used remain locked.

4.2.1.2.9.4.3 Checking Planning Results

Procedure

10. Open the e-mail with which you were informed that planning in one of your assigned areas of responsibility is ready *For Approval / Review*. Click on the link contained in this e-mail in order to start the Web browser with the Status and Tracking System.

    Depending on how your system environment is configured, a dialog box is displayed that requests you to enter your logon data. (Also see Maintaining Web Services for STS [Page 314].) This concerns the user names and password for the SAP System in which you operate BW-BPS.
11. The system displays the overview page of the planning session in the Web browser. This is personalized for you. Select the area of responsibility under which the task that you have to check is to be found.

    If only one task exists, the overview page is not displayed. Instead, you go straight to the detail page for your task.
12. Choose *Open Report* to observe the result of the planning to be checked.
13. In the bottom half of the screen, select the task for which you want to change the status.
14. Set the status of the task corresponding to the degree of processing achieved:
    - Choose *Approve* if the plan data matches your expectations.
    - Choose *Back to Revision* if you think a revision is necessary.

If you have classified a planning task as needing revision then where possible, you should enter a comment briefly explaining the reason to the person responsible.

You can the change a task that you have already approved back to the status *For Approval / Review* (resubmission function). Here the change lock is retained for the data affected. You also have the option of assigning the status *Back to Revision* directly to a task that has already been approved.

With each non-cascading status switch, the system automatically:

- saves the status and (depending on the status change)
- sets or lifts the lock for the selected area of responsibility
- triggers the automatic notification system of the Status and Tracking System, if it has been activated. See: Executing Customizing [Page 317]. Notifications assigned to a status are sent by e-mail. See Adjusting Notification Texts [Page 329].
- starts the global planning sequence for this status switch synchronously or as a background job. This happens as long as global planning sequences have been activated (see Executing Customizing [Page 317]) and a global planning sequence has been assigned to this status switch and your area of responsibility (see Executing Global Planning Sequences upon Status Switch [Page 327]).

15. Where necessary, enter *Comments* for your area of responsibility and, if you are a superior, for the employees who report to you, in order to record any special features. Choose *Save* to save the comments.

16. Choose *Log Off* to close the Status and Tracking System.

It is important that you log off from the planning session explicitly since otherwise the connection to the SAP System application server is kept and the planning objects you used remain locked.

4.2.1.2.9.4.4 Monitoring Planning Processes

1. To start the Web browser for the monitor specific to the required subplan and planning session, click on the link.

   Depending on how your system environment is configured, a dialog box is displayed that requests you to enter your logon data. (Also see Maintaining Web Services for STS [Page 314].) This concerns the user names and password for the SAP System in which you operate BW-BPS.

   The system displays the hierarchical view of the overview page in the Web browser. (See also Overview: Planning with the Status and Tracking System [Page 334]).

2. A number of control and intervention functions are now available to you. For example:

- Sending e-mail to persons responsible or to the current processor of a task
- Entering comments
- Locking or unlocking
- Switching to the tabular view with the option of filtering according to various criteria

3. (Also see Overview: Planning with the Status and Tracking System [Page 334]).
4. If a particular area of responsibility requires your particular attention, click on *Detail* to call up the detail page for the selected area of responsibility in display mode. Here you can find detailed information on the selected area of responsibility.
5. If you want to perform a status switch, click on the *Edit* button, below the detailed information for the selected area of responsibility. You are then in processing mode.
6. Choose the status switch that you require.

   With each non-cascading status switch, the system automatically:

   - saves the status and (depending on the status change)
   - sets or lifts the lock for the selected area of responsibility
   - triggers the automatic notification system of the Status and Tracking System, if it has been activated. See: Executing Customizing [Page 317]. Notifications assigned to a status are sent by e-mail. See Adjusting Notification Texts [Page 329].
   - starts the global planning sequence for this status switch synchronously or as a background job. This happens as long as global planning sequences have been activated (see Executing Customizing [Page 317]) and a global planning sequence has been assigned to this status switch and your area of responsibility (see Executing Global Planning Sequences upon Status Switch [Page 327]).

7. Where necessary, enter *Comments*, to record any special features. Choose *Save* to save the comments.
8. If necessary, choose further actions on the detail page or the overview page.
9. Choose *Log Off* to close the Status and Tracking System.

   It is important that you log off from the planning session explicitly since otherwise the connection to the SAP System application server is kept and the planning objects you used remain locked.

4.2.1.2.9.5 Integrating BI Integrated Planning

Use

You can use the BW-BPS Status and Tracking system (STS) with BI Integrated Planning. This is useful if you are already using BW-BPS with the Status and Tracking system but want to use additional functions from BI Integrated Planning.

Activities

If you want to apply the STS lock check to Web applications created in BI Integrated Planning, proceed as follows:

1. Open the *Planning Modeler.*
2. Open the InfoProvider for which you want to use the STS lock check.
3. Open the *Data Slice* tab page for this InfoProvider.
4. Choose *Create.*
5. Choose the *Exit Class* entry from the selection list for *The Data Slice Is Based On*.
6. Choose exit class CL_UPS_LOCK_DATASLICE.
7. Save your entries.

In the Customizing of STS you can then enter the URLs of Web applications created in BI Integrated Planning for the planning application and control report.

In addition, you can enter either a global planning sequence created within BW-BPS or a planning sequence (with a variant, if applicable) created within BI Integrated Planning upon status switch. Input help is available.

4.2.1.2.10 Transport

Use

To transport the elements that you create and edit in the planning environment from the Customizing system into a production system, you must specify the objects to be transported (for example, planning areas and planning levels) and add a Customizing request. The objects collected in this request are transported into the target system as soon as you release the request.

Integration

Use *Tools* → *Transport* to create Customizing requests that can be processed further with the Transport Organizer of the ABAP Workbench (*Tools* → *ABAP Workbench* → *Overview* → *Transport Organizer*).

Features

Compiling planning objects for transport requests

You have various possibilities to combine a transport request:

- Enter a **planning area**, the lower-level objects of which you want to transport.

  New Transport Including All Planning Levels: You are transporting all planning levels and their subobjects.

  Change Transport of Individual Planning Levels: You are only transporting a particular planning level. You enter the name of this planning level.

The following applies to transporting planning areas: The smallest transportable unit is the planning level with all its lower-level objects. Separately transporting a single planning layout or a planning function, for example, is **not** possible.

- Specify whether you want to transport **global planning sequences**. Here you can also choose whether you want to transport all available planning sequences or only a sequence specified by you.

  Planning sequences created in the context of a planning level are automatically transported with the level, if the higher-level planning level is transported.

- Specify whether you also want to transport **planning profiles**. You define whether all profiles should be transported or only a certain one.

You can combine any three of the approaches listed for creating a transport request with each other (for example, a planning level and all planning profiles or only a certain planning sequence).

Saving combinations as variants

You can save any combination of objects to be transported as a variant, and simply reproduce it at a later point in time. This can be useful in the following scenarios:

- You are still in the implementation phase, and are constantly creating new planning objects but want to make the current status available in the test client or system as early as possible.
- You intend to make the planning objects available gradually in different system landscapes at greater intervals.

Further transport possibilities in the BW-BPS Environment

You have the following possibilities to transport objects that could be relevant in connection with BW-BPS using the functions described before about transporting within the planning environment:

- **Exit functions**: You can create or edit exit functions in the *Function Builder* of the ABAP Workbench in order to assign them to a planning function of the type Exit [Page 257]. In this case the system asks you to create a transport request for the function(s). A transport does not take place if you enter **$TMP**, the name reserved for local objects, as the package [External].
- **Business Server Pages and ABAP-OO classes**: The same applies to these objects from the Web Interface Builder [Page 285] environment as to the exit functions. If you make a different entry for the package other than **$TMP**, the system asks you to create a transport request.
- **Layout template**: There is a special report to transport the central layout template for the planning layout of manual planning. For more information, see Editing the Layout Template [Page 163].
- **Planning session and subplan**: You can transport the attributes of a combination of planning session and subplan from the Status and Tracking System [Page 310]. To do this, start the **UPS_TRANSPORT_STS** report and enter the desired combination of planning session and subplan.

Note that URL entries, which are a part of Customizing (for example, a link to a BW report), are transported unchanged into the target system. Since the formal structure of such URLs is partly dependent on the system (the name of the application server is part of the URL), you must manually adapt these URLs after the transport to the conditions in the target system.

The system generates a transport request for the planning session during creation.

- **Planning folders**: For information about transporting planning folders, see Editing a Planning Folder [Page 271].
- **Web templates**: If you refer to Web templates [External] in SAP BW in the environment of a Web-enabled planning application then you must make sure manually that these templates are also available in the target environment. You transport Web template using the Administrator Workbench *Transport Connection* (see Transports [External]).

Activities

Transporting planning objects from the planning environment

To transport planning objects within the planning environment, proceed as follows:

1. Choose *Tools* → *Transport* in the planning environment. The *Transporting Planning Objects* screen appears.
2. If you want a combination of transports to be executed as a background job, enter the desired transport request in the *Customizing Task* field.
3. Select the objects you want to transport. Note that for objects you want to transport, you must set the corresponding indicator for the higher-level object type. Alternatively, you can select the objects with a variant. To do this, choose *Goto* → *Variants* → *Get*.
4. Choose to assign the specified planning objects to a transport request.
    a. If you have already specified a Customizing task, the objects are written to this transport request.
    b. If you have not specified a Customizing task, you reach a dialog box where you can specify a transport request. Choose an appropriate transport request or create a new one. Confirm your entries.

       If you want to transport a large number of objects, choose *Program* → *Execute in Background*. Using the selection entered the system starts the program for executing and printing as a background job. A dialog box appears where you enter the print parameters.

   You receive a log of the actions you execute.
5. To transport the planning objects you must release the transport request in the Transport Organizer at a specific point in time (*Tools* → *ABAP Workbench* → *Overview* → *Transport Organizer*).

Client copy within the same system

After you have released the transport request, the objects of the request are available in the target system and the target clients. If you want to carry out a client copy within the same system, proceed as follows:

1. Log onto the system in the desired target client.
2. Start the client copy (*Tools* → *Administration* → *Administration* → *Client Administration* → *Special Functions* → *Copy Transport Request*; transaction **SCC1**) and enter the source client and the number of the transport request.
3. Then start importing the objects.

4.2.1.2.11 Translation

Use

In order to display the descriptions of elements that you create and edit in the planning environment in different logon languages, you have to translate the objects. Otherwise you would only see the (language-independent) technical names of the planning objects in the planning environment.

Integration

You can use the integrated SAP NetWeaver translation functionality oriented towards the translation of planning objects in the planning environment by choosing *Tools* → *Translation.*

Prerequisites

The target language that you want to translate into must be set up in the system.

Features

Compiling planning objects for translation

You have different possibilities to group together objects for translation:

- You enter a **planning area** whose lower-level objects you want to translate.

  New Translation Including All Planning Levels: You are translating all planning levels and their subobjects.

  Change Translation of Individual Planning Levels: You are only translating a particular planning level. You enter the name of this planning level.

  > 💡
  >
  > The following applies for the translation of planning areas: The smallest translatable unit is the planning level with all subordinate objects. The separate translation of a single planning layout or a planning function, for example, is **not** possible.

- You specify whether you want to translate **global planning sequences**. Using selection conditions, you can enter which planning sequences you want to translate.

> Planning sequences that are created in the context of a planning level are automatically translated with the level, if the superior planning level is translated.

- Specify whether you also want to translate **planning profiles**. Using selection conditions, you can enter which planning profiles you want to translate.

You can combine any of the above approaches with each other, in order to group together the objects to be translated (for example a planning level and all planning profiles, only one certain planning sequence).

Saving combinations as variants

You can save any combination of objects to be translated as a variant, and simply reproduce it at a later point in time. This can be useful in the following scenarios:

- You are still in the implementation phase, and are constantly creating new planning objects, however, you also want to make the current status available in the other languages as early as possible.
- You intend to make the planning objects available in other languages gradually over greater intervals.

Activities

The current log-on language is the source language. To translate planning objects from the source language into the target language, proceed as follows:

1. In the planning environment, choose *Tools* → *Translation*. The *Translation of Planning Objects* screen appears.
2. Enter the target language in which you want to translate the objects.
3. Make the selection of objects that you want to translate. Alternatively, you can select the objects with a variant. To do this choose *Goto* → *Variants* → *Get*.
4. Choose ⊕, to load the specified objects into the translation tool.
5. The system starts the translation tool, and offers you the selected objects for translation.
6. Save the translation.

See also:

Tools for Translators [External]

4.2.1.2.12 Data Modeling: Key Figure Model and Account Model

Definition

You can use two different modeling strategies to carry out a planning project in BW-BPS. To illustrate these strategies in terms of a data model, the following terms are used:

- Key figure model

- Account model

You have to consider the data modeling aspects for the entire procedure as soon as you begin to create InfoCubes.

This simple example shows how the terms key figure model and account model are defined.

You want to assign several key figures (**`sales price`**, **`manufacturer price`**, **`mean price`**) to a characteristic (product *P1*). The key figures have the same unit (currency *EUR*).

The key figure model includes all the key figures in one data record:

Product	Sales Price	Manufacturer Price	Mean Price
P1	100	50	75

However, in the account model, you add a new characteristic which separates the key figures (here: price types) within the data record, and which you can use to determine which key figure is being used for planning.

Product	Price Type	Price
P1	Sales Price	100
P1	Manufacturer Price	50
P1	Mean Price	75

Use

Use the *key figure model* if:

1. You are working with a restricted and consistent number of key figures, and
2. Each, or at least a large proportion of the key figures in the data record are also being used in the other data records.

Use the *account model* if:

1. You are working with a large and changing number of key figures, and
2. Most of the key figures would have no value in a data record in the key figure model.

If your planning model requires it, you can use different data models in different InfoCubes. In this case, you are able to convert or transform data from one data model into another data model using FOX formulas, which assign the respective combinations.

As a general rule: Subsequently converting the data model (from the key figure model to the account model and vice versa) involves a large amount of work. Instead, it is better to create a new InfoCube.

In the account model, adding an additional price type (see the example above) does not pose a problem as the data structure does not have to be changed. However, in the key figure model, an additional key figure has to be added to the structure. The extent to which this difference is to be seen as an advantage or a disadvantage depends on the frequency of changes expected.

Performance and Required Memory Capacity

Transaction data is stored in the BW system. Analysis has shown that the read time depends essentially on the number of data records and not their size. Even in the example above you can see that if you change from a key figure model to an account model, the number of data records increases by a factor of 3; consequently, longer read times can be expected.

When determining the factor, also consider that in the key figure model, not all key figures are used in every data record. This means that upon switching to an account model, only those data records are generated that have key figures that are not zero.

If, in the example above, the key figure **`sales price`** was not being used, only two data records would be generated and the factor is reduced to 2.

After being read from the BW system, the transaction data is stored in the working memory of the application server. Among other things, the memory space required depends on the number and length of the data records. As discussed, the length of the data record is influenced by the choice of data model; in the key figure model data records are longer than in the account model, and they use more memory space per data record. Whether less memory space would be used by the key figure model or the account model depends on the ratio between the number and length of the data records.

If the data model is changed, the planning functions also have to be revised. According to the sort of calculation you want to perform, one model may have advantages over the other.

For more information on forecasts, see SAP Note 407563.

Direct Comparison of Key Figure Model and Account Model

The following table shows the advantages and disadvantages of each data model.

Key Figure Model

PRO	CONTRA
• Key figures are created for a specific usage. • When configuring the system you do not have to think about whether a key figure has a reference to characteristic properties. • The planning layout is easily structured. • Formulas are simple.	• A separate key figure is required for each account and summation level. • Creating and deleting key figures in an InfoCube that has already been loaded is not straightforward. • Selection variables cannot be used in BEx and BW-BPS. • BEx queries cannot be structured as flexibly. • Possible negative effect on performance due to the size of data records in the database table and planning buffer.

Account Model

PRO	CONTRA

• Fewer key figures in the InfoCube, fewer columns in the flat file. • Advantages offered by ability to use characteristic hierarchies. • Configuration is simpler; Simply add hierarchy nodes. • Use of selection variables in BEx and BW-BPS. • BEx queries can be structured flexibly.	• Account master data has to be entered from customer data. • Planning layouts in manual planning have to contain hierarchies in order to retain the order of account characteristics in the lead column. As a result, unnecessarily long hierarchies have to be entered. An understanding of the InfoObject hierarchy concept is necessary. • You always get subtotals in the planning layout, unless you deactivate *Total*. • Possible negative effect on performance due to the number of data records in the database table and planning buffer.

Data Flow

You can use multi-planning areas to transfer and convert data from one InfoCube to another. The different data models (key figure model, account model) are no obstacle. For more information, see Multi-Planning Area [Page 97].

See also:

Architecture and System Landscape (BW-BPS) [Page 349]

4.2.1.2.12.1 Architecture and System Landscape (BW-BPS)

Definition

There are three basic possibilities for configuring BW and BW-BPS systems.

1. **Centralized**: BW system and BW-BPS share data, structure and database.
2. **Remote**: BW-BPS (local) has a remote connection to the BW system (remote).
3. **Separate**: Separation of BW system functions and BW-BPS functions.

These configuration options do not restrict the functionality or the features of BW-BPS, they simply address different business requirements.

Use

There are three areas which need particular consideration when defining and implementing BW-BPS applications:

- System availability
- Performance
- Patches and upgrades

It is also important to remember that the BW system is also used by non-BW-BPS users. As a result, the following key issues arise:

- Cost-efficiency, legal and security requirements
- Periodic usage of planning tools
- Data redundancies
- Data integration - combining and using plan data and non-BW-BPS data conjointly
- Routine work and work related to patches and upgrades
- Administration work and costs
- System costs

Choosing the right system configuration is very important. It is possible to switch from one configuration to another but this involves a large outlay in terms of time, costs, and administrative efforts.

The most important characteristics of the different system configuration options named above will now be listed.

Centralized

- The BW system and BW-BPS are on the same server/system; they use the same system landscape (DEV, QA, PROD).
- The data and structures are stored on the same server in the same database.
- BW patches and upgrades are implemented simultaneously for the BW system and for BW-BPS.

Remote

- The BW system and BW-BPS are on different servers/systems. BW-BPS (local) has a remote connection to the BW system (remote).
- BW-BPS is configured in the local BW-BPS system.
- Data and structures are stored on the same database.
- BW-BPS only uses the structures of the remote BW system.
- BW-BPS reads and writes data straight from the remote BW system.
- BW patches and upgrades are implemented on the respective servers for the BW system and for BW-BPS and can therefore be implemented independently of each other.

Separate

- The BW system and BW-BPS run on different systems that are independent of each other.
- Configuration, structures and data from the BW system and BW-BPS are stored on different (that is their own) servers and databases.
- Users of BW-BPS and the “general” BW systems use different (that is their own) system landscapes (DEV, QA, PROD).
- Data from BW-BPS and the BW system can be loaded to and from each system using BW data marts.

- BW patches and upgrades are implemented on the respective servers for the BW system and for BW-BPS and can therefore be implemented independently of each other.

Advantages and Disadvantages of System Configuration Options

The following table offers an overview of the advantages and disadvantages of these three options:

	Centralized	Remote	Separate
Data integration	+	+	-
No data redundancy	+	+	-
Ease of system maintenance	+	-	-
Access to planning and reporting	+	-	-
Low costs for system landscape	+	-	-
Separation of BW and BW-BPS	-	+	+
Ease of security setup/maintenance	-	+	+
Reduced release-dependency	-	+	+
Ability to fine-tune	-	+	+

4.2.1.2.13 Application Programming Interface (API)

Definition

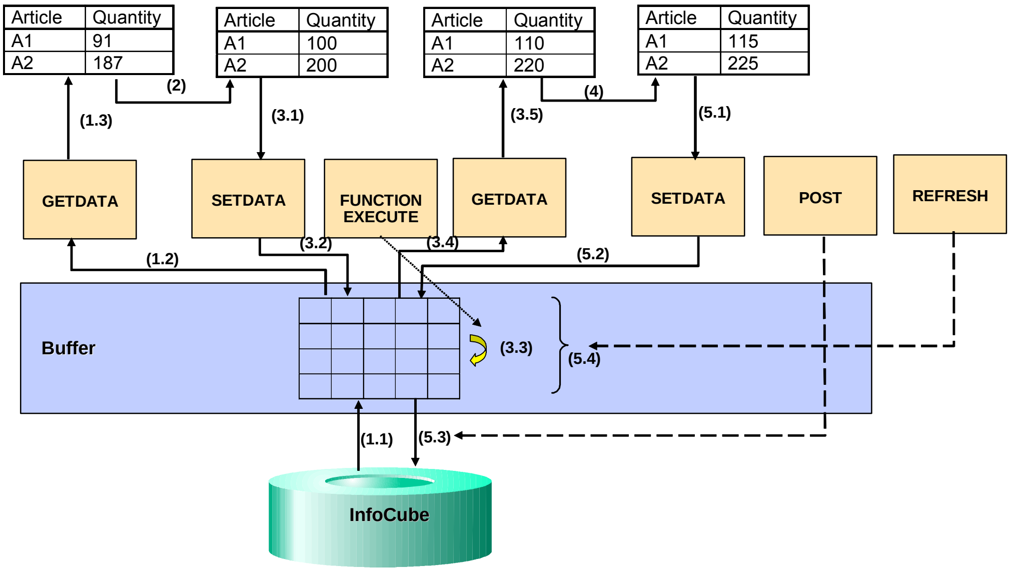
The BW-BPS application programming interface comprises a large number of functions that you can use to access the BW-BPS planning objects and functions from your own applications.

Use

The BW-BPS application programming interface enables you to provide program interfaces created by you, orientated towards the end user, with the functionality of BW-BPS. As a result, you connect the efficiency of BW-BPS with the simple operation of a familiar interface, for example Microsoft Excel or a Web application.

You can offer the users of the applications you have created a subset of planning functionality determined by you, where the use of the functionality can take place completely transparently. This means that BW-BPS works together “invisibly” with your application. An advantage of this is that your users to not have to learn how to use BW-BPS.

The following graphic shows a typical sequence of access to the application programming interface. As a client application which accesses the interface, a self-defined transaction is accepted with the appropriate programs, which can display plan data in a separate table that is ready for input (for example ALV grid control [External]). The purpose of this application is to execute a simple revaluation, and to offer the possibility of manual data change.



The numbers in the graphic describe the following steps:

1. Data collection from the data basis.

   When the function **API_SEMBPS_GETDATA** is first called up, the BW-BPS buffer storage is empty. Therefore, the function reads the data directly from the InfoCube that the addressed planning area is assigned to (1.1). It prepares the values corresponding to the planning layout transferred as a parameter, automatically fills the buffer with the read data, and transfers the data into the table that is ready for input (1.3).

2. Manually changing plan data

   The user of the client application changes the values in the table that is ready for input. Subsequently he or she chooses one of the functions “Percentage Increase” offered by the application, with which the current value should be increased by 10%.

3. Revaluation

   The function “Percentage Increase” of the client application (3.1) is designed so that the data currently displayed with the function **API_SEMBPS_SETDATA** is written into the buffer storage managed by BW-BPS. Subsequently with a corresponding parameterized call of function **API_SEMBPS_FUNCTION_EXECUTE**, the desired revaluation is executed (3.3). The actual revaluation is not carried out by the interface function. Rather, this calls up a planning function of the type Revaluation [Page 215], which was created before in the planning environment under entry of a parameter group, and which contains the revaluation factor (10%). The values that are increased by 10% are now in the buffer and, by calling up the function **API_SEMBPS_GETDATA** (3.4), are transferred into the table ready for input (3.5), so that the result of the revaluation is visible. Since at this point data already exists in the buffer, this time the function only accesses the buffer (and not the InfoCube). This is different to the first call (1.1). This way a considerable improvement is achieved in the response time.

4. Manually changing the plan data again

   The client application user changes the now increased values again in the table that is ready for input.

5. Exiting the application

Finally the user exits the client application. The application makes sure that the current status of the data from the ready for input table is transferred into the buffer again (5.2) by calling up the function **`API_SEMBPS_SETDATA`**. The data found in the buffer is saved permanently in the data basis by a final call up of the function **`API_SEMBPS_POST`**. Following the BW-BPS transaction concept, the objects concerned remain locked with access using the API, until the connection to the system is removed (usually by exiting your external application). This normally occurs when you exit your external application. If you want to release the objects earlier, you can do this with the function **`API_SEMBPS_REFRESH`** (5.4).

The applications created by you are not restricted to user-defined transactions, reports or other SAP programming objects. Rather, you can also contact the BW-BPS application programming interface from an external PC application using the corresponding SAP technologies. The prerequisite for this is that the external application can take on the role of an OLE Automation Controller. This is, for example, the case for all products of the Microsoft Office family, which have a macro language or an integration with Visual Basic for Applications (VBA) at their disposal.

To make troubleshooting easier when developing your program, BW-BPS offers a special trace function with which you can log the internal function calls of BW-BPS. This function is not integrated into the planning environment; it must be called explicitly with the transaction **`BPS_TRACE`**. The following functions are offered to you there:

- *User Overview*: You can choose, on a user by user basis, whether trace lists should be generated or not. In addition, you receive an overview about all users for whom the trace function is activated.
- *Trace Overview*: You receive an overview of all traces that are saved in the system. You can sort the trace list by user or by date/time. By double-clicking on a trace, the content of the trace is displayed, every call of a function module is displayed in its own row. With the function *Run Trace*, you can run a function call in the debugger, which is recorded in the trace log, at exactly the time of recording.

Structure

The application programming interface consists of a large number of functions that are grouped together in the function group **`UPC_API.`** The function group contains functions for the following tasks:

- Reading planning architecture objects (planning areas, levels and packages; see Planning Environment [Page 90]), using one of the planning layouts to be transferred as a parameter of manual planning [Page 127]

  The planning layout that you want to use as a data source must be configured so that the selection in the header area is clear. Otherwise the system cannot determine which data should be delivered by the interface.

- Reading master data
- Reading and writing transaction data
- Reading from BW characteristic hierarchies (“external hierarchies”, see Hierarchically Structured Planning [Page 135])
- Reading variables
- Executing planning functions

If you want to use the programming interface not only to display data from BW-BPS but also to change it, use the planning functions defined in the planning environment (instead of local functions within your application). The reason for this is that the data delivered by the API is not passed on unchanged from the data basis, instead it is formatted using a planning layout of manual planning. The settings prevailing in the layout definition regarding scaling factors, decimal places, and reversed credit/debit sign [Page 172] have an immediate effect on the transferred key figure values, without it being recognizable from the external application whether such formatting has taken place or not. In addition, formatting with few decimal places can lead to unnoticed rounding errors.

- Explicit release of all locked objects

  You will find more information (for example on the interaction of various functions or the parameter supply) in the system, in the documentation on the function modules of function group **`UPC_API`**.

Integration

The BW-BPS programming interface enables the integration of planning functionality in as many other applications as you want. If you want to access the programming interface functions from an external PC application, you can achieve this with the help of the SAP Automation Interface. You can find information on using this service under SAP Automation RFC and BAPI Interfaces [External]. The following objects are of particular interest:

- Logon Control [External]
- Table Control [External]
- Function Control [External]

4.2.1.2.14 Workflow Scenarios for BW-BPS

Purpose

By linking enterprise planning with BW-BPS to the SAP Business Workflow, you achieve optimal support when coordinating your planning processes.

For example, you can distribute planning tasks to many users from a central position and monitor the procedure using the SAP Business Workflow [External], and for example, execute a work item analysis [External].

Process Flow

1. First set up the planning environment [Page 90] with the planning objects and planning functions that you require.
2. Assign agents to the planning packages defined in the planning environment. You can find more information under Preparation and Customizing [Page 359].
3. Start one of the workflows delivered using *Tools* → *SAP Business Workflow* → *Runtime Tool* → *Start Workflow* from the *SAP Easy Access Menu* screen.

4. The system generates the necessary work items, and provides the assigned agents with them in the Business Workplace inbox.
5. The agent enters the plan data. According to the workflow definition the agent either closes certain workflow steps manually and makes sure that further steps of the workflow can be processed by the system, or the system automatically closes the workflow.

Result

You can find out whether the tasks were correctly processed in the automatically generated workflow logs.

See also:

Technical Realization [Page 355]

Preparation and Customizing [Page 359]

Workflow: Manual Planning and Planning Function [Page 360]

Workflow: Planning Session with One Layout [Page 361]

Workflow: Planning Session with Several Layouts [Page 361]

4.2.1.2.14.1 Technical Realization

Object Types

The following object type is relevant for the workflow support of business planning with BW-BPS:

Object Type BUS1145 (BPSPlanningPackage)

Planning packages [Page 119] from BW-BPS can be processed with the object type BUS1145. The key fields of the object types are planning area, planning level, and planning package.

You define planning areas, planning levels, and planning packages in the BW-BPS planning environment [Page 90].

The object type BUS1145 has the following methods:

- **`ChangePlanData`** to change plan data
- **`DisplayPlanData`** to display plan data
- **`ExecutePlanningFunction`** to execute a planning function
- **`DisplayLog`** to display the error log after executing a planning function

Workflow Tasks

The following workflow tasks delivered by SAP are ready as modules and can be used unchanged in individual workflow scenarios (see Tasks and Task Groups [External]). The tasks are used for entering or displaying plan data, executing planning functions, and displaying logs.

Task BPS01

Definition

Task: 40107918

ID: BPS01

Description: Enter BPS plan data

Structure

Object type: BUS1145 (BPSPlanningPackage)

Referenced object methods: ChangePlanData

Attributes: Confirm object methods with dialog, end of processing

Agent assignment: No restriction of possible agents

> The possible agents are specified here and the responsible agent is selected from them. The agent responsible for this task is determined by the rule. The agent can be any employee. Therefore, it is necessary to classify the task within Customizing as a *general task.* See Preparation and Customizing [Page 359].

Task BPS02

Definition

Task: 40107919

ID: BPS02

Description: Display BPS plan data

Structure

Object type: BUS1145 (BPSPlanningPackage)

Referenced object methods: DisplayPlanData

Attributes: Confirm object methods with dialog, end of processing

Agent assignment: no restriction of possible agents (see task BPS01)

Task BPS03

Definition

Task: 40107920

ID: BPS03

Description: Execute BPS planning function

Structure

Object type: BUS1145 (BPSPlanningPackage)

Referenced object methods: ExecutePlanningFunction

Attributes: Confirm object methods with dialog, end of processing

Agent assignment: does not apply as concerned with background processing

> The system automatically assigns a technical agent to the task for background processing. This can lead to problems if the workflow task is carried out for a planning package that contains characteristic value selection using user-specific

Variables [Page 102]. In such a case, the discrepancy between the technical agent and the user to which the variable refers, means that the workflow steps on the basis of task BPS03 **cannot** be carried out.

Task BPS04

Definition

Task: 40107921

ID: BPS04

Description: Display BPS log

Structure

Object type: BUS1145 (BPSPlanningPackage)

Referenced object methods: DisplayLog

Attributes: Confirm object methods with dialog, end of processing

Agent assignment: no restriction of possible agents (see task BPS01)

Rule for Agent Assignment BPS01

Definition

Rule with which the agent of a planning package can be determined. This rule is used in the delivered workflow *Planning Session in One Layout* (BPS51).

Rule: 40100136

ID: BPS01

Description: Person responsible for BPS planning package

Rule parameter

All key fields of the object type BUS1145 can be transferred as rule parameters. You can assign an agent in dependency on these fields. See Preparation and Customizing [Page 359].

Workflow BPS50

Definition

Workflow: 40100004

ID: BPS50

Description: Manual planning and planning function (see Workflow: Manual Planning and Planning Function [Page 360])

Workflow container

The import parameters of a workflow are provided in the workflow container.

The essential information that has to be available for the process flow of workflow BPS50 is:

- An object reference to the planning package to be processed. This is provided in the container element *BPSPlanningPackage.*

- A planning layout to execute the method *ChangePlanData.* This is provided in the container element *Layout*. The layout is created in the planning environment like the planning packages.

- A planning function with parameter group belonging to it for the method *ExecutePlanningFunction*

Specify this information when starting the workflow.

Workflow steps

The BPS50 workflow consists of several steps, the tasks *Enter BPS Plan Data, Execute BPS Planning Function*, and *Display BPS Log.*

Agent of the workflow steps *Enter BPS Plan Data* and *Display BPS Log*

In the delivered workflow BPS50, the agent can be found using the rule BPS01.

Starting the workflow

Start workflow BPS50 from the *SAP Easy Access* screen under *Tools* → *Business Workflow* → *Runtime Tools* → *Start Workflow*.

Specify a planning package, planning layout, planning function, and parameter group when starting the workflow.

Workflow BPS51

Definition

Workflow: 40100005

ID: BPS51

Description: Planning session with one layout (see Workflow: Planning Session with One Layout [Page 361])

Workflow container

The import parameters of a workflow are provided in the workflow container.

The essential information that has to be available for the process flow of workflow BPS51 is:

- A table with object references for the planning packages to be processed. This is provided in the multiline container element *BPSPlanningPackage.*

- A planning layout to execute the method *ChangePlanData.* This is provided in the container element *Layout*. The layout is created in the planning environment like the planning packages.

You must enter this information when starting the workflow.

Workflow steps

The BPS51 workflow only consists of one single step, the task *Enter BPS Plan Data.* However, this step is indicated by a dynamic parallel processing (tab page *Miscellaneous* in the step definition). A work item with a planning package is generated for every item of the multiline container element *BPSPlanningPackage.*

Agent of the workflow step *Enter BPS Plan Data*

Plan data for a planning package can be entered in a workflow step. There are many different ways to determine the responsible agent of this workflow step. In the delivered workflow BPS50, the agent can be found using the rule BPS01.

Starting the workflow

Start workflow BPS51 from the *SAP Easy Access* screen under *Tools* → *Business Workflow* → *Runtime Tools* → *Start Workflow*.

When starting the layout, specify one or more planning packages and a planning layout.

Workflow BPS52

Definition

Workflow: 40100008

ID: BPS52

Description: Planning session with several layouts (see Workflow: Planning Session with Several Layouts [Page 361])

Workflow container

The import parameters of a workflow are provided in the workflow container.

The essential information that has to be available for the process flow of workflow BPS52 is:

- An object reference to the planning package to be processed. This is provided in the container element *BPSPlanningPackage.*
- A table with layouts to execute the *ChangePlanData* method. This is provided in the container element *Layout*.
- The desired processing end (optional).

Workflow steps

The BPS52 workflow only consists of one single step, the task *Enter BPS Plan Data.* However, this step is indicated by a dynamic parallel processing (tab page *Miscellaneous* in the step definition). A work item with a planning package is generated for every item of the multiline container element *Layout.*

Agent of the workflow step *Enter BPS Plan Data*

Plan data for a planning package can be entered in a workflow step. There are many different ways to determine the responsible agent of this workflow step. In the delivered workflow BPS52, the agent can be found using the rule BPS01.

Starting the workflow

The workflow BPS52 is started with the report **`UPC_WORKFLOW_BPS52_START`**.

4.2.1.2.14.2 Preparation and Customizing

Before you can carry out the workflows *Execute Planning Session in BW-BPS* and *Manual Planning and Planning Function*, you must carry out the preparations described below.

Activities

1. Create a planning level, planning package, and planning layout in the BW-BPS planning environment. Also create a planning function with parameter group for the workflow *Manual Planning and Planning Function*.
2. Classify the task *Enter Plan Data* in the Basis Implementation Guide under *SAP NetWeaver → SAP Web Application Server → Business Management → SAP Business Workflow → Perform Task-Specific Customizing*.
   a. Choose *Execute*.
   b. Navigate to the *BW → BW-PLA → BW-PLA-BPS* folder.

c. Choose *Assign Agent*. The *Tasks of an Application Component: Assign Agent* screen appears.

d. Select the *Enter BPS Plan Data* task and choose *Properties*. The *Task* dialog box appears.

e. Choose the *General Task* option.

3. Specify which agent should be provided with which planning package for plan data entry in the inbox. Maintain the agent assignment in the BPS01 rule. This takes place in the workflow development menu.

   a. Choose *Tools* → *Business Workflow* → *Definition Tools* → *Rules for Agent Assignment* → *Change* from the *SAP Easy Access* screen.

   b. Enter rule BPS01 in the field that is ready for input and choose *Change.*

   c. Choose the tab page *Responsibilities*. You enter the planning area, planning level, and planning package in a responsibility. You can then assign one or several agents to this responsibility.

4.2.1.2.14.3 Workflow: Manual Planning and Planning Function

Purpose

The delivered workflow *Manual Planning and Planning Function* (BPS50) shows how manual planning and planning functions can be linked in workflow.

Prerequisites

You have carried out the activities described in the Preparation and Customizing [Page 359] section.

Process Flow

1. Define a layout for manual planning and a planning function with a parameter group in the BW-BPS planning environment.
2. Assign agents to the planning packages defined in the planning environment. You can find more information under Preparation and Customizing [Page 359].
3. Start the workflow *Manual Planning and Planning Function* in the *SAP Easy Access* screen under *Tools* → *Business Workflow* → *Runtime Tools* → *Test Workflow*. Enter planning package, layout, planning function, and parameter group.
4. The system generates a work item and automatically provides the assigned agent with it in the Business Workplace inbox.
5. The agent enters the plan data – possibly with interruptions - and completes the workflow step manually.
6. Then the planning function is automatically executed in the background by the workflow system.
7. If the planning function reports errors, the agent automatically receives a further work item in the inbox with which he or she can display the error messages.
8. After correcting the cause of the error, the planning function is automatically started again.

9. The workflow is completed when the planning function runs without any errors.

4.2.1.2.14.4 Workflow: Planning Session with One Layout

Purpose

You can realize a planning session within BW-BPS business planning in which plan data can be entered by several agents with the delivered workflow *Planning Session with Several Layouts* (BPS52). In contrast to the workflow planning session with several layouts [Page 361] only **one** larger workflow is started here that contains the work items of all agents. This is only possible with a small number of agents (< 99).

You can distribute planning tasks to many users from a central position and monitor the procedure using the SAP Business Workflow [External], and for example, execute a work item analysis [External].

Prerequisites

You have carried out the activities described in the Preparation and Customizing [Page 359] section.

Process Flow

1. Define a layout for manual planning [Page 127] in the planning environment.
2. Assign agents to the planning packages defined in the planning environment. You can find more information under Preparation and Customizing [Page 359].
3. Start the workflow *Manual Planning and Planning Function* manually in the *SAP Easy Access* screen under *Tools* → *Business Workflow* → *Runtime Tools* → *Start Workflow* or *Test Workflow*. Enter the planning packages and the layout that should be used for planning.
4. The system generates a work item per planning package and automatically provides the assigned agents with them in the Business Workplace inbox.
5. The agents enter the plan data and finish the workflow manually. Since the workflow is completed manually, plan data entry can be executed in several phases over a longer period.

4.2.1.2.14.5 Workflow: Planning Session with Several Layouts

Purpose

You can realize a planning session within BW-BPS business planning in which plan data can be entered by several agents with the delivered workflow *Planning Session with Several Layouts* (BPS52). In contrast to the planning session with one layout [Page 361] workflow, here a separate workflow is generated for each planning package to be processed. This is suitable for more comprehensive planning work.

For example, you can distribute planning tasks to many users from a central position and monitor the procedure using the SAP Business Workflow [External], and for example, execute a work item analysis [External].

Prerequisites

You have carried out the activities described in the Preparation and Customizing [Page 359] section.

Process Flow

1. In the planning environment define several planning packages and one or several layouts for manual planning.
2. Assign agents to the planning packages defined in the planning environment. You can find more information under Preparation and Customizing [Page 359].
3. Start the report **UPC_WORKFLOW_BPS52_START** in the ABAP Editor. Enter the planning area and planning level in which you have created the packages and layouts. If you enter a date in the field *Processing to*, you receive an e-mail if the agent has not processed your work item on time.
4. The system generates a workflow per planning package; each workflow contains a work item per layout that is automatically provided to the assigned agents in the Business Workplace inbox.
5. The agents enter the plan data and finish the workflow manually. Since the workflow is completed manually, plan data entry can be executed in several phases over a longer period.

4.2.2 Analysis Process Designer

Use

In the BI system data is collected from the heterogeneous databases in the various systems that the enterprise is using and is consolidated, managed, and staged for evaluation purposes. There is often further, valuable potential in this data.

This is completely new information that is displayed in the form of meaningful correlations between data but that is too well hidden or complex to be discovered by simple observation or intuition.

The Analysis Process Designer (APD) allows you to find and identify these hidden or complex relationships between data in a simple way. Various data transformations are provided for this purpose, such as statistical and mathematical calculations, data cleansing and structuring processes.

The results of the analysis are saved in BI InfoProviders or in CRM systems. They are available for all decision and application processes and thus can be decisive (strategically, tactically, and operatively).

Examples of analysis processes include the calculation of ABC classes, determination of frequency distribution or of scoring information.

Integration

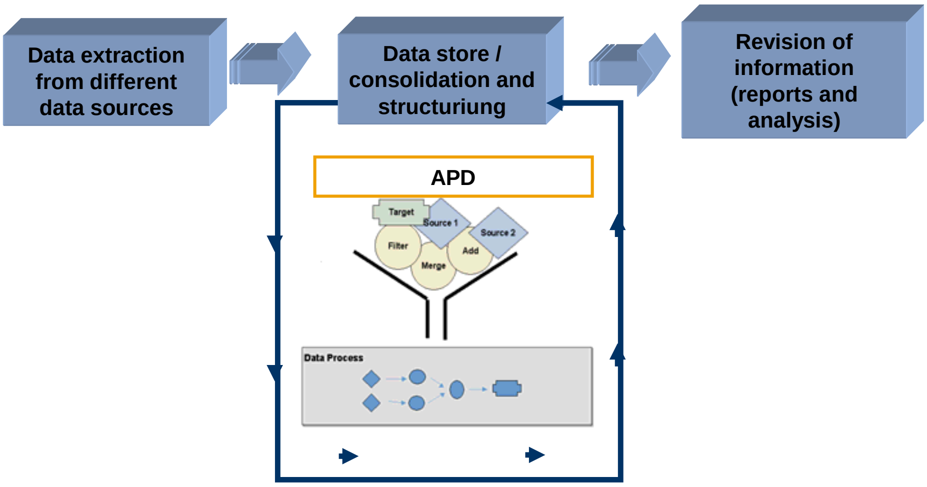
The Analysis Process Designer is the application environment for the SAP data mining solution. The following data mining functions are integrated into the APD:

- Creating and changing data mining models

- Training data mining models with various BI data (data mining model as data target in the analysis process)
- Executing data mining methods such as prediction with decision tree, with cluster model, and integration of data mining models from third parties (data mining model as a transformation in the analysis process)
- Visualizing data mining models

For more information, see Data Mining [Page 393].

The APD is integrated into the Data Warehousing Workbench:


Data extraction from different data sources
Data store / consolidation and structuriung
Revision of information (reports and analysis)
APD
Target
Source 1
Source 2
Filter
Add
Merge
Data Process

Constraints

The following restrictions apply to the integration into the Data Warehousing Workbench:

- The node texts are not language dependent.
- You can only integrate an analysis process into a process chain using the process type *ABAP program*. To do this, choose the ABAP report RSAN_PROCESS_EXECUTE.
- Analysis processes are not displayed in the data flow display.
- The where-used list only functions from the analysis process to other objects and from data mining models to the analysis process, not from other objects such as InfoProviders.

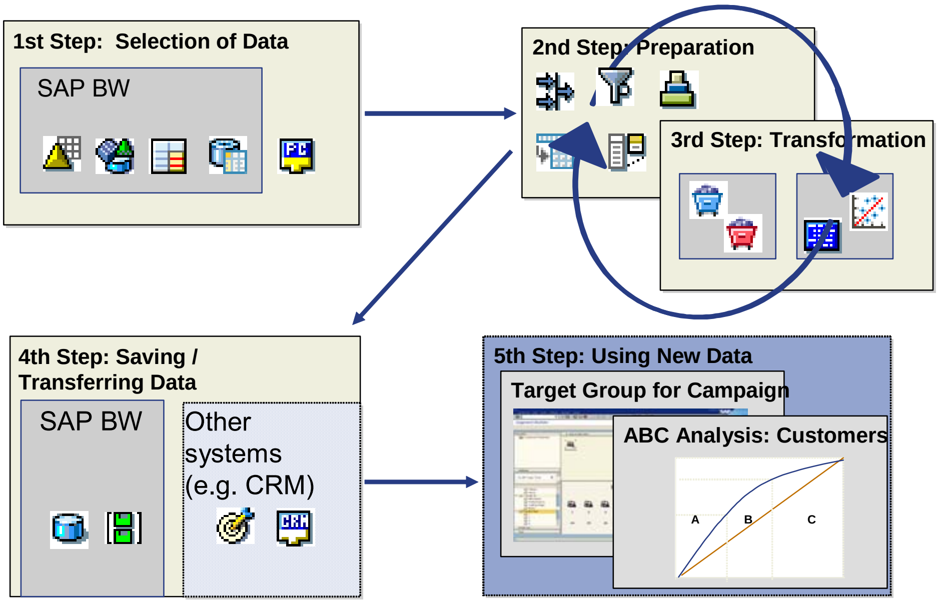
Features

The Analysis Process Designer is a workbench with an intuitive graphic user interface for the creation, execution, and monitoring of analysis processes. Analysis processes can be created using Drag&Drop. Data from different data sources in the BI system can be combined, transformed, and staged for analysis in several individual steps so that it can be resaved in data targets in the BI system (DataStore objects for direct update or InfoObjects with attributes), or in a CRM system. Available to you are various

- Data sources [Page 366]
- Transformations [Page 371] and
- Data targets [Page 381]

Various additional functions support you in modeling and executing an analysis process, as well as in interpreting the analysis results. See also Checking Data [Page 389].

The following graphic shows the various steps in the Analysis Process Designer.



First select a data target that contains the required data. This data is then staged and transformed. The transformed data is saved in a BI object or in another system. For analysis, you can display the data in a query in the Business Explorer.

For more information, see Modeling an Analysis Process [Page 387].

Versioning

Analysis processes are integrated into the versioning concept (active, inactive version, content version and delivery).

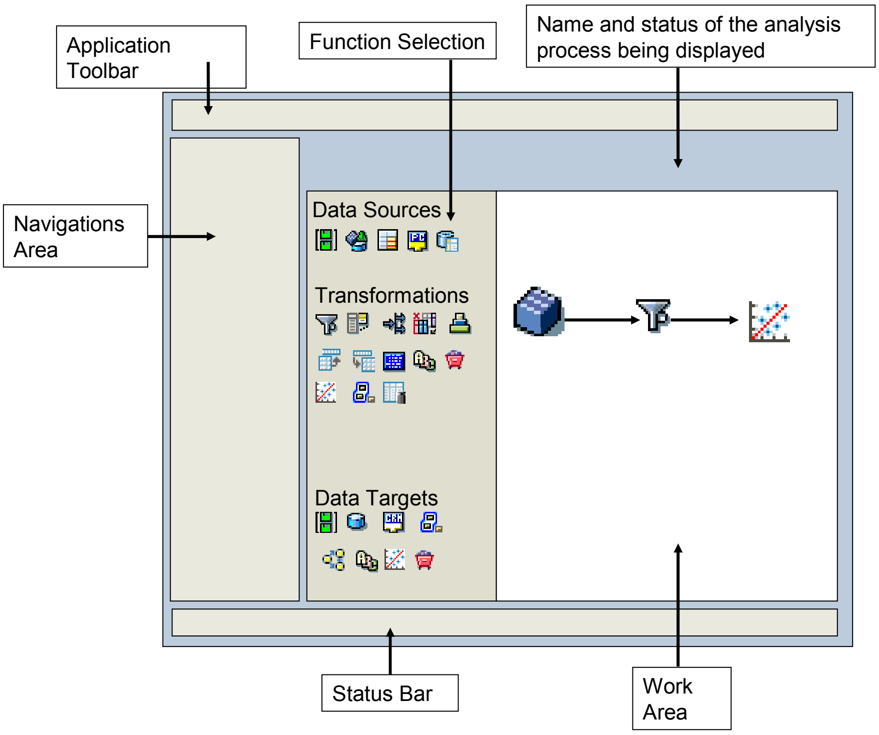
Transport Connection

Analysis processes are connected to the BI transport system as TLOGO objects. For more information, see Transport Systems [External].

4.2.2.1 Structure of the Analysis Process Designer

Functions

As is shown in the following graphic, the interface of the analysis process designer is comprised of an application toolbar, a name and status bar for the analysis process being displayed, the navigation area, function selection and the work area.



Application Toolbar:

The application toolbar offers functions that are described under Additional Functions in the APD [Page 392].

Navigation area:

The navigation area is comprised of an application toolbar and a tree structure in which all available analysis processes are displayed. Each analysis process is assigned to an application component *Fill CRM Attributes, Surveys* or *General.* These application components are provided by SAP, you cannot create any application components yourself. Different functions are available depending on the application component under which you model your analysis process. An authorization check is possible for each application component.

Using the application toolbar, you can create an analysis process and search for an existing analysis process.

Function selection:

In the function selection area, all of the available data sources, transformations and data targets are displayed in the form of icons.

Work area:

In the work area you model or change your analysis process.

Status bar:

The status bar shows warnings and error messages.

4.2.2.2 Data Sources for an Analysis Process

Definition

Provides the input data for an analysis process.

Use

Depending on the defined problem or task that you wish to solve with an analysis process, it is necessary to provide all relevant data to the analysis process from the beginning. This means there is a sufficient comprehensive "raw data" basis upon which the subsequent steps can be based.

The following types of data sources are available in the analysis process designer:

- Attribute of a Characteristic [Page 366]
- InfoProviders [Page 367]
- Query [Page 369]
- File [Page 370]
- Database Table [Page 371]

You can find descriptions of the data sources in the analysis process designer as well in the dialog window for the respective node under *Help*.

4.2.2.2.1 Attributes of a Characteristic

Use

With the *Read Attributes of a Characteristic* node, the master data table for the specified characteristic is read. In this way, attributes can be determined for a characteristic.

Functions

The active version of the data is read from the master data table.

Example

You can read the ABC class from the master data for the business partner (characteristic 0BPARTNER).

4.2.2.2.2 InfoProviders

Use

The data source *Read Data from InfoProvider* allows you to use data from a BI InfoProvider as a data source in the analysis process. InfoCubes, DataStore objects, InfoObjects, MultiProviders and InfoSets are InfoProviders. See InfoProviders [External].

Data is provided in the form of a simple table. You can determine which columns the table contains by selecting characteristics and key figures. If you omit fields when you select characteristics, key figures are aggregated using the excluded characteristics. Aggregation is performed using the standard aggregation behavior for the selected key figure. Aggregates for InfoCubes are used if they are available.

Features

Reading data from an InfoProvider is comparable to the direct selection of data according to the schema.

```
SELECT          <selected characteristics and key figures>
      FROM            <InfoProvider>
      GROUP BY <selected characteristics>
```

Some basic functions of the query are not possible here and have to be reproduced, as required, using subsequent transformations.

Active Data

Only data available for reporting is read from the data source InfoProviders. With InfoCubes, the data is available immediately as long as the requests are loaded successfully. You can see this by checking the *Request is Available for Reporting* icon in InfoCube administration. With master data, the attribute change run has to have finished. With DataStore objects, the data has to have been activated. With DataStore objects of type *direct update* the data can be seen immediately.

Compound Characteristics

All characteristics are considered independent fields. If, for example, the characteristic *Fiscal Year / Periods* (0FISCPER) is selected and the compound characteristic *Fiscal Year Variant* (0FISCVARNT) is not selected as well, the data is aggregated using all fiscal year variants.

Units / Currencies

Units of key figures are considered independent fields. No currency translation takes place. If, for example, you aggregate using the key figure *Revenue* (0REVENUE), the currency is not considered when aggregation takes place. If necessary, include the relevant currency or unit field in the list of selected characteristics.

InfoObjects That Are "Attributes Only"

InfoObjects that are defined as *Attributes Only* are not offered as fields for selection if they are used in InfoProviders, especially with DataStore objects and master data tables.

Exception Aggregation

Key figures are only aggregated according to their standard aggregation. If a key figure with exception aggregation is present among the selected key figures, the associated reference characteristic for the exception aggregation has to be selected as a characteristic. This is to prevent a situation where a key figure *Number of Employees* with exception aggregation LAST VALUE is also totaled using periods. This is not worthwhile.

Exception aggregation can be reproduced using a subsequent grouping step:

- If the exception aggregation type is SUM, MIN, MAX, AVG, AV0 or NOP, you can use the *Aggregate Data* transformation. See Aggregating Data [Page 373].
- If the aggregation type is different to those listed above, you have to use the transformation ABAP routine and program aggregation yourself. See ABAP Routines [Page 379].

Non-Cumulative Key Figures

Non-cumulatives can be modeled in the BI system using a non-cumulative key figure with the corresponding fields for changing the non-cumulative or the corresponding fields for receipts or issues. The current non-cumulative is then saved on a marker.

If a non-cumulative key figure is selected, the closing stock balance for the month is provided for each period within the selected timeframe. This value is determined when data is read from the marker and the non-cumulative changes. If the selected period is outside of the area of validity, no non-cumulative is returned.

For more information see Modeling Non-Cumulatives with Non-Cumulative Values [External].

Features of Non-Cumulative Key Figures:

- The reference characteristic for time-based aggregation (time-reference characteristic) must always be selected as a characteristic. If additional validity-determining characteristics have been selected in non-cumulative parameter maintenance, these also have to be added to the selected characteristics.
- Since non-cumulatives are returned for all periods in the selected timeframe, a restriction for the time-reference characteristic has to be defined in a subsequent filter. Only one interval or a list of single values is supported here. Other restrictions for the time-reference characteristic will lead to errors during execution.

Activities

6. On the *Data Source* tab page, select an InfoProvider.
7. On the *Field Selection* tab page, select individual fields from which the system is to read data during the analysis process.

   On the right-hand side of the dialog box, all of the fields for the InfoProvider are listed, separated according to characteristics (top list) and key figures (bottom list). You have to select at least one key figure and one characteristic from these.

Example

The following characteristics and key figures are selected from DataStore object *RFM Response Rates* (0CRM_OFCV):

- RFM response rate model (0CRM_RFMFCV)
- RFM segmentation model (0CRM_RFMSGV)
- RFM R segment (0CRM_RFM_R)
- RFM F segment (0CRM_RFM_F)

- RFM number of responses (0CRM_RFM_RE)

Characteristic RFM M segment (0CRM_RFM_M) of the InfoProvider is not selected.

When the data is read, the data passed on to the subsequent nodes (the next step of the analysis process) is grouped together with the standard aggregation behavior of the key figure on the level of the selected characteristics. The aggregation for the key figure is SUM; all records with the same combination of characteristics are added together.

Example data:

- RFM response rate model (0CRM_RFMFCV) = test
- RFM segmentation model (0CRM_RFMSGV) = test

R segment	F segment	M segment	Number of responses
1	1	1	17
1	2	2	16
2	1	1	15
2	1	2	14
2	1	3	13
3	1	1	12

The data output for the nodes in this example is:

R segment	F segment	Number of responses
1	1	17
1	2	16
2	1	42
3	1	12

4.2.2.2.3 Query

Use

With the node *Read Data from File*, you can use data from a BW query in an analysis process.

At its exit, the node makes all fields of the query available as source fields for subsequent nodes.

Functions

When an analysis process is executed, the required data is read via the ODBO interface of SAP BW.

Activities

8. Choose a query from which the data should be read.

9. If the query has variables, select a variant.

4.2.2.2.4 File

Use

With the node *Read Data from File*, you are able to use data from a file in an analysis process. The file can either be found on an application server or on the presentation server.

Functions

The technical name of the InfoObject has to be in the first line of the file, for example, 0BPARTNER for the business partner. The data is in the following lines. The values are separated from one another with semicolons (;). Each line in the file becomes a data record. The data are transferred with ABAP MOVE logic into type-related fields. The data is expected in internal data format.

> Internal date format for displaying date: YYYYMMDD
>
> Example: 02. January 2004 = 20040102
>
> Internal date format for numeric entries: leading zeros are added until the field length is filled.
>
> Example for an eight digit field: 1234 = 00001234

If you require several columns with the same InfoObject, you can add a field name prefix when you specify the InfoObejct: <fieldname> <InfoObject>. Separate the field name from the name of the InfoObject with a colon (:).

> You want to read a file that contains the answers of a questionnaire in each line. The first line of the file can contain the following information:
>
> 0BPARTNER ; ANSWER1:0WS_ANSWER ; ANSWER2:0WS_ANSWER ; ANSWER3:0WS_ANSWER

Only data from the application server can be used in background processing.

Example

A file should contain the following data:

Business partner	Sales Revenue	Currency
1000	1200	EUR
1002	1500	EUR
1080	1100	EUR

The file must also have the following content:

0BPARTNER;0REVENUE;0CURRENCY

1000;1200;EUR

1002;1500;EUR

1080;1100;EUR

4.2.2.2.5 Database Table

Use

With the node *Read Data from Database Table*, you can use data from a database table in an analysis process.

4.2.2.3 Transformations for an Analysis Process

Use

Under *Transformations*, you will find functions for the preparation of data, as well as functions for transformation of data.

Preparation of data:

A complete, error-free data basis is decisive for the good quality of results of an analysis process. In order to be able to guarantee this, there are functions available that you can use to prepare the data basis accordingly.

Transformation of the data:

Using "real" transformations, it is then possible to uncover and map hidden information.

Functions

The following functions are available for the preparation of data in the analysis process designer:

- Limiting Data Volume [Page 372]
- Aggregating Data [Page 373]
- Joining Data from Multiple Sources (Join) [Page 374]
- Hiding Columns [Page 375]
- Sorting Data [Page 375]
- Transforming Lists into Data Records [Page 376]
- Transforming Data Records into Lists [Page 377]

The following functions are available for the transformation of data in the analysis process designer:

- ABAP Routines [Page 379]
- Data Mining Functions [Page 381]:
  - ABC classification
  - Weighted table scoring
  - Regression analysis
  - Prediction with decision tree
  - Prediction with cluster model
  - Prediction with data-mining model from third parties

You can find descriptions of the transformations in the analysis process designer as well in the dialog window for the respective node under *Help*.

4.2.2.3.1 Limiting Data Volume

Use

You use the node *Limiting Data Volume* to restrict the volume of data to be processed. You do this by specifying selection conditions. The columns in the table do not change. Single values and intervals are available to specify the condition.

Activities

10. On the *Field Selection* tab page, select the fields for which you wish to specify conditions.

11. On the *Filter Conditions* tab page, enter the conditions for these fields.

Example

You only want to read the data for specific customers from a DataStore object. For this purpose, you add a filter node to the analysis process, in addition to the *InfoProvider* data source. If you define the interval 1000-2000 as a filter condition for the *Customer* field, this condition is passed on to the data source during execution. This reduces the volume of data read.

Customer	Sales
1000	$ 500
2000	$ 300
3000	$ 250
4000	$ 600

→

Customer	Sales
1000	$ 500
2000	$ 300

4.2.2.3.2 Aggregating Data

Use

With the *Aggregate Data* node, you are able to group the data according to the values in specific fields (grouping fields, database function group by) and can aggregate the data within this group into other fields (aggregation fields). You are also able to only group data using these nodes, by only selecting grouping fields and not aggregation fields.

The node can be integrated into any point in an analysis process.

Functions

Since the data is going to be sorted, first all of the data has to be read. After sorting has taken place, the data is aggregated and passed on to the next node in groups. Only those fields are passed on that were either aggregated or used as grouping fields.

Activities

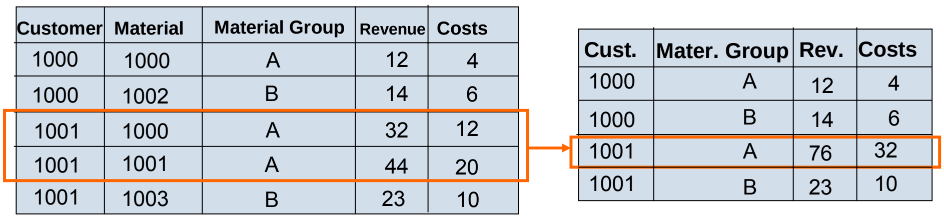
12. Select at least one grouping field from the available fields and if necessary, one or more aggregation fields.
13. As necessary, change the sorting sequence from ascending to descending in the grouping fields.
14. Specify the aggregation behavior for each aggregation field. The following values are currently supported:
    - SUM: All values in a group, that is those with identical values in all grouping fields, are added up and the total is passed on to the next node.
    - MIN: Only the minimum of the values in the group is passed on.
    - MAX: Only the maximum of the values in the group is passed on.
    - AVG: The average of all values in a group is passed on.
    - AV0: The average of all values in a group is passed on. But, zero values in the fields are not considered in this aggregation.
    - NOP: No aggregation.

Depending on the type of aggregation field, not all of the listed aggregation behaviors are supported. For example, for a field of type *Character*, only the values MIN and MAX are useful.

Example

If the characteristics *Material* (0MATERIAL), *Material Group* (0MATL_GROUP) and *Customer* (0CUSTOMER) along with the key figures *Costs* (0COSTS) and *Revenue* (0REVENUE) are available as input fields for the node, but you only transfer *Customer* and *Material Group* as grouping fields and the key figures as aggregation fields, then the data records look like this:

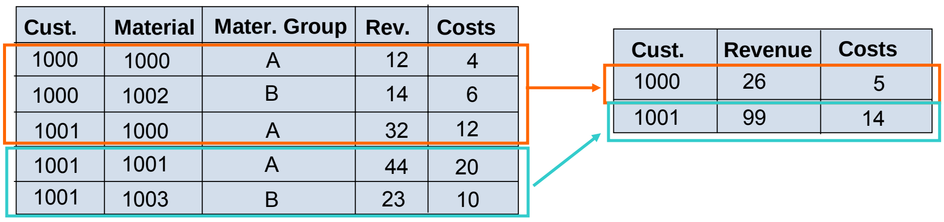
Aggregation on the customer level, material group with aggregation SUM for both key figures:



Customer	Material	Material Group	Revenue	Costs
1000	1000	A	12	4
1000	1002	B	14	6
1001	1000	A	32	12
1001	1001	A	44	20
1001	1003	B	23	10

Cust.	Mater. Group	Rev.	Costs
1000	A	12	4
1000	B	14	6
1001	A	76	32
1001	B	23	10

Aggregation on customer level with aggregation SUM for revenue and AVG for costs:



Cust.	Material	Mater. Group	Rev.	Costs
1000	1000	A	12	4
1000	1002	B	14	6
1001	1000	A	32	12
1001	1001	A	44	20
1001	1003	B	23	10

Cust.	Revenue	Costs
1000	26	5
1001	99	14

4.2.2.3.3 Joining Data from Multiple Sources (Join)

Use

You use the node *Joining Data from Multiple Sources* to join two different data sources with one or more common InfoObjects (database function Join [External]). Only the selected fields are transferred to the subsequent nodes.

Features

The join is performed in ABAP with a sort-merge-join logic. First, the data from the first data source is read and then sorted according to the fields in the join condition. Then the data from the second data source is read in turn and sorted according to the fields in the join condition. Finally, the data from the two tables is merged (merge).

If you have set the key figure *Process Data in Memory* (*Goto* → *Performance Settings)*, all the data is kept in the main memory when it is sorted. If this key figure is not set, the data is stored in temporary database tables.

> If, in the analysis process, you are joining large sets of data from the master data or from DataStore objects, SAP recommends that you use an InfoSet. This improves performance, since fewer temporary tables are required and the join is executed in the database itself.
>
> See InfoSets [External].

Activities

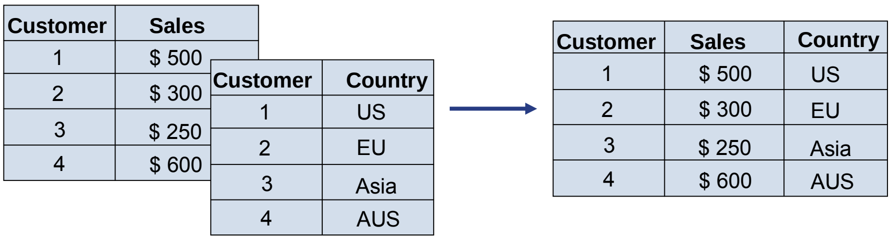
15. Select the fields that you want to pass on to subsequent nodes using the indicator in front of each field.

16. Between the fields for the various data sources which are to be linked, create a join condition by drawing a connecting line between the fields with the mouse.

17. In the context menu for the connecting line, define the type of connection (inner or outer join). In most cases, the inner join is relevant. For more information on the differences between the two types of join, see Left Outer Join with InfoSets [External].

Example

Data on customers in an InfoProvider is to be enhanced with attributes from the master data for the *Customer* characteristic (0CUSTOMER). To do this, you link the InfoProvider and the characteristic 0CUSTOMER with an inner join with a join condition via the field 0CUSTOMER. The required attributes from the master data and the selected fields from the InfoProvider form the output structure of the node. At runtime, the master data from the characteristic are added to each record in the InfoProvider.



Customer	Sales
1	$ 500
2	$ 300
3	$ 250
4	$ 600

Customer	Country
1	US
2	EU
3	Asia
4	AUS

Customer	Sales	Country
1	$ 500	US
2	$ 300	EU
3	$ 250	Asia
4	$ 600	AUS

4.2.2.3.4 Hiding Columns

Use

With the *Hide Columns* (*Projection)* transformation, you can hide fields or rename field names and text. Use the transformation if a data source has too many fields or if field names are incomprehensible.

Activities

18. On the *Field Selection* tab page, select fields that are to be transferred.
19. Enter new names for the selected fields on the *Rename Fields* tab page.

Example

Data from the master data table for business partner is being read. However, too many fields appear in the field selection. Use the projection transformation to hide columns.

With a join, the same column name occurs in both data sources. Use the projection transformation to rename the fields so that you can tell them apart more easily.

4.2.2.3.5 Sorting Data

Use

With the *Sort Data* transformation, you can sort data according to the selected fields. The node can be integrated into any point in an analysis process. Use the node if you would like to display the sorted data with the *Display Data* function.

The data structure does not change during sorting.

Functions

To sort the data, first all data in the data source is read, then it is buffered in the main storage or in the database and lastly it is made available in sorted form the transformations that follow.

Because almost all transformations perform sorts according to grouping fields, performing the sort beforehand with such transformations only leads to a decrease in performance because the data has been sorted once unnecessarily. The sorting transformation is particularly unsuitable for sorting data for the call of a routine transformation.

Activities

20. Select the fields according to which you want to sort.
21. Establish the sorting sequence. By default, the system sorts in ascending order, buy you can set the *Descending* indicator.

Example

Revenue is to be read from an InfoProvider in sorted form. In addition to the *InfoProvider* data source, a node for sorting is to be added to the model. If you select the *Revenue* field as the sorting field and set the *Descending* indicator, when the analysis process is executed, the data at the exit of the node will be sorted by revenue in descending order.

Customer	Revenue
1	$ 500
2	$ 300
3	$ 250
4	$ 600

Sort according to revenue

Customer	Revenue
4	$ 600
1	$ 500
2	$ 300
3	$ 250

4.2.2.3.6 Transforming Lists into Data Records

Use

The transformation *Transforming Lists into Flat Data Records* enables a change to the internal data structure. This creates a one-line data record at the node exit from a list of table rows at the node entrance.

Transferring a list of data records to a data record is done using a transformation field, usually a characteristic. You also select a transposition field - usually a key figure - the content of which is put in the new fields of the output structure depending on the characteristic value of the transformation field.

Every combination of a characteristic value of the transformation field and the selected transposition field can be assigned a new field in the output structure of the node. The new field has the same technical definition as the transposition field. The values for all of the non-selected fields are passed on to subsequent nodes without being changed.

Functions

Input and output structures differ in transformation of a list into a data record. The transformation field and the transposition field are not passed on to the subsequent node. Instead only the selected fields and the new fields that are defined in the details are passed on. The structure of the data is wider in the data record form, this means that the internal tables for the analysis process contain fewer data records, but they have more fields. Due to this, problems can especially occur when characteristics with lots of characteristic values are to be used as transformation field. If in this case new field names have to be specified for all or a number of characteristic values, the maximum permitted length of the data structures can be exceeded. In this case you can expect error messages from the process.

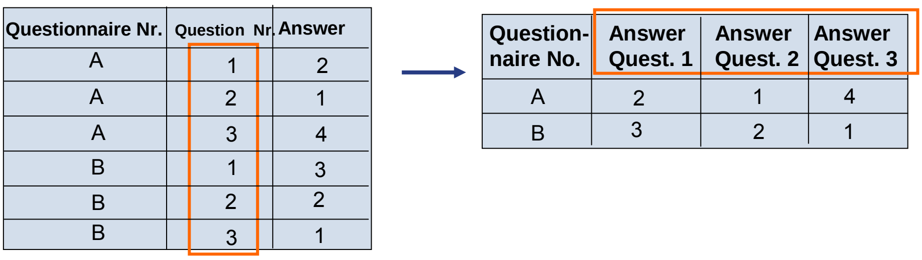
Activities

22. On the *Definition* tab page, from all of the available fields, select a transformation field that can be used to fill the newly generated fields in the output structure.
23. Select a transposition field.
24. On the *Details* tab page, assign a new field for the flat data record to every combination of a characteristic value of the transformation field and a transposition field. Specify a technical name for each new field. If you do not specify a field name, the value for this combination will not be passed on and will no longer be available at the exit.

Example

The questionnaire for a survey contains questions and answers in tabular form, that is, every record contains the number of the question as characteristic and the value of the answer for this question as key figure. However, for the report, the data needs to be in a transformed form in which the answer to each question has its own field. To do this, the transformation into a data record can be used.

In this example, the questionnaire contains three questions. If you transfer the questionnaire number directly, and define the question number as the transformation field and the answer as the transposition field, then this output structure for this node would have the following form:



Questionnaire Nr.	Question Nr.	Answer
A	1	2
A	2	1
A	3	4
B	1	3
B	2	2
B	3	1

Question-naire No.	Answer Quest. 1	Answer Quest. 2	Answer Quest. 3
A	2	1	4
B	3	2	1

4.2.2.3.7 Transformation of Data Records into Lists

Use

Using the transformation *Transform Data Record into List*, you can transform repeat columns in a table into several rows. Here the system generates several records in the output table for each individual data record in the input table.

This transformation is similar to transposition of a table. During transposition, rows and columns in the whole table are swapped. This transformation, however, only transposes selected columns in the table; the other columns are transferred without changes.

Functions

The type of input column should concur with the type of transposition field. The field content is transferred with ABAP-MOVE logic and converted as needed. If conversion is not possible, for example because the FLOAT value `1.00E30` is to be transferred into a decimal field of length 10, it causes the analysis process to terminate due to overflow.

Activities

25. On the *Definition* tab page, select the fields to be transformed. These are the fields that are to be filled by the transformation. In the example below, these are the columns *Answer Question 1, Importance Question 1, Answer Question 2, Importance Question 2, Answer Question 3* and *Importance Question 3*. The *Questionnaire* characteristic is transferred unmodified. These columns are deleted from the output structure.

26. On the *Transformation* tag page, define the transformation field. The new column that adds the characteristic value to the column name is also defined here (with name, description and InfoObject for type). In the example below, this is the new column *Question No.*.

27. Also define at least one transposition field on the *Transformation* tab page that is to be filled by the transformation. The fields defined here are inserted into the output structure of the node. In the example, these are the new columns *Answer* and *Importance.*

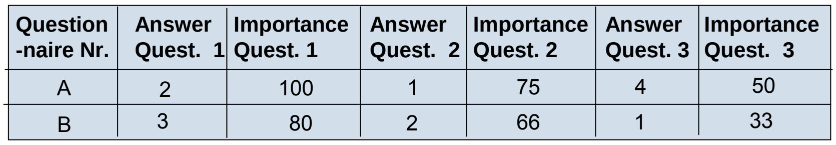
28. With the previous steps, you described the new output structure. Now the transformation rules that describe how the field contents are to be transferred from the old into the new structure are missing.

    On the *Details* tab page, for each field from the original data record, specify the row and column in the output table into which the value is to be placed. The row is specified with a characteristic value for the transformation field and the column through the selection of a transposition field.

    For the column *Importance Question 1*, the characteristic value **1** is entered for the characteristic value of *Question No.* and for the column, the new column *Importance* is entered*.*

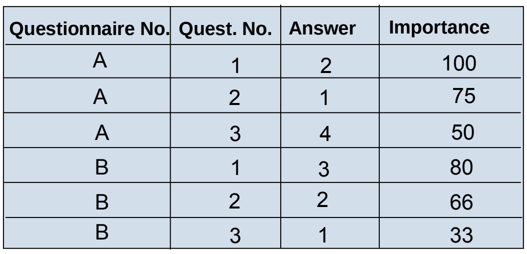
Example

The answers to a questionnaire are available in table form. The table contains all of the answers to the questionnaire by line:



Question -naire Nr.	Answer Quest. 1	Importance Quest. 1	Answer Quest. 2	Importance Quest. 2	Answer Quest. 3	Importance Quest. 3
A	2	100	1	75	4	50
B	3	80	2	66	1	33

For the following process, this table should be put into a format in which each data record contains the answer and importance of one question. When executing the analysis process, a part of the column name (question 1/question 2/question 3) is converted into a characteristic value for a new column *Question No.*. The repeating fields *Answer* and *Importance* are transferred into the new columns *Answer* and *Importance*:



Questionnaire No.	Quest. No.	Answer	Importance
A	1	2	100
A	2	1	75
A	3	4	50
B	1	3	80
B	2	2	66
B	3	1	33

4.2.2.3.8 ABAP routines

Use

With the transformation *ABAP Routine*, you can transform the data for an input table into an output table. You can implement any transformation with ABAP yourself.

Prerequisites

You need authorization for the authorization object RSANPR and activity 36 (enhanced maintenance) to insert ABAP routines into an analysis process or to change them.

Functions

You can access all rows in the input table in this ABAP routine. The output table can contain any number of rows. This makes the following calculations possible, for example:

- Determining an ABC class
- Determination of a lifetime period
- Statistic functions such as *Number of Customers That Have Made Purchases per Period*

You can use different types of fields in your ABAP routines:

- Grouping fields:

  Grouping fields are available in the routine as input fields. They are output to the output table of the node unchanged.

  You use the grouping fields to define a group of data records. All data records with the same values in the group field form one group. The data in the input table for the node are also grouped according to grouping fields and the routine is only called with the data for each group. If, for example, determination of the ABC class is to be implemented per customer group, the partitioning of the data according to customer groups will already be done by the routine node. The routine is then called up multiple times with the data for each customer group.

- Source fields:

Source fields are available in the routine as input fields. However, they are not output to the output table of the node.

You can transform source fields into other fields in the routine. If you want to output a source field unchanged into the output table of the node it has to be defined as a target field as well. All selected source fields are transferred to the list of target fields as a suggestion.

- Target fields:

  Target fields are not available in the routine as input fields. However, they are only output to the output table of the node.

  You can use a target field, for example, to include an additional column in the output table. You define a target field by specifying a unique field name, a description and an InfoObject. The fields appear in the subsequent nodes with the field names and the description. The InfoObject is used to assign type properties to the field (type, field length, check table).

Implementation of the Routine:

The grouping fields in a structure are available in the routine. They cannot be changed. Thus the fields are not contained in the output structure of the routine. The columns with the grouping fields are automatically enhances in the output table of the node afterwards.

The implementation of the data transformation occurs on the *Routine* tab page. Here you are implementing a subprogram that is called for each group of data records. Data transfer takes place using three parameters:

Parameter	Description
`is_group`	Contains the structure with the grouping fields
`it_source`	Contains the input table without the grouping fields
`et_target`	The output table for the routine. This table must be filled in the routine. After the routine has ended, the data in this table is enhanced with the columns with the grouping fields and the subsequent nodes are transferred.

The following restrictions apply:

- The sequence of data records in the data package is not guaranteed. Before the transformation is called, the data is sorted according to grouping fields. Especially due to this sorting, a sequence that already existed from a previous transformation is often destroyed.
- The sequence of the data package is not guaranteed.
- Data for global variables may not be swapped between different calls of routines.
- The sequence of fields in the data package is not guaranteed.

Activities

29. Determine the grouping and source fields by filling the appropriate lists on the *Source Fields* tab page.
30. Define the target fields on the *Target Fields* tab page by specifying a unique field name, a description and an InfoObject.

If you have previously selected source fields from the *Source Fields* tab page, the fields are transferred to the list of target fields as a proposal. You can delete or enhance these fields.

31. Implement the data transformation on the *Routine* tab page.

    Program code that copies the data for the input table into the data for the output table is generated as a proposal.

4.2.2.3.9 Data Mining Methods

Use

The following data mining methods are available to you:

- *ABC Classification:*

  For more information see ABC Classification [Page 401].

  For procedures, see Executing ABC Classification [Page 418]

- *Weighted Table Scoring*:

  You can find more information under Scoring [Page 396].

  For procedures, see Transformation for Weighted Table Scoring [Page 417].

- *Regression analysis:*

  You can find more information under Scoring [Page 396] and Regression Analysis [Page 397].

- *Prediction with decision tree*:

  You can find more information under Decision Trees [Page 395].

  For procedures, see Executing Predictions [Page 415] and Evaluating Decision Trees [Page 416].

- *Prediction with cluster model:*

  You can find more information under Clustering [Page 405].

  For procedures, see Executing Predictions [Page 415].

- *Prediction with data-mining model from third parties*

  For procedures, see Executing Predictions [Page 415].

See also:

Data Mining [Page 393]

4.2.2.4 Data Targets for an Analysis Process

Use

The staged or transformed data for an analysis process is saved in a data target.

Features

The following types of data target are available in the Analysis Process Designer:

- Attributes of a characteristic [Page 382]
- DataStore object [Page 383]
- CRM attributes [Page 385]
- Target groups for SAP CRM [Page 386]: This data target is only available in the *Survey* application.
- Data mining models [Page 387]:
    - Training the decision tree
    - Training the clustering model
    - Training the scoring model (regression)
    - Training data-mining models from third parties
    - Creating association analysis models

You can find descriptions of the data targets in the analysis process designer, as well in the dialog box for the respective node under *Help*.

4.2.2.4.1 Attributes of a Characteristic

Use

With *Changing the Attributes of a Characteristic* as the data target, you can change the attribute values in the master data table of a characteristic.

Note the following:

- Only attribute values in records that already exist can be changed. It is not possible to insert new records for the master data table with a new analysis process.
- The referential integrity of the attribute values is checked. This means that for all attributes with master data tables, the system checks whether the attribute values exist in the master data table. Checking of attribute values is done by package and compared to the newest data in the master data tables. If there is an error, the analysis process terminates.
- Attribute values are only changed in records that are contained in the source data.
- If the input table contains several records with the same key, from which record the attribute value is filled is not defined. As needed, use another grouping step before the update.
- With compounded characteristics, the compounded characteristics also have to be filled.
- Units and currencies are treated like normal characteristics. Key figures and units can be changed independently of one another.
- An analysis process locks the master data table against competing changes.

Functions

In order to be able to process mass data, the data are written by package. After every package, the data is written to the database (COMMIT). In case of error, the master data could already have been partially changed. A reset occurs when the analysis process restarts, which cause all of the attribute values to be overwritten again.

The process chain can help you to automate the execution of the analysis process and activation of changes. You add an analysis process to a process chain using the general service *ABAP Program*. Use the program RSAN_PROCESS_EXECUTE in it. For the second step, the activation of the master data, use an attribute change run (under *Other BW Processes*).

Activities

32. On the *Data Target* tab page, select the characteristic whose attributes are to be changed.
33. On the *Attribute* tab page, select fields that are to be changed.
34. Explicitly define the field assignment for this data target by double clicking on the inbound data flow arrow.
35. After executing the changes on the master data table, activate.

4.2.2.4.2 DataStore Object

Use

With *Writing Data to a DataStore Object* as the data target, you can save calculation results from an analysis process in a DataStore object for direct update. With every run of the analysis process, first all of the data in the DataStore object is deleted, then the result is calculated and written to the DataStore object again. Each calculation of an analysis process thus corresponds to a full update with the data that already exists being deleted beforehand.

If you want to compare multiple calculation results with one another, you can use a DataStore object for direct update as a temporary buffer for the calculation result of the analysis process. Execute the analysis process. Check the data in the DataStore object as needed. If the calculation result is OK, use an update rule to write the data to an InfoCube or a normal DataStore object.

Features

On the *Target Area* tab page, you can partition the DataStore object into several subareas. The analysis process then only deletes and writes data into one partition at a time. You use the target area when you want to manually parallelize the processing of an analysis process.

One ABC classification of customers should be implemented per business area. Each business area also gets in own analysis process. Then each analysis process only reads the data for a business area and writes the calculation result for exactly one business area. In this case, select the business area characteristic to restrict the partition and then specify a characteristic value to restrict the partition. The analysis process then only writes data to this one partition. Other analysis processes can write to other partitions. If you create an analysis process for each business area, processing can be started in parallel.

Updating in Detail:

- When writing the data, the records are always only inserted (INSERT). If the inbound table contains several records for the same key, the system terminates the analysis process. As needed, use an aggregation transformation before the data target to aggregate the data to fit the key fields of the DataStore object.
- So that you can process mass data, the data from the analysis process is processed internally in technical packages. When the data is written to the target area, the system saves the data of each package in the database.

  ⚠

  If the analysis process terminates during execution, it is not clear which data has already been written up to that point. The data in the target area may be entirely deleted or only partly there. In this case, eliminate the cause of the error and start execution again.
- When writing to a DataStore object for direct update, a lock is currently not set. This means that several analysis processes can write in various partitions in the DataStore object. You should make sure that two analysis processes do not write to the same partition to the same DataStore object.
- Currently, no validity check is performed on the field values before they are written to the target area. This means that the data is expected to have the following format:
  - With characteristics with master data tables, only valid characteristic values may be transferred.
  - The data are in internal format: NUMC fields contain leading zeroes, all digits are filled for dates, and the conversion has already been implemented on fields with Conversion Routines [External].

The Following Restrictions Apply:

- From a technical viewpoint, the same DataStore object can be used as a data source and as a data target in one analysis process simultaneously. However, because the data in the target area is always deleted first, and then the data is read, this approach cannot be used to change a data field in a DataStore object.
- Only key fields of the DataStore objects can be used to define a target area.
- Fields that were already defined in the settings for the target area to restrict the target area are no longer offered in the field assignment for this data target.
- Entering a value in the *Target Area* tab page occurs in a similar way to entering a constant in the field assignment on the incoming dataflow arrow. The difference lies in the restriction of the target area, in which the data is deleted and written again. For this, the only characteristics considered are those specified in the *Target Area* tab page. If the characteristic value is defined in the field assignment, all data will still be deleted.
- Currently queries are not supported on a DataStore object for direct update because master data IDs (SID) are not determined during update of this type of DataStore object. For this reason, create for reporting an InfoSet containing the DataStore object. See DataStore Object for Direct Update [External].

Activities

36. On the *Data Target* tab page, select the DataStore object for direct update that you want to fill.
37. As needed, enter values to restrict the partition on the *Target Area* tab page.
38. Explicitly define the field assignment for this data target by double clicking on the inbound data flow arrow.

Example

You have modeled an analysis process that implements a customer evaluation. The result of the valuation is expressed as the attribute *Customer Class*, which takes the values gold, silver, and bronze. The analysis process delivers a table with two columns: *Business Partner* and *Customer Class*:

Business Partner	**Customer Classification**
4711	Gold
4712	Silver
4713	Bronze

In order to be able to track changes between customer classifications, this evaluation should be run once a month and the result should be saved for each month.

To do this, create a DataStore object for direct update with the key fields *Calendar Month* and *Business Partner*. In the data part, include the field *Customer Class*. You use this field to store the customer classification of the business partner every month.

In this example, the target area is all the data for one month. Enter a value for the month in the *Target Area* tab page, for example, June 2003. On an inbound data flow arrow In the field assignment, you can now only assign the fields *Business Partner* and *Customer Classification*.

Execute the analysis process. Afterwards, the evaluation results are available in the DataStore object under *June 2003.* If you want to perform the evaluation in July, copy the analysis process and change the value in the data target on the *Target Area* tab page from *June 2003* to *July 2003.*

4.2.2.4.3 CRM Attributes

Use

With *Update CRM Attributes* as the data target, you can write attribute values for the business partner into SAP CRM. In this way you can, for example, fill in the ABC class of the business partner in the Interaction Center of the SAP CRM.

Functions

The attributes that are available depend on which data targets and attributes are defined in SAP CRM and also on which Release status your SAP CRM has.

The following types of data targets can be filled in SAP CRM 3.1, for example:

- IC attributes for the business partner

- Marketing characteristics for business partner segmentation
- Enhancements depending on business partner

If no data targets or attributes are offered on the *Data Target* tab page for your SAP CRM, in SAP CRM Customizing, create data targets under *Settings in SAP Business Information Warehouse* → *Data Transfer from SAP BW.*

Activities

39. On the *Data Target* tab page, select the logical system into which the data is to be written.

    The data targets defined in the specified system are determined.

40. Select a concrete data target - and depending on the type of data target - a subordinate object as well.
41. On the *Attribute* tab page, select the attributes that are to be filled.

4.2.2.4.3.1 ata Targets for SAP CRM

Use

You can transfer the values determined in the SAP BW to SAP CRM and place these values in attributes – essentially business partner attributes. You can fill the following types of data targets in SAP CRM:

- **Marketing Attributes for Business Partner Segmentation**: You can create marketing attributes for business partner segmentation and assign the attribute to an attribute set. For more details, see *SAP Customer Relationship Management* → *Enterprise Marketing* → *Customer Segmentation* → *Segmentation Preparation* → *Characteristics* → *Working with Attributes*
- **CRM Target Group:** To enable selective marketing, you can classify your business partners into different target groups on the basis of certain shared marketing-relevant attributes (for example: age, occupation, hobbies, income and so on). For more details, see *SAP Customer Relationship Management* → *Enterprise Marketing* → *Customer Segmentation* → *Target Group Creation* → *Target Groups* → *Working with Target Groups*
- **IC Attributes**: You create IC attributes directly as data targets. For more details, under SAP Customizing see *Settings for SAP Business Information Warehouse* → *Data Transfer from SAP BW* → *Maintain Attributes and Select BW Key Figures for Business Partners*
- **Enhancements Specific to Business Partners:** You define **tables** in the *ABAP Dictionary*, which you can call up in the SAP menu under *Tools* → *ABAP Workbench* → *Development* → *Dictionary*.

  The table has to fall within the customer namespace (that is, the table name has to begin with Y or Z) and it has to have the business partner number as its key.

- **Analytical Data Storage:** The analytical data storage (ADS) enables SAP Customer Relationship Management (SAP CRM) business applications to **efficiently** access information from SAP Business Intelligence (SAP BI). For more details, see *SAP*

Customer Relationship Management → *Application Services* → *Analytical Methods* → *Optimize Relationships* → *Analytical Data Storage* in the SAP Library.

You can transfer data from the following sources in SAP BW:

- InfoObject attributes, such as attributes for the InfoObject *Business Partner* (0BPARTNER)
- Queries, in which the master data – such as the business partner – is unique

For transferring data from SAP BW, start by defining a data target in SAP CRM, and then use this data target in SAP BW to model the data transfer process in the form of an analysis process.

For more details, see *SAP Customer Relationship Management* → *Application Services* → *Analytical Methods* → *Optimize Relationships* → *Transferring Attribute Data from SAP BW* in the SAP Library.

4.2.2.4.4 Data Mining Models

Use

Data mining models, which first have to be trained (created) are treated like a data target. In this case, execution (transformation) of the data mining model takes place only after training.

The following data mining models are available to you:

- *Training Decision Trees:* You can find more information under Decision Trees [Page 395].
- *Training Cluster Models:* You can find more information under Clustering [Page 405].
- *Creating Association Analysis Models:* You can find more information under Association Analysis [Page 406].
- *Training Scoring Models:* You can find more information under Scoring [Page 396].
- *Training data-mining models from third parties*

For procedures, see Training Models [Page 413].

See also:

Data Mining [Page 393]

4.2.2.5 Modeling an Analysis Process

Use

With the help of an analysis process, you can attain new insights from your data. Various types of data sources, data targets and transformations are available for modeling your analysis process. When you execute the modeled analysis process, the data are written to the data target and transformed as needed.

You model an analysis process in a specific context, that is, for a specific application such as *Fill CRM Attribute* or *Create Target Group for BW Survey.* Each analysis process is only valid for the application for which it was created. The authorizations are also assigned specific to an application (authorization object RSANPR). Depending on application, various data sources, data targets and transformations are available to you. If you select *General* as the application, almost all data sources, data targets and transformations are available to you.

Data sources provide the input data for an analysis process. See also Data Sources for an Analysis Process [Page 366].

You can change the data in a data source using a **transformation**. See also Transformations for an Analysis Process [Page 371].

You can write any of the prepared or transformed data to different **data targets**. See also Data Targets for an Analysis Process [Page 381].

Functions

Modeling of an analysis process takes place with the following graphic elements:

- Nodes
- Data flow arrows

Nodes

Data sources, transformations and data sources are represented by nodes that are connected with data flow arrows. The various types of nodes are indicated using various decorators:

- Data sources are indicated in the work area with a small square
- Transformations are indicated in the work area with a small triangle
- Data targets are indicated in the work area with a small circle

These decorators enable you to differentiate between the BW object *InfoProvider* and the function of the analysis process *Read Data from the InfoProvider.*

By double clicking on a selected node, you call up a dialog box in which you can make detailed settings. You can call up documentation on each dialog box with *Help*. Each node has one or more connectors (small red triangles) with which you can connect the nodes using the mouse. To the left of the node is the inbound connector (not available for data sources) and to the right is the outgoing connector (not for data target).

Data flow arrows

The nodes are connected to one another with data flow arrows. For data flow arrows with a symbol, you have to define a field assignment explicitly between inbound and outbound nodes. Call up a dialog box by double clicking on the arrow. For the other data flow arrows, field assignment takes place automatically.

All of the analysis processes for the current application are listed on the left side of the screen. Under templates, you will find all of the analysis processes provided by SAP as examples. These templates provide you with a starting point on how you can define an analysis process for a specific application. You cannot change a template; instead you may copy it and then edit it.

On the side of the screen to the left of the work area, you can see the nodes that are available for an application. You can drag them into the work area using Drag&Drop and connect them with the mouse. To delete a node again, select it using the *Delete* context menu.

The context menu if a node provides you with a variety of functions that you support you during modeling and when you are executing an analysis process, as well as helping you with interpretation of the analysis results. See also Checking Data [Page 389].

When you execute the analysis process (directly or in the background) the determined data is posted to the data target.

See also:

Create Analysis Process [Page 389]

4.2.2.5.1 Creating an Analysis Process

Procedure

You are in the SAP Easy Access SAP Business Information Warehouse. In the SAP menu, choose *Special Analysis Processes* → *Analysis Process Designer*. In order to create and execute a simple analysis process with transformation, proceed as follows:

1. Choose *Create.*
2. Select an application from the dropdown menu and select ✓ *Okay.* Your analysis process will be assigned to the appropriate folder on the left side of the screen.
3. Specify a description.
4. Drag a data source into the work area and make the following detailed settings in the dialog box that appears.
5. Drag a transformation into the work area. By double clicking on the transformation node, you can make the settings.
6. Drag a data target into the work area and make the following settings in the dialog box that appears.
7. Connect the nodes with the mouse.
8. To make an explicit field assignment, double click on the data flow arrow that connects the nodes.
9. Save your analysis process. Specify a technical name.
10. Before you execute your analysis process, you have the option of checking the data and of calculating intermediate results for performance optimization. See also Checking Data [Page 389].
11. Choose *Check.*
12. Choose *Activate.*
13. Execute the analysis process. The data is written to the data target and the log is displayed.

4.2.2.5.2 Checking Data

Use

A complete, error-free data basis is decisive for the results of an analysis process. This is realized step by step with the APD. It is then possible to check each individual processing step. The APD offers you the option of display the data for each step in the analysis process, to calculate intermediate results and to analyze the quality of the data for some nodes. You can perform this check of the data even before you execute the analysis process.

Functions

Display data

Using the *Display Data* function in the context menu for a node, you can display the data contained in a data source in a table. If an intermediate result was already calculated, it is displayed.

Basic statistics

Using the *Display Basic Statistics* function in the context menu for a node, you can display the statistics for the selected fields. This information on the data includes histograms, distribution and frequency calculations, simple statistic key figures, such as arithmetical means, standard deviations or correlations.

Here the information is differentiated with the values according to value type of the field. It differentiates between discrete (DST) and continuous (CNT) fields:

- *Discrete* means that there are a number of countable values for the field. This applies to almost all characteristics with check table. For characteristics with a large number of values, such as *Business* Partner, a report for each single value does not make much sense.

  Basic statistics for discrete fields: A frequency table of the most frequent values is displayed.

- *Continuous* means there are is an undefined number of values. A typical example of this is the key figure *revenue*.

  Basic statistics for continuous fields: A frequency table of the most frequent values is displayed. A value distribution in intervals, the average value, the standard deviation (based on the population) and additional figures are displayed. You can see how these figures are calculated in Formulas for the Calculation of Statistics [Page 391].

You can select the value type for each selected field. However, the system always suggests a reasonable value type: *continuous* for numeric fields and *discrete* for non-numeric fields.

The *Color* field, with the values red, blue... receives the value type *discrete* as the proposal. The *Environment* field, with values between 0 and 1000, receives the value type *continuous* as the proposal. If *Gender* is coded as an integer (1 for *male*, 2 for *female*, 0 for *sex unknown*), you should change the suggested value type from *continuous* to *discrete* because calculating the average value does not make sense. If you have chosen *continuous* for a non-numeric field, the system automatically changes the value type to discrete during execution.

Intermediate result

Using the *Calculate Intermediate Result* function in the context menu for a node, you can calculate the data up to this node. The result is saved in a temporary database table and is helpful, for example, if you want to try out different options after this node during modeling of the analysis process. The intermediate results are also helpful with performance optimization during execution of the analysis process with large amounts of data. If an intermediate result is available for a node, this is displayed with an icon ▦. The intermediate result becomes invalid and is no longer displayed if the node was changed. In this way you can also delete the intermediate result if it is no longer current.

Calculation summary

After you have executed the analysis process, you can display additional information about the calculation of the data using the *Calculate Calculation Summary* function in the context menu for a node. This information can only be called for data mining methods. Depending on the type of transformation, they are comprised of statistical data, probability information or similar. They help to improve evaluation of data quality.

Notes

In order to display data and statistics, you need authorization for the simulate activity (48) in authorization object RSANPR.

With the *Display Data* and *Display Basic Statistics* functions the complete calculation is performed up to the specified nodes with all data. This can lead to a short dump due to exceeding the maximum allowed runtime for the dialog process. In this case, create an intermediate result in the background for the selected node. Start the simulation again when the intermediate result has been calculated.

For larger volumes of data, short dump can also occur due to memory overflow. In this case, you select *Goto* → *Performance Settings* and delete the *Process Data in Memory* indicator. This indicator specifies whether the data is maintained in the main memory during the analysis process or whether data is temporarily stored in the database. This indicator is set by default, that is, the data is processed in the main memory. This setting is ideal when small amounts of data are to be processed. For larger volumes of data, the program can terminate when the data no longer fit in the main memory. If this occurs, deactivate this indicator. Then the data is temporarily stored in temporary tables in the database during the analysis process to reduce the main memory requirements. The generated tables begin with /BIC/000AP.

Tips for processing large amounts of data:

- Optimize the performance by inserting a filter directly behind the data source.
- Test with mass data: Insert a filter for testing to select a partial quantity of the data. With this restricted volume of data, select *Display Data* or *Display Basic Statistics*. Before executing the analysis process, delete the conditions in the filter.
- If calculation takes too long, you can end the simulation in the Windows system menu of the new window using *Cancel Transaction*.

4.2.2.5.2.1 Formulas to Calculate the Statistics

The following formulas are used to calculate statistical numbers for continuous fields:

Statistical Numbers	Formula
Average	$\bar{x} := \frac{1}{n} \cdot \sum_{i=1}^{n} x_i$ For a series of values $x_i, i = 1,\ldots,n$
Median	The median of a series of values is the central value (or the average of the two central values, if n is even) for the ordered series.

Quartile	für $\alpha \in [0,1]$ is the α -quantile $q_\alpha := \frac{1}{2}\left(\min\left\{x_j \middle\| \#\left\{x_i \mid x_i \le x_j\right\} \ge \alpha \cdot n\right\} + \min\left\{x_j \middle\| \#\left\{x_i \mid x_i \le x_j\right\} > \alpha \cdot n\right\}\right)$ The quartiles are the α-quantiles for α = 1/4, 2/4 and 3/4. Note that the 0.5-quantile is the median.
Standard deviation	The formula for standard deviation is based on the entirety of the values used. $\sigma := \sqrt{\frac{1}{n} \cdot \sum_{i=1}^{n} (x_i - \bar{x})^2}$
Variation coefficient	The variation coefficient is the normalized standard deviation. $\frac{\sigma}{\bar{x}}$ if $\bar{x} \neq 0$
Relative skewness	$\gamma := \frac{1}{n} \cdot \sum_{i=1}^{n} (x_i - \bar{x})^3 \Big/ \sigma^3$ The relative skewness is a key figure that contains the information about the asymmetry of the data. If the distribution of data around the median is symmetrical, then the skewness is 0 (zero). If the distribution has a peak with a short tail on the left side and long tail on the right side, then the skewness > 0. If the long tail is on the left side and the short tail is on the right, then the skewness is < 0.
Outlier	x_i is an high outlier if $x_i > q_{.75} + 1.5 \cdot (q_{.75} - q_{.25})$ x_i is a low outlier if $x_i < q_{.25} - 1.5 \cdot (q_{.75} - q_{.25})$ x_i is an outlier if it is a high or low outlier.

4.2.2.6 Additional APD Functions

Features

In the main menu, you can choose *Environment* to jump to the Data Mining Workbench, DataStore object maintenance, the Data Warehousing Workbench or the BEx Analyzer.

The following functions are available in the toolbar:

Display / Change

If you call up an analysis process that already exists, it appears in display mode first.

Active <-> Inactive

With this function, you can switch back and forth between versions.

Full Screen On/Off

Use this function to hide the hierarchical display of the analysis process in order to get a larger work area for modeling the analysis process.

Check

Use this function to check your analysis process is complete and consistent.

Schedule Job

Use this function to schedule the execution of your analysis process in the background.

Job Overview

Use this function to jump to the analysis process monitor. The analysis processes for a selected period are displayed on the left-side of the screen area in a tree structure. You can change the period selected using *Filter*. You can adjust the tree structure to your requirements using *Configure Tree*.

When you double click on an analysis process, the details are displayed on the right-hand side of the screen. General information on the analysis process is displayed on the *Header* tab page. The *Log* tab page displays the messages for each processing step.

Undo / Restore

You can use this function to undo previous actions and to restore actions you have undone.

Importing and Exporting Analysis Processes

You can export a model into an XML file and import this file back into the system. In this way, you can transfer a model into another application. However, this is only possible when all of the nodes used are available in the application into which you want to import them.

You import the XML file by first creating a model in the application and then entering the properties of the model. Then choose *Import Analysis Process*.

Create Data Mining Models

Use *Create Data Mining Models* to get to the data mining workbench in order to create data mining models. For more information, see Creating, Changing and Activating Models [Page 408].

4.2.3 Data Mining

Purpose

You can use data mining to automatically determine significant patterns and hidden associations from large amounts of data. Data mining provides you with insights and correlations that had formerly gone unrecognized or been ignored because it had not been considered possible to analyze them.

Since each company has different data mining requirements, it is not possible to deliver fixed models for producing prediction results. However, the data mining methods available in SAP BW allow you to create models according to your requirements and then use these models to draw information from your SAP BW data to assist your decision-making. For example, you can analyze patterns in customer behavior and predict trends by identifying and exploiting behavioral patterns. Data mining models can be used to provide answers to decision-making questions like the following:

- Which offer is most appropriate to which customers and when should that offer be made?
- Which customers are liable to churn?
- How high is the cross-selling potential for a new product?

Implementation Considerations

You can access data mining methods from the SAP Easy Access menu under *Enhanced Analytics* → *Data Mining Models*.

The data mining methods can also be accessed from the menu for the role *Customer Behavior Analysis* (SAP_BWC_CUSTOMER_BEHAVIOR).

Integration

Alongside SAP's own data mining methods, you can alternatively use the role *Customer Behavior Analysis* (SAP_BWC_CUSTOMER_BEHAVIOR) to access an interface to the IBM Intelligent Miner.

Features

SAP delivers the following SAP-owned data mining methods, which can be supplemented by the models that you create:

- Decision Trees [Page 395]
- Clustering [Page 405]
- Association Analysis [Page 406]
- Scoring [Page 396]
- Weighted Score Tables [Page 399]
- ABC Classification [Page 401]

Decision trees display data using (non-continuous) category quantities. The display rules are determined in training using those sections of historic data where the assignment to categories is already known.

Clustering is used to split data into homogeneous groups. The model looks for a global structure for the data with the aim of partitioning the data into clusters.

Association analysis can be used to establish composite effects and thereby identify cross-selling opportunities, for example. The search for associations considers objects with information content that is remotely comparable. Statements are formulated about partial structures in the data and take the form of rules.

In contrast to decision tree classification, clustering and association analysis determine the models using the data itself.

In **scoring**, data is displayed using continuous quantities. If required, discretization can then be applied to split the data into classes. The scoring function can either be specified using weighted score tables or be determined by training using historic data as linear or nonlinear regression of a target quantity.

ABC Classification displays data grouped into classes of A, B, C and so on, using thresholds and classification rules. The classified results are displayed in the form of ABC chart or list.

You can use historic data to train the models that you create for these data mining methods. This data helps the model to learn by establishing formerly unrecognized patterns. You can either export the result of this learning process into another system (association rules) or you apply the result during prediction to other data that lacks certain information (clustering, decision trees).

You use BW queries to train the model and perform the prediction. You assign these BW queries to the model as sources for the respective business transaction.

4.2.3.1 Decision Trees

Use

Decision trees are used to learn from historic data and to make predictions about the future. Prediction involves establishing rules using historic data and applying these rules to new data. These rules are displayed graphically as a hierarchy.

> Your customer data typically contains attributes such as gender, age, income, region, and occupation as well as information about whether a customer is a satisfied customer or not (possibly drawn from a survey). You can use such historic data to train a decision tree. You find out as a result that customers exhibiting certain attributes are generally satisfied customers while customers exhibiting other attributes tend to be dissatisfied customers. You can use rules determined in this way to assess the satisfaction of other customers in cases where this information is not available.

Integration

The data that you use to train the model can be taken from any other system, provided that the system can extract data into SAP BW. Similarly, you can apply the identified rules to any data that has been extracted into SAP BW. In SAP BW, you can use queries to access data with known statements and then use this data to find out statements about other data.

Features

You can make the following settings in a model for the *Decision Trees* method*:*

You use the **model fields** to specify which characteristic is to be considered with which attributes (such as the characteristic *Customer* with the attributes *Occupation, Gender, Age*, and so on). Moreover, you specify for which attribute the dependency on other attributes should be determined (such as the attribute *Customer Satisfaction*). The system then determines which of the attributes influences the dependent attribute most and takes the most influential attribute as the basis for building the decision tree.

You can use the **model parameters** to specify, for example, whether training should be executed using all data or whether the windowing technique should be applied to select just a representative part of the data. Furthermore, you can enhance the quality of the tree by specifying conditions for when the system should stop building the decision tree as well as by activating relevance checks and pruning.

You can display the result graphically as a hierarchy or in the form of rules. For the graphical display, you can set filters for nodes and call up detailed statistical information for individual nodes. You can also view the specific rule corresponding to a particular node in the decision tree. You must also create an analysis process to execute the prediction.

See Also

Creating, Changing and Activating a Model [Page 408]

Creating Analysis Process for Training [Page 413]

Analysis Process for Executing the Prediction [Page 415]

Evaluating a Decision Tree Model [Page 416]

Analysis Process Designer [Page 362]

4.2.3.2 Scoring

Use

The purpose of scoring is to valuate data records. Here you can choose from three function types with which to perform the valuation:

- Weighted score tables [Page 399]
- Regression Analysis [Page 397]
  - Linear regression
  - Nonlinear regression (implemented with multilinear splines)

Integration

The data that you use to train a model can be taken from any other system, provided that the system can extract data into SAP BW. Likewise, you can apply the same valuation to any data that has been extracted into SAP BW.

Prerequisites

You can use queries in SAP BW to access the data with which you wish to calculate scores. For the function types *Linear Regression* and *Nonlinear Regression*, training requires data where the numeric target value is known.

Functions

You can make the following settings in a model for the *Scoring* method*:*

You use the **model parameters** to specify the function type.

You use the **model fields** to specify the attributes you wish to use in each case. You specify for each model field whether the corresponding attribute takes discrete or continuous values or whether it is a key field. You also specify the target value.

Some more settings depend on the function type:

- In the case of *weighted score tables*, you also need to specify the range of values for the individual model fields and enter a weighting.

- For both the regression analysis functions, you can enter or automatically determine an interval for continuous model fields, and you can specify for discrete model fields the attribute values to be considered or simply select those occurring most frequently. Furthermore, several options are open to you for dealing with outliers and missing values.

When using historic data to train the function types *Linear Regression* and *Nonlinear Regression*, the system determines the effect of the model field values on the numeric target value. When you train the function type *Weighted Score Tables*, the function is determined directly, independently of historic data. With all three function types, the system applies the calculated function on the data from the training source. You can display the result in graphical format and export the visualization data into an Excel workbook.

See Also

Creating, Changing and Activating a Model [Page 408]

Creating Analysis Process for Training [Page 413]

Analysis Process for Executing the Prediction [Page 415]

Transformation for Weighted Score Tables [Page 417]

Analysis Process Designer [Page 362]

4.2.3.2.1 Regression Analysis

In *Regression Analysis,* you can use *Linear Regression* and *Nonlinear Regression* to automatically define valuation functions and thereby determine numeric target values. If you wish to generate the valuation functions, you need to train the analysis process using historic data.

After you have determined the valuation functions either by defining them directly or by training them on the basis of historic data, you can then apply them to other datasets as part of a prediction.

A beverage outlet wants to attract the younger end of the market by introducing a product from a higher price category into its product range.

Linear Regression
The beverage outlet wants to estimate its revenue potential in the drinks market. Assuming that the revenue from the sale of drinks has a linear dependency on income and household size, a *linear regression* is performed on data where the revenue is already known. Training determines the influence that income, household size, and region have on the revenue from the sale of drinks. The function that is trained using this data can now be applied to prospects in order to calculate the potential revenue from such customers in this market.

Nonlinear Regression

- The beverage outlet also wants to investigate the relevance of the attribute "age" for its potential revenue in the drinks market. Revenue here is unlikely to have a linear dependency on age. Nonlinear dependencies can also be analyzed by using *nonlinear regression*.

- A newspaper publisher wants to identify customers with a high propensity to churn (in this instance, a strong likelihood of canceling their newspaper subscriptions). The publisher's customer database contains details relating to age, income, household size, academic qualifications, length of the subscription, and region, as well as a field for canceled subscriptions. If a customer canceled their subscription in the past quarter, this field contains the value 1, otherwise it contains the value 0. The function *Nonlinear Regression* is then trained using this data. The result of training should show the relationship between the different customer attributes and the canceled subscription field. The trained function then generates a value for each customer in the customer database, and this value can be used to reflect that customer's propensity to churn.

See Also

Special Settings for Regression Analysis [Page 398]

4.2.3.2.1.1 Special Settings for Regression Analysis

Linear Regression

With the function type *Linear Regression*, the system trains the scoring function using data with known target values. You need to set the value type of the target value as continuous. At least one of the other model fields must also be continuous. The system defines a separate linear function for each combination of values in discrete model fields that occur in the training data. If alongside the continuous fields the model also contains, for example, the discrete fields "Gender" and "Region", which take the values "m"/"f" or "North"/"Center"/"South" respectively in the training data, then a separate linear function is defined for each combination - (m, North), (m, Center), (m, South), (f, North), (f, Center), (f, South) - for which training data exists. To exclude combinations with a minimal amount of data records, you can use the model parameter *Minimum Number of Records*. If you set this parameter to 100, for example, and there are 200 training records with (m, North) and 50 with (m, Center), then a linear regression is only performed for (m, North) but not for (m, Center), since the later falls below the minimum number. All data records with (m, Center) thus fall outside of the trained function's definition area (for the domain). If you select the indicator *Skip input outside of trained domain*, then no score value is calculated for such data records. If you do not select this indicator, the default score value is assigned to these data records.

With the parameters of the model fields, you can specify for discrete fields whether all values, just special values, or just the most frequent values should be considered. For continuous fields, you can explicitly specify both limits of a value range or have them specified automatically by choosing the option *Complete Data Range*. With the automatic option, the limits are determined by rounding off the maximum and minimum values of the field in the training data. When the function is applied to other data, values occurring outside of this range are then treated as outliers.

Nonlinear Regression

With the function type *Nonlinear Regression* (using multilinear splines), the system defines a separate multilinear spline function for each combination of discrete model field values occurring in the training data.

As with linear regression, you need to specify the value type of the target value and of at least one other model field as continuous. To prevent the function from overadjusting areas of the training data with a low density of data, you can use the model parameter *Smoothing Factor*. The greater the smoothing factor, the more the function will smooth out areas with a low density of data.

As with linear regression, you can specify for discrete model fields whether the system should consider all values, just special values, or just the most frequent values. You have to split the value ranges of continuous model fields into intervals. As with linear regression, you can have both of the *outer* interval limits determined automatically or you can enter them explicitly. You can then specify the desired number of intervals of equal size within those outer limits. Alternatively, you can set these intervals within the outer limits explicitly.

The greater the number of intervals, the greater the extent to which the function can adjust itself to accommodate nonlinear data. At the same time, more intervals mean an increase in processing effort. The number of model fields increases the complexity of the calculation to a greater degree than with linear regression. For this reason, narrower limits are set when nonlinear regression is used.

4.2.3.2.2 Weighted Score Tables

You can use *weighted score tables* to define your own valuation function by entering the valuations for the individual model fields and then using these valuations to create a weighted total.

Weighted Score Table

A beverage outlet wants to attract the younger end of the market by introducing a product from a higher price category into its product range. To determine potential customers, customer data (including attributes like age, income, and drink expenditure) is valuated directly using the *weighted score tables*. The age group 10-19 is valuated with 15, 20-29 with 10, 30-39 with 5, and so forth. The customer incomes are valuated continuously by taking the respective figure as the value in each case (that is, an income of 50,000 is valuated with 50,000). A weighting is then assigned to each attribute: 2 for age and 0.0001 for income. Thus, the score for a 25-year-old customer with an income of 40,000 is calculated as follows: (2 x 10) + (0.0001 x 40,000) = 24.

See Also

Special Settings for Weighted Score Tables [Page 399]

4.2.3.2.2.1 Special Settings for Weighted Score Tables

When you use the function type *Weighted Score Tables*, a model field weighting needs to be specified for each model field. Furthermore, a function in the model field needs to be specified so that a partial weighting can be assigned to each value of the field. The weighting of the model fields then determines the share of partial weightings that the score value has. This is calculated as follows: Score value (field1, field2, ...) = weighting1 x partial weighting1 (field1) + weighting2 x partial weighting2 (field2) + ...

How the partial weightings are determined depends on the type of model field:

- With **discrete** model fields, you specify partial weightings for individual values of the field. You can determine an additional partial weighting that is assigned to the remaining values. This partial weighting is only applied if you selected the *Treat as separate instance* indicator during outlier handling. For more detailed information about outlier handling, see *Settings for All Function Types* below.

- With **continuous** model fields, you enter partial weightings for individual threshold values. You also have to specify the function process for the partial weightings **between** the threshold values. You have the following options:
  - One option is for the function process to be constant between the threshold values, in which case the partial weighting of the left or right threshold value (depending on the setting) is taken. In this case, the function process of the partial weighting is piecewise constant.
  - Alternatively, the partial weighting is calculated continuously by linear interpolation.
  - Furthermore, you also have the option of setting the partial weighting as equal to the value of the field.

  In each case, you need to enter at least **two** threshold values because the value range applied in outlier handling (see below) lies outside of these values (that is, above the highest threshold value and below the lowest threshold value).

The score value to be defined is dependent on the discrete model field *Customer Category* and the continuous model field *Income*. The weighting of these two model fields should be 3 and 1 respectively. In the model field *Customer Category*, the data to be processed takes the values gold, silver, bronze, copper, and iron. The following partial weightings could then be specified:

Value	**Partial Weighting**
Gold	10
Silver	6
Bronze	4

The partial weighting 2 can be assigned to the remaining values. For the model field *Income*, the threshold values and corresponding partial weightings could be assigned as follows:

Threshold Value	**Partial Weighting**
0	0
10 000	10
25 000	20
50 000	30

The partial weightings function should be piecewise constant and take the partial weighting of the left threshold value in the interval between two threshold values. In this way, the score value (silver, 40 000) = 3 x 6 + 1 x 20 = 38 is obtained. If the partial weightings function for the income should be continuous instead of piecewise constant, then it produces the score value (silver, 40 000) = 3 x 6 + 1 x 26 = 44.

If the *Treat as separate instance* option was selected in outlier handling for the model field *Customer Category*, then the function produces the score value (iron, 10 000) = 3 x 2 + 1 x 10 = 16.

If the *Constant extrapolation* option was chosen in outlier handling for the model field *Income*, then the function produces the score value (silver, 60 000) = 3 x 6 + 1 x 30 = 48. If the *Extrapolation* option is chosen, this produces the score value (silver, 60 000) = 3 x 6 + 1 x 34 = 52.

4.2.3.2.3 Settings for All Function Types

In the case of each of the three function types, the model field parameters offer options for controlling how outliers and missing values are treated.

For discrete model fields, values not belonging to those entered explicitly or to the most frequent values are considered **outliers**. For continuous model fields, outliers are those values falling outside of the outer limits that were either entered explicitly or determined automatically when the value ranges were defined. You can set one of the following system reactions for dealing with occurrences of a data record containing an outlier:

1. Cancel processing
2. Ignore the data record
3. Assign the default score value.

With continuous model fields, you can specify that outliers are extrapolated. With discrete model fields, you can specify that all outliers are treated like a single value (as a remainder).

You first need to identify **missing values** before the system can treat them. If a model field, such as *Occupational Group*, takes a two-digit number code and code 99 stands for unknown occupational group, then you can enter 99 as the missing value to treat such values separately. You can set one of the following system reactions for dealing with occurrences of a data record containing a value defined in this way:

1. Cancel processing
2. Ignore the data record
3. Assign the default score value.

You can also set a substitute value.

4.2.3.3 ABC Classification

Use

The *ABC Classification* is a frequently used analytical method to classify objects (Customers, Products or Employees) based on a particular measure (Revenue or Profit). For example, you can classify your customers into three classes A, B and C according to the sales revenue they generate.

ABC classification allows you to classify your data based on specified classification rules. The data to be classified is generated by a query in the SAP BW. The classification rules refer to a single key figure value in your data and implicitly specify which absolute or relative key figure values map to which classes.

Integration

The data to be classified using ABC classification can be taken from any system, if that system can extract data into SAP BW. You can apply the same ABC classification to any data that has been extracted into SAP BW.

Prerequisites

You require a query in SAP BW delivering 'suitable' objects that can be classified by ABC classification. In particular, the query must contain the key figure to which the classification rules can be applied.

Only numerical key figure fields can be used for the classification type *Cumulated Percentage of Classification Criterion,* as summation of non-numerical fields is not possible.

Features

You specify the following for the ABC classification model:

- The **characteristic** for which the classification is to be performed. This entails specifying the characteristic values to be classified (such as *Customer*).
- The **key figure** that is to form the basis for classifying the characteristic values (such as *Profit made from that Customer*)
- The **attribute** of the characteristic that should receive the result (*the ABC class*)
- The **query** for determining the data (such as *Profitability Data from Customer*)
- The **threshold values** for the individual ABC classes. For example, all customers generating a profit of 0 to 20,000 belong to class C, those generating a profit between 20,001 and 80,000 to class B, and those generating more than 80,001 to class A.

See Also

Creating, Changing and Activating a Model [Page 408]

Transformation for ABC Classification [Page 418]

Classification Rules [Page 418]

Model Settings [Page 402]

4.2.3.3.1 Model Settings

You use the **model fields** to specify those fields that are part of the ABC classification. You must specify the following three model field values.

- *Classified Object*. You must specify which characteristic values to be classified (for example, Info Object *0BPARTNER* for Business Partners to be classified)
- *Classification Criterion.* You must specify the key figure field, which is used as the basis for the classification (for example, Info Object *0NETVALORD* for the Sales Revenue generated by the customer)
- *Class Info Object:* You must specify the Info Object used as attribute for the classification result (for example, Info Object *0ABC_CLASS* representing the possible classification values 'A', 'B', and 'C').

You use the **model parameters** to specify the type of Classification Rules [Page 403] and the classification rules with their mappings. The classification rules consist of a threshold and the corresponding class as shown in the table below.

Threshold	**Class**
Thres_1	C_1
Thres_2	C_2
Thres_3	C_3
…	…

A threshold of *Thres_n* in the table specifies that the left-open value interval (*Thres_n-1, Thres_n*] maps to class *C_n*. *Thres_0* denotes infinite values.

Consequently, the first row in the table specifies that all records with values less than or equal to *Thres_1* will be mapped to class *C_1.* The second row specifies that all values less than or equal to *Thres_2* but higher than *Thres_1* will be mapped to *C_2*, and so on.

The referenced value depends on the specified type of classification rules.

4.2.3.3.2 Classification Rules

ABC classification provides the following four types of classification rules:

- ***Absolute Values of Classification Criterion*:** Each record in the dataset is classified based on the absolute value of its referred key figure value.

  You need to classify your *Customers* according to the *Sales Revenue* they generated last year. You require the customers with sales revenue of:

  - At least $ 80,000 and more to be classified as **A**
  - Less than $ 80,000 but more than $ 20,000 to be classified as **B**, and
  - Less than $ 20,000 to be classified as **C**

  This classification can be done using classification rule type *Absolute Values of Classification Criterion* containing the corresponding mappings. Each mapping specifies which key figure value interval (*Sales Revenue* in this case) maps to which classification value.

- ***Cumulated Percentage of Classification Criterion*:** All data records are ranked by their key figure value and starting with the record with maximum key figure value. In the order of descending key figure values, each record is classified based on the ratio:

  Cumulated Key Figure Value / Total of Key Figure Values.

This is in terms of percentage. The *Cumulated Key Figure Value* denotes the sum of all key figure values up to and including the record to be classified. The Total of key figure values denotes the sum of all key figure values.

You want the *most profitable* customers with highest sales revenue and together generating sales revenue of at most 10 percent of the total sales revenue to be classified as **A**.

The *medium profit* customers with next highest sales revenue and together generating the next 50 percent of the total sales revenue last year are to be classified as **B**.

The remaining *low profit* customers with lowest sales revenue are to be classified as **C**.

You can do this classification by using classification rule type *Cumulated Percentage of Classification Criterion.*

A single record hitting the border line, that is, a record falling in between classes is always assigned to the class with the higher threshold. The following example illustrates this.

Following is the table of classification rules for an ABC classification of type *Cumulative Percentage of Classification Criterion.*

Threshold	**Class**
10	A
40	B
100	C

The first line specifies that those customers with highest revenue and together leading to 10 percent of total revenue map to class *A*. If we assume that the following three customers have to be classified according to these rules:

Customer	**Revenue**
Miller	10.000 $
Smith	5.000 $
Green	5.000 $

In this case, the result of this classification is that all the three customers are classified to class C. This is because Miller contributes 50 percent to total revenue, Miller and Smith together contribute 75 percent of total revenue and all three customers together contribute to 100 percent of total revenue. All percentages of cumulated revenues are higher than 10 percent and even higher than 40 percent and so they are all classified to class *C*.

- ***Cumulated Percentage of Classified Object***: All data records are ranked by their key figure value and starting with the top-ranked record with maximum key figure value. In the order of descending key figure values, each record is classified based on the ratio

  Ranking Position / Number of Records

  This is in terms of percentage. The *Ranking Position* denotes position in the key-figure-based ranking while the top-ranked has ranking position 1; the secondly best-ranked record has ranking position 2 and so on. The *Number of Records* denotes the total number of records to be classified.

  As in the example used in type *Cumulated Percentage of Classification Criterion*, you can classify your customers into top 10% as class **A,** next 50% as **B** and the remaining as **C** profitability customers.

- ***Absolute Values of Classified Object***: All data records are ranked by their key figure value, and starting with the top-ranked record with maximum key figure value in the descending order. Each record is classified based on the absolute ranking position.

  In this case, the top 10 records, depicting the top 10 most profitable customers with highest sales revenue will be classified as **A**. The next 2000 records depicting the medium profit customers will be under class **B** and the remaining low profit customers generating the least sales revenue would be classified as **C**.

4.2.3.4 Clustering

Use

Clustering allows you to segment data automatically into clusters. In a subordinate dataset, the system groups together associated data by forging formerly unknown links. This entails determining the criteria for clustering as well as the mappings between datasets.

You execute clustering by training a model on the basis of historic data. You can use a prediction to apply the same segmentation to another dataset.

The customer data for a fruit juice outlet contains attributes such as gender, age, income, region, occupation, and product bought most. During clustering, the system determines which combinations of attributes frequently occur together and uses this information to build clusters, that is to say, customer segments. A customer segment could consist of male customers aged between 30 and 40, with high incomes, and whose most frequent purchase is orange juice. Another customer segment could represent female customers aged between 20 and 40, without occupation, and whose most frequent purchase is apple juice.

Integration

The data that you use to train the model can be taken from any other system, provided that the system can extract data into SAP BW. Likewise, you can apply the same segmentation to any data that has been extracted into SAP BW.

Prerequisites

The queries available in SAP BW allow you to access data for which the statements are known and which you can use to find out similar statements about other data.

Features

You can make the following settings in a model for the *Clustering* method*:*

You use the **model fields** to specify which characteristic is to be considered with which attributes (such as the characteristic *Customer* with the attributes *Occupation, Gender, Age*, and so on). You can specify in the field parameters different weightings for the individual attributes. The system then establishes formerly unknown associations between the attribute values.

You can use the **model parameters** to specify, for example, how many clusters the system should create during training. By specifying conditions for interrupting the segmentation, you enhance the quality and performance of the segmentation.

During training, the system determines not only the clusters but also which cluster each characteristic (such as a customer) belongs to and what distance separates the clusters. You can display the result in graphical format and export it into an Excel workbook.

See Also

Creating, Changing and Activating a Model [Page 408]

Creating Analysis Process for Training [Page 413]

Analysis Process for Executing the Prediction [Page 415]

4.2.3.5 Association Analysis

Use

The purpose of association analysis is to find patterns in particular in business processes and to formulate suitable rules, of the sort "If a customer buys product A, that customer also buys products B and C".

If a customer buys mozzarella at the supermarket, that customer also buys tomatoes and basil.

Association analysis also helps you to identify cross-selling opportunities, for example. You can use the rules resulting from the analysis to place associated products together in a catalog, in the supermarket, or in the Web shop, or apply them when targeting a marketing campaign for product C at customers who have already purchased product A.

Association analysis determines these rules by using historic data to train the model. You can display and export the determined association rules.

Integration

The data that you use to train the model can be taken from any other system, provided that the system can extract data into SAP BW.

Prerequisites

The queries available in SAP BW allow you to access the necessary data with which to determine association rules for the transactions concerned.

Features

You can make the following settings in a model for the *Association Analysis* method*:*

You use the **model fields** to specify which transactions and which items should be considered.

You can use the **model parameters**, for example, to specify what percentage of the transactions with the leading item (product A) should also contain the dependent item (product B or C) to define a valid rule (confidence [External]). Coupled with this, you can use the parameters Lift [External] and Support [External] to improve the quality of the determined rules. Moreover, you can specify the number of leading and dependent items.

You can display the association rules determined in training and export them to an Excel workbook. If you have determined product association rules, you can export these to an SAP Customer Relationship Management System (SAP CRM) for use as product proposals for cross-selling.

See Also

Creating, Changing, and Activating a Model [Page 408]

Creating Analysis Process for Training [Page 413]

Maintaining DataSource for Association Rules [Page 412]

4.2.3.6 Activities in the Data Mining Workbench

You can perform the following activities in the Data Mining Workbench:

Create, Change and Activate a Model [Page 408]

Reset or Delete a Model [Page 410]

Execute What-If Analysis [Page 411]

Use the Where Used List [Page 411]

Export Model as PMML [Page 412]

Maintain DataSource for Association Analysis [Page 412]

4.2.3.6.1 Creating, Changing, and Activating a Model

Use

You create a model for a data mining method so that you can apply the method according to your business requirements. You use model fields in a model to specify what is to be predicted and which data should form the basis of the prediction.

You can create a data mining model using the *Data Mining Workbench* or the *Analysis Process Designer* (APD). Once you have created and saved the model to meet your requirements, you can activate it.

Prerequisites

You must have been assigned to the role *Customer Behavior Analysis* (SAP_BW_CUSTOMER_BEHAVIOR) and you must have chosen *Customer Behavior Analysis* → *Customer Behavior Modeling* in the user menu.

Creating a Model in the Data Mining Workbench

1. Position the cursor on a data mining method and use the right-hand mouse to choose *Create* in the context menu.
2. In the step *Create Model*, enter a name and a description for the model. The method name for which you are creating a model is displayed. You have three options for model field selection:

- To create the model fields manually, select the *Manual* option.
- If you want to create a model that is similar to an existing model created previously, you can copy it choosing the *Use Model as Template* option. You can make minor changes to the copied version manually to suit your requirements.
- To create a model from a query, choose *Model Field Selection* and select the query which you want use as a source for model fields.

    Selecting a query at this point will assist you in creating the model. It is therefore **recommended** to enter the same query that you would like to use subsequently, while training the model. However, this is **not essential**. You can also use any other query as a template for your model.

  The InfoObjects contained in the selected query are available in the next step as model fields.

3. In the step *Select InfoObjects*, select from the query those InfoObjects that you would like to use as model fields.

    If you would like to use other fields from the query as calculated or restricted key figures in your model, you need to include an SAP BW dummy InfoObject as a model field for each one. You can then assign the corresponding field to this model field in the *Change* mode.

4. In the step *Edit Model Fields*, specify the attributes for each field.

- The description you give the model field does not necessarily have to be identical with that of the InfoObject.
- The system automatically copies the attributes *Data Type* and *Length* from InfoObject (these cannot be modified).
- The *value types* valid for a model field are dependent on the method that you are creating the model for and on the data type of the model field.

  The value type specified for a model field determines which entries can be made as *Field Parameters* and *Field Values*.

  > The attributes for a model field that are listed below do not apply in the data mining method *Association Analysis*. No prediction is involved with this method. Instead, the association rules are determined by training and form the result. Consequently, the settings for the field parameters and field values do not apply.

- Set the *Prediction Variable* indicator for the model field for which the subsequent prediction is to be made. Select as a prediction variable that model field for which you wish to gain more information (via the model).

  With the data mining method *Clustering*, the cluster is always the prediction variable. Consequently, you cannot specify a prediction variable for this method.

- The *field parameters* are dependent on the value type of the model field and on the data mining method.

  You cannot select any parameters for model fields where the value type KEY has been set.

- Under *Field* Values, you can specify how the system should interpret specific values that can be taken by a model field but have no bearing on the result.

5. In the *Model Parameters* step, enter the parameters that are valid for the entire model. The model parameters are dependent on the data mining method.
6. *Save* the model.

Result

You have performed all necessary steps for the creation of a model. The created model appears in the tree beneath the relevant method.

Changing the Model

You can make changes to the model that you have created.

1. Position the cursor on a model that you wish to change and use the right-hand mouse to choose *Change* in the context menu.
2. In the *Model Fields* tab page, make your changes to the model fields. You can change the attributes for the model fields or add more model fields.
3. In the *Model Parameters* tab page, make your changes to the model parameters.
4. Save your changes.

Activating the Model

Once a model meets your requirements, you can activate it. The active model is then used for creating other versions. This means that, when you change a model that has been activated, the active version remains unchanged and the changes are saved in a *Revised* version. The active version is only overwritten when you activate the modified version.

You can only train or valuate a model or use it for the prediction if the model has been activated.

If a model has a modified version, the model name in the tree is marked in blue.

To activate a model, proceed as follows:

1. Position the cursor on a model that you wish to activate and use the right-hand mouse to choose *Activate* in the context menu.

   The version displayed under *Model Information* is changed to *Active*.

2. Make any necessary changes to the model and save your changes.

   The version displayed under *Model Information* is changed to *Revised*.

3. To navigate between the active and modified versions, place the cursor in the model and choose .

4.2.3.6.2 Resetting or Deleting a Model

Use

If a model contains errors or has become obsolete, you can delete it. This deletes the definition of the model in the process but not the queries that you took as the basis for the definition.

If you do not want to delete the definition of the model and only want to delete the result of training, valuation, or prediction, you can reset the model.

Resetting the Model

To reset a model, choose *Reset* from the respective context menu.

When a model is reset, all results are deleted. The status of the model is adjusted accordingly.

If you reset the model for training, the training result is deleted. The system resets the status of the model changing it from *Trained* to *New*.

Deleting the Model

To delete a model, choose *Delete* from the respective context menu. When a model is deleted, the model, all results are deleted.

4.2.3.6.3 Executing What If Analysis

Use

You can use the *What If Analysis* to predict prospective customer behavior based on specific attributes of that customer. This is applicable in situations where online predictions are required. For example, when a customer applies for a loan, you can predict if the customer is creditworthy or not, based on certain attributes such as income, profession or transaction history.

What If Analysis can be used for the data mining methods *Clustering* and *Decision Tree Analysis.*

Prerequisites

- The model for the analysis needs to be fully created and activated (see Creating, Changing, and Activating a Model [Page 408])
- The model must be trained (see Creating Analysis Process for Training [Page 413])

Procedure

1. Position the cursor on the relevant data mining model, use the right-hand mouse to choose *What If Analysis* in the context menu. Alternatively, you can double-click on the relevant model and choose the *What If* option from the tool bar.
2. This invokes the *What If Analysis* screen.
3. Enter the model values for the prediction fields.

   💡

   The model field values for prediction must be selected from the given set of values. If you enter a value that is not present in the training data, a warning message will be generated and this value will not be considered for prediction.
4. Choose *Predict* to view the results.

Result

In a decision tree, the prediction result is displayed as a histogram in both tabular and graphical form. In the histogram, all possible class outcomes are shown along with their probability and support in the training data. For example, in the case of a customer applying for a loan, the what-if prediction shows the probability of the customer getting accepted or rejected.

In clustering, there is no graphical representation of the *What If Analysis*. The result contains the first and the second best cluster under which the predicted case falls.

4.2.3.6.4 Using the Where Used List

Procedure

1. Select the model and use the right-hand mouse to choose Where Used List. This displays the details of the analysis processes in which the model is used.

2. Select the analysis process to navigate to the *Analysis Process Designer* and make any changes, if required.

4.2.3.6.5 Exporting Model as PMML

Use

You can export the decision tree and clustering results from data mining models into a local file in the PMML format. *Predictive Model Markup Language* (PMML) is an XML-based language that enables applications to define statistical and data mining models. It also enables sharing of models between PMML compliant applications. For more details on PMML, see Predictive Model Markup Language.

Prerequisites

To generate PMML documents, the data mining model must be *Trained.*

Procedure

To display the output in PMML format, choose the *Display as PMML* option in the decision tree or clustering results screen.

To export the PMML results to a local file, select the model and choose the *Export as PMML* option from the context menu.

4.2.3.6.6 Maintaining DataSource for Association Rules

Use

You can use the extraction process to load the results from mining models using DataSources. The generation of DataSources is based on the InfoObjects that are used as model keys. Hence, any two models having the same keys will share the same DataSource.

- The extraction process takes into account the results that have already been uploaded into SAP BW. Using the SAP BW staging process, you can upload the extracted data obtained from association rules.

Prerequisites

- The data mining model must activated
- Only DataSources for transaction data (TRAN) are generated

Procedure

1. In the context menu for the association analysis model, choose *Maintain DataSource*
2. If you have already maintained a DataSource, then:
   a. Details of the associated DataSource are displayed

   b. You will get a message stating that the model source is already associated to a DatSource

3. If no DataSource is maintained for the mining source, then:

   a. The system prompts you to specify the validity options for extracted data. Choose the *Validity of Extracted Results* to get the valid date interval. This indicates the time period up to which the uploaded data will be valid in the InfoCube

   b. Select the relevant option and choose *Create*

   c. This displays the details of the associate DataSource

For further details on data extraction, refer to:

Association Analysis [Page 406]

Data Extraction from SAP Source Systems [External]

Assigning DataSources to InfoSources and Fields to InfoObjects [External]

Creating Update Rules for Data Targets [External]

Scheduling InfoPackages

[External]

4.2.3.7 Activities in the Analysis Process Designer (APD)

You can perform the following data mining activities in the Analysis Process Designer:

Create an Analysis Process for Training [Page 413]

Create Analysis Process for Data Mining Transformations [Page 415]

Load Mining Results into the SAP BW [Page 419]
Load Mining Results into the SAP CRM [Page 419]

4.2.3.7.1 Creating Analysis Process for Training

Use

You must create an analysis process for a data mining method to train a data mining model. The purpose of training a model using historic data is to allow the model to learn from the historic data. The training result can then be used for a prediction or in the operational system. You can create a training process for the data mining methods:

Decision Tree

Clustering

Regression Analysis

Association Analysis.

Prerequisites

To train a data mining model:

- The model must be fully created and activated

- To execute an analysis process for training, the analysis process must be activated and model status must not be *Trained*

Procedure

To create and execute an analysis process for training:

4. Choose *Create*
5. Select an application from the drop-down menu and select *Okay.* Your analysis process will be assigned to the appropriate folder on the left side of the screen
6. Enter the description for the analysis process
7. Drag a data source into the work area and enter the settings in the dialog box that appears
8. For the data target, drag the icon for the relevant data mining method in the work area
9. Connect the two nodes with the mouse
10. Select the data mining method and choose *Properties* from the context menu. Alternatively, you can double click on data mining node to make the settings in the dialog box that appears
11. Enter the description of the model. The relevant data mining method would be displayed
12. In the Model field:
    a. If you have already created the model in the data mining workbench, you can either enter the name of the model in this field or choose F4 to select the model from the list of existing models
    b. If you want to create a new model, enter the name of the model and choose *Create.* Enter the description of the model and make the model settings. *Save* the changes and *Activate* the model. Choose *Environment* → *Analysis Process Designer* to return from the Data Mining Workbench. To create a model in the data mining workbench, see Creating, changing and Activating Model [Page 408]
13. To make an explicit field assignment, double click on the data flow arrow that connects the nodes
14. Save the changes
15. To activate the analysis process, choose *Analysis Process* → *Activate* or use the *Activate* option from the toolbar
16. To execute the training process, choose Analysis Process → *Execute* use the *Execute* option from the toolbar. The data is written to the data target and a log is displayed
17. Using the context menu on the data target, select *Intermediate Results* → *Calculate-Directly* or *Calculate- Scheduling.* This is how you can display data that was posted when executing into the data target

Result

- The data mining model acquires the status *Trained.* To view the training results, in the context menu of data target, choose *Data Mining Model* → *View Model Results.*

Now you can execute a prediction or - during association analysis - export the association rules determined during training to SAP CRM.

See Also

Analysis Process Designer [Page 362]

Data Sources for an Analysis Process [Page 366]

Data Targets for an Analysis Process [Page 381]

4.2.3.7.2 Analysis Process for Data Mining Transformations

You must create an analysis process for any data transformation using data mining models. For performing prediction and evaluation (in case of decision trees), you must create an analysis process for executing the prediction. For more information, see Analysis Process for Executing the Prediction [Page 415] and Evaluating a Decision Tree Model [Page 416].

For more information on transformation of ABC Classification and Weighted Table Scoring, see Transformation for ABC Classification [Page 418] and Transformation for Weighted Table Scoring [Page 417].

See Also

Transformations for an Analysis Process [Page 371]

4.2.3.7.2.1 Analysis Process for Executing the Prediction

Use

A model that you trained using historic data from a source can now be applied to a different set of data. By doing so, a prediction is made for the model field that you selected as the prediction variable. You can execute the prediction process for the data mining methods decision tree, clustering and regression analysis. For *clustering*, the prediction output would be the best three clusters. The predicted output, in case of a decision tree, is the best three predicted classes and the corresponding probability of the prediction. For the scoring method, *regression analysis*, the predicted output is the score.

You cannot execute a prediction for *association analysis* because training is already the result of this data mining method. You can export the association rules determined in training to SAP CRM (see the section Association Analysis [Page 406]).

Prerequisites

- The model must be trained
- To execute the prediction, the analysis process must be activated

Procedure

To create an analysis process for prediction:

18. Choose *Create*

19. Select an application from the drop-down menu and select *Okay.* Your analysis process will be assigned to the appropriate folder on the left side of the screen
20. Enter the description for the analysis process
21. Drag a data source into the work area and enter the detail settings in the dialog box that appears
22. Drag the relevant prediction icon, that is, source for transformation, in the work area
23. Connect the two nodes with the mouse
24. Select the data mining method and choose *Properties* from the context menu. Alternatively, you can double click on data mining node to make the settings in the dialog box that appears
    1. Specify the prediction parameters:
        a. Specify the model that you want to use for that data mining method
        b. Specify the mapping between model fields and the data input fields
        c. Select the prediction output fields
    2. Save the analysis process and activate it
    3. Execute the analysis process
    4. To display the prediction results, choose *Display Data* from the context menu
    5. To display the summary of the prediction results, choose *Calculation Summary*
    6. To display the stored results, choose *Intermediate Results*

For more details, see Transformations for an Analysis Process [Page 371].

Result

The prediction result is determined by the system and then displayed in graphic form.

You can load the determined values into the master data in SAP BW (see Loading Mining Results into SAP BW [Page 419])

4.2.3.7.2.1.1 Evaluating a Decision Tree Model

Use

You can evaluate the results of a decision tree model. The purpose of valuation is to verify the validity or accuracy of the training result using historic data. You can do this by using the tree to classify a separate set of data whose outcomes are already known. If you compare the predicted outcome with the known outcome, you can easily discover the number of correct predictions and ones that were not correctly predicted. This information can then be displayed in the form of a matrix, called the *Error Matrix.*You use this matrix to know which outcome values the tree predicts well and the values that the tree doesn’t predict properly.

Prerequisites

- The model must be fully created and activated
- The analysis process must be activated

Procedure

To evaluate a model, you must follow the steps described in Executing the Prediction. [Page 415] The only difference is that you need to check the option *Run in Evaluation Mode.*

4.2.3.7.2.2 Transformation for Weighted Score Tables

Use

You must create an analysis process for transformation to perform scoring calculation using weighted score tables.

Prerequisites

- The model must be activated
- To execute the transformation, the analysis process must be activated

Procedure

To create an analysis process for prediction:

25. Choose *Create*
26. Select an application from the drop-down menu and select *Okay.* Your analysis process will be assigned to the appropriate folder on the left side of the screen
27. Enter the description for the analysis process
28. Drag a data source into the work area and enter the detail settings in the dialog box that appears
29. Drag the weighted table scoring icon, that is, source for transformation, in the work area
30. Connect the two nodes with the mouse
31. Select the data mining method and choose *Properties* from the context menu. Alternatively, you can double click on this node to make the settings in the dialog box that appears
    1. In the prediction parameters screen:
        1. Enter the description and specify the model that you want to use for the Weighted Score Table. You can also create a new model based on the input data, using the *Create* option
        2. Specify the name and description of the field for the prediction score in the Prediction Output
        3. Specify the mapping between the model fields and data input fields in Prediction Input. For the mapping, only non-key content type are used
    2. Save the analysis process and activate it
    3. Execute the analysis process
    4. To display the transformation results, choose *Display Data* from the context menu
    5. To display the statistics, choose *Display Basic Statistics*
    6. To display the summary of results, choose *Calculation Summary*
    7. To display the stored results, choose *Intermediate Results*

For more details, see Transformations for an Analysis Process [Page 371].

Result

The prediction score is the result from weighted score tables. As a result of scoring prediction, the input data gets enriched by a column containing this predicted score. You can rename the generated data field name and descriptions that stores the score output.

For more details, see Weighted Score Tables [Page 399]

4.2.3.7.2.3 Transformation for ABC Classification

Use

You must create an analysis process for transformation to perform classification using the ABC classification model.

Prerequisites

- The model must be activated
- To execute the transformation, the analysis process must be activated

Procedure

In order to create and execute an analysis process with transformation, proceed as follows:

1. Choose *Create*
2. Select an application from the drop-down menu and select *Okay.* Your analysis process will be assigned to the appropriate folder on the left side of the screen
3. Enter the description for the analysis process
4. Drag a data source into the work area and enter the detail settings in the dialog box that appears.
5. Drag the ABC classification icon in the work area
6. Connect the two nodes with the mouse
7. Select the data mining method and choose *Properties* from the context menu. Alternatively, you can double click on this node to make the settings in the dialog box that appears
   1. In the prediction parameters screen:
      1. Enter the description and choose the data mining model. Specify the model that you want to use for ABC classification. You can also create a new model based on the input data, using the *Create* option
      2. Specify the ABC output fields
      3. Specify the mapping between the model fields and data input fields in Prediction Input. For the mapping, only non-key content type and non-predictable model fields are used
   2. Save the analysis process and activate it
   3. Execute the analysis process
   4. To display the transformation results, choose *Display Data* from the context menu

5. To display the statistics, choose *Display Basic Statistics*
6. To display the summary of results, choose *Calculation Summary*
7. To display the stored results, choose *Intermediate Results*

For more details, see Transformations for an Analysis Process [Page 371]

Result

The transformation output is the ABC Class. For each selected output field, the input data gets enriched by one column. You can rename the generated data field name and description that stores the ABC outputs.

For additional information, see ABC Classification. [Page 401]

4.2.3.8 Loading Mining Results into SAP BW

Use

You can load the values that the system determined during transformation or prediction into the SAP BW objects using the Analysis Process Designer. The values determined for the prediction as well as other values (depending on the data mining method) are updated. For example, with the *Decision Trees* method, you can update the *Predicted Value* and *Probability* of the prediction value. For more details on transformation and prediction output for the various data mining methods, see:

Analysis Process for Executing the Prediction. [Page 415]

Transformation for ABC Classification [Page 418]

Transformation for Weighted Table Scoring [Page 417]

Procedure

For information on how to upload results into SAP BW, see:

Data Targets for an Analysis Process [Page 381]

Master Data [Page 382]

ODS Object [Page 383]

4.2.3.9 Loading Mining Results into SAP CRM

Use

You can transfer the values determined in the SAP BW to SAP CRM and place these values in attributes – essentially business partner attributes in the Analysis Process Designer.

Procedure

For more information, see:

CRM Attribute

[Page 385]Data Targets for SAP CRM [Page 386]

SAP Customer Relationship Management → *Application Services* →*Analytical Methods* →*Optimize Relationships* →*Transferring Attribute Data from SAP BW* in the SAP Library.

4.3 Developing Persistency

Purpose

Integrated Planning and BW-BPS: Data Basis

To store planning data persistently in the BI system, use real-time-enabled InfoCubes.

For more information, see Real-Time InfoCubes [Page 420].

For BW-BPS only:

Before modeling the InfoCube, you need to consider the strategy you want to use in your data model.

For more information, see Data Modeling: Key Figure Model and Account Model [Page 346].

Analysis Process Design

The staged or transformed data of an analysis process is saved in a data target.

For more information, see Data Targets for an Analysis Process [Page 381].

4.3.1 Real-Time InfoCubes

Definition

Real-time InfoCubes differ from Standard InfoCubes in their ability to support parallel write accesses. Standard InfoCubes are technically optimized for read accesses to the detriment of write accesses.

Use

Real-time InfoCubes are used in connection with the entry of planning data. See also Planning with BW-BPS in Overview [Page 88]. The data is simultaneously written into the InfoCube by multiple users. Standard InfoCubes are not suitable for this. You should use Standard InfoCubes for read-only access (for example, when reading reference data).

Structure

Real-time InfoCubes can be filled with data using two different methods: Using the transaction of BW-BPS to enter planning data and using BI staging, whereas planning data then cannot be loaded simultaneously. You have the option to convert a real-time InfoCube Select *Convert Real-Time InfoCube* using the context menu in your real-time InfoCube in the InfoProvider tree. By default, *Real-Time Cube Can Be Planned, Data Loading Not Permitted* is selected. Switch this setting to *Real-Time Cube Can Be Loaded With Data; Planning Not Permitted* if you want to fill the cube with data via BI Staging.

During entry of planning data, the data is written to a real-time InfoCube data request. As soon as the number of records in a data request exceeds a threshold value, the request is closed and a rollup is carried out for this request in defined aggregates (asynchronously). You can still rollup and define aggregates, collapse, and so on, as before.

According to the database on which they are based, real-time InfoCubes differ from Standard BasisCubes in the way they are indexed and partitioned. For an Oracle DBMS this means, for example, no Bitmap indices for the fact table and no partitioning (initiated by BI) of the fact table according to the packet dimensions.

Reduced read-only performance is accepted as a drawback of real-time InfoCubes, in favor of the option of parallel (transactional) writing and improved write performance.

Creating a Real-Time InfoCube

When creating a new InfoCube in the Data Warehousing Workbench, mark the *Real-Time indicator.*

Converting a Standard InfoCube into a Real-Time InfoCube

Conversion with Loss of Transaction Data

If the Standard InfoCube already contains transaction data but you no longer need it (for example, test data from the implementation phase of the system), you proceed as follows:

1. In the InfoCube maintenance in the Data Warehousing Workbench choose, from the main menu, *InfoCube* → *Delete Data Content.* The transaction data is deleted and the InfoCube is set to *inactive*.
2. Continue with the same procedure as with creating a real-time InfoCube.

Conversion with the Retention of Transaction Data

If the Standard InfoCube already contains transaction data from the production operation that you still need, proceed as follows:

Execute the SAP_CONVERT_TO_TRANSACTIONAL ABAP report under the name of the corresponding InfoCube. You should schedule this report as a background job for InfoCubes with more than 10,000 data records. This is to avoid a potentially long run-time.

Integration

The following typical scenarios arise for the use of real-time InfoCubes in BW-BPS.

1. Scenario:

Actual data (read-only access) and planned data (read-only and write access) have to be held in different InfoCubes. Therefore, use a Standard InfoCube for actual data and a real-time InfoCube for planned data. Data integration is achieved using a multi-planning area that contains the areas that are assigned to the InfoCubes. Here, access to the two different InfoCubes is controlled by the characteristic *Planning area*, which is automatically added.

Scenario:

In this scenario, the planned and actual data have to be together in one InfoCube. This is the case, for example, with special rolling forecast variants. Here you have to use a real-time InfoCube, since both read-only and write accesses take place. You can no longer load data directly that has already arrived in the InfoCube by means of an upload or import source. To be able to load data nevertheless, you have to make a copy of the real-time InfoCube that is identified as a Standard InfoCube and not as real-time. Data is loaded as usual here and subsequently updated to the real-time InfoCube.

4.4 Using Connectivity and Interoperability

Purpose

BW-BPS

The BW-BPS programming interface allows you to integrate planning functionality in as many external applications as required.

For more information, see Programming Interfaces [Page 351].

4.5 Enabling Globalization

Purpose

BW-BPS

In order to display the descriptions of elements that you create and edit in the planning environment in different logon languages, you have to translate the objects.

For more information, see Translation [Page 345].

5 Ensuring Quality

This section deals with the quality measures involved in development such as testing, logging and tracing, and debugging.

5.1 Testing

Purpose

Analysis Process Designer

The Analysis Process Designer (APD) allows you to display the data for each step in the analysis process, calculate intermediate results, and analyze the quality of the data for some nodes.

For more information, see Checking Data [Page 389].

5.2 Logging and Tracing

Purpose

BW-BPS Trace-Mode

The trace mode allows you to obtain detailed information about which data objects were changed and how when a planning function was executed. This function is especially useful when you set up new planning functions and parameter groups and want to check whether the function has the desired effect on the existing data.

For more information, see Trace Mode for Planning Functions [Page 205].

6 Reference

This section contains a list of all available APIs and tutorials, along with check lists that help you to avoid design errors.

6.1 API Documentation

Purpose

The following APIs are available in the *Business Planning and Analytical Services* scenario.

BW-BPS

Programming interface of BW-BPS:

API	**Use**
API_SEMBPS_GETDATA	This API reads the data from the InfoCube that the addressed planning area is assigned to. It prepares the values corresponding to the planning layout transferred as a parameter, automatically fills the buffer with the read data, and transfers the data into the input-ready table.
API_SEMBPS_SETDATA	This API writes the data displayed presently to the buffer memory that is managed by BW-BPS.
API_SEMBPS_FUNCTION_EXECUTE	This API runs the revaluation you require.
API_SEMBPS_POST	This API saves the data in the buffer persistently in the data basis.
API_SEMBPS_REFRESH	This API releases the objects.

7 Copyright

7.1 SAP Copyrights and Trademarks

SAP Library document classification: PUBLIC

SAP Copyrights and Trademarks