



**PUBLIC**

SAP Analysis for Microsoft Office

Document Version: 2.8 – 2019-12-19

# Analysis Plug-in User Guide

# Content

- 1 About this guide. . . . . 8**
- 1.1 Who should read this guide? . . . . . 8
- 1.2 User profiles. . . . . 8
- 1.3 About the documentation set. . . . . 8
- 2 Getting Started. . . . . 10**
- 2.1 What is SAP Analysis for Microsoft Office?. . . . . 10
- 2.2 Introduction to the Analysis Plug-in. . . . . 10
- 2.3 Working with Analysis in Microsoft Excel. . . . . 11
  - Analysis in the File tab. . . . . 12
  - Analysis tab. . . . . 13
  - Analysis Design tab. . . . . 17
- 2.4 Working with Analysis in Microsoft PowerPoint. . . . . 20
- 2.5 Customizing the User Interface . . . . . 24
- 3 Creating and Administrating Workbooks. . . . . 26**
- 3.1 To insert a data source in a workbook. . . . . 27
- 3.2 To insert a data source into a default workbook. . . . . 29
  - To define a default workbook. . . . . 30
- 3.3 To open a workbook. . . . . 30
- 3.4 To save a workbook. . . . . 32
- 3.5 To rename a workbook. . . . . 34
- 3.6 To delete a workbook. . . . . 34
- 3.7 To protect a workbook or worksheet. . . . . 35
- 3.8 To launch Analysis from a BW system. . . . . 36
- 3.9 To convert a BEx workbook. . . . . 37
  - Conversion log. . . . . 39
- 4 Designing Workbooks. . . . . 40**
- 4.1 Inserting other components. . . . . 41
  - To insert a dynamic chart. . . . . 41
  - To insert a waterfall chart. . . . . 41
  - To insert an info field. . . . . 43
  - To insert a filter. . . . . 43
- 4.2 Defining Conditional Formatting. . . . . 44
  - To define a conditional format. . . . . 45
  - To edit conditional formats. . . . . 46
- 4.3 Editing crosstabs with Table Design. . . . . 47

	To apply formats. . . . .	48
	To add new lines. . . . .	49
	To add texts in new member cells. . . . .	50
	To add formulas . . . . .	51
	Changing Table Design rules. . . . .	52
4.4	Defining style sets for crosstabs. . . . .	52
	SAP cell styles. . . . .	53
	To apply a style set. . . . .	55
	To create cell styles and style sets. . . . .	55
	To share a style set. . . . .	56
	To delete a style set. . . . .	57
4.5	Grouping crosstabs. . . . .	58
4.6	Linking dimensions. . . . .	59
4.7	Inserting data sources using BW Workspaces. . . . .	60
	To create a local provider. . . . .	60
	To reload data in a local provider. . . . .	62
	To create a CompositeProvider. . . . .	62
4.8	Converting crosstab cells to formula. . . . .	64
	To convert a crosstab to formula. . . . .	65
4.9	Creating Web Applications. . . . .	66
4.10	Launching the Query Designer for editing a data source. . . . .	67
4.11	To smart copy/paste a data source. . . . .	68
4.12	To save a query view. . . . .	69
4.13	Working with formulas. . . . .	70
	To create a formula. . . . .	71
	Working in formula-optimized mode. . . . .	72
	SAPGetData. . . . .	74
	SAPGetDimensionDynamicFilter. . . . .	75
	SAPGetDimensionEffectiveFilter. . . . .	75
	SAPGetDimensionInfo. . . . .	76
	SAPGetDimensionStaticFilter. . . . .	77
	SAPGetDisplayedMeasures. . . . .	77
	SAPGetInfoLabel. . . . .	78
	SAPGetMeasureFilter. . . . .	79
	SAPGetMeasureInfo. . . . .	79
	SAPGetMember. . . . .	80
	SAPGetSourceInfo. . . . .	81
	SAPGetUniformScaling. . . . .	82
	SAPGetVariable. . . . .	82
	SAPGetWorkbookInfo. . . . .	83
	SAPListOf. . . . .	84

	SAPListOfDesignRules. . . . .	85
	SAPListOfDimensions. . . . .	85
	SAPListOfDynamicFilters. . . . .	86
	SAPListOfEffectiveFilters. . . . .	87
	SAPListOfMeasureFilters. . . . .	87
	SAPListOfMembers. . . . .	88
	SAPListOfMessages. . . . .	89
	SAPListOfStaticFilters. . . . .	90
	SAPListOfVariables. . . . .	90
	SAPSelectMember. . . . .	91
	SAPSetData. . . . .	92
	SAPSetFilterComponent. . . . .	93
4.14	Working with macros. . . . .	94
	SAPAddMessage. . . . .	95
	SAPCallMemberSelector. . . . .	96
	SAPExecuteCommand. . . . .	98
	SAPExecutePlanningFunction. . . . .	104
	SAPExecutePlanningSequence. . . . .	104
	SAPGetCellInfo. . . . .	105
	SAPGetProperty. . . . .	106
	SAPLogOff. . . . .	107
	SAPLogon. . . . .	108
	SAPMoveDimension. . . . .	109
	SAPOpenWorkbook. . . . .	110
	SAPSetFilter. . . . .	111
	SAPSetPlanParameter. . . . .	113
	SAPSetRefreshBehaviour. . . . .	114
	SAPSetVariable. . . . .	114
	SAPSuppressMessage. . . . .	116
	Table Design API. . . . .	117
	Using Analysis functions. . . . .	123
	Hiding / Showing Analysis UI components. . . . .	124
	Enhancing the Analysis Ribbon. . . . .	125
	Using BAdI information. . . . .	126
	Using Callbacks. . . . .	128
	Using LastError information. . . . .	132
	Syntax for Entering Values. . . . .	134
<b>5</b>	<b>Analyzing Data. . . . .</b>	<b>136</b>
5.1	Analyzing data with the design panel. . . . .	136
	The Analysis tab. . . . .	137
	The Property View on the Analysis tab. . . . .	139

	The Information tab. . . . .	141
	The Components tab. . . . .	142
	The Design Rules tab. . . . .	147
	The Comments tab. . . . .	148
	The Versions tab. . . . .	149
	Pausing Refresh. . . . .	149
	Managing components in the design panel. . . . .	150
	Managing system connections in the design panel. . . . .	152
5.2	Prompting. . . . .	153
	To define prompt values. . . . .	154
	Using Variants. . . . .	156
	To select workbook properties for prompting. . . . .	160
5.3	Filtering data . . . . .	162
	Filtering members. . . . .	162
	Filtering measures. . . . .	170
	To work with BEx conditions in Analysis. . . . .	174
	To show/hide zeros in rows and columns. . . . .	175
5.4	Sorting data. . . . .	176
	To sort values. . . . .	177
	To sort members. . . . .	178
5.5	Working with hierarchies. . . . .	178
	To include dimensions with hierarchies in an analysis. . . . .	180
	To display single dimensions as hierarchy. . . . .	182
	To group members. . . . .	183
5.6	Calculating new measures . . . . .	184
	To calculate a new measure based on available measures. . . . .	185
	To add a new measure based on one available measure. . . . .	186
	To add a new measure based on free-form calculation. . . . .	187
	To add a restricted measure . . . . .	189
5.7	Defining the display of members, measures and totals. . . . .	190
	To define the members display. . . . .	190
	Defining the measures display. . . . .	191
	Defining the totals display. . . . .	194
5.8	To jump to a target with Goto. . . . .	197
5.9	Commenting data cells. . . . .	198
	To comment a data cell. . . . .	199
	To save comments in SAP BW/4HANA. . . . .	199
	To save comments on the BI platform. . . . .	201
	To save comments locally in the workbook. . . . .	202
5.10	Planning Data. . . . .	202
	To plan data manually. . . . .	203

	To plan data with planning objects . . . . .	206
	To recalculate planning data . . . . .	208
	To save planning data . . . . .	209
	Locking Cells. . . . .	209
	Setting the Work Status. . . . .	212
	To undo changes while planning data . . . . .	214
	To switch between display and change mode. . . . .	214
	Editing short texts in queries. . . . .	215
<b>6</b>	<b>Analyzing SAP HANA Data. . . . .</b>	<b>217</b>
6.1	Analyzing SAP HANA data sources. . . . .	217
	Calculating new measures based on SAP HANA views. . . . .	218
<b>7</b>	<b>Analyzing SAP Analytics Cloud Data. . . . .</b>	<b>219</b>
7.1	Analyzing SAP Analytics Cloud models. . . . .	219
7.2	Planning Data with SAP Analytics Cloud models. . . . .	220
<b>8</b>	<b>Creating Presentations. . . . .</b>	<b>222</b>
8.1	To smart paste a data source. . . . .	223
8.2	Presentation properties. . . . .	224
<b>9</b>	<b>Using Analysis in other SAP Applications. . . . .</b>	<b>225</b>
9.1	To save an analysis view. . . . .	225
9.2	To insert an analysis view. . . . .	226
9.3	To export a data source to SAP Lumira / SAP BusinessObjects Design Studio. . . . .	227
<b>10</b>	<b>Scheduling and Publishing. . . . .</b>	<b>228</b>
10.1	Precalculation Restrictions. . . . .	229
<b>11</b>	<b>Customizing Analysis. . . . .</b>	<b>230</b>
11.1	Customizing the User Interface. . . . .	230
	Introduction to Customizing the User Interface. . . . .	230
	Profiles. . . . .	231
	Creating a profile. . . . .	232
	Renaming a profile. . . . .	240
	Setting a profile as a default profile. . . . .	240
	Embedding a profile in a workbook. . . . .	240
	Creating a Company Profile. . . . .	241
	Using user profiles as read-only profiles. . . . .	242
11.2	Maintaining settings in Analysis. . . . .	243
	Analysis Settings. . . . .	244
11.3	Options. . . . .	301
	User Options. . . . .	301
	Advanced Options. . . . .	302

	Platform Options. . . . .	303
	Conversion Options. . . . .	305
<b>12</b>	<b>Troubleshooting. . . . .</b>	<b>307</b>
12.1	Troubleshooting in Analysis. . . . .	307
12.2	To enable the Analysis Add-In after system crash. . . . .	309

# 1 About this guide

## 1.1 Who should read this guide?

This guide is intended for users interested in building and analyzing workbooks using SAP Analysis for Microsoft Office.

## 1.2 User profiles

There are three user profiles for SAP Analysis for Microsoft Office:

- **Workbook Creator**  
Users who create and maintain workbooks based on different data sources.
- **Data Analyst**  
Users who navigate through existing workbooks and analyze the data they contain. They can also include workbooks in a Microsoft PowerPoint presentation and continue the analysis there.
- **Administrator**  
IT specialists who install, configure and administer SAP Analysis for Microsoft Office. They also assign security rights and authorizations to workbook creators and analyzers.

If your existing profile needs to be modified, contact your IT administrator.

## 1.3 About the documentation set

The documentation set for SAP Analysis for Microsoft Office, comprises the following guides and online help products:

### → Tip

The guides and tutorials are regularly updated and enhanced. Make sure that you have the latest version by checking the SAP Help Portal and SAP Community Network on a regular basis.

## Administrator Guide

The Administrator Guide contains detailed information that a user needs to install, configure and administer Analysis for Microsoft Office. The guide is available on the SAP Help Portal.

## User Guide

The User Guide contains the conceptual information, procedures and reference material that a user needs to create and analyze Microsoft Excel workbooks and Microsoft PowerPoint slides with Analysis for Microsoft Office. There are three user guides for Analysis: the Analysis Plug-in User Guide, the BPC Plug-in User Guide and the EPM Add-in User Guide. The guides are available on the SAP Help Portal.

## Online Help

The online help contains the same information as the User Guides. It links directly to the documents on the SAP Help Portal.

## What's New Guide

The What's New guide for SAP Analysis for Microsoft Office, provides a complete list of the new and modified features for SAP Analysis since the previous release. The guide is available on the SAP Help Portal.

## eLearning Tutorials

The tutorials show you how to use SAP Analysis. They give you a quick introduction to different features so that you can learn the basics of working with the Add-In. They also give you a first impression of the look and feel. The tutorials are available in the SAP Community Network at <http://scn.sap.com/docs/DOC-7679?refer=product-help>.

## 2 Getting Started

### 2.1 What is SAP Analysis for Microsoft Office?

SAP Analysis for Microsoft Office, is a Microsoft Office Add-In that allows multidimensional analysis of OLAP sources. It consists of the following components:

- Analysis Add-in including the Analysis plug-in and the Business Planning and Consolidation plug-in
- Enterprise Performance Management Add-in

The add-ins include versions for Microsoft Excel and Microsoft PowerPoint. They can be installed in one common installation. After the installation, the Analysis plug-in and the Enterprise Performance Management (EPM) add-in are available as separate tabs in the ribbon. The Business Planning and Consolidation plug-in is available in the interface as a pane named *Activity*.

In the Analysis for Microsoft Excel, you can use the add-ins in one workbook.

In Analysis for Microsoft PowerPoint, you can add data sources with the add-ins into one presentation and define the analysis with the respective plug-in.

The Business Planning and Consolidation plug-in pane can be displayed in the Ribbon: Analysis.

### More Information

For more information on the Analysis add-in (and its plug-ins), see the SAP Help Portal at [SAP Analysis for Microsoft Office](#).

For more information on the EPM add-in, see the SAP Help Portal at [SAP EPM solutions, add-in for Microsoft Office](#).

### 2.2 Introduction to the Analysis Plug-in

The Analysis plug-in allows multidimensional analysis of OLAP sources in Microsoft Excel, MS Excel workbook application design, and intuitive creation of BI presentations with MS PowerPoint. The Plug-in is available for the following Microsoft Office versions:

- Office 365 (Excel and PowerPoint)
- Microsoft Office 2019 (Excel and PowerPoint)
- Microsoft Office 2016 (Excel and PowerPoint)
- Microsoft Office 2013 (Excel and PowerPoint)
- Microsoft Office 2010 (Excel and PowerPoint)

In Microsoft Excel, Analysis is available in two separate tabs in the ribbon: *Analysis* and *Analysis Design*. In Microsoft PowerPoint, it is available in one tab: *Analysis*.

In the Analysis plug-in, you can use SAP Queries, query views and InfoProvider as data sources. The data is displayed in the workbook in crosstabs. You can insert multiple crosstabs in a workbook with data from different sources and systems. If the workbook will be used by different users, it is also helpful to add info fields with information on the data source and filter status.

Using the design panel, you can analyze the data and change the view on the displayed data. You can add and remove dimensions and measures to be displayed easily with drag and drop. To avoid single refreshes after each step, you can pause the refresh to build a crosstab. After ending the pause, all changes are applied at once.

You can refine your analysis using conditional formatting, filter, prompting, calculations and display hierarchies. You can also add charts to your analysis. If you want to keep a status of your navigation, you can save it as an analysis view. Other users can then reuse your analysis.

For more sophisticated workbook design, the Analysis plug-in contains a dedicated set of functions in Microsoft Excel to access data and meta data of connected BW systems. There are also a number of API functions available that you can use with the Visual Basic Editor, to filter data and set values for BW variables.

You can also plan business data based on the current data in your data source. You can enter the planning data manually and you can enter planning data automatically using planning functions and planning sequences of SAP BW Integrated Planning.

The Analysis plug-in, must be installed on your local machine. You can connect directly to a SAP BW system or you can connect via a to include data sources. You can use the following platforms to store and share workbooks and presentations: SAP BusinessObjects business intelligence platform and SAP BW (SAP BW/4HANA).

### **i** Note

Microsoft Office documents contain free text fields. These text fields are not intended to store personal data without additional technical or organizational measures to safeguard data protection and privacy.

In order to ensure that analytical data which is personal data and which is retrieved from the servers with Analysis functionality, does not get stored in documents, the Analysis workbook property *Remove Data Before Saving* can be active in those workbooks.

Using the business intelligence platform enables you to save workbooks and presentations with their navigation state in a central management system and to reuse these analysis views in other applications such as SAP Crystal Reports or Analysis, OLAP edition.

To get a first impression of the look and feel of the Plug-in, you can have a look at the Analysis eLearning tutorials. They are available in the SAP Community Network at <http://scn.sap.com/docs/DOC-7679>.

## **2.3 Working with Analysis in Microsoft Excel**

In Microsoft Excel, Analysis is available in two separate tabs in the ribbon: *Analysis* and *Analysis Design*.

The ribbon is part of the Microsoft Office user interface above the main work area that presents commands and options. Some Analysis options are available in the ribbon tab under **File > Analysis** in Microsoft Excel.

This guide describes procedures using the ribbon. Most of the options are also available via the context menu.

## Related Information







[Analysis in the File tab \[page 12\]](#)




[Analysis tab \[page 13\]](#)

[Analysis Design tab \[page 17\]](#)

### 2.3.1 Analysis in the File tab

The following table describe the options that are available for Analysis in the ribbon tab under ► *File* ► *Analysis* ►.

Icon	Description
	Open Data Source More information: <a href="#">To insert a data source into a default workbook [page 29]</a>
	Open Workbook This icon is displayed if only one platform is enabled. More information: <a href="#">To open a workbook [page 30]</a>
	Open Workbook Open Workbook from SAP BusinessObjects BI Platform. This icon is displayed if both platforms are enabled.
	Open Workbook Open Workbook from SAP BW. This icon is displayed if both platforms are enabled.
	Convert BEx Workbook Convert BEx workbook to Analysis workbook. More information: <a href="#">To convert a BEx workbook [page 37]</a>
	Save Workbook This icon is displayed if only one platform is enabled. More information: <a href="#">To save a workbook [page 32]</a>

Icon	Description
	<p>Save Workbook</p> <p>Save Workbook to SAP BusinessObjects BI Platform.</p> <p>This icon is displayed if both platforms are enabled.</p>
	<p>Save Workbook</p> <p>Save Workbook to SAP BW.</p> <p>This icon is displayed if both platforms are enabled.</p>
	<p>Comments</p> <p>You can save comments on the BI platform.</p> <p>More information: <a href="#">To save comments on the BI platform [page 201]</a></p>
	<p>Protect Workbok</p> <p>Protect workbook or sheet.</p> <p>More information: <a href="#">To protect a workbook or worksheet [page 35]</a></p>
	<p>Adapt Analysis</p> <p>You can customize the user interface, change the technical configuration and options</p> <p>More information: <a href="#">Introduction to Customizing the User Interface [page 230]</a></p> <p>More information: <a href="#">Options [page 301]</a></p>
	<p>Examine Analysis</p> <p>Run Analysis in different problem examination modes.</p>
	<p>About Analysis</p> <p>View details of the installed version of Analysis.</p>
	<p>Help</p> <p>Launch help.</p>



## 2.3.2 Analysis tab

The *Analysis* tab contains the following groups:





- Data Source
- Actions
- Data Analysis
- Display
- Comments
- Planning
- Design Panel

The following tables describe the groups in the Analysis tab and their options.



## Data Source group




Icon	Description
	<p>Insert Data Source</p> <p>Insert data from a source system into a crosstab.</p> <p>More information: <a href="#">To insert a data source in a workbook [page 27]</a></p>
	<p>Refresh All</p> <p>Refresh all data sources. The data sources are updated with the corresponding data from the server and the crosstabs are redrawn.</p> <p>More information: <a href="#">The Components tab [page 142]</a></p>

## Actions group




Icon	Description
	<p>Undo</p> <p>Undo last Analysis step.</p>
	<p>Redo</p> <p>Redo last Analysis step.</p>
	<p>Messages</p> <p>Display a dialog with error, warning and information messages.</p> <p>Select  <i>Messages</i>  to display the messages in a dialog.</p>

## Data Analysis group

Icon	Description
	<p>Prompts</p> <p>Enter values for query parameters and variables.</p> <p>More information: <a href="#">Prompting [page 153]</a></p>
	<p>Filter</p> <p>Define filter criteria for data.</p> <p>More information: <a href="#">To filter data by measure [page 171]</a> / <a href="#">To filter data by member [page 163]</a></p>

Icon	Description
	Sort Sort data. More information: <a href="#">Sorting data [page 176]</a>
	Hierarchy Define hierarchy options such as expansion level and parent member positions. More information: <a href="#">Working with hierarchies [page 178]</a>
	Calculations Define simple calculations (+, -, *, /) and dynamic calculations (for example, ranking and cumulation). More information: <a href="#">Calculating new measures [page 184]</a>

## Display group







Icon	Description
	Member Display Configure display for members (key/text). More information: <a href="#">To define the members display [page 190]</a>
	Measure Display Define display options for measures (for example, decimal places, scaling factors and currencies). More information: <a href="#">Defining the measures display [page 191]</a>
	Totals Configure display, position and calculation of totals. More information: <a href="#">Defining the totals display [page 194]</a>

## Comments group


Icon	Description
	Save Comments
	New Comment
	Delete
	Show/Hide Comments


## Planning group

This group is optional. You can configure in the *Customize User Interface* dialog if this group should be displayed in the ribbon or not.

Icon	Description
	Save Data Save plan values to InfoProvider. More information: <a href="#">To save planning data [page 209]</a>
	Recalculate Recalculate plan values. More information: <a href="#">To recalculate planning data [page 208]</a>
	Lock Cells Lock and unlock input-ready cells. More information: <a href="#">To lock cells [page 211]</a>
	Work Status Change the work status of the data source/selected cells. More information: <a href="#">Setting the Work Status [page 212]</a>
	Display Switch all data sources to display mode. More information: <a href="#">To switch between display and change mode [page 214]</a>
	Change Switch all data sources to change mode. More information: <a href="#">To switch between display and change mode [page 214]</a>
	Back Reset edited cells. More information: <a href="#">To undo changes while planning data [page 214]</a>

## Design Panel group

Icon	Description
	Display Show/hide Design Panel More information: <a href="#">Analyzing data with the design panel [page 136]</a>

Icon	Description
	<p>Pause Refresh</p> <p>Activate/deactivate automatic refresh after each navigation step in the Design Panel.</p> <p>More information: <a href="#">Analyzing data with the design panel [page 136]</a></p>




## 2.3.3 Analysis Design tab

The *Analysis Design* tab contains the following groups:



- Insert
- Actions
- Format
- Build
- Tools
- View
- Design Panel

The following tables describe the groups in the Analysis Design tab and their options.



### Insert group

Icon	Description
	<p>Chart</p> <p>Insert dynamic chart.</p> <p>More information: <a href="#">To insert a dynamic chart [page 41]</a></p>
	<p>Info Field</p> <p>Insert information on data sources (for example, name and last data update).</p> <p>More information: <a href="#">To insert an info field [page 43]</a></p>
	<p>Filter</p> <p>Insert component for simple data filtering.</p> <p>More information: <a href="#">To insert a filter [page 43]</a></p>




## Actions group

Icon	Description
	Undo Undo last Analysis step.
	Redo Redo last Analysis step.
	Messages Display a dialog with error, warning and information messages. Select ► <b>Messages</b> ► <b>Show Messages</b> ► to display the messages in a dialog.


## Format group

Icon	Description
	Conditional Formatting Define rules for highlighting values using colors and symbols. More information: <a href="#">To define a conditional format [page 45]</a>
	New Lines Insert/delete new rows or new columns. More information: <a href="#">To add new lines [page 49]</a>
	Format Cells Format new cells or existing cells. More information: <a href="#">To apply formats [page 48]</a>
	Crosstab Configure display options for the crosstab. The following options are available: <ul style="list-style-type: none"><li>• Apply Default Formats</li><li>• Display Symbols for Parent Members</li><li>• Optimum Cell Width/Height</li><li>• Repeat Members</li></ul>
	Style Manage crosstab styles. More information: <a href="#">Defining style sets for crosstabs [page 52]</a>

## Build group

Icon	Description
	<p>Combine</p> <p>Group crosstabs or link dimensions.</p> <p>More information: <a href="#">Grouping crosstabs [page 58]</a></p> <p>More information: <a href="#">Linking dimensions [page 59]</a></p>
	<p>Workspaces - Create Local Provider</p> <p>More information: <a href="#">To create a local provider [page 60]</a></p>
	<p>Workspaces - Reload Local Provider</p> <p>More information: <a href="#">To reload data in a local provider [page 62]</a></p>
	<p>Workspaces - Add Local Provider to Data Source</p> <p>More information: <a href="#">To create a CompositeProvider [page 62]</a></p>



## Tools group

Icon	Description
	<p>Convert to Formula</p> <p>Convert a crosstab into Excel formulas to retrieve the data.</p> <p>More information: <a href="#">Converting crosstab cells to formula [page 64]</a></p>
	<p>Create Web Application</p> <p>Transfer components in an Analysis workbook to the SAP BusinessObject Design Studio to create a web application.</p> <p>More information: <a href="#">Creating Web Applications [page 66]</a></p>
	<p>Launch Query Designer</p> <p>Launch a query designer to check the query definition and to change the definition according to your needs.</p> <p>More information: <a href="#">Launching the Query Designer for editing a data source [page 67]</a></p>

## View group

Icon	Description
	Smart Copy Copy data source to clipboard. More information: <a href="#">To smart copy/paste a data source [page 68]</a>
	Smart Paste Paste data source from clipboard as table. More information: <a href="#">To smart copy/paste a data source [page 68]</a>
	Save View Save data source as view. More information: <a href="#">To save a query view [page 69]</a>

## Design Panel group

Icon	Description
	Display Show/hide Design Panel More information: <a href="#">Analyzing data with the design panel [page 136]</a>
	Pause Refresh Activate/deactivate automatic refresh after each navigation step in the Design Panel. More information: <a href="#">Analyzing data with the design panel [page 136]</a>

## 2.4 Working with Analysis in Microsoft PowerPoint

In Microsoft PowerPoint, Analysis is available as a separate tab in the ribbon. The ribbon is part of the Microsoft Office user interface above the main work area that presents commands and options. Some Analysis options are available in the ribbon tab under **File > Analysis** in Microsoft PowerPoint.

This guide describes procedures using the ribbon. Most of the options are also available via the context menu.









The *Analysis* tab contains the following groups: :

- Data Source
- Actions
- Data Analysis

- Display
- Insert
- Tools
- Presentation



The following table describe the options that are available in the ribbon tab under **File > Analysis**.

## Analysis in the File tab

Icon	Description
	Open Presentation This icon is displayed if only one platform is enabled.
	Open Presentation Open Presentation from SAP BusinessObjects BI Platform. This icon is displayed if both platforms are enabled.
	Open Presentation Open Presentation from SAP BW. This icon is displayed if both platforms are enabled.
	Save Presentation This icon is displayed if only one platform is enabled.
	Save Presentation Save Presentation to SAP BusinessObjects BI Platform. This icon is displayed if both platforms are enabled.
	Save Presentation Save Presentation to SAP BW. This icon is displayed if both platforms are enabled.
	Settings Edit settings. More information: <a href="#">Options [page 301]</a>
	About Analysis View details of the installed version of Analysis.
	Help Launch help.



The following tables describe the groups in the Analysis tab and their options.

## Data Source group





Icon	Description
	<p>Insert Data Source</p> <p>Insert data from a source system into a crosstab.</p> <p>More information: <a href="#">To insert a data source in a workbook [page 27]</a></p>
	<p>Refresh All</p> <p>Refresh all data sources.</p> <p>More information: <a href="#">The Components tab [page 142]</a></p>

To open and save existing presentations saved on the business intelligence platform, use the corresponding options in the Microsoft Office button.




## Actions group

Icon	Description
	<p>Undo</p> <p>Undo last Analysis step.</p>
	<p>Redo</p> <p>Redo last Analysis step.</p>
	<p>Messages</p> <p>Display a dialog with error, warning and information messages.</p> <p>You have the following options:</p> <ul style="list-style-type: none"><li>• Select <b>Messages &gt; Show Messages</b> to display the messages in a dialog.</li><li>• Select <b>Messages &gt; Show Workbook Profiling Statistics</b> to display the workbook profiling statistics.</li><li>• Select <b>Messages &gt; Show Client Profiling Statistics</b> to display the client profiling statistics.</li></ul>



## Data Analysis group

Icon	Description
	<p>Prompts</p> <p>Enter values for query parameters and variables.</p> <p>More information: <a href="#">Prompting [page 153]</a></p>
	<p>Filter</p> <p>Define filter criteria for data.</p> <p>More information: <a href="#">To filter data by measure [page 171]</a> <a href="#">To filter data by member [page 163]</a></p>
	<p>Sort</p> <p>Sort data.</p> <p>More information: <a href="#">Sorting data [page 176]</a></p>
	<p>Hierarchy</p> <p>Define hierarchy options such as expansion level and parent member positions.</p> <p>More information: <a href="#">Working with hierarchies [page 178]</a></p>



## Display group

Icon	Description
	<p>Member Display</p> <p>Configure display for members (key/text).</p> <p>More information: <a href="#">To define the members display [page 190]</a></p>
	<p>Measure Display</p> <p>Define display options for measures (for example, decimal places, scaling factors and display currency).</p> <p>More information: <a href="#">Defining the measures display [page 191]</a></p>
	<p>Totals</p> <p>Configure display, position and calculation of totals.</p> <p>More information: <a href="#">Defining the totals display [page 194]</a></p>

## Insert group

Icon	Description
	<p>Chart</p> <p>Insert dynamic chart.</p> <p>More information: <a href="#">To insert a dynamic chart [page 41]</a></p>
	<p>Info Field</p> <p>Insert information on data sources (for example, name and last data update).</p> <p>More information: <a href="#">To insert an info field [page 43]</a></p>

## Tools group

Icon	Description
	<p>Fit Table</p> <p>Abbreviate a table to fit one slide, or split the table across multiple slides.</p>
	<p>Move to</p> <p>Move the selected Analysis object (table, chart or info field) from its current location to different slide in the presentation.</p>
	<p>Smart Paste</p> <p>Paste data source from clipboard.</p> <p>More information: <a href="#">To smart paste a data source [page 223]</a></p>

## Presentation group

Icon	Description
	<p>Properties</p> <p>Configure the properties of this presentation.</p> <p>More information: <a href="#">Presentation properties [page 224]</a></p>

## 2.5 Customizing the User Interface

You can customize certain user interface areas of Analysis to meet your business needs.

## i Note

In Analysis versions less than 2.4, the display of the following commands was specified with file system settings: Convert to Formula, Create Web Application, Launch Query Designer and Planning group.

When switching to Analysis 2.5 from a version less than 4, for example 2.3, Analysis takes the definition from the setting in the file system. If the command was displayed in the 2.3 installation, it will also be displayed in the 2.5 installation.

To see how you can customize user interface areas using profiles, check out this how-to video: <https://youtu.be/X1jXGF8NiFs> ➔

## Related Information

[Introduction to Customizing the User Interface \[page 230\]](#)

# 3 Creating and Administrating Workbooks

You can insert SAP BEx Queries, query views and SAP BW InfoProvider as data sources into a workbook. These data sources are stored in a SAP BW system. You can add multiple crosstabs to a worksheet or workbook. The crosstabs can contain data from the same data source or from different sources. You can also use data sources that are stored in different systems in one workbook.

You can also insert and analyze SAP HANA data sources and SAP Analytics Cloud models in an Analysis workbook.

To add a crosstab with data to a workbook, you select a data source in a SAP BW system. You need the appropriate authorizations for the platform you want to use (SAP BW, SAP BW/4HANA or SAP BusinessObjects Business Intelligence), and the relevant SAP BW systems to insert a data source into a workbook. For more information, contact your IT administrator.

You can insert a data source into a standard workbook or default workbook. If you use a default workbook, styles, formatting and the workbook settings that you defined in the default workbook are used when you open a data source.

You can also insert data sources with local data into a workbook. To be analyzed, the local data is uploaded to a BW workspace.

You can save new workbooks either on the SAP BusinessObjects business intelligence platform, a SAP BW platform or in a SAP BW/HANA system. Stored workbooks can be opened from the corresponding platforms. They can also be renamed or deleted on the platforms.

You can also convert workbooks created with the SAP BEx Analyzer 3.5 and SAP BEx Analyzer 7.0 to an Analysis workbook.

## Related Information

[To insert a data source in a workbook \[page 27\]](#)

[To insert a data source into a default workbook \[page 29\]](#)

[To save a workbook \[page 32\]](#)

[To open a workbook \[page 30\]](#)

[To rename a workbook \[page 34\]](#)

[To delete a workbook \[page 34\]](#)

[To convert a BEx workbook \[page 37\]](#)

[Analyzing SAP HANA data sources \[page 217\]](#)

[Analyzing SAP Analytics Cloud models \[page 219\]](#)

## 3.1 To insert a data source in a workbook

### Procedure

1. Select the cell in the worksheet where the crosstab with the data from the selected data source should be inserted.
2. Select **► Insert Data Source ► Select Data Source for Analysis ►**.  
The *Log on to SAP BusinessObjects BI platform* dialog box appears.
3. Log on to the platform you want to use:
  - Enter your *User*, *Password* and the *WEB Service URL* to log on to the business intelligence platform.
  - Select *Skip* to use a SAP BW system, a SAP HANA system or a SAP Analytics Cloud (import data connections and live data connections).  
Log on to one of the platforms directly without using the business intelligence platform. Continue with step 7/9 if using this log on.

#### i Note

If you log on with an initial password to a BW system, or your password is expired and needs to be reset, the dialog for changing the password opens automatically.

4. Optional step: Enter *System* and *Authentication*.  
You will normally not be asked to supply this information. However, if you are asked to log on to a special Central Management System (CMS), you can add these two additional fields to the dialog box by selecting *Options*. Enter the name of your Central Management System in the *System* field and the authentication type in the *Authentication* field.
5. Press *OK*.  
The *Select Data Source* dialog box appears.
6. Select a connection in the *Show Connections* list:
  - If you select *All*, all available systems, Cubes / InfoProvider and Query / Query views on the business intelligence platform are displayed.
  - If you select *System*, all available systems on the business intelligence platform are displayed.
  - If you select *Cube / InfoProvider*, all available Cubes and InfoProvider on the business intelligence platform are displayed.
  - If you select *Query / Query View*, all available Queries and query views on the business intelligence platform are displayed.
  - If you select *Local System*, all systems in your local *SAP Logon* are displayed.
7. Select a system and press *Next*.  
To select a Query, query view or InfoProvider directly, double-click the object you want to select.  
The *Logon to System* dialog box appears.  
  
You can refresh the displayed system list by choosing *Refresh* in the context menu or pressing *F5*, for example after adding a new system to the SAP Logon.
8. Enter *Client*, *User* and *Password* in the fields and press *OK*.  
If you want to specify the system language, select *Options* and enter the language in the *Language* field.
9. Select a data source in the *Select Data Source* box and press *OK*.

- For BW systems, the *Select Data Source* dialog consists of the following tabs:
  - *Search*  
You can select if you want to search for the *Description*, *Technical Name* or *All*.  
To retrieve data sources that begin with a specific string, you can type \* after a partial string.
  - *Area*
  - *Role*
  - *Workspaces*
 On the *Search* and *Role* tabs, you can specify the objects to be displayed: *All*, *InfoProvider*, *Query* or *Query View*.  
Note that query views have only a technical name.
- For HANA systems, the *Select Data Source* dialog consists of the following tabs:
  - *Search*  
You can select if you want to search for the *Description*, *Technical Name* or *All*.  
To retrieve data sources that begin with a specific string, you can type \* after a partial string.
  - *Area*
 On both tabs, you can specify the objects to be displayed: *All*, *Cube*, *Attribute View* or *Dimension View*.
- For SAP Analytics Cloud, the *Select Data Source* dialog consists of the following tabs:
  - *Search*  
You can select if you want to search for the *Description*, *Technical Name* or *All*.  
To retrieve data sources that begin with a specific string, you can type \* after a partial string.
  - *Area*

## Results

A new crosstab with the data of the selected data source is inserted into the worksheet. The SAP style set *SAP Black&White* is used as default. You can now analyze the data and change the displayed data set according to your needs. You can also add other components to your analysis, charts for example.

### i Note

You can also activate *Pause Refresh* before inserting a data source. You can then insert a data source and define your analysis on the *Analysis* tab in the design panel (columns, rows and filter). The crosstab with the data source and the defined analysis will only be inserted after *Pause Refresh* is deactivated again.

## Related Information

[Pausing Refresh \[page 149\]](#)

## 3.2 To insert a data source into a default workbook

You can define a workbook as the default workbook. The styles, formatting and the workbook settings that you defined in the default workbook are used when you open a data source. If you save the default workbook on a server, you can share it with other users.

If you do not define a default workbook, the data source is opened with the SAP standard settings.

### To open a data source with the default workbook

1. Open a data source in Analysis.  
Choose ► *File* ► *Analysis* ► *Open Data Source* ►.
2. Log on to a platform and system.
3. Select a data source.
4. Press *OK*.

A new workbook with the data from the selected data source is created using the styles, formatting and workbook settings that you defined in the default workbook for the first inserted data source. All elements that you have defined for the first data source of your workbook are displayed with the data from the new selected data source. If the default workbook contains more than one data source, the elements from the other data sources are also displayed with the data from the corresponding data sources.

### To open a data source without a default workbook

If you do not define a default workbook in the user options, the data source is opened with the following Analysis default settings:

- The crosstab is inserted starting in cell A1 in the worksheet.
- The default style set is used for formatting. SAP style set *SAP Black&White* is the initial default.

## Related Information

[To define a default workbook \[page 30\]](#)

## 3.2.1 To define a default workbook

### Procedure

1. Open a workbook and insert a data source.
2. Define the styles, formatting and workbook settings that you want to use for your default workbook.  
You can insert other elements like charts or formulas for the data source. You can also change the data source alias.

#### i Note

You can insert more than one data source into a default workbook. The elements and settings from the first data source that you insert, are used as a reference when you open a data source with the default workbook. The elements from the other data sources in the workbook are displayed unchanged when you open a new data source with the default workbook.

3. Save the workbook locally or on a server.
4. Choose *Options* on the *Analysis* file tab to open the *User* dialog.
5. Define the location of your default workbook.  
Select *Browse* to browse to your default workbook.
6. Press *OK*.

### Results

The selected workbook is defined as the default workbook.

### Related Information

[User Options \[page 301\]](#)

## 3.3 To open a workbook

### Context

You can open a workbook that is stored on the business intelligence platform or on a SAP BW platform. This allows you to see the navigation state that was saved with the workbook. You also see the workbook properties that were saved with the workbook. These properties are part of the *Component* tab in the design panel.

You can navigate through the data and design different views on the data. You can also add new data sources or components.

## Procedure

1. Choose ► *File* ► *Analysis* ► *Open Workbook* ►.

### i Note

If all platforms are enabled in your Analysis installation, you will find several icons for opening a workbook, one for each platform.

2. Log on to the selected platform and BW system.
3. Select a workbook.

You can search for a workbook name or select a workbook in the platform folder structure.

You can refresh the displayed document list by choosing *Refresh* in the context menu or pressing *F5*.

- On the business intelligence platform, the *Open Document* dialog consists of the following tabs:
  - *Search*  
You can select if you want to search for the *Description*, *Technical Name* or *All*.  
If you search a workbook on the BI platform, you may find two different object types: Analysis Workbook or Microsoft Excel Workbook.
  - *Folders*
  - *Favorite Folders*
- On the SAP BW platform, the *Open Document* dialog consists of the following tabs:
  - *Search*  
You can select if you want to search for the *Description*, *Technical Name* or *All*.
  - *My Documents*  
This tab shows the workbooks that you (the current user) have saved to the My Documents folder on the BW platform.  
Note that this does not mean that only you can access the workbooks in that folder. Other users can find the workbooks using the search and open them as Analysis does not support private workbooks.  
Your administrator can build namespaces to enable private workbooks. For more information, contact your system administrator.
  - *Role*
  - *Environment*  
You can open EPM workbooks saved in the SAP Business Planning and Consolidation (BPC) repository.  
You can access all environments with the respective EPM folder structure (private, public, teams).  
To change an EPM workbook or save it to the BPC repository, you have to use the EPM plug-in in Analysis.

4. Press *OK*.  
The workbook is opened in Analysis.
5. Choose *Refresh All* to refresh the data sources.

### i Note

If the workbook property *Refresh Workbook on Opening* is selected on the *Components* tab in the design panel, the refresh is executed automatically. In this case, you do not need to refresh the data sources manually.

## Results

The selected workbook is opened in Analysis, and you can start your analysis.

## Related Information

[Working with Analysis in Microsoft Excel \[page 11\]](#)

[The Components tab \[page 142\]](#)

[Platform Options \[page 303\]](#)

## 3.4 To save a workbook

### Context

You can save a workbook on the business intelligence platform, on a SAP BW platform or as a local file.

If you save a workbook to the BI platform (as of version 4.1), the workbook is saved as an Analysis Workbook object. If you save a workbook to the BI platform 4.0 or former releases, the workbook is saved as Microsoft Excel object.

The workbook is saved with the current navigation state and selected properties. The workbook properties are part of the *Component* tab in the design panel.

If you want to save a workbook as a local file, you can use the Microsoft Excel functionality choosing ► *File* ► *Save As* ►.

#### i Note

Microsoft Office documents contain free text fields. These text fields are not intended to store personal data without additional technical or organizational measures to safeguard data protection and privacy.

In order to ensure that analytical data which is personal data and which is retrieved from the servers with Analysis functionality, does not get stored in documents, the Analysis workbook property *Remove Data Before Saving* can be active in those workbooks. T

You can activate the property *Remove Data Before Saving* on the *Components* tab in the design panel.

Administrators can set the file setting *RemoveDataBeforeSaving* to true to activate the workbook property *Remove Data Before Saving* in the design panel by default.

The following procedure shows how you can save a workbook on a platform.

## Procedure

1. Choose **File > Analysis > Save Workbook**.

### i Note

If all platforms are enabled in your Analysis installation, you will find several icons for saving a workbook, one for each platform.

2. Log on to the selected platform and BW system.
3. Select the location where you want to save the workbook.
4. Enter a name and technical name for the workbook.
5. Optional step: Select the checkbox *Save as 1.x format*.

You can save workbooks with the format of former Analysis releases. Then you can open the workbook in a 1.x release. Note that all changes made with functions that are only available as of Analysis 2.0 (for example, Table Design formatting), are deleted if you save workbook as 1.x format.

Note that after saving a workbook in 2.x format, it is no longer possible to save the same workbook to 1.x format. If you want to save a 2.x workbook as 1.x again, you can delete the 2.x version with the context menu in the *Save Workbook* dialog and then save the workbook in 1.x format.

This option can only be enabled with administration rights. For more information, contact your IT administrator.

6. Press *Save*.  
The workbook is saved to the selected platform.

You can refresh the displayed document list by choosing *Refresh* in the context menu or pressing *F5*.

## Results

The workbook is saved. You and other users with access to the server can open and analyze it.

## Related Information

[Working with Analysis in Microsoft Excel \[page 11\]](#)

[The Components tab \[page 142\]](#)

[Platform Options \[page 303\]](#)

## 3.5 To rename a workbook

### Context

You can rename a workbook on the business intelligence platform and on a SAP BW platform.

### Procedure

1. Choose ► *File* ► *Analysis* ► *Open Workbook* ▾.

#### i Note

If all platforms are enabled in your Analysis installation, you will find several icons for opening a workbook: one for each platform.

2. Log on to the selected platform and BW system.
3. Select the workbook that you want to rename.
4. Open the context menu and select *Rename*.
5. Enter a new name for the workbook.  
The technical name of a workbook cannot be changed.

### Results

The workbook is renamed. You and other users with access to the server can open and analyze it again.

## 3.6 To delete a workbook

### Context

You can delete a workbook on the business intelligence platform and on a SAP BW platform.

### Procedure

1. Choose ► *File* ► *Analysis* ► *Open Workbook* ▾.

### **i** Note

If all platforms are enabled in your Analysis installation, you will find several icons for opening a workbook, one for each platform.

2. Log on to the selected platform and BW system.
3. Select the workbook that you want to delete.
4. Open the context menu and select *Delete*.

## **Results**

The workbook is deleted.

## **3.7 To protect a workbook or worksheet**

### **Context**

You can protect a workbook or a worksheet in Analysis. The concept of protection in Analysis allows you to perform Analysis OLAP navigation actions when a protection is defined.

You can protect either a workbook or a worksheet.

### **Procedure**

1. Choose **File > Analysis > Protect Workbook**.
2. You can protect the complete workbook or the current worksheet.  
To protect the workbook, select **Protect Workbook > Protect Workbook**.  
To protect a worksheet, select **Protect Workbook > Protect Current Sheet**.  
The *Workbook Protection* or *Worksheet Protection* dialog appears.
3. Enter the password for the protection twice: in the *Protection Password* field and in the *Confirm Password* field.
4. Select the activities users are allowed to execute in spite of the protection.  
If you select *Insert Rows* in the list, for example, all users are able to insert rows in the workbook or sheet. They don't need to unprotect the workbook or sheet.
5. Press *OK*.

The workbook or worksheet protection is created.

## Results

The workbook or worksheet is protected as specified.

To unprotect the workbook or worksheet again, enter the password.

If you use the standard Microsoft Excel functions to protect a worksheet or workbook, OLAP navigation will not be possible for these objects.

You cannot use the Analysis workbook protection and the MS Excel workbook protection for an object at the same time, for example a sheet in a workbook. But you can protect some sheets in a workbook with the Analysis workbook protection and other sheets in the same workbook with MS Excel workbook protection.

If you only protect sheets of type Non-COF in a workbook, the sheets of type Analysis in the workbook can be used with full functionality.

## 3.8 To launch Analysis from a BW system

### Context

You can launch Analysis from a BW system in the SAP GUI. Before launching, you can also select the data source that should be displayed in Analysis.

### Procedure

1. Log on to a BW system.
2. Open the Analysis Launcher.  
Execute transaction RAAOE to open the *Analysis Excel Launcher*.  
Execute transaction RAAOP to open the *Analysis PowerPoint Launcher*.
3. Select the *Type* you want to open with Analysis:  
The following types can be selected:
  - Client Only  
If you select this option, Analysis is opened and the connection information (system, client, language, user and password) is transferred to Analysis. Therefore the connected user can open data sources and workbooks (respectively presentations) without logging on again.
  - Document  
If you select this option, you can select a workbook or presentation in the next step.
  - Query  
If you select this option, you can select a query in the next step.

- Query View  
If you select this option, you can select a query view in the next step.
- 4. Select an object.  
You can use input help to select an object. Input help lists the objects available in the system depending on the object type selected.
- 5. Press *Execute*.

## Results

Analysis is opened, and the selected object is displayed. If you have selected type Client Only, no objects are displayed, but you can open a data source or workbook/presentation without logging on to the BW system again.


The BW system switches automatically back to the start menu.

## 3.9 To convert a BEx workbook

### Context

You can convert workbooks created with SAP BEx Analyzer 3.5 and SAP BEx Analyzer 7.0 to an Analysis workbook. In the Conversion options dialog, you can specify which BEx objects should be converted and you can also specify whether a conversion log should be created.

### Procedure

1. Choose **File > Analysis > Convert BEx Workbook**.
2. Select the SAP BW system and press *Next*.
3. Enter your user and password to log on to the system.
4. Select the workbook you want to convert.  
You can search for the name or technical name of the workbook or you can select it in a folder. BEx Analyzer workbooks have the following icon in the workbook list: .
5. Press *OK*.  
The conversion starts directly. Depending on your selection in the conversion options, the following objects will be converted:

BEx Analyzer object	Analysis object
Workbook	Workbook

<b>BEx Analyzer object</b>	<b>Analysis object</b>
Data Provider	Data Source
Item: Grid	Crosstab
Chart	Chart
Item: List Of Filters	Formulas: SAPGetDimensionEffectiveFilter and SAPGet-DimensionInfo
Item: Dropdown Box	Component: Filter with single member selection
Item: Checkbox Group	Component: Filter with multiple member selection
Item: Radio Button	Component: Filter with single member selection
Item: Text	Different info fields and formulas
Item: Message	Formula: SAPListOfMessages
Planning Objects used in item Button	Planning Objects
Styles	The style is not converted. The converted workbook is displayed with the default Analysis style set.
Item: List of Conditions	The list is not converted. The conditions are available as BEx conditions in Analysis and can be activated and deactivated with the menu.
Item: List of Exceptions	The list is not converted. The exceptions are available as Conditional Formatting in Analysis and can be activated and deactivated with the menu.
VBA API	Customer-specific code is not converted, but it is available in the workbook after the conversion.
Navigation Pane	No conversion. You can use the design panel in Analysis.
Item: Dropdown Box (displaying query views)	No conversion.
Item: Button	No conversion.

## Results

After the conversion, the converted workbook is opened in Analysis. You can save the newly created workbook to a platform and continue your analysis.

## Related Information

[Conversion log \[page 39\]](#)

[Conversion Options \[page 305\]](#)

## 3.9.1 Conversion log

The conversion log contains the conversion status for all objects in the BEx workbook. In the conversion options, you can specify whether a conversion log should be created during the conversion and whether it should be stored in a visible or hidden sheet in the workbook.

At the top of the conversion log, a message is displayed informing you whether conversion was successful. The ID of the BEx workbook and the selected conversion option are shown too. Possible selections are to convert all objects, data sources and crosstabs or data sources only.

The objects of the BEx workbook are then displayed with the conversion status in three lists: one for data providers, one for planning objects and one for items (all other objects on the workbook). The list contains the following columns:

- Status  
The status can be *Converted*, *Not Converted* or *Ignored*.
- Type  
The objects type is displayed. Examples are DATAPROVIDER, GRID or NAVIGATION\_PANE.
- Text  
The technical name of the object is displayed.
- Data Provider / Alias  
The alias of the data provider is displayed.
- Message  
A message is displayed if the conversion status is *Not Converted* or *Ignored*.

### Example

An object with status *Ignored* can have the following message: *Item not converted due to current conversion settings*.

- Sheet Range  
You can select the sheet range to go to the object in the workbook.

# 4 Designing Workbooks

After inserting a crosstab with data, you have various options to design your workbook in Analysis.

You can apply one of the SAP style sets to the workbook and you can also define new style sets. To illustrate the data, you can insert charts, dynamic and waterfall charts. If different users use the workbook, it could be helpful to add info fields with information on the data source and filter status.

You can use conditional formatting to highlight important values or unexpected results in your data. With Table Design, you can edit single elements in a crosstab. For example, you can add a new line to the crosstab. The changes made using Table Design persist after navigation steps that force a rebuild of the crosstab, like a refresh.

With grouping crosstabs, you can combine several crosstabs in the workbook to one crosstab. If you link dimensions in a workbook, an action on a linked dimension affects the same dimension in other data sources in the same way.

Furthermore, you can use BW workspaces, convert crosstab cells to formula, create Web Applications and smart copy/paste data sources.

For more sophisticated workbook design, the Analysis plug-in contains a dedicated set of functions in Microsoft Excel to access data and meta data of connected BW systems. There are also several API functions available that you can use with the Visual Basic Editor, to filter data and set values for BW variables.

## Related Information

[Analysis Design tab \[page 17\]](#)

[Inserting other components \[page 41\]](#)

[Defining Conditional Formatting \[page 44\]](#)

[Editing crosstabs with Table Design \[page 47\]](#)

[Defining style sets for crosstabs \[page 52\]](#)

[Grouping crosstabs \[page 58\]](#)

[Linking dimensions \[page 59\]](#)

[Inserting data sources using BW Workspaces \[page 60\]](#)

[Converting crosstab cells to formula \[page 64\]](#)

[Creating Web Applications \[page 66\]](#)

[Launching the Query Designer for editing a data source \[page 67\]](#)

[To smart copy/paste a data source \[page 68\]](#)

[To save a query view \[page 69\]](#)

[Working with formulas \[page 70\]](#)

[Working with macros \[page 94\]](#)

## 4.1 Inserting other components

In addition to crosstabs, you can add the following components to your analysis:

- [Charts](#) for providing a graphical representation of the data in the crosstab.  
You can insert a dynamic chart (using the Microsoft Excel functionality) or a waterfall chart.
- [Info fields](#) for providing metadata information
- [Filters](#) or providing simplified filtering for end users

### 4.1.1 To insert a dynamic chart

#### Procedure

1. Select a cell of the crosstab you want to visualize in a chart.  
By inserting a chart with Analysis, the data of the entire crosstab is visualized in the chart. If you want to visualize only a subset of the crosstab data, you can use Microsoft Excel functionality.  
If you change the data displayed in the crosstab, e.g. with a filter, the chart is adapted automatically.
2. Choose **► Chart > Insert Chart** in the ribbon.  
The chart is added to the analysis. You can position it in the worksheet using drag and drop.
3. Modify the chart.  
To modify the chart, you can use Microsoft Excel options for charts. For example, you can change the chart type or define a data range for the chart.
4. You can move the chart to another worksheet in the workbook.  
On the [Component](#) tab in the design panel, select the chart you want to move, and open the [Move to](#) dialog. Select the sheet that should contain the chart and press [OK](#).

#### Results

The chart is added to the analysis according to your configuration. The chart is updated automatically when you change the displayed data in the crosstab.

### 4.1.2 To insert a waterfall chart

#### Context

A waterfall chart is a specialized type of bar chart. Typically, it can be used to show how an initial value is affected by a series of intermediate positive or negative values. The start and end values are always displayed in a column as totals. The individual interim values are not displayed as subtotals, but as delta values.

In Analysis, the crosstab values of the first key figure are drawn into the waterfall chart cell by cell from top to bottom. Total rows in the crosstab are always ignored. The start and end column are displayed in gray color, negative delta values in red, and positive delta values in green.

You can also insert a waterfall chart for a query with hierarchies. The hierarchy nodes should be expanded upwards. Expanded nodes are displayed as columns and are also displayed in gray color.

To insert a meaningful waterfall chart, the crosstab view must meet specific requirements. You can define structures for a query in the BEx Query Designer to create queries that meet these requirements:

- Only one dimension should be in the view.
- The first data row should be the desired start column in the waterfall chart.
- The last data row should be the desired end column in the waterfall chart.
- The intermediate data rows should represent delta values showing the changes between the start and end column.

## Procedure

1. Select a cell of the crosstab that you want to visualize in a waterfall chart.

For the waterfall chart, the data of the first key figure in the crosstab will be used.

2. To insert a vertical waterfall chart, choose **Chart > Insert Waterfall Chart** from the ribbon.

To insert a horizontal waterfall chart, choose **Chart > Insert Pivoted Waterfall Chart**.

The chart is added to the analysis. You can position it in the worksheet using drag and drop.

If you change the data of the first key figure displayed in the crosstab, e.g. by expanding a hierarchy node, the chart is adapted automatically.

### i Note

You should not modify the waterfall chart using Microsoft Excel options for charts.

3. You can move the chart to another worksheet in the workbook.

On the *Component* tab in the design panel, select the chart you want to move, and open the *Move to* dialog. Select the sheet that should contain the chart and press *OK*.

## Results

The chart is added to the analysis according to your configuration. The chart is updated automatically when you change the displayed data in the crosstab.

## 4.1.3 To insert an info field

### Context

You can insert information fields to provide additional information on data displayed in the workbook sheets.

### Procedure

1. Select an empty cell where you want to place the info field.
2. Select the info field you want to insert.
  - Choose *Info Field* and one of the listed fields: Data Source Name, Last Data Update, Key Date, Effective Filters, Variables. If you want to insert other info fields, use the second option.  
The info field is added to worksheet. If you use more than one data source in your analysis, you are prompted to select a data source.  
For MultiProviders, two dates are inserted as Last Data Update. The date when all InfoProviders of the MultiProvider were updated successfully for the last time (LastDataUpdate) and the date when a single InfoProvider of the MultiProvider was updated for the last time (LastDataUpdateMaximum).
  - You can also drag and drop the info fields from the *Information* tab in the design panel to a cell in the worksheet.  
Select the data source on the top of the tab and drag and drop the information you want to insert as info field. For dynamic info fields (filters and variables), you have to use the first option.

### Results

The info fields are inserted with label and source information. The functions used for the formulas are SAPGetInfoLabel and SAPGetSourceInfo. The formulas are created automatically.

## 4.1.4 To insert a filter

### Context

You can insert a filter component to your analysis to simplify the filtering. This helps you to quickly change the view of the displayed data, for example to different periods of time.

### Procedure

1. Select an empty cell where you want to place the filter component.

2. Choose *Filter* and select one of the listed dimensions to insert a filter component for this dimension. The dimension name and a filter component formula are inserted in the worksheet. The functions used for the formulas are SAPGetDimensionInfo and SAPSetFilterComponent. The formulas are created automatically.
3. Optional Step: Specify the filter component formula.

The formula that is inserted automatically, allows the user to select multiple members for filtering. It looks like this: =SAPSetFilterComponent("DS\_2"; "OCALYEAR";"ALL").

You can add one of the following parameters to the formula: SINGLE, MULTIPLE, LOWERBOUNDARY, UPPERBOUNDARY, MEMBERSELECTOR, MEMBERSELECTORBYRANGE to specify the filtering options.

If you add the parameter SINGLE, the user can only select one member for filtering. The formula looks like this: =SAPSetFilterComponent("DS\_2"; "OCALYEAR";"ALL";"SINGLE").

You can also insert filter components to enable a range selection. Insert two filter components for the same dimension and add to one the parameter LOWERBOUNDARY and to the other the parameter UPPERBOUNDARY. You can now filter for the lower and upper bounds of a range.

If you add the parameter MEMBERSELECTOR or MEMBERSELECTORBYRANGE, the standard filter dialog with individual or range selection is used for the filter component.
4. Optional step: Format the filter component.

You can use the formatting options of Microsoft Excel to format cells of the filter component.
5. Select the filter icon to define a filter.

All tables on the current sheet that contain this dimension, will be filtered according to the selected filter. On the *Components* tab in the design panel, you can define which tables should be affected if not all tables should be filtered accordingly.

## Results

The filter is added to the analysis according to your configuration.

## Related Information

[SAPGetDimensionInfo \[page 76\]](#)

[SAPSetFilterComponent \[page 93\]](#)

## 4.2 Defining Conditional Formatting

You can use conditional formatting to highlight important values or unexpected results in your data.

With conditional formatting, you define rules to select different or critical values in a crosstab. Results that fall outside a set of predetermined threshold values are highlighted in color or designated with symbols. This enables you to identify immediately any results that deviate from the expected results. For example, you directly see answers to questions, such as which products have over 10% revenue increase this year or which regions have sales volume higher than a defined amount.

You can add one or more rules to a conditional format and prioritize the application of the rules in your crosstab. You can also define more than one conditional format in a worksheet and decide for each of them when it should be active by toggling it on and off. Existing conditional formats can be edited or deleted.

### **i** Note

The conditional formatting described in this guide is part of the Analysis plug-in. There is also a conditional formatting available in Microsoft Excel.


## Related Information

[To define a conditional format \[page 45\]](#)

[To edit conditional formats \[page 46\]](#)

## 4.2.1 To define a conditional format

### Procedure

1. Choose **► Conditional Formatting ► New** .  
The *New Conditional Formatting* dialog box appears.
2. In the *Name* field, enter a name for the new conditional format.
3. In the *Based on* list, select the measure you want to apply the conditional format to.
4. In the *Format* list, select the style you want to use for highlighting values.

The following styles are available:

- *Background*: the cells background is colored.
- *Values*: the text of the cells is colored.
- *Status Symbol*: a symbol is displayed in the cells.
- *Trend Ascending*: a colored arrow (green to red) is displayed in the cells.
- *Trend Descending*: a colored arrow (red to green) is displayed in the cells.
- *Trend Grey*: a grey arrow is displayed in the cells.

Every style consists of nine members. The numbers 1 to 9 represent the priority of the rule.

5. Define one or more rules on the *Definition* tab.
  - a. Select the priority.

In the first list, select one of the nine priorities of the style selected above. If you define more than one rule in a conditional format, and the rules overlap for values in your crosstab, the priority determines which rule is applied to these cells. You should therefore use a priority only for one rule in a conditional format. Priority one is the highest priority and priority 9 the lowest.
  - b. Select an operator.

You can choose one of the following operators: Between, Outside, Greater Than, Greater Than Or Equal To, Less Than, Less Than Or Equal To and Equal To.
  - c. Enter a threshold value or value range.

d. Press *Add*.

The rule is added to the *Rules* area.

6. Optional step: Define the selection for the conditional format.

On the *Selection* tab, you can define for each dimension if the conditional formatting should be applied to *All Members and Totals* or only to the *Members* or *Totals*. The default value is All Members and Totals.

7. Optional step: Define the visualization on the *Display* tab.

Under *Apply conditional formatting to:*, you can define which parts of the crosstab are affected by conditional formatting: Data Cells, Row Headers and/or Column Headers.

Under *Apply Visualization to Another Measure:*, you can specify that the conditional format defined for one measure (sales volume for example), is visualized in the data of another measure (sales order items for example). To use this function, select the *Applied to* check box.

8. Press *OK* to apply your rules to the crosstab.

## Results

You have created one or more conditional formats. They are saved with the workbook and are available to other users or in other sessions.

If you change the scaling factor for the measure, the rules are changed automatically to apply as before. Rules created with an Analysis version < 2.5 will not change automatically, only after editing a rule once.

## 4.2.2 To edit conditional formats

### Context

Once you have defined conditional formats, you can open and change them, delete them or activate and deactivate them:

### Procedure

1. Choose ► *Conditional Formatting* ► *[Name of Conditional Formatting]* ▾.
2. Select one of the following options:
  - Select *Edit* if you want to change the current definitions of a conditional format. You can add new rules, change existing rules, , or delete existing ones.

To change existing rules, select the *Change* icon in the rule that you want to change, change the definition of the rule and select the *Apply* icon to change the rule.

To delete existing rules, select the *Remove* icon in the rule that you want to delete.

To apply the changed rule(s) to your data, press *OK*.
  - Select *Delete* if you want to delete a conditional format.

- Select or deselect *Active* if you want to toggle the formatting on and off.

## 4.3 Editing crosstabs with Table Design

Table Design enables you to edit single elements in a crosstab. For example, you can add a new line to the crosstab. The changes made using Table Design persist after navigation steps that force a rebuild of the crosstab, like a refresh or swapping axes.

You can edit crosstabs with the following Table Design options:

- Applying formats to single cells, rows and columns
  - Adding new lines (rows and columns)
  - Adding texts in new member cells
  - Adding formulas in new member and data cells
- You can also add formulas to input-enabled planning data cells.

You can manage existing Table Design modifications (rules) on the *Design Rules* tab in the design panel. You can also copy the rules from one data source to another in the design panel.

Using the Table Design API, you can create and apply Table Design rules with API methods.

You can also remove all Table Design rules manually by choosing *Reset Data Source* in the context menu. If Table Design rules are available for the data source, a pop-up will ask if you want to keep the rules. With *Yes*, the rules will be available after the reset. With *No*, the rules will be removed.

### i Note

There are some limitation when working with grouped crosstabs. For more information, see SAP Note [2631644](#).

## Related Information

[The Design Rules tab \[page 147\]](#)

[Table Design API \[page 117\]](#)

## 4.3.1 To apply formats

### Context

You can apply defined cell styles as format to single cells, rows and columns. You can use and select standard Microsoft Excel cell styles, SAP cell styles and user-defined cell styles.

You cannot apply Table Design formats if Pause Refresh is active.

### Procedure

1. Select a cell in the crosstab.

Multiple cells cannot be selected.

2. Select *Format cells* in the Format group in the ribbon.
3. In the Format dialog, select the cell range the new format should be applied to.

For a selected dimension header, you have the following options:

- Dimension Header  
The format will be applied to the selected cell.
- Dimension Header and All Members  
The format will be applied to the selected cell and all members of the dimension.
- All Members  
The format will be applied to all dimension members of the selected cell, but not to the selected dimension header cell.

For a selected dimension member, you have the following options:

- Dimension Member  
The format will be applied to the selected cell. A dimension member in a crosstab can be a member of a dimension or a measure header.
- Member and all Dependants on axis  
The format will be applied to the selected cell and all its members on the same axis.
- All Dependants on the Axis for this Member  
The format will be applied to all dimension members of the selected cell, but not to the selected dimension member cell.

For a selected data cell, you have the following option:

- Data Cell  
The format will be applied to the selected cell.

4. Select the format that should be applied to the defined cells.
5. Select *OK*.

## Results

The format is applied to the selected cell range in the crosstab. If a cell is affected by more than one format, the last defined format is visible.

You can change an applied format by selecting the cell and **► Format cells ► <defined format> ► Change/Delete ►**.

The new format is also added as a rule of type *Formats* to the *Design Rules* tab in the design panel. In the design panel, you can activate and deactivate the new rule. You can also delete the rule.

## Related Information

[The Design Rules tab \[page 147\]](#)

## 4.3.2 To add new lines

### Context

You can insert new lines, rows and columns, into a crosstab. These new lines can be used as layout elements to visually separate report areas. The style information is taken from the triggered context. You can adapt column width, row height and formats for new lines.

New lines can be inserted to the following crosstab elements:

- next to the header section  
The header section comprises the dimension and measures at the left border of a crosstab before the data cell area. You can insert a row below the header and a column beside the header.
- before and after every dimension
- before and after every dimension member  
If a dimension member appears several times in a crosstab, the new line is added to each occurrence.
- before and after the members of a dimension (dimension group)
- before and after a combination of members of different dimensions
- before and after every measure
- before and after a result line

The new line is connected to the selected element. If you add a new line after a dimension member and you change your analysis so that the member is no longer part of the crosstab, the new line will also no longer be visible. But it is not deleted. The new line will be displayed again if the dimension member is again in the crosstab.

If you connect a new line to all members of a dimension (dimension group), you can remove members from the dimension in the crosstab and the new line will still be in the crosstab as it is connected to the dimension group

New lines cannot be inserted next to a data cell.

## Procedure

1. Select a cell in the crosstab where you want to add a new line.
2. Select *New Lines* and the option where you want to locate the new line in the Format group in the ribbon.

## Results

The new line is inserted into the crosstab. The context menu for cells in new lines is restricted to table design functions.

To delete a new line, select the added line and **►► *New Lines* ► *Delete* ▾** in the Format group.

The new line is also added as a rule of type *New Lines* to the *Design Rules* tab in the design panel. In the design panel, you can activate and deactivate the new rule. You can also delete the rule.

## Related Information

[The Design Rules tab \[page 147\]](#)

### 4.3.3 To add texts in new member cells

## Context

You can add texts to member cells of new inserted lines.

The display of the new text corresponds to the currently selected display in the crosstab. If the display *Text* is selected for the dimension, the new text will also be created with display *Text*. If you change the display in the crosstab to *Key*, no text will be displayed in the new line. You can add again a new text that will then be created with display *Key*.

## Procedure

1. Select a member cell in the crosstab where you want to add a text.

2. Enter the text manually.

## Results

The text is added to the cell in the crosstab.

The inserted text is added as a rule of type *Texts* to the *Design Rules* tab in the design panel. In the design panel, you can activate and deactivate the new rule. You can also delete the rule.

## Related Information

[The Design Rules tab \[page 147\]](#)

### 4.3.4 To add formulas

## Context

You can add formulas to data and member cells of new inserted lines.

You can also add formulas to a data cell of an input-ready planning data source in change mode.

## Procedure

1. Select a cell in the crosstab where you want to add a formula
2. Enter a function.

Enter **=SUM** to create a sum, for example.

3. Select the cells that you want to use in the function

You can select single cells or a range of cells.

You can also select cells outside the crosstab or use Microsoft Excel reference functions like VLOOKUP.

If you select a range of cells, you can remove one or several members that are in the range. The formula will then use the data of the members that are still available in the crosstab.

If you select single cells and you change your analysis later so that one or several of the selected cells are no longer displayed in the crosstab, the formula will return NA.

4. Optional step: Copy the formula to adjacent cells by using Microsoft Excel *Fill Handling*.

Select the cell(s) that you want to use as a basis for filling additional cells, and then drag the fill handle across or down the cells that you want to fill.

The fill handling works with relative references and absolute references (\$).

5. Press *Return*.

## Results

The formula is calculated and added to the cell in the crosstab.

The formula is added as a rule of type *Formulas* to the *Design Rules* tab in the design panel. In the design panel, you can activate and deactivate the new rule. You can change the rule and delete it. You can edit the formula definition and the formula location. For example, if for a dimension text and key are displayed in two columns, you can change the location from text to key or vice versa.

Formulas added to planning data cells are not editable in the design panel.

## Related Information

[The Design Rules tab \[page 147\]](#)

### 4.3.5 Changing Table Design rules

All single actions you have done with Table Design as applying formats, adding new lines, text or formulas are creating Table Design rules. These rules are shown on the tab *Design Rules* in the Design Panel.

In the design panel, the rules are listed by rule type *New Lines*, *Formats*, *Formulas* and *Text*. You can edit the single rules by selecting a rule and changing the details in the *Properties* section. Whereas the properties for formats and new lines are the same as in the corresponding dialogs for creating these rules, the properties for formulas offer more detailed options for redefining the formulas.

## Related Information

[The Design Rules tab \[page 147\]](#)

## 4.4 Defining style sets for crosstabs

A style set is a collection of Microsoft Excel cell styles that is applied by Analysis to format the cells of a crosstab. Whenever you insert a new crosstab in a workbook, the styles in the current default style set are used

to format the crosstab cells. You can change the applied style set in your analysis. With Analysis, the following style sets and their cell styles are installed:

- SAP Blue (default)
- SAP Black&White
- SAP Tradeshow Plus
- SAP Blue Crystal

By modifying the cell styles of these style sets, you can create your own style sets and share them with other users.

## 4.4.1 SAP cell styles

### SAP standard styles

SAP standard styles are available after the installation. You can modify them in the *Styles* group on the *Home* tab in Microsoft Excel. They affect the formatting as described in the following table:

Style Name	Description
SAPDimensionCell	Format for dimension header cells.
SAPMemberCell	Format for member cells (non-hierarchical dimensions).
SAPHierarchyCell	Format for hierarchical member cells (even levels 0, 2, ...).
SAPHierarchyOddCell	Format for hierarchical member cells (odd levels 1, 3, ...).
SAPMemberTotalCell	Format for member total cells.
SAPDataCell	Format for data cells.
SAPDataRemoved	Format for empty crosstab cells whose data was removed as the workbook property <i>Remove Data Before Saving</i> is selected.
SAPDataTotalCell	Format for data total cells.
SAPExceptionLevel1-9	Format for highlighted cells due to conditional formats (rule priorities 1-9).
SAPEmphasized	Format for highlighted data cells (as per query definition).
SAPEmphasizedTotal	Format for highlighted total cells (as per query definition).
SAPBorder	Format for borders around a crosstab and between header/member and data cells (format for left border is taken).  After changing this style, for example the color, you have to execute <i>Refresh All</i> in the Analysis ribbon to apply the changes.
SAPMessageText	Format for the message that is displayed whenever a crosstab cannot be rendered and displayed in the sheet.

Style Name	Description
SAPFormula	Format for inserted single cell fomulas: Insert Filter and Insert Info Field (Data Source Name, Last Data Update and Key Date).  The fomate is not applied for array formulas (Insert Info Field for Effective Filters or Variables) and for fomulas inserted directly into the sheet (manual or as MS Excel function).
SAPGroupingFillCell	Format for header fill cells when grouping crosstabs.
SAPEditableDataCell	Format for input-ready data cells in planning change mode.
SAPEditableDataTotalCell	Format for input-ready total cells in planning change mode.
SAPReadOnlyDataCell	Format for read-only data cells in planning change mode.
SAPReadOnlyDataTotalCell	Format for read-only total cells in planning change mode.
SAPLockedDataCell	Format for locked data cells in planning change mode.
SAPLockedDataTotalCell	Format for locked total cells in planning change mode.

## SAP custom styles

The following SAP custom styles are not availabe after the installation, but you can create them in the [Styles](#) group on the [Home](#) tab of Microsoft Excel. They affect the formatting as described in the following table:

Style Name	Description
SAPMemberCellX	Format for member cells on columns (overriding SAPMemberCell).
SAPMemberTotalCellX	Format for member total cells on columns (overriding SAPMemberTotalCell).
SAPHierarchyCellX	Format for hierarchical member cells on columns, even levels (overriding SAPHierarchyCell).
SAPHierarchyOddCellX	Format for hierarchical member cells on columns, odd level (overriding SAPHierarchyOddCell).
SAPHierarchyCell0-9	Format for hierarchical member cells on specific level (overriding SAPHierarchyCell and SAPHierarchyOddCell).
SAPHierarchyCell0-9X	Format for hierarchical member cells on specific level on columns (overriding SAPHierarchyCellX and SAPHierarchyOddCellX).

### Example: SAPMemberCellX

The column headings are defined as SAPMemberCell. If you want a different format for these cells than for member cells in rows, you can duplicate the SAPMemberCell, name it SAPMemberCellX and change the

format definition. If you save this as style set, the member cells in column headings are displayed in the new defined format. The member cells in rows continue to be displayed as defined in the SAPMemberCell style.

## Example: Styles for Hierarchy Levels

The standard styles for hierarchies are SAPHierarchyCell and SAPHierarchyCellOdd. With these styles, the hierarchy levels have alternating styles. If you want to have three different styles for the first three hierarchy levels of an hierarchy, create the styles SAPHierarchyCell0, SAPHierarchyCell1 and SAPHierarchyCell2. The system ignores the standard styles and applies the styles for specific styles. If an hierarchy have more than three levels, the levels after level 2 are displayed with the SAPHierarchyCell2 style.

## 4.4.2 To apply a style set

### Context

You can apply one of the SAP style sets or any new defined style set to a workbook.

### Procedure

1. Choose **► Styles ► Apply Style Set... ▾**.  
The *Apply Style Set* dialog box appears.
2. In the list box, select the style set you want to apply.
3. Select the *Set as Default* check box if the style set should be applied as default in your workbooks. The default style set is used when you open a new workbook and insert a data source.
4. Press *OK*.

### Results

The style set is applied to all crosstabs in your workbook.

## 4.4.3 To create cell styles and style sets

### Context

Based on available cell styles, you can define a new style set. You can also define new cell styles and use them in a style set.

In Analysis, you can create your own cell styles according to your needs using the Microsoft Excel style functionality. After the creation, these styles are available in the *Custom* section of the styles list. You can then save these new defined styles in a style set. You can also use the user-defined cell styles for editing crosstabs with Table Design.

## Procedure

1. On the *Home* tab, in the *Styles* group, choose *Cell Styles*.  
The available cell styles are listed.
2. Modify the existing cell styles or create new cell styles according to your needs with the Microsoft Excel style functionality.
3. On the *Analysis Design* tab, in the *Format* group, choose ► *Styles* ► *Save Style Set...* ▾.  
If there are user-defined cell styles available, the *Add to Style Set* dialog appears.  
In the list, you can select the cell styles that you want to add to the style set. Press *Okay* and the *Save Style Set* dialog box appears.  
If there are no user-defined cell styles available, the *Save Style Set* dialog box appears directly.
4. Enter a *Style Set Name*.
5. Select the *Set as Default* check box if the style set should be applied as default in your workbooks. The default style set is used when you open a new workbook and insert a data source.
6. Press *OK*.

## Results

The new defined style set is created and available in the list of style sets that can be applied to a workbook.

## Related Information

[Editing crosstabs with Table Design \[page 47\]](#)

### 4.4.4 To share a style set

#### Context

You can share a style set with other users by exporting the style set to a local fileshare. Other users can import the style set and use it for the analysis.

## Procedure

1. Apply the style set that you want to export.
2. Choose ► *Styles* ► *Export Style Set...* ►.
3. Save the style set as XML format.

The XML file contains the cell styles of the three SAP style sets and your currently applied style set.

4. Choose ► *Styles* ► *Import Style Set...* ►.
5. Select a style file from the server and press *Open*.
6. Save the imported styles as new style set.

## Results

You have exported a style set to be used by other users and / or you have imported a style set to use it in your analysis.

## 4.4.5 To delete a style set

### Context

You can delete all user-defined style sets. The standard SAP style set that is installed with the Add-In can not be deleted.

## Procedure

1. Choose ► *Styles* ► *Delete User Style Set* ►.  
The *Delete User Style Set* dialog box appears.
2. In the list box, select the style set you want to delete.
3. Press *OK*.

## Results

The style set is deleted and no longer available in the list of style sets that can be applied to a workbook.

## 4.5 Grouping crosstabs

### Context

You can stick two crosstabs together so that they look and behave as they would be one crosstab. Therefore you have to group these crosstabs.

With grouping crosstabs, they are moved close together that there is no space in between. Furthermore the row axis is building the common axis of this grouped crosstab. The crosstab from which you trigger the grouping is defined as master and determines the structure of the common crosstab, whereas the dependent crosstab is providing its column axis.

From a user interaction point, the grouped crosstab behaves as one entity. The functionality provided within ribbon and context menu is adjusted to the unified behavior of the grouped crosstab. For example:

- You can only move dimensions to or within the master crosstab.
- Filtering dimensions is only possible for dimensions of the master crosstab.
- If you use or change the display in the master crosstab to *Compact Display in Rows*, the dependent crosstabs behave accordingly.  
If you undo the grouping, the dependent crosstabs will be set back to the behavior before the grouping. If you ungroup the crosstabs, the dependent crosstabs will have the behavior of the grouped crosstab.

Only the rows axis can build the common axis. The crosstabs needs to have a similar row structure so that grouping the crosstabs is feasible.

You can group several crosstabs.

#### i Note

There are some limitations when working with grouped crosstabs. For more information, see SAP Note [2631644](#).

### Procedure

1. Insert at least two data sources.
2. Select a dimension which is on the rows axis and choose **► Combine ► Group Crosstab ►** on the *Analysis Design* tab.

### Related Information

[SAPMoveDimension \[page 109\]](#)

[SAPSetFilter \[page 111\]](#)

## 4.6 Linking dimensions

### Context

You use several data sources in a workbook and you want that an action on a dimension affects the same dimension in other data sources in the same way. For example, you want to have this dimension to be filtered by the same member or have the same hierarchy drill operation.

Therefore, you need to link these dimensions.

With the next action, the linked dimensions behave simultaneously, especially with regards to filter and hierarchy navigation.

Note that you can only link dimensions for data sources that:

- are located in exactly the same system, and not only the same system type, like SAP BW or SAP Analytics Cloud.
- have the same client in case of SAP BW systems.
- have the same port in case of SAP HANA systems.

### Procedure

1. Select a dimension in the crosstab.
2. Choose **Combine > Link Dimension > <dimension of data source you want to link to>** on the *Analysis Design* tab.

### Results

The dimensions are linked.

On the Analysis tab in the design panel, you can recognize linked dimensions by an icon that is displayed next to the dimension name.

You can undo the linking by choosing **Combine > Link Dimension > <dimension of data source you link to>**.

## 4.7 Inserting data sources using BW Workspaces

A BW workspace is a special area in which new models can be created based on a central data provider from the BW system and local data. The aim of workspaces is to bridge the gap between the central requirements and the flexibility required locally.

Queries are usually based on InfoProviders. If you need to add new fields, you have to change these InfoProviders in the BW back-end. BW workspaces enable you to react quickly to new analytical requirements. This means you can add fields to a query without changing the objects in the BW system.

BW workspaces are usually created by the central IT department making a set of InfoProviders available in a workspace and assigning the authorizations for workspaces to business users. By uploading local data, you can create a local provider in a workspace. In a CompositeProvider, you can connect your own data in the local provider with BW data in the workspace. By reloading data to a local provider, the data in the local provider can be updated. This allows you to react quickly to new requirements. The central InfoProviders stored on the BW server expose their data to a workspace. The data in these providers is updated with the newest server data with every refresh.

You can insert all providers of a workspace as a data source in Analysis. They are available in the workspace view on the folder tab when you select a data source in a BW system.

### i Note

To create and work with BW workspaces, you need to use a SAP Business Warehouse Accelerator (BWA) or a SAP HANA database.

For more information about creating BW workspaces and authorizations for BW workspaces, see the SAP BW documentation at [BW Workspace](#).

## Related Information

[To create a local provider \[page 60\]](#)

[To reload data in a local provider \[page 62\]](#)

[To create a CompositeProvider \[page 62\]](#)

### 4.7.1 To create a local provider

#### Context

You can upload local data to create a local provider. The data in the local provider can then be analyzed with Analysis and you can use the local provider to create a CompositeProvider.

## Procedure

1. Open the Microsoft Excel file containing the data you want to upload to a local provider.
2. Select the cells you want to upload.

The entries in the first row of the selected range will be defined as column names automatically. You can change these column names in the local provider settings.
3. Select *Create* in the *Build* group on the *Analysis Design* tab.
4. Log on to the BW system.

The new local provider will be added to a workspace. Log on to the BW system containing the workspace where you want to add the local provider.
5. Select the workspace and press *Next*.

The Details dialog appears.
6. Enter a name for the local provider in the *Description* field (optional step).
7. Enter a technical name for the local provider in the *Technical Name* field (optional step).

The first part is predefined by the system containing the elements @3 and the workspace prefix.
8. In the *Column Definition* section, you can define the following settings:
  - Column Name  
The column name is predefined with the entries in the first row of the uploaded range. You can change the name here. If you change the name, you should also change the entries in your Excel source file to be able to reload data later.
  - Measure  
Select the check box if the column contains measure data.  
A default entry for the measure is made. It can however be changed manually.
  - Type  
For dimensions, select one of the following types: Time, Date, Dimension, Text (default), Text with Leading Zeros.  
For measures, select one of the following types: Integer, Decimal, Floating Point Number, Measure.  
A default entry for the type is made. It can however be changed manually.
  - Type Details  
If you want to use this local provider to create a CompositeProvider, you can select the InfoProvider object of the workspace where the uploaded data should be linked to. The selection is possible for type Dimension and Measure. For dimensions, you get a list of all dimensions in the workspace to select an object. For measures, the list contains all measures available in the workspace.
9. Press *Next*.

In the *Target Location* dialog, you can specify whether the new local provider should be inserted into the current workbook sheet or a new workbook sheet after being created. You can also select that the new local provider is not inserted in a sheet after the creation.
10. Press *OK* to create the local provider.

## Results

The local provider is created in the workspace. You can add it to a workbook as a data source and analyze the data with Analysis.

## 4.7.2 To reload data in a local provider

### Context

You can reload data to a local provider to update the data in the provider.

### Procedure

1. Open the Microsoft Excel file containing the data you want to upload to a local provider.
2. Select the cells you want to reload.  
The selected range must contain the same number of columns you used for creating the local provider. The column titles must also be identical.
3. Select *Reload* in the *Build* group on the *Analysis Design* tab.
4. Log on to the BW system.  
Log on to the BW system containing the local provider you want to reload.
5. Select the workspace and press *Next*.
6. Select the local provider you want to reload.  
In the *Data Source Details* section, you see the defined settings for the local provider. You cannot change the settings when reloading data.  
If the columns in the selected local provider do not match the selected columns for reloading, a message is displayed.
7. Press *Next*.  
In the *Target Location* dialog, you can specify whether the reloaded local provider should be inserted into the current workbook sheet or a new workbook sheet after reloading. You can also specify that it is not inserted into a sheet after reloading.
8. Press *OK* to reload the local provider.

### Results

The local provider is updated with the reloaded data.

## 4.7.3 To create a CompositeProvider

### Context

In a CompositeProvider, you can combine all the data that you want to view in your query. You can use all central BW InfoProviders assigned to the workspace as data providers, as well as your own data assigned to the workspace as local providers.

## Procedure

1. Insert the query that you want to enhance with a local provider.  
The query must belong to the same workspace as the local provider.
2. Select a cell of the crosstab.
3. Choose **► Add ► Dimension ►** or **► Add ► Data Records ►** in the *Build* group on the *Analysis Design* tab.

### i Note

If you want to add measure data to a query, we recommend adding it as data records.

The *Define Composite Provider Settings* dialog appears.

4. Select a local provider from the list.  
You can specify whether the local providers for all dimensions in the query should be listed, or just for one dimension that you select in the drop down list.  
If you select a local provider, the data source details are displayed.
5. Press *Next*.  
In the *Details* dialog, you can enter a description and technical name for the new CompositeProvider and the new data source (composite provider query).  
The technical name of the composite provider has the same predefined prefix as the local provider. The technical name of the composite provider query starts with the predefined prefix containing the elements \$ and the workspace prefix.  
If you add a local provider to a composite provider, you cannot change the descriptions and technical names. These can only be assigned when the first local provider is added to a query.
6. Press *Next*.  
In the *Target Location* dialog, you can specify whether the CompositeProvider should be inserted into the current workbook sheet or a new one. You can also specify that it should not be inserted into a sheet.
7. Press *OK* to create the CompositeProvider.

### i Note

You can use the graphical view or the table view in the BW Workspace Designer to check the field connections.

For more information on the BW Workspace Designer, see the SAP NetWeaver BW documentation at [BW Workspace Designer](#).

## Results

The composite provider is created in the workspace. You can insert it as a data source into a workbook and analyze the data. You can also add new dimension or data records to the composite provider.

## 4.8 Converting crosstab cells to formula

You can convert all cells of a crosstab into formulas with one step. This deletes the crosstab object and defines every row in the table as a Microsoft Excel formula. The result values called from the server with the formula are still displayed in the table. The formula of the selected cell is displayed in the formula bar. In formula mode, you can edit the analysis table using Microsoft Excel formatting and formula functions and make further calculations using the existing data.

In formula mode, you can use all Microsoft Excel formatting functions. With the deletion of the design item, the individual formatting of the data will not be overwritten by the standard formatting in the crosstab the next time you update this data. For example, if you select a color to highlight interim results in the table and then navigate in this table, only the data for the values from the server is called and not the standard formatting from the crosstab. Your individual formatting is retained.

You can use the Microsoft Excel formula functions to make further calculations on the basis of existing data. You can also copy the formula for a cell to another cell outside the table and thus work independently of the original table. If the workbook contains two crosstabs based on different data providers, you can combine the data from both data providers for your calculations.

### The Formulas

Formulas with the following functions are composed in the formula mode:

- SAPGetData
- SAPGetMember
- SAPGetDimensionInfo
- SAPGetUniformScaling

### Examples for working in formula mode

In formula mode, you can use various functions to modify the layout and perform additional calculations.

- You can highlight cells by formatting the font and background color.
- You can insert spaces to make the display easier to read.
- You can copy parts of the table or individual cells to another position in the workbook in order to compare particular values.
- You can re-use cells.
- You can overwrite a members with another one, or add one in order to call data that you need from the BI server. If member "3.2007" is used to read the sales revenue for March 2007, for example, you can replace the 3 with a 4, thus using member "4.2007" to obtain the sales revenue for April 2007, provided that the data provider contains this data.
- You can also calculate additional subtotals.
- You can create offers based on data from various data providers.

## Restrictions

Converting to formula mode has the following consequences:

- Navigation using Drag & Drop is no longer possible.
- The context menu is not available.

## Related Information

[To convert a crosstab to formula \[page 65\]](#)

### 4.8.1 To convert a crosstab to formula

#### Procedure

1. Insert a crosstab into a workbook
2. Choose *Convert to Formula*.

If the icon *Convert to Formula* is not displayed in the ribbon Tools group, you can enable it in the *Customize User Interface* dialog.

Converting to formula performs the following steps:

- Texts that are not displayed because they occur several times in a column or row, are repeated in each cell automatically to produce valid formulas. You can also execute this step manually by selecting the *Repeat Members* check box for the crosstab on the *Components* tab in the design panel.
- Every cell in the crosstab is defined as a Microsoft Excel formula.

#### **i** Note

All currently displayed cells of the crosstab are converted to formula. Cells in a hierarchy that are currently not expanded, are not converted.

- For dimensions and members displayed as text in the crosstab, the key is added to the data source during conversion to formula. This doesn't change the display in the original crosstab. You will only see the added key if you insert the crosstab with the same data source again in your workbook. Then columns and rows are added to display the key.
- Crosstab object is deleted.

## Results

All currently displayed cells of the crosstab are converted to a formula using the functions *SAPGetData*, *SAPGetMember*, *SAPGetDimensionInfo* and *SAPGetUniformScaling*.

### i Note

As long as you haven't changed the data in the table, you can go back to analysis mode by choosing [Undo](#).

## Related Information

[Customizing the User Interface \[page 24\]](#)

## 4.9 Creating Web Applications

### Context

You can transfer components in an Analysis workbook to SAP Lumira / SAP BusinessObjects Design Studio to create a web application. These tools enable application designers to create analysis applications and dashboards for browsers and mobile devices.

In Analysis, you can use methods that are not available in Lumira / design studio. For example, you can use calculations, conditional formatting, and exceptions to change a data source. You may then want to create a web application to use this changed data source in Lumira / the design studio.

The following Analysis components can be transferred:

- Crosstabs
- Charts  
The following chart types are enabled: Pie, Line, Column, Bar, Surface, Radar, Bubble, and Scatter.
- Filters

### Procedure

1. Ensure that Lumira / design studio is closed.
2. In Analysis, select the workbook sheet with the components that you want to transfer to Lumira / design studio.
3. Choose [Create Web Application](#) in the ribbon.  
If the icon is not displayed in the ribbon Tools group, you can enable it in the [Customize User Interface](#) dialog.  
The 'Logon' dialog for Lumira / design studio appears.
4. Enter your logon data and choose [OK](#) to log on to Lumira / design studio.  
Lumira or Design Studio is opened and the components of the workbook sheet are added to the template.

## Results

You can now modify the web application with application design:

For more information about working with SAP Lumira, see the Application Designer guide at [SAP Lumira](#)

For more information about working with the Design Studio, see the Application Designer guide at [SAP BusinessObjects Design Studio](#).

## Related Information

[Customizing the User Interface \[page 24\]](#)

# 4.10 Launching the Query Designer for editing a data source

## Context

Directly from Analysis, you can access the BEx Query Designer or the BW Modeling Tools for queries or the SAP HANA editor for HANA calculation views to open a data source that has been inserted in the current workbook.

You can use this function to check the query definition and to change the definition according to your needs. After saving the changes, you can update the data source in the workbook immediately.

Prerequisites for using this function with SAP BW:

- You use a BEx query as data source.
- BEx Query Designer with minimum release 7.20 SP10 or 7.30 SP3 is installed on your client PC.
- BW Modeling Tools with minimum release 7.40 SP16 or 7.50 SP5 (See SAP note 2343003).

Prerequisites for using this function with BW4HANA:

- BW Modeling Tools with minimum release 7.40 SP16 or 7.50 SP5 (See SAP note 2343003 and 2246699).

Prerequisites for using this function with SAP HANA:

- You use a HANA calculation view as data source.
- You use a SAP HANA Web-based Development Workbench editor with minimum release SPS 10.

## Procedure

1. Select a crosstab cell.

2. Choose *Call Query Designer* in the ribbon.  
If the icon is not displayed in the ribbon Tools group, you can enable it in the *Customize User Interface* dialog.  
The Logon dialog for the BEx Query Designer, the BW Modeling Tools or the SAP HANA system appears.
3. Enter your logon data and choose *OK* to log on.  
The query editor is opened and the data source selected in the workbook is already open and ready to be edited.
4. Edit the query definition and save your changes.
5. In Analysis, select a crosstab cell and choose *Reset Data Source* in the context menu to display the changed data source.

## Results

The data is displayed corresponding to the new initial state defined in the BEx Query Designer, BW Modeling Tools or SAP HANA Web-based Development Workbench editor.

## Related Information

[Customizing the User Interface \[page 24\]](#)

## 4.11 To smart copy/paste a data source

### Prerequisites

You have inserted a data source in Analysis for Microsoft Excel.

### Context

You can copy a data source (crosstab or chart) that is currently displayed in the workbook. You can then insert the copied data source in Analysis for Microsoft Excel and Analysis for Microsoft PowerPoint.

## Procedure

1. In Analysis for Microsoft Excel, select the crosstab or chart that should be copied.
2. Choose *Smart Copy*.
3. Select an empty cell in the sheet and choose *Smart Paste*.  
The copied data source is inserted as crosstab.

## Results

The current navigation state of the copied data source is displayed, but it exists independently from the origin. Navigation steps in the original object do not affect the pasted crosstab and vice versa.

Note that most of the properties managed on the Components tab in the design panel, are specified as not copy-relevant. That means that the properties for the copied data source will have the default values and not the values of the original object. The only exception is the number of new lines for planning.

In Analysis for Microsoft PowerPoint, you can paste a data source as crosstab or chart.

## Related Information

[Managing components in the design panel \[page 150\]](#)

[To smart paste a data source \[page 223\]](#)

## 4.12 To save a query view

### Context

You can save a navigation state of a data source as a query view in a BW system.

### Procedure

1. Select a cell of the crosstab.
2. Choose *Save View*.  
The Save Query View dialog box appears. If there are already query views based on the same query, they are listed in the dialog box.

### i Note

If you are connected to a BI platform server, the view will be saved as analysis view on the BI platform.

3. Enter a name and a description.
4. Select *Save Variable Values* if the values should be saved with the query view.  
This option is only selectable if the data source contains variables.
5. Press *Save* to save the query view.

## Results

The query view is saved on the same server as the data source.

You can open the query view with *Insert Data Source*.

Note that most of the properties managed on the Components tab in the design panel are specified as not view-relevant. That means that the properties for the saved view will have the default values and not the values of the original object. The only exception is the number of new lines for planning.

You can delete a query view with the context menu in the *Save Query View* dialog. The system checks if the query view is used in other objects saved on the same server. If the query view is used in an object saved on another server, no message appears.

## Related Information

[To save an analysis view \[page 225\]](#)

## 4.13 Working with formulas

In Analysis for Microsoft Office, you can use the standard functions of Microsoft Excel to build formulas. The Plug-in also contains an own set of functions that you can use to build formulas. You can use these functions to include data and meta data of used data sources into your analysis. For example, you can insert information fields on data source properties, display the measure filter or list the variables of a data source. With the SAPGetData function, you can also define measure values for certain member combinations.

A Microsoft Excel formula for Analysis consists of a function and references to the data source, measures and/or dimensions. You can use the text or the key of an object to use it as reference. You can also use a cell value like B10 as reference.

The formula alias of a data source is displayed and can be changed in the data source properties on the *Components* tab in the design panel. For measures, dimensions and their members text references are better to read, but if you want to create a multi-language enabled analysis or there are duplicate texts in the meta data of your data source, you should reference these objects with their keys.

You can also use these functions in VBA macros.

## Analysis functions

The following functions are available in the Analysis category:

- SAPGetData
- SAPGetDimensionDynamicFilter
- SAPGetDimensionEffectiveFilter
- SAPGetDimensionInfo
- SAPGetDimensionStaticFilter
- SAPGetDisplayedMeasures
- SAPGetInfoLabel
- SAPGetMeasureFilter
- SAPGetMeasureInfo
- SAPGetMember
- SAPGetSourceInfo
- SAPGetUniformScaling
- SAPGetVariable
- SAPGetWorkbookInfo
- SAPListOf
- SAPListOfDesignRules
- SAPListOfDimensions
- SAPListOfDynamicFilters
- SAPListOfEffectiveFilters
- SAPListOfMeasureFilters
- SAPListOfMembers
- SAPListOfMessages
- SAPListOfStaticFilters
- SAPListOfVariables
- SAPSelectMember
- SAPSetData
- SAPSetFilterComponent

## Related Information

[Using Analysis functions \[page 123\]](#)

### 4.13.1 To create a formula

#### Context

To create a formula with Analysis functions:

## Procedure

1. Select the cell in which you want to enter the formula.
2. To start the formula with a function, press the *Insert Function* button on the formula bar.  
The *Insert Function* dialog box appears.
3. Select *Analysis* in the *Select a category* box.
4. Select a function.
5. Press *OK*.  
The *Function Arguments* dialog box appears.
6. Enter the arguments.  
To enter cell references as an argument, press the *Collapse Dialog* button (which temporarily hides the dialog box), select the cells on the worksheet, and then press the *Expand Dialog* button.
7. When you complete the formula, press *OK*.

## 4.13.2 Working in formula-optimized mode

You can open data sources in a formula-optimized mode. This allows the creation of highly flexible and formatted reports.

You can use all data sources connected to Analysis (SAP BW, SAP HANA and SAP Analytics Cloud). The data sources must have at least one key figure structure and they can contain restrictions, restricted key figures, calculated key figures and hierarchies.

To open a data source in formula-optimized mode:

1. Select a cell in the sheet and choose ► *Data Source* ► *Select Data Source for Defining Formulas...* ▾.
2. Log on to a system and open a data source.

When you open a data source in formula-optimized mode, no crosstab is displayed. Therefore, the Analysis tab in the design panel does not show the Columns and Rows sections. You can use the sheet to define formulas.

On the Components tab in the design panel, you can use the property *Use Data Source Formula-Optimized* to switch between the formula-optimized mode and the analysis mode.

In the recently used list, different icons are used for data sources opened for analysis and data source opened in formula-optimized mode. You can open the data source directly from the list in the desired mode.

## Tips

If you encounter any issue, you can check the following to solve it:

- Switch off parallel formula execution in Microsoft Excel.  
Choose ► *File* ► *Options* ► *Advanced* ► *Formulas* ▾ and deselect *Enable mutli-threaded calculation*.
- If the member value is a number, check if formatting it as text can avoid the issue.  
You can format it as text by adding an apostrophe before the number, for example.

## Examples

The following examples show the handling of SAPGetData and SAPSelectMember formulas in the formula mode. You can also reference cells and use the cell content to define measures and restrictions on dimensions in formulas.

### Reading the total for a measure

Cell B3: =SAPGetData("DS\_1";"Sales Revenue")

The total for measure Sales Revenue is displayed in cell B3.

### Restricting one dimensions

Cell B3: =SAPGetData("DS\_1";"Sales Revenue";"OCountry";"France")

The total for measure Sales Revenue for France is displayed in cell B3.

### Restricting multiple dimensions

Cell B3: =SAPGetData("DS\_1";"Sales Revenue";"OCountry";"France";"OYear";"2017")

The total for measure Sales Revenue for country France and the year 2017 is displayed in cell B3.

### Adding a Filter icon to select members

Cell B3: =SAPSelectMember("DS\_1";"";"OCountry";"TEXT";"Full")

The Filter icon is displayed next to cell B3. You can select it to open the filter dialog for dimension OCountry and select members of the dimension. The members will be displayed as text and the full version of the filter dialog will be opened.

## Related Information

[SAPGetData \[page 74\]](#)

[SAPSelectMember \[page 91\]](#)

## 4.13.3 SAPGetData

This function returns the measure value for a specific dimension member combination.

The formula can return values only for member combinations that are part of the current navigation state of the data source. To be part of the navigation state, the member combinations must be used in rows or columns. If you filter a dimension, you can return values only for member combinations contained in the filter.

For example, if the navigation state of the data source displays the dimension 'Region' in rows and the measure 'Sales Volume' in columns, you can create a formula to return a value for a particular region, but you cannot return a value for a special customer, even if customer information is available in the data source. To be able to return values for a special customer, you have to add the dimension to the navigation state.

Note that dimensions you have defined as background filter, cannot be used in member combinations for this formula. However, they can have an influence on the returned measure value.

The formula consists of at least three parameters and is made up of the following arguments:

- **Data Source**  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- **Measure**  
Enter the name of measure, for example **"Incoming Orders"**.
- **Member combination**  
There are three methods for entering the member combination:
  - Enter one parameter as member combination, for example **"Region=France;Product=Services"**.
  - Enter several parameters as the member combination, for example **"Region";"France";"Product";"Services"**. This method can only be entered manually. It is recommended for member combinations that use cell references.
  - Enter a mixture of both: one parameter per dimension/member combination, for example **"Region=France";"Product=Services"**. This method is used for converting to a formula.

You can use the function SAPSelectMember to simplify the syntax for the formula. When referring to cells using SAPSelectMember, the meta data context is defined, so it is sufficient to only refer to one particular cell. The simplified syntax looks like this: SAPGetData(<data source alias>; <measure>; <cell reference with SAPSelectMember>). Example: H20=SAPGetData("DS\_1";"OAmount";\$B1).

### Example: 3 Parameters Formula

Cell H20: =SAPGetData("DS\_1";"Incoming Orders";"Region=France;Product=Services")

The data for the value in this cell comes from data source DS\_1. The name of the measure is 'Incoming Orders'. The member combination is 'France' and 'Services'. The formula in cell H20 therefore uses the data from DS\_1 to calculate the incoming orders for region 'France' and product 'Services'. If you change 'France' to 'Germany' in the formula, the incoming orders for Germany and Services are displayed in cell H20.

## Example: >3 Parameters Formula with Cell Reference

Cell H20: =SAPGetData("DS\_1";"Incoming Orders";"Region";B10;"Product";"Services")

The data for the value in this cell comes from data source DS\_1. The name of the measure is 'Incoming Orders'. The member combination is the region entered in cell B10 and 'Services'. For example, if you enter 'Spain' in cell B10, the formula in cell H20 uses the data from DS\_1 to calculate the incoming orders for region 'Spain' and product 'Services'. If you change 'Spain' to 'France' in cell B10, the incoming orders for 'France' and 'Services' are displayed in cell H20.

## Related Information

[SAPSelectMember \[page 91\]](#)

## 4.13.4 SAPGetDimensionDynamicFilter

This function returns the dynamic filter of a dimension. Dynamic filters are defined by the user.

The formula consists of three parameters and is made up of the following arguments:

- *Data Source*  
Enter the formula alias of the data source or planning function. You can set the alias when configuring the data source or planning function on the *Components* tab in the design panel.
- *Dimension*  
Enter the technical name of the dimension.
- *Member Display*  
You can enter **TEXT**, **KEY**, or **INPUT\_STRING** to define how the filtered members should be displayed in the workbook.  
You can use **INPUT\_STRING\_AS\_ARRAY** if the formula is called from VBA to get the input string as a string array.

## Example

Cell F20: =SAPGetDimensionDynamicFilter("DS\_1";"ODIVISION";"TEXT")

You add a filter for dimension ODIVISION and the following members are displayed in the analysis: Paints, Lighting, Foods. If you enter the formula in cell F20, the three filtered members are displayed in cell F20 as text.

## 4.13.5 SAPGetDimensionEffectiveFilter

This function returns all effective filters of a dimension: dynamic filters defined by the user, static filters defined in the underlying source, and filters by measure defined for the selected dimension.

The formula consists of three parameters and is made up of the following arguments:

- **Data Source**  
Enter the formula alias of the data source or planning function. You can set the alias when configuring the data source or planning function on the *Components* tab in the design panel.
- **Dimension**  
Enter the technical name of the dimension.
- **Member Display**  
You can enter **TEXT**, **KEY**, or **INPUT\_STRING** to define how the filtered members should be displayed in the workbook.  
You can use **INPUT\_STRING\_AS\_ARRAY** if the formula is called from VBA to get the input string as a string array.

## Example

Cell F20: =SAPGetDimensionEffectiveFilter("DS\_1";"ODIVISION";"TEXT")

If you enter the formula in cell F20, the members of ODIVISION that are currently filtered by the user, the static filters that are defined in the data source, and the filters by measure for this dimension are displayed in cell F20 as text. If no static filters are defined for the data source, only the dynamic filter members and filters by measure are displayed.

### 4.13.6 SAPGetDimensionInfo

This function returns the name of a dimension or the name of an active hierarchy.

The formula consists of 3 parameters and is made up of the following arguments:

- **Data Source**  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- **Dimension**  
Enter the technical name of the dimension.
- **Property Name**  
You can enter the following property names:
  - NAME
  - ACTIVEHIERARCHY
  - ATTRIBUTE
  - ISCHARACTERISTICSTRUCTURE  
This property returns the value True if the dimension is a structure containing dimension members. If this is not the case, it returns the value False.
  - ISMEASURESTRUCTURE  
This property returns the value True if the dimension is a structure containing measures. If this is not the case, it returns the value False.

## Example

Cell F20: =SAPGetDimensionInfo("DS\_1";"ODIVISION";"NAME")

If you enter the formula in cell F20, the name of dimension ODIVISION is displayed in cell F20.

## 4.13.7 SAPGetDimensionStaticFilter

This function returns the static filter of a dimension. Static filters are defined in the underlying source and cannot be changed by the user.

The formula consists of three parameters and is made up of the following arguments:

- *Data Source*  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- *Dimension*  
Enter the technical name of the dimension.
- *Member Display*  
You can enter **TEXT**, **KEY**, or **INPUT\_STRING** to define how the filtered members should be displayed in the workbook.  
You can use **INPUT\_STRING\_AS\_ARRAY** if the formula is called from VBA to get the input string as a string array.

## Example

Cell F20: =SAPGetDimensionStaticFilter("DS\_1";"OMATERIAL";"KEY")

If you enter the formula in cell F20, the static filter of dimension OMATERIAL is displayed in cell F20.

## 4.13.8 SAPGetDisplayedMeasures

This function returns a list of all measures displayed in the analysis as text.

The formula is made up of the following argument: *Data Source*.

Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.

## Example

Cell G10: =SAPGetDisplayedMeasures("DS\_1")

If you enter the formula in cell G10, all measures that are currently displayed in the crosstab are listed in cell G10. If you add or remove a measure from the crosstab, the list in cell G10 is updated accordingly.

## 4.13.9 SAPGetInfoLabel

This function returns the language-dependant label for an info field. The property names correspond to the info fields that are available for workbook and data sources on the *Information* tab in the design panel. Using this function, the info field labels are displayed in the selected UI language. The info field values can be inserted with function SAPGetWorkbookInfo and SAPGetSourceInfo.

The formula is made up of the following argument: *Property Name*.

For workbook related info fields, you can enter the following property names:

- WorkbookName
- CreatedBy
- CreatedAt
- LastChangedAt
- LastRefreshedAt
- LogonUser

For data source related info fields, you can enter the following property names:

- DataSourceName
- LastDataUpdate
- LastDataUpdateMaximum

This property is related to MultiProviders. It returns the date of the last update of all InfoProviders in a MultiProvider.

Example: A MultiProvider contains three InfoProviders. The date when all three InfoProviders were updated succesfully for the last time, is the LastDataUpdateMaximum.

- KeyDate
- QueryTechName
- QueryCreatedBy
- QueryLastChangedBy
- QueryLastChangedAt
- QueryLastRefreshedAt
- InfoProviderTechName
- InfoProviderName
- System
- LogonUser

### Example

Cell D20: =SAPGetInfoLabel("System")

The label of the info field is displayed in the selected UI language, for example in English: System.

## 4.13.10 SAPGetMeasureFilter

This function returns a list of filtered measures created with Filter by Measure (All Dimensions Independently).

### i Note

As of Analysis 1.4 SP7, you can use the function SAPListOfMeasureFilters to get a list of the all filtered measures. This function also returns more detailed information and includes BEx conditions.

The formula is made up of the following argument: *Data Source*.

Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.

### Example

Cell G10: =SAPGetMeasureFilter("DS\_1")

If you enter the formula in cell G10, all measures that have a filter definition and the corresponding rules are displayed in a list in cell G10. If you add or remove a filter to a measure, the list in cell G10 is updated accordingly.

### Related Information

[SAPListOfMeasureFilters \[page 87\]](#)

## 4.13.11 SAPGetMeasureInfo

This function returns information about a measure. It can return the information if a measure is restricted, the selection behind a restricted measure or the info objects that is represented by this measure.

The formula consists of 3 parameters and is made up of the following arguments:

- *Data Source*  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- *Measure*  
Enter the technical name of the measure. By default, that is an unreadable GUID that is available in the properties view on the *Analysis* tab in the design panel. If a technical name was defined in the query designer, this name can also be used.
- *Property Name*  
You can enter the following property names:
  - ISRESTRICTED

This property returns the value `True` if the measure is a restricted measure. If the measure is not restricted, the property returns the value `False`.

- **SELECTION**  
This property returns the selection behind a restricted measure. If there is no selection behind the measure, an empty string is returned.
- **BASEINFOBJECT**  
This property returns the info object that is represented by this measure.

## Example

Cell F20: `=SAPGetMeasureInfo("DS_1";"TN_IncomingOrders";"ISRESTRICTED")`

If you enter the formula in cell F20, the return value `True` is displayed in cell F20 if the measure `TN_IncomingOrders` is restricted. If it is not restricted, the return value in cell F20 is `False`.

## 4.13.12 SAPGetMember

This function returns the dimension member or attribute.

The formula can only return values for dimension members or attributes that are part of the current navigation state of the data source. To be part of the navigation state, the members must be used in rows, columns or as background filter. If you filter a dimension, you can only return values for members that the filter contains.

The formula consists of 3 parameters and is made up of the following arguments:

- **Data Source**  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- **Dimension Member**  
Enter the technical name of a dimension and assign a member key, for example `"ODIVISION=R1"`.
- **Member Display**  
You can enter **TEXT, KEY** or **Internal\_Key** to define how the filtered members should be displayed in the workbook.

## Example

Cell G15: `=SAPGetMember("DS_1";"ODIVISION=R1";"TEXT")`

You want to display the text for member Retail. The key for Retail is R1. If you enter the formula in cell G15, the text of member R1 (Retail) is displayed in cell G15.

## 4.13.13 SAPGetSourceInfo

This function returns an info field value for a data source. The info field label can be inserted using the SAPGetInfoLabel function. The property names correspond to the info field values that are available for data sources on the *Information* tab in the design panel.

The formula consists of 2 parameters and is made up of the following arguments:

- **Data Source**  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- **Property Name**  
You can enter the following property names:
  - DataSourceName
  - LastDataUpdate
  - LastDataUpdateMaximum  
This property is related to MultiProviders.  
For MultiProviders, the property LastDataUpdate returns the date of the last update of all InfoProviders in the MultiProvider.  
The property LastDataUpdateMaximum returns the date of the last update of a single InfoProvider of the MultiProvider.  
Example: A MultiProvider contains three InfoProviders. The date when all three InfoProviders were updated successfully for the last time, is the LastDataUpdate. The date when a single InfoProvider was updated for the last time, is the LastDataUpdateMaximum.
  - KeyDate
  - QueryTechName
  - QueryCreatedBy
  - QueryLastChangedBy
  - QueryLastChangedAt
  - QueryLastRefreshedAt
  - InfoProviderTechName
  - InfoProviderName
  - System
  - LogonUser

### Example

Cell D20: =SAPGetInfoLabel("DataSourceName")

Cell E20: =SAPGetSourceInfo("DS\_1";"DataSourceName")

In cell D20, the **Data Source Name** label is displayed. In cell E20, the name of the data source with alias DS\_1 is displayed, for example **Sales Volume Europe**.

## 4.13.14 SAPGetUniformScaling

This function returns the unit/scaling information for a specific dimension member combination.

The formula can only return values for member combinations that are part of the current navigation state of the data source. To be part of the navigation state, the member combinations must be used in rows, columns, or as a background filter. If you filter a dimension, you can return values only for member combinations contained in the filter.

The formula consists of at least three parameters and is made up of the following arguments:

- *Data Source*  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- *Measure*  
Enter the name of measure, for example **"Sales Volume"**.
- *Member combination*  
There are two methods for entering the member combination:
  - Enter one parameter as member combination, for example **"Region=France;Product=IT"**. This method is used for converting to a formula.
  - Enter several parameters as the member combination, for example **"Region";"France";"Product";"IT"**. This method can only be entered manually. It is recommended for member combinations that use cell references.

### Example: 3 Parameters Formula

Cell H20: =SAPGetUniformScaling("DS\_1";"Sales";"Region=France;Product=IT")

The data for the value in this cell comes from data source DS\_1. The name of the measure is 'Sales Volume'. The member combination is 'France' and 'IT'. The formula in cell H20 therefore uses the data from DS\_1 to calculate the unit/scaling factor for the sales volume for region 'France' and 'IT' (e.g. EUR). If you change 'France' to 'USA' in the formula, the unit/scaling factor for the sales volume of 'USA' and 'IT' is displayed in cell H20 (e.g. USD).

## 4.13.15 SAPGetVariable

This function returns the value of the specified property for a specific variable.

The formula consists of three parameters and is made up of the following arguments:

- *Data Source*  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- *Variable Name*  
Enter the name or technical name of the variable.
- *Property Name*  
You can enter the following property names:

- VALUE  
If you enter VALUE, the current value of the variable is displayed.
- VALUEASKEY  
If you enter VALUEASKEY, the current value of the variable is displayed as a key.
- VALUEASTEXT  
If you enter VALUEASTEXT, the current value of the variable is displayed as text.
- INPUT\_STRING  
If you enter INPUT\_STRING, the current value of the variable is displayed as an input string.
- INPUT\_STRING\_AS\_ARRAY  
This property name can only be used if the formula is called from VBA. It returns the input string as a string array.
- DESCRIPTION  
If you enter DESCRIPTION, the variable name is displayed.
- ISINPUTENABLED  
If you enter ISINPUTENABLED, the function checks if the variable is input-enabled.
- ISMANDATORY  
If you enter ISMANDATORY, the function checks if the variable is mandatory.
- TECHNICALNAME  
If you enter TECHNICALNAME, the technical name of the variable is displayed.
- BASEINFOBJECT  
If you enter BASEINFOBJECT, the technical name of the dimension the variable is based on is displayed.

## Example

Cell F20: =SAPGetVariable("DS\_2";"OBW\_VAR";"DESCRIPTION")

If you enter the formula in cell F20, the name of variable OBW\_VAR is displayed in cell F20.

## 4.13.16 SAPGetWorkbookInfo

This function returns an info field value for the current workbook. The info field label can be inserted with the SAPGetInfoLabel function. The property names correspond to the info field values that are available for workbooks on the *Information* tab in the design panel.

This formula is made up of the following argument: *Property Name*.

You can enter the following property names:

- WorkbookName
- WorkbookDescription
- CreatedBy
- CreatedAt
- LastChangedAt
- LastRefreshedAt

- LogonUser
- PlanningEnvironment
- PlanningModel

## Example

Cell D20: =SAPGetInfoLabel("WorkbookName")

Cell E20: =SAPGetWorkbookInfo("WorkbookDescription")

In cell D20, the **Workbook Name** label is displayed. In cell E20, the description name used for saving the workbook is displayed, for example **Sales in Europe**.

## 4.13.17 SAPListOf

This function returns a list of available objects for a specific object type. The list contains two columns for every entry (Alias and Description).

This formula consists of 3 parameters and is made up of the following arguments:

- *Object Type*  
You can enter the following object types:
  - DATASOURCES
  - CROSSTABS  
For object type CROSSTABS, the function returns a list with three columns (Alias, Description and data source Alias).
  - PLANNINGFUNCTIONS
  - PLANNINGSEQUENCES
  - CUSTOMAPPLCONTEXT  
For object type CUSTOMAPPLCONTEXT, the function returns a list with three columns (SystemID, RowCount and Content).
- *Search*  
You can enter a search string. Note that the search is case sensitive. If you do not enter a search string, all available objects of the selected object type will be listed.
- *Search in*  
You can search in ALIAS or DESCRIPTION. This parameter is optional. The default value is ALIAS.

## Example

Cell G15: =SAPListOf("CROSSTABS";"Cross")

Starting in Cell G15, a list with three columns (crosstab Alias, crosstab Description, data source alias) is displayed with all crosstabs that contain the search string Cross in their Alias.

## Related Information

[Using BAdI information \[page 126\]](#)

### 4.13.18 SAPListOfDesignRules

This function returns a list of design rules. Depending on the search entries, the list can display all design rules for a data source or the design rules for one defined rule type.

This formula consists of 2 parameters and is made up of the following arguments:

- **Data Source**  
Enter the formula alias of the data source or planning function. You can set the alias when configuring the data source or planning function on the *Components* tab in the design panel.
- **Rule Type**  
You can enter one of the following rule types to define which rules should be displayed:
  - ALL  
This is the default rule type.
  - NewLine
  - NewDimensionLine
  - NewMemberLine
  - NewHeaderLine
  - Format
  - LineSize
  - Text
  - Formula

### Example

Cell G11: =SAPListOfDesignRules("DS\_1";"NewLine")

Starting in Cell G11, a list with all rules for new lines for data source DS\_1 is displayed.

### 4.13.19 SAPListOfDimensions

This function returns a list of dimensions. Depending on the search entries, the list can display all dimensions with their current drill state, dimensions that match a defined search string with their current drill state or just the dimensions used in rows, columns or background filter.

For planning functions, the function returns a list of all dimensions defined on the aggregation level of the planning function. The axis type is always ROW.

This formula consists of 3 parameters and is made up of the following arguments:

- **Data Source**  
Enter the formula alias of the data source or planning function. You can set the alias when configuring the data source or planning function on the *Components* tab in the design panel.
- **Search In**  
You can determine which columns the search should be executed in by selecting TECHNICALNAME, DESCRIPTION or AXIS. The default selection is TECHNICALNAME.
- **Search**  
If you have selected TECHNICALNAME or DESCRIPTION, you can enter a search string. Note that the search is case sensitive. If you do not enter a search string, all available dimensions will be listed. If you have selected AXIS, you can use the axis type to define which dimensions should be displayed:
  - Enter ROWS to list the dimensions that are used in rows in the current analysis.
  - Enter COLUMNS to list the dimensions that are used in columns in the current analysis.
  - Enter FILTER to list the dimensions that are used as background filter in the current analysis.
  - Enter ALL to list all available dimensions with their current drill state. For dimensions that are not used in the current analysis, the drill state column is empty.

If the selected axis type contains no dimensions in the current analysis, the list is empty.

## Example

Cell G11: =SAPListOfDimensions("DS\_1";"TECHNICALNAME";"OSOLD")

Starting in Cell G11, a list with three columns (Technical Name, Description, Drill state) is displayed with all dimensions that contain OSOLD in their technical name.

## 4.13.20 SAPListOfDynamicFilters

This function returns a list of all dynamic filters of a data source. Dynamic filters are defined by the user using *Filter by Member*.

The formula consists of 2 parameters and is made up of the following arguments:

- **Data Source**  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- **Member Display**  
You can enter **TEXT**, **KEY** or **INPUT\_STRING** to define how the filtered members should be displayed in the workbook.

## Example

Cell G15: =SAPListOfDynamicFilters("DS\_1";"TEXT")

If you enter the formula in cell G15, the dynamic filters of data source DS\_1 are displayed as text.

## 4.13.21 SAPListOfEffectiveFilters

This function returns a list of all effective filters of a data source: Dynamic filters defined by the user, static filters defined in the underlying source, and filters by measure defined for the data source dimensions.

The formula consists of 2 parameters and is made up of the following arguments:

- **Data Source**  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- **Member Display**  
You can enter **TEXT**, **KEY** or **INPUT\_STRING** to define how the filtered members should be displayed in the workbook. This parameter is optional. The current display selected for members is the default setting. If two options exist, the first one is applied.

### Example

Cell F20: =SAPListOfEffectiveFilters("DS\_1";"TEXT")

You have added dimension Region from data source DS\_1 to your analysis. You filter this dimension and the following members are part of the analysis: California, Arizona, Florida, Nevada. If you enter the formula in cell F20, the name of the dimension is displayed in cell F20, and the four filtered members are listed as text in cell G20.

## 4.13.22 SAPListOfMeasureFilters

This function returns a list of all measure filters including BEx conditions. The list contains two columns: the first column lists the filter type (BEX, DIMENSION, MEASURE) and the second with filter values.

The formula consist of three parameters and is made up of the following arguments:

- **Data Source**  
Enter the formula alias of the data source or planning object. You can set the alias when configuring the data source or planning object on the *Components* tab in the design panel.
- **Display**  
You can enter ALL, DIMENSION, MEASURE or BEX to define which measure filters should be displayed in the workbook:
  - ALL returns a list of all measure filters.
  - DIMENSION returns a list of filter by measure on all dimensions or a defined one.
  - MEASURE returns a list of filter by measure on data cells.  
The list also returns the filter type: All Dimensions Independently, Most Detailed Dimension in Rows or Most Detailed Dimension in Columns.
  - BEX returns a list of BEx conditions.  
This parameter is optional. ALL is the default value. You can also combine values of this parameter.
- **Name**

Enter the technical name of the dimension if you have selected DIMENSION in the display parameter. This parameter is optional. If you do not enter a dimension name, all dimensions are listed.

## Example

Cell G20: =SAPListOfMeasureFilters("DS\_1";"BEX")

Starting in cell G20, a list is displayed with all BEX conditions in data source DS\_1.

### 4.13.23 SAPListOfMembers

This function returns a list of members for a selected dimension or prompt. Each list entry contains the key, text, and key as filter string of a member. The key as filter string differs from the key if the member is compounded to another dimension.

You cannot use this function to list the members for a dimension with hierarchies.

The formula consists of five parameters and is made up of the following arguments:

- **Data Source**  
Enter the formula alias of the data source or planning object. You can set the alias when configuring the data source or planning object on the *Components* tab in the design panel.  
You have to define this parameter, irrespective of whether the variables in the workbook are merged or not.
- **Source**  
Enter one of the following sources:
  - PROMPT to get a list of the prompt members.  
If you use PROMPT as source, you have to define the data source alias, irrespective of whether the variables in the workbook are merged or not.
  - FILTER to get a list of the dimension members.
  - FILTER\_MASTERDATA to get a list of all members of the master data for the selected dimension.
  - PLAN\_PARAMETER to get a list of all members for a variable in a planning object.
- **Name**  
Enter the technical name of a flat dimension or prompt whose members should be listed.
- **Search**  
You can enter a search string to restrict the list of members. This parameter is optional.
- **Search in**  
Enter whether the search should be executed in the KEY or TEXT column. This parameter is optional. The default entry is KEY.
- **Count**  
You can specify (restrict or enlarge) the number of members that can be returned by this formula. The number specified in user option *Maximum Number of Members Displayed in the Filter Dialog* is used as default.

## Example

Cell G15: =SAPListOfMembers("DS\_1";"FILTER";"OCALMONTH";"12";"KEY")

Starting in cell G15, a list is displayed with all members of dimension OCALMONTH that contain search string '12' in their technical name (KEY).

### 4.13.24 SAPListOfMessages

This function includes messages into the sheet. You can select which messages, according to their level of severity, and details should be displayed in the sheet.

The formula consists of 2 parameters and is made up of the following arguments:

- *Severity*

You can enter the following severities: INFORMATION, WARNING, ERROR and CRITICAL.

INFORMATION is the least severe category, CRITICAL the most severe. If you select a severity, all messages are shown which have this severity or higher. If you select WARNING, for example, all messages with severity WARNING, ERROR and CRITICAL are displayed.

- *Show Details*

If you enter FALSE, the messages are listed with severity and short text.

If you enter TRUE, the messages are listed with additional information. The details are listed in the following sequence:

1. Message ID

The message ID is a unique identifier for a message that is valid for the current messages dialog. If the same message is displayed again at a later time, it has a different message ID.

The message ID can be used as identifier in the API methods SAPAddMessage and SAPSuppressMessage.

2. Short Text

The message short text is displayed.

3. Message Class

If the message comes from a BW back-end system, the message class is displayed.

4. ID

For Analysis messages, the notification ID is displayed. For messages from a BW back-end, a concatenation of the message class and the message number is displayed.

5. Severity

The message severity is displayed.

6. Variables: VARIABLE1-VARIABLE4

If the message comes from a BW back-end system, the message can comprise up to four variables to support dynamic message texts.

7. System ID

If the message comes from a BW back-end system, the system ID is displayed.

## Example

Cell H11: =SAPListOfMessages("ERROR";"FALSE")

In cell H11, all messages with severity ERROR and CRITICAL are displayed with severity and text.

## Related Information

[Using Analysis functions \[page 123\]](#)

[SAPAddMessage \[page 95\]](#)

[SAPSuppressMessage \[page 116\]](#)

### 4.13.25 SAPListOfStaticFilters

This function returns a list of all static filters of a data source. Static filters are defined in the underlying source and cannot be changed by the user.

The formula consists of 2 parameters and is made up of the following arguments:

- *Data Source*  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- *Member Display*  
You can enter **TEXT**, **KEY** or **INPUT\_STRING** to define how the filtered members should be displayed in the workbook.

### Example

Cell G15: =SAPListOfStaticFilters("DS\_1","TEXT")

If you enter the formula in cell G15, the static filters of data source DS\_1 are displayed as text.

### 4.13.26 SAPListOfVariables

This function returns a list of all variables of a data source or planning object.

The formula consists of 3 parameters and is made up of the following arguments:

- *Data Source*  
Enter the formula alias of the data source or planning object. You can set the alias when configuring the data source or planning object on the *Components* tab in the design panel.  
You have to define this parameter if the variables in the workbook are not merged. If the variables in the workbook are merged, you can define this parameter but you do not have to define it.
- *Member Display*  
You can enter **TEXT**, **KEY** or **INPUT\_STRING** to define how the filtered members should be displayed in the workbook.

This parameter is optional. The current display selected for members is the default setting. If two options exist, the first one is applied.

- *Display*

Enter one of the following values:

- ALL to display all variables (filled and unfilled) including variables not visible on the prompts dialog.
- PROMPTS to display all variables (filled and unfilled) visible on the prompts dialog.
- ALL\_FILLED to display all filled variables including variables not visible on the prompts dialog.
- PROMPTS\_FILLED to display all filled variables visible on the prompts dialog.
- PLAN\_PARAMETER to display all variables (filled and unfilled) of a planning object.

This parameter is optional. The default value is ALL\_FILLED.

## Example

Cell F20: =SAPListOfVariables("DS\_2";"TEXT")

If you enter the formula in cell F20, all variables with values in data source DS\_2 are listed with their values in the worksheet. The first variable name is displayed in cell F20, the next in cell F21 and so on. The corresponding values are listed in G20, G21 and so on.

## 4.13.27 SAPSelectMember

This function returns a member of a dimension.

You can use the returned value to simplify the syntax for function SAPGetData.

The formula consists of five parameters and is made up of the following arguments:

- *Data Source*  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- *Member*  
Enter the name, technical name or generated member (by value help) of a dimension member.
- *Dimension*  
Enter the dimension name or technical name.
- *Member Display*  
You can enter TEXT, KEY or INPUT\_STRING to define how the members should be displayed in the workbook.
- *Selection Mode*  
With the selection mode, you can define which filter dialog (as icon) is selectable next to the cell. The options are: NONE, SIMPLE or FULL. Using NONE, no filter icon will be selectable next to the cell. Using SIMPLE, a simple version of the filter dialog will be opened where you can select other members of the dimension. Using FULL, the full version of the filter dialog will be opened where you can also search for members and change the display (from Text to Key, for example). The default value is SIMPLE. This parameter is optional.

## Example

Cell D10: =SAPSelectMember("DS\_1";"France";"OCOUNTRY";"TEXT";"SIMPLE")

The formula in cell D10 selects the member of a dimension of data source DS\_1. The name of the dimension is OCountry. The name of the member France. The value (France) that is set is selected and displayed in cell D10. With selection mode SIMPLE, a filter icon is available next to cell D10 that you can use to open the filter dialog and select another member of dimension OCOUNTRY.

## Related Information

[SAPGetData \[page 74\]](#)

### 4.13.28 SAPSetData

With this function, you can set planning values. You can either enter the value(s) in the formula directly or you can refer to another cell in Microsoft Excel that contains the value(s).

The formula consists of five parameters and is made up of the following arguments:

- **Data Source**  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- **New Value**  
You can enter a value manually or reference a cell in Microsoft Excel that contains the value.

#### **i** Note

If you reference a cell, the cell must not be part of a crosstab in Analysis.

- **Option**  
You can use options to define the behavior of the formula. If you use more than one option in a formula, use a semicolon-separated list to enter the options.  
This parameter is optional. If you do not need to define an option for the formula, you can enter an empty parameter.  
The following options are available:
  - **SetOnly**  
With this option set, the formula does not adopt values in a referenced cell if the values in the data cell were changed, for example after executing a planning function.
  - **NoScaling**  
With this option set, the scaling factor of the data cell in the crosstab is not included for setting the value.  
Example: The scaling factor for a data cell is set to 100. The data cell shows the value 3, the 'real' value of the data cell is 300. If you want to change the 'real' value to 400, you can set the value 4 or you set the value 400 using the option NoScaling. Due to the scaling factor, the data cell will show value 4.

- *Measure*  
Enter the name of measure, for example **"Incoming Orders"**.
- *Member combination*  
There are three methods for entering the member combination:
  - Enter one parameter as the member combination, for example **"Region=France;Product=Services"**.
  - Enter several parameters as the member combination, for example **"Region";"France";"Product";"Services"**. This method can only be entered manually. It is recommended for member combinations that use cell references.
  - Enter a mixture of both: one parameter per dimension/member combination, for example **"Region=France";"Product=Services"**.

When a data source is initially refreshed, the formula is evaluated and the value after the formula is set in the data source. If the target value is changed, manually or in the referenced cell, the new value is set in the data source. But a recalculation is not triggered automatically and the crosstab still shows the old value. With this behavior, you can change several values and then recalculate all changes in one step by executing the recalculation manually.

## Example

Cell D10: =SAPSetData("DS\_1"; H5; "Incoming Orders"; "Region"; "France"; "Product"; "Services")

The formula in cell D10 defines the value for a measure and member combination of data source DS\_1. The name of the measure is 'Incoming Orders'. The member combination is 'France' and 'Services'. The value that is set is taken from cell H5. Therefore, cell H5 must not be part of the crosstab.

The parameter option is not defined in the formula.

## Related Information

[SAPGetData \[page 74\]](#)

[Syntax for Entering Values \[page 134\]](#)

## 4.13.29 SAPSetFilterComponent

This function creates a filter component and sets the members selected by the user as a filter. You can click the filter icon to change your filter definition in a dialog box. This dialog box is a simplified version of the standard filter dialog.

This formula consists of 4 parameters and is made up of the following arguments:

- *Data Source*  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.

- *Dimension Name*  
Enter the technical name of the dimension.
- *Target Data Source*  
You can enter **ALL** or a list of formula aliases for data sources that should be affected.
- *Selection Type*  
Enter one of the following selection types:
  - SINGLE  
With this selection type, you can select only one member for filtering.
  - MULTIPLE  
With this selection type, you can select multiple members for filtering. This is the default selection type.
  - LOWERBOUNDARY  
With this selection type, you can define a member as a lower boundary, for example a date.
  - UPPERBOUNDARY  
With this selection type, you can define a member as an upper boundary, for example a date. You can also insert two filter components in your analysis to define a period of time with a lower boundary date and an upper boundary date.
  - MEMBERSELECTOR  
With this selection type, you can define that the standard filter dialog with individual selection is used for the filter component.
  - MEMBERSELECTORBYRANGE  
With this selection type, you can define that the standard filter dialog with range selection is used for the filter component.

## Example

Cell E25: =SAPSetFilterComponent("DS\_1";"ODIVISION";"ALL";"MULTIPLE")

If you enter the formula in cell E25, the members that are currently filtered are displayed in cell E25. If you select the filter icon next to cell E25, you can change your filter definition.

## 4.14 Working with macros

Analysis contains API methods that can be used in VBA macros that are embedded in Microsoft Office Excel workbooks. Macros are created in the Visual Basic Editor. The Visual Basic Editor can be used to write and edit a macro that is attached to a Microsoft Office Excel workbook. The macros can be connected to UI elements that are available on the *Developer* tab in the menu.

You can also use the Analysis functions for creating formulas in VBA macros.

Microsoft Office documentation provides information about creating and using VBA macros. The following section describes the Analysis API methods.

The following methods are available in Analysis:

- SAPAddMessage
- SAPCallMemberSelector
- SAPExecuteCommand
- SAPExecutePlanningFunction
- SAPExecutePlanningSequence
- SAPGetCellInfo
- SAPGetProperty
- SAPLogOff
- SAPLogon
- SAPMoveDimension
- SAPOpenWorkbook
- SAPSetFilter
- SAPSetPlanParameter
- SAPSetRefreshBehaviour
- SAPSetVariable
- SAPSuppressMessage
- Table Desing API
  - SAPSetFormat
  - SAPInsertLine
  - SAPFixLineSize
  - SAPListOfDesignRules
  - SAPDeleteDesignRule

## 4.14.1 SAPAddMessage

With this API method, you can define messages and add them to the standard message dialog.

To call the method, use **Application.Run** and specify the following input parameters:

- **Message**  
Enter the message text.
- **Severity**  
You can enter one of the following severities: INFORMATION, WARNING, ERROR or CRITICAL. This parameter is optional. The default value is INFORMATION.
- **Details**  
You can enter a long text with additional information to the message. This parameter is optional.
- **Message ID**  
With the message ID, you can define a specific position in the message dialog where the new message should be added. The new message is inserted before the specified message ID. If the specified message ID does not exist in the message dialog, the new message is added as last message.  
The message ID is a unique identifier for a message that is valid in the current messages dialog. It can be retrieved with function [SAPListOfMessages](#).  
This parameter is optional.

The system returns one of the following output parameters for each function execution:

- 0 = execution failed.

- 1 = execution successful.

## Example

```
Dim lResult As Long
```

```
lResult= Application.Run("SAPAddMessage", "This is a new error message!", "ERROR")
```

The message 'This is a new error message' with severity Error is displayed in the message dialog.

## Related Information

[SAPListOfMessages \[page 89\]](#)

[SAPSuppressMessage \[page 116\]](#)

## 4.14.2 SAPCallMemberSelector

With this API method, you can call the standard input help (member selector) from the worksheet. The standard input help is the dialog that is used for filtering in Analysis. If you define the ReferenceCell parameter, the inplace selector is called. The inplace selector is a smaller version of the member selector. You can use it to select members of a dimension, but you cannot change how members are displayed in the selector, for example to change the display from key to text.

To call the method, use **Application.Run** and specify the following input parameters:

- **Formula Alias**  
Enter the formula alias for the data source or planning object. You can set the alias when configuring the data source on the **Components** tab in the design panel.  
You have to define this parameter if the variables in the workbook are not merged. If the variables in the workbook are merged, you can define this parameter but you do not have to define it.
- **Selector Type**  
Enter one of the following selector types:
  - PROMPT to get a selector with the prompt members.
  - FILTER to get a selector with the dimension members.
  - FILTER\_MASTERDATA to get a selector with all members of the master data for the selected dimension.
  - PLAN\_PARAMETER to get a selector with the members of a variable (filled and unfilled) of a planning object.
  - FILTER\_NEW\_LINES to get a selector with member access mode P (Planning) in leaf member view. Leaf members are members that do not have any child members.
  - FILTER\_NEW\_LINES\_WITH\_MASTER\_DATA to get a selector with all members of the master data for the selected dimension in leaf members view. Leaf members are members that do not have any child members.

- *Name*  
Enter the name of the variable or dimension.
- *ReferenceCell*  
Define the cell in which the inplace selector is displayed. This parameter is optional.
- *Selection Type*  
Enter one of the following selection types:
  - SINGLE  
With this selection type, you can select only one member for filtering.
  - MULTIPLE  
With this selection type, you can select multiple members for filtering.
  - MULTIPLEBYRANGE  
With this selection type, you can select a range of members for filtering.
 This parameter is optional.

The system returns one of the following output parameters for each function execution:

- InputString = selected data.
- Error = if invalid data is selected.
- False = if the dialog is canceled.

## Example: SAPCallMemberSelector Filter

```
Dim lResult As Variant
```

```
lResult= Application.Run("SAPCallMemberSelector", "DS_1", "FILTER", "OMATERIAL")
```

The member selector for dimension OMATERIAL is displayed, and you can select the members you need. You can also select a range of members or change how the members are displayed in the selector.

## Example: SAPCallMemberSelector Filter used for SAPSetFilter

```
Public Sub CallMemberSelectorFilter()
  Dim lResult As Variant
  lResult = Application.Run("SAPCallMemberSelector", "DS_1", "FILTER",
"OMATERIAL")
  If Not IsError(lResult) Then
    If lResult <> False Then
      Call Application.Run("SAPSetFilter", "DS_1", "OMATERIAL",lResult)
    End If
  End If
End Sub
```

The member selector for dimension OMATERIAL is displayed, and you can select the members you need. If the selected data is valid and the dialog is not canceled, the filter for dimension OMATERIAL is set to the selected members.

## Example: SAPCallMemberSelector Inplace

```
Dim lResult As Variant
```

```
lResult= Application.Run("SAPCallMemberSelector", "DS_1", "FILTER", "OMATERIAL",  
ActiveCell)
```

The inplace selector for dimension OMATERIAL is displayed in the active cell, and you can select the members you need.

## Example: SAPCallMemberSelector Inplace with single selection

```
Dim lResult As Variant
```

```
lResult= Application.Run("SAPCallMemberSelector", "DS_1", "FILTER", "OMATERIAL",  
ActiveCell, "SINGLE")
```

The inplace selector for dimension OMATERIAL is displayed in single selection mode in the active cell, and you can select the member you need.

## Related Information

[SAPSetFilter \[page 111\]](#)

## 4.14.3 SAPExecuteCommand

You can use this API method as a generic command to execute the commands described below.

The system returns one of the following output parameters for each command execution:

- 0 = execution failed.
- 1 = execution successful.

### PlanDataTransfer

Use this command to recalculate entered planning data.

### PlanDataSave

Use this command to save entered planning data.

## ❖ Example

```
Dim lResult As Long
```

```
lResult= Application.Run("SAPExecuteCommand", "PlanDataSave")
```

If you execute this command, the entered planning data is saved.

## PlanDataReset

Use this command to reset planning data entered since the last save of data.

## PlanDataClientReset

Use this command to reset planning data entered since the last successful recalculation of data.

## PlanDataToDisplayMode

Use this command to quit input-ready mode and switch to display mode while planning data.

## PlanDataToChangeMode

Use this command to switch to input-ready mode while planning data.

## PauseVariableSubmit

Use this command to pause the execution of variables using the parameter "On". To end the pausing, use the command with parameter "Off".

When PauseVariableSubmit is in mode "On", all Analysis formulas and most of the Analysis macros will not be executed. The following macros will be executed: SAPAddMessage, SAPSetVariable, PauseVariableSubmit "Off".

## Hide/Show

Use this command to hide and show Analysis UI components for a workbook. You can use the command for the types "Ribbon", "ContextMenu" and "TaskPane".

You can hide the complete ribbon ("All") and show it again ("Default"). For the context menu, you can hide the complete menu and show it again. For the design panel, you can hide the Analysis and the Components tab and show the complete design panel again.

The following commands are supported:

- `Application.Run("SAPEXecuteCommand", "Hide", "Ribbon", "All")`
- `Application.Run("SAPEXecuteCommand", "Show", "Ribbon", "Default")`
- `Application.Run("SAPEXecuteCommand", "Hide", "ContextMenu", "All")`
- `Application.Run("SAPEXecuteCommand", "Show", "ContextMenu", "Default")`
- `Application.Run("SAPEXecuteCommand", "Hide", "TaskPane", "Analysis")`
- `Application.Run("SAPEXecuteCommand", "Hide", "TaskPane", "Components")`
- `Application.Run("SAPEXecuteCommand", "Show", "TaskPane", "Default")`

## ShowPrompts

Use this command to display the prompts dialog. You can call the workbook or data source prompts dialog. To call the dialog, the data source(s) need to be refreshed.

### ❖ Example

Show prompts dialog for one data source

```
lResult= Application.Run("SAPEXecuteCommand", "ShowPrompts", "DS_1")
```

If you execute this command, the prompts dialog for DS\_1 is displayed. This is only possible if the variables are not merged in the workbook.

### ❖ Example

Show prompts dialog for all data sources

```
lResult= Application.Run("SAPEXecuteCommand", "ShowPrompts", "ALL")
```

If you execute this command, the prompts dialog for all data sources in the workbook is displayed.

## Refresh

Use this command to initially refresh the data in the workbook. You can specify one data source or one planning object as a parameter that should be refreshed. If you do not enter a parameter or you enter the string 'ALL' as parameter, all data sources and planning objects will be refreshed. If you execute this command for a data source which is already refreshed, all corresponding crosstabs are redrawn.

Additionally, you can refresh a list of data sources.

### i Note

If a planning function 'PF\_X' has a filter assigned, you can include this filter with the alias 'PF\_X' into the list. However, the planning function itself will not be refreshed and has to be refreshed with an own command.

### ❖ Example

Refresh one data source

```
Dim lResult As Long
```

```
lResult= Application.Run("SAPExecuteCommand", "Refresh", "DS_1")
```

If you execute this command, the data for data source DS\_1 is refreshed.

### ❖ Example

Refresh all data sources

```
Dim lResult As Long
```

```
lResult= Application.Run("SAPExecuteCommand", "Refresh")
```

If you execute this command, the data of all data sources is refreshed.

### ❖ Example

Refresh a list of data sources

```
Dim lResult As Long
```

```
lResult= Application.Run("SAPExecuteCommand", "Refresh", "DS_1;DS_2")
```

If you execute this command, the data of data source DS\_1 and DS\_2 is refreshed.

## RefreshData

Use this command to refresh the transaction data for all or defined data sources in the workbook. You can specify the data sources that should be refreshed. If you do not enter a parameter or you enter the string 'ALL' as parameter, all data sources will be refreshed. If you execute this command for a data source, the corresponding transaction data is updated from the server and the crosstabs are redrawn.

To use this command, the following prerequisites must be met:

- SAP BW 7.30 or higher
- The defined query read mode provides current data.
- Entered planning data was saved.

### ❖ Example

Refresh two defined data sources

```
Dim lResult As Long
```

```
lResult= Application.Run("SAPExecuteCommand", "RefreshData", "DS_1;DS_3")
```

If you execute this command, the data for data source DS\_1 and DS\_3 is refreshed from the server.

## Restart

Use this command to technically restart all or defined data sources in the workbook. Technically restarting a data source has the same effect than deleting and reinserting it with keeping the current navigation state and the selected prompt values. The complete process of adding or initially refreshing a data source is executed. After the restart, new data from the database will be included in the corresponding crosstabs. Before you can restart a data source, the data source has to be initially refreshed, for example with the command Refresh. If you do not enter a parameter or you enter the string 'ALL' as parameter, all data sources will be restarted. If you want to define a set of data sources that should be restarted, you can enter them as parameter with a semicolon separated list.

### ❖ Example

```
Dim lResult As Long
```

```
lResult= Application.Run("SAPExecuteCommand", "Restart", "DS_1;DS_2")
```

If you execute this command, the data of data sources DS\_1 and DS\_2 is restarted.

## AutoRefresh

Use this command to stop one or several data sources from redisplaying with parameter 'Off'. To reactivate the redisplaying, use the command with parameter 'On'. A data source can be set to off at any time. It can also be done in the callback Workbook\_SAP\_Initialize, for example. All parameters are mandatory and may not be blank.

If the command is used with parameter On, any crosstabs of the requested data sources that have pending updates, will be redisplayed automatically.

The configuration AutoRefresh of a data source is always independent of the 'Pause Refresh' of the workbook. A crosstab based on a data source is redisplayed if the workbook is not paused and the data source is not paused. The AutoRefresh value of a data source can be received with macro SAPGetProperty (SAPGetProperty("IsAutoRefresh","DS\_1").

### ❖ Example

```
Dim lResult As Long
```

```
lResult= Application.Run("SAPExecuteCommand", "AutoRefresh", "Off", "DS_1;DS_2")
```

If you execute this command, any crosstabs on top of data sources DS\_1 and DS\_2 are not redisplayed automatically anymore.

## RegisterCallback

Use this command to register a callback. The following callbacks can be registered with SAPExecuteCommand: AfterRedisplay, BeforePlanDataSave, BeforePlanDataReset, BeforeMessageDisplay.

## UnregisterCallback

Use this command to unregister the callbacks listed above.

### ❖ Example

```
Dim lResult As Long
```

```
lResult= Application.Run("SAPExecuteCommand", "UnregisterCallback",  
"AfterRedisplay")
```

If you execute this command, the AfterRedisplay callback is unregistered.

## SetFunctionalArea

Use this command to add a functional area (similar to a tag) to an Analysis workbook. It is limited to a length of 30 characters.

As it is typically called in the callback Workbook\_SAP\_Initialize, the functional area must be set before Analysis has established a connection to the BW system.

## NumberOfNewLines

Use this command to set the number of new lines for a crosstab.

If you enter a number less than 0, the number of new lines in the crosstab is set as defined in the file system setting NumberOfNewLines.

To get the number of new lines for a crosstab, you can use the API method SAPGetProperty with property NumberOfNewLines.

### ❖ Example

```
Dim lResult As Long
```

```
lResult= Application.Run("SAPExecuteCommand", "NumberOfNewLines", "Crosstab_1",  
"8")
```

If you execute this command, 8 new lines are added to crosstab Crosstab\_1. Crosstab\_1 is the formula alias of the crosstab.

You can set the alias when configuring the crosstab on the *Components* tab in the design panel.

## Related Information

[Using Callbacks \[page 128\]](#)

[Hiding / Showing Analysis UI components \[page 124\]](#)

[SAPSetVariable \[page 114\]](#)

[Using BAdI information \[page 126\]](#)

[SAPGetProperty \[page 106\]](#)

## 4.14.4 SAPExecutePlanningFunction

With this API method, you can execute a planning function. Planning functions are maintained in the *Components* tab in the design panel.

To call the method, use **Application.Run** and specify the following input parameters:

- *Planning Function Alias*  
Enter the formula alias of the planning function. You can set the alias when configuring the planning function on the *Components* tab in the design panel.

The system returns one of the following output parameters for each function execution:

- 0 = execution failed.
- 1 = execution successful.

### Example

```
Dim lResult As Long
```

```
lResult= Application.Run("SAPExecutePlanningFunction","PF_1")
```

If you select this command, planning function PF\_1 is executed.

## 4.14.5 SAPExecutePlanningSequence

With this API method, you can execute a planning sequence. Planning sequences are maintained in the *Components* tab in the design panel.

To call the method, use **Application.Run** and specify the following input parameters:

- *Planning Sequence Alias*  
Enter the formula alias of the planning sequence. You can set the alias when configuring the planning sequence on the *Components* tab in the design panel.

The system returns one of the following output parameters for each function execution:

- 0 = execution failed.
- 1 = execution successful.

## Example

```
Dim lResult As Long
```

```
lResult= Application.Run("SAPExecutePlanningSequence","PS_1")
```

If you select this command, planning sequence PS\_1 is executed.

### 4.14.6 SAPGetCellInfo

With this API method, you can define a command to get information on a crosstab cell. You can define the commands for a selection and for a dimension.

For a Dimension call, the result contains the data source alias and the technical name of the dimension.

For a Selection call, the result contains a list of dimensions. Each list entry contains the data source alias, the technical name of the dimension and the technical name of the member. For measures, a list entry contains the data source alias, the measure structure and the technical name of the measure.

To call the method, use **Application.Run** and specify the following input parameters:

- *Cell*  
Enter the cell information for one cell of the worksheet. If the currently selected cell should be evaluated, enter `ActiveCell`.  
If you have selected a range containing more than one cell, the method returns the cell information for the upper left cell in the range.
- *Property*  
Enter one of the following properties: SELECTION, DIMENSION, DATASOURCE or CROSSTAB.
- *Filter*  
For the SELECTION property, you can enter a list of dimensions to be evaluated, using semicolons as separator. If you do not enter any dimensions, all available dimensions are evaluated.  
For DIMENSION, DATASOURCE or CROSSTAB, you do not have to enter a value.

#### → Tip

If you want to include a check in your program to check if the selected cell is part of the crosstab, you can use the named range of the crosstab as reference. You can find the named range in the Name box in the formula bar. You can modify it on the *Components* tab in the design panel.

## Example: Dimension

```
Dim lResult As Variant
```

```
lResult= Application.Run("SAPGetCellInfo", ActiveCell, "DIMENSION")
```

If you select a dimension cell in the crosstab and execute the command, it returns the data source alias and the technical name of the dimension.

## Example: Selection

```
Dim lResult As Variant
```

```
lResult= Application.Run("SAPGetCellInfo", ActiveCell, "SELECTION")
```

In your current analysis, you have filtered the dimensions OMATERIAL and OCOUNTRY in rows and the measure OSALESVOLUME in columns. If you select a data cell and execute the command, it returns a list with three rows: One with the measure information, one with the information for dimension OMATERIAL and one with the information for dimension OCOUNTRY.

## Related Information

[The Components tab \[page 142\]](#)

### 4.14.7 SAPGetProperty

You can use this API method to read properties of a data source, a crosstab or a workbook.

To call the method, use `Application.Run` and specify the following input parameters:

- **Property Name**  
Enter one of the following properties:
  - `IsDataSourceActive` to check whether a data source is active.
  - `IsDataSourceEditable` to check whether the data source is input-ready.
  - `HasChangedPlanData` to check whether the workbook contains changed planning data.
  - `IsAutoRefresh` to check whether the automatic refresh is active or paused on workbook level or for the specified data source.
  - `IsConnected` to check whether a data source is already connected.
  - `LastError` to get information on an error that occurred while executing Analysis API methods and functions.
  - `ChangedCrosstabs` to get the crosstabs that have been changed in the last round trip with callback `AfterRedisplay`.
  - `ChangedDatasources` to get the data sources that have been changed in the last round trip with callback `AfterRedisplay`.
  - `NumberOfNewLines` to get the number of new lines for a crosstab.  
To set the number of new lines for a crosstab, you can use the API method `SAPExecuteCommand` with command `NumberOfNewLines`.
- **Formula Alias**  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the [Components](#) tab in the design panel. This parameter is optional.  
If you use the [NumberOfNewLines](#) property, enter the formula alias for the crosstab. You can set the alias when configuring the crosstab on the [Components](#) tab in the design panel.  
If you use the [HasChangedPlanData](#) or [LastError](#) property, you should not enter a data source alias, as this property is not related to a specific data source but to the complete workbook.

## Example: IsDataSourceEditable

```
Dim lResult As Variant
```

```
lResult= Application.Run("SAPGetProperty", "IsDataSourceEditable", "DS_1")
```

If you execute the command, it returns the Boolean value. If the data source DS\_1 is editable, the value is True.

## Example: IsConnected

```
Iret = Application.Run("SAPGetProperty", "IsConnected", "DS_1")
```

If you execute the command, it returns the Boolean value. If the data source DS\_1 is connected, the value is True.

## Example: NumberOfNewLines

```
Iret = Application.Run("SAPGetProperty", "NumberOfNewLines", "Crosstab_1")
```

If you execute the command, it returns the number of new lines added to crosstab Crosstab\_1.

## Related Information

[Using LastError information \[page 132\]](#)

[SAPExecuteCommand \[page 98\]](#)

## 4.14.8 SAPLogOff

You can use this API method to disconnect a workbook from the server or to trigger a restart to the system.

You can also log off and reconnect a workbook manually in Analysis. The options are available in the context menu for workbooks on the *Components* tab in the design panel.

To call the method, use **Application.Run** and specify the following input parameters:

- Boolean Parameter for restarting a system  
This parameter is optional. You can enter the following values:
  - True: With this value, the system will be restarted.
  - False: With this value, the system will not be restarted.

The system returns one of the following output parameters for each command execution:

- 0 = execution failed.
- 1 = execution successful.

## Example

```
Dim lResult As Long  
lResult= Application.Run("SAPLogOff", True )
```

If you execute this command, the workbook connection to the system is restarted.

## Related Information

[Managing system connections in the design panel \[page 152\]](#)

### 4.14.9 SAPLogon

You can use this API method to trigger a logon to a system for a specified data source.

To call the method, use **Application.Run** and specify the following input parameters:

- *Formula Alias*  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- *Client*  
Enter the system client. This parameter is only relevant for logon to an SAP system.
- *User*  
Enter the user.
- *Password*  
Enter the password.
- *Language*  
Enter the logon language. This parameter is optional.

The system returns one of the following output parameters for each command execution:

- 0 = execution failed.
- 1 = execution successful.

## Example

```
lret = Application.Run("SAPLogon", "DS_1", "000", "Anzeiger", "Display")
```

If you execute the command, the system of data source DS\_1 with client 000 will be connected for user Anzeiger with password Display.

## 4.14.10 SAPMoveDimension

With this API method, you can define the position of a dimension in the crosstab, swap the dimension with another one or position a dimension relative to another one.

Note: In a grouped crosstabs, you can only move dimensions to or within the master crosstab.

To call the method, use **Application.Run** and specify the following input parameters:

- *Data Source Alias*  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- *Dimension Key*  
Enter the technical name of the dimension.
- *PositionBy*  
Enter one of the following constants:
  - ROWS to move a dimension to the rows.
  - COLUMNS to move a dimension to the columns.
  - FILTER to move a dimension to the background filter.
  - AFTER to move a dimension after a relative dimension.
  - BEFORE to move a dimension before a relative dimension.
  - SWAP to swap a dimension with a relative dimension.
- *Identifier*  
If PositionBy is set to AFTER, BEFORE or SWAP, enter the technical name of the relative dimension.  
If PositionBy is set to ROWS, COLUMNS or FILTER, enter the position as an index. The index is 1-based and optional. If the index is not valid, the command will not be executed. If no index is specified, the dimension is added at the end.

The system returns one of the following output parameters for each function execution:

- 0 = execution failed.
- 1 = execution successful.

### Example

```
Dim lResult As Long
```

```
lResult= Application.Run("SAPMoveDimension", "DS_1", "OCOUNTRY", "AFTER",  
"OCALMONTH" )
```

If you execute this command, dimension OCOUNTRY will be inserted in the crosstab to the position after dimension OCALMONTH.

## Example

```
Dim lResult As Long
```

```
lResult= Application.Run("SAPMoveDimension", "DS_1", "OCOUNTRY", "ROWS", "2")
```

If you execute this command, dimension OCOUNTRY is added as second entry to the rows in the crosstab.

## Related Information

[Grouping crosstabs \[page 58\]](#)

### 4.14.11 SAPOpenWorkbook

You can use this API method to open an Analysis workbook. The workbook to be opened can be stored on a SAPBW server or on a BI platform.

The workbook is opened in the same Microsoft Excel instance. After the command execution, the opened workbook is active.

To call the method, use **Application.Run** and specify the following input parameters:

- **Name**  
Enter the technical name of the document.  
For documents stored on a SAP BW server, this is the technical name as is it is shown in the Open dialog.  
For documents stored on a BI platform, it is the CUID that can be found in the general properties of an Analysis workbook in the CMC.
- **Formula Alias**  
Enter the formula alias for the data source or planning object, which provides the reference connection for the target workbook. The target workbook is the workbook that you want to open.  
If the referenced connection is a local connection to a BW system, the workbook is opened from this system. If the referenced connection is a system (BW or HANA) on a BI platform, the workbook is opened from the BI platform.  
This parameter is mandatory if several connections are used in the workbook, which provides the reference connection for the target workbook. If only one connection is used in the workbook, it is optional.
- **Parameters for variables**  
This parameter is only relevant if variables are used in the data source.  
Enter the data source alias, the variable name and values.  
The data source alias is only required if a variable is used in several data sources, the variables are not merged and you want to specify different values for the same variable.  
There are three methods for entering the variable name and values:
  - Enter one parameter, for example "**[DS\_1]OCOUNTRY=France;OPRODUCT=P01; [DS\_2]OCOUNTRY=Germany**".
  - Enter several parameters, for example "**[DS\_1]OCOUNTRY";"France";" [DS\_1]OPRODUCT";"P01";" [DS\_2]OCOUNTRY";"Germany"**".

- Enter a mixture of both: one parameter per variable/value combination, for example  
`" [DS_1] OCOUNTRY=France" ; " OPRODUCT=P0 "1 ; " [DS_2] OCOUNTRY=Germany"`

Limitations:

The first method only allows single members as variable values, complex input strings are not allowed. As the Microsoft Excel command `Application.Run` is limited to 30 arguments, the second and third method only allow you to enter 14 variable or 28 value combinations.

If you need to set more than 28 variable/value combinations with a complex input string (for example, multiple members separated by a semicolon), the third method can be changed to specify multiple variables in one string. In this case the data source alias is mandatory but can be left blank, for example  
`" [DS_1] OCOUNTRY=France" ; " [ ] OPRODUCT=P01 ; P02 ; P03 ; [DS_2] OCOUNTRY=Germany"`.

The system returns one of the following output parameters for each command execution:

- 0 = execution failed.
- 1 = execution successful.

## Example

```
Dim lResult as long
```

```
lResult = Application.Run("SAPOpenWorkbook", "TEST_OpenWB", "DS_1",  
"OCOUNTRY";"France")
```

With this example, the workbook TEST\_OpenWB is opened from the system of data source DS\_1. The variable OCOUNTRY is set to value France.

## Related Information

[Syntax for Entering Values \[page 134\]](#)

### 4.14.12 SAPSetFilter

With this API method, you can define which members of a dimension should be filtered. You can also use this method to define the filter for a planning function.

Note: In a grouped crosstab, you can only filter the dimensions of the data source of the master crosstab.

To call the method, use `Application.Run` and specify the following input parameters:

- *Formula Alias*  
Enter the formula alias for the data source or planning function. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- *Dimension*  
Technical name of the dimension to be filtered.
- *Member*

String that represents the member filter for the dimension, for example technical names or a variable. The "ALLMEMBERS" string or an empty string clears the filter and selects all members. Note the syntax rules for entering values.

- **Member Format**
  - Text  
Single member as text.
  - Key  
Single member as key.
  - INTERNAL\_KEY  
Single member with its internal key.
  - INPUT\_STRING  
Complex selection of members.
  - INPUT\_STRING\_AS\_ARRAY  
Returns the input string as string as array.
  - LITERAL  
Single member with its internal key.  
With this value format, no immediate validation with the back-end system is executed. This improves the performance.  
This format is only valid if you set the prompt value with its internal key and you use the syntax rule [Equal To](#).  
If the entered member is not valid, a message is displayed.  
This parameter is optional. The default value is INPUT\_STRING.

#### **i** Note

The KEY and INTERNAL\_KEY depend on the InfoObject modeling in SAP BW.

The system returns one of the following output parameters for each function execution:

- 0 = execution failed.
- 1 = execution successful.

## Example

```
Dim lResult as long
```

```
lResult= Application.Run("SAPSetFilter", "DS_1", "0SOLD_TO__0COUNTRY", "CA;US;DE",  
"INPUT_STRING")
```

With this example, you set the filter for dimension 0SOLD\_TO\_\_0COUNTRY of data source DS\_1 to the countries USA, Canada and Germany using member format INPUT\_STRING.

## Related Information

[Syntax for Entering Values \[page 134\]](#)

[Grouping Crosstabs \[page 58\]](#)

## 4.14.13 SAPSetPlanParameter

With this API method, you can define values for input-ready variables of planning objects.

To call the method, use `Application.Run` and specify the following input parameters:

- **Formula Alias**  
Enter the formula alias for the planning object. You can set the alias when configuring the planning object on the *Components* tab in the design panel.
- **Variable Name**  
Name or technical name of the variable to be filtered.
- **Variable Value**  
String that represents the value for the variable, for example the key. Note the syntax rules for entering values.
- **Member Format**
  - Text  
Single member as text.
  - Key  
Single member as key.
  - INTERNAL\_KEY  
Single member with its internal key.
  - INPUT\_STRING  
Complex selection of members.
  - LITERAL  
Single member with its internal key.  
With this value format, no immediate validation with the back-end system is executed. This improves the performance.  
This format is only valid if you set the prompt value with its internal key and you use the syntax rule *Equal To*.  
If the entered member is not valid, a message is displayed.  
This parameter is optional. The default value is INPUT\_STRING.

### i Note

The KEY and INTERNAL\_KEY depend on the InfoObject modeling in SAP BW.

The system returns one of the following output parameters for each function execution:

- 0 = execution failed.
- 1 = execution successful.

## Example

```
Dim lResult as long
```

```
lResult= Application.Run("SAPSetPlanParameter", "PF_1", "OTARGET_YEAR", "2013",  
"INPUT_STRING")
```

With this example, you set the variable OTARGET\_YEAR in planning function PF\_1 to the year 2013.

## Related Information

[Syntax for Entering Values \[page 134\]](#)

### 4.14.14 SAPSetRefreshBehaviour

With this API method, you can use different API methods one after the other without the result set being refreshed and without the message dialog being shown.

To call the method, use **Application.Run** and specify the following input parameters:

- *Mode*  
Define whether the mode is *On* or *Off*.
- *Message Display*  
Define whether the message display is *On* or *Off*. The default setting is *Off*.

The system returns one of the following output parameters for each function execution:

- 0 = execution failed
- 1 = execution successful

### Example

The default API call would look like this:

```
Public Sub SetGermanyAndProduct01
    Application.Run("SAPSetRefreshBehaviour","Off")
    Application.Run("SAPSetFilter","DS_01","0COUNTRY","DE")
    Application.Run("SAPSetFilter","DS_01","0PRODUCT","P01")
    Application.Run("SAPSetRefreshBehaviour","On")
End Sub
```

### 4.14.15 SAPSetVariable

With this API method, you can define values for input-ready variables (prompts). If you want to set multiple variables, you can use the `PauseVariableSubmit` command with the `SAPExecuteCommand` method.

To call the method, use **Application.Run** and specify the following input parameters:

- *Prompt Name*  
Name or technical name of the variable to be filtered.
- *Prompt Value*  
String that represents the value for the prompt, for example the key. Note the syntax rules for entering values.
- *Value Format*

- Text  
Single member as text.
- Key  
Single member as key.
- INTERNAL\_KEY  
Single member with its internal key.
- INPUT\_STRING  
Complex selection of members.
- INPUT\_STRING\_AS\_ARRAY  
Returns the input string as string as array.
- LITERAL  
Single member with its internal key.  
With this value format, no immediate validation with the back-end system is executed. This improves the performance.  
This format is only valid if you set the prompt value with its internal key and you use the syntax rule *Equal To*.  
If the entered member is not valid, a message is displayed.

### **i** Note

The KEY and INTERNAL\_KEY depend on the InfoObject modeling in SAP BW.

- *Formula Alias*  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.  
You have to define this parameter if the variables in the workbook are not merged. If the variables in the workbook are merged, you can define this parameter but you do not have to define it.

The system returns one of the following output parameters for each function execution:

- 0 = execution failed.
- 1 = execution successful.

## Example

```
Dim lResult As Long
```

```
lResult=Application.Run("SAPSetVariable", "OBWVC_COUNTRY", "DE", "INPUT_STRING", "DS_1")
```

With this example, you set the variable OBWVC\_COUNTRY in data source DS\_1 to country Germany.

## Example: Setting multiple variables

```
Call Application.Run("SAPSetRefreshBehaviour", "Off")
Call Application.Run("SAPExecuteCommand", "PauseVariableSubmit", "On")
Call Application.Run("SAPSetVariable", "OBWVC_COUNTRY", "US")
Call Application.Run("SAPSetVariable", "OBWVC_MATERIAL", "Hardware")
```

```
Call Application.Run("SAPExecuteCommand", "PauseVariableSubmit", "Off")
Call Application.Run("SAPSetRefreshBehaviour", "On")
```

With this example, you set the variable OBWVC\_COUNTRY to country USA and the variable OBWVC\_MATERIAL to material Hardware. Both variables will be executed at once after setting off the command PauseVariableSubmit.

## Related Information

[Syntax for Entering Values \[page 134\]](#)

[SAPExecuteCommand \[page 98\]](#)

### 4.14.16 SAPSuppressMessage

With this API method, you can suppress messages in the standard message dialog. It can only be used in the callback BeforeMessageDisplay.

To call the method, use **Application.Run** and specify the following input parameter:

- *Message ID*  
The message ID is a unique identifier for a message that is valid in the current messages dialog. It can be retrieved with function [SAPListOfMessages](#).

The system returns one of the following output parameters for each function execution:

- 0 = execution failed.
- 1 = execution successful.

## Example

```
lret = Application.Run("SAPSuppressMessage", "5")
```

The message with message ID 5 is suppressed in the message dialog.

## Related Information

[Using Callbacks \[page 128\]](#)

[SAPListOfMessages \[page 89\]](#)

[SAPAddMessage \[page 95\]](#)

## 4.14.17 Table Design API

Table Design API enables you to edit single elements in a crosstab with API methods (for example, you can add a new line to the crosstab).

The Table Design API offers methods for the following options:

- Applying formats
- Adding new lines
- Adding texts to new lines
- Fixing the line size
- Listing Table Design API methods for a specific data source
- Deleting Table Design API methods for a specific data source

### Related Information

[Editing crosstabs with Table Design \[page 47\]](#)

### 4.14.17.1 SAPSetFormat

With this API method, you can apply defined cell styles as formats to single cells and cell ranges. You can use and select standard Microsoft Excel cell styles, SAP cell styles and user-defined cell styles.

To call the method, use **Application.Run** and specify the following input parameters:

- *RuleID*  
If you define a rule ID, you can address it in other API methods, for example for deleting the rule with `SAPDeleteDesignRule`.  
If you do not define an ID, the system generates one automatically.
- *Data Source Alias*  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- *Style*  
Enter the style that should be applied.
- *First Line Type and attributes*  
With the first and second line type and their attributes, you can define the anchor for the area in the crosstab, which you want the selected format to be applied to. The operation anchor, or operation context, is defined as a crossing point of two lines. Each line is defined using two parameters: line type and line context (attributes). Different context information is expected, depending on the line type.  
You can enter one of the following constant types and their relevant attributes:
  - DIMENSION  
To define a dimension, enter the constant DIMENSION and the technical name of the dimension as the attribute.
  - MEMBER

To define a member, enter the constant MEMBER and the technical names of the dimension and the member as attributes.

- RESULTMEMBER

To define a result member, enter the constant RESULTMEMBER and the technical name of the dimension as the attribute.

- HIERARCHYNODE

To define a hierarchy node, enter the constant HIERARCHYNODE and the technical names of the dimension, the member and the node as attributes.

- VIRTUALHEADER

To define a new inserted header line, enter VIRTUALHEADER and the rule ID of the inserted line.

- VIRTUALDIMENSION

To define a new inserted dimension line, enter VIRTUALDIMENSION and the rule ID of the inserted line.

- VIRTUALMEMBER

To define a member of a new inserted dimension line, enter VIRTUALMEMBER and the rule ID of the inserted line.

- TUPLE

The constant TUPLE can be used to define a combination of members of various dimensions on an axis.

To define a TUPLE, enter one of the following member types with the corresponding attributes:

- MEMBER with the technical names of the dimension and the member as attributes.
- VIRTUALMEMBER with the rule ID of the inserted line as the attribute.
- RESULTMEMBER with the technical name of the dimension as the attribute.
- HIERARCHYNODE with the technical names of the dimension, the member and the node as attributes.

- *Second Line Type and attributes*

You can enter the same constants and attributes as for the first line type. If you enter a second line type, you can specify the anchor more precisely.

- *Scope Area*

This parameter is optional. You can enter one of the following values:

- CONTEXT

The parameter CONTEXT is the anchor defined with first and second line type. Context is the default value for the scope area parameter.

- FIRST

The parameter FIRST defines the area on the left or top from the anchor as the scope area.

- SECOND

The parameter SECOND defines the area on the right or bottom from the anchor as the scope area.

- BOTH

The parameter BOTH defines the areas of the parameters FIRST and SECOND as the scope area.

- *Scope Axis*

This parameter is optional. You can enter one of the following values:

- PRIMARY

PRIMARY applies style to the primary axis, depending on the context.

- SECONDARY

SECONDARY applies style to the secondary axis, depending on the context.

- ROW

ROW applies the style to the specified area on the row axis.

- COLUMN

COLUMN applies style to the specified context on the column axis.

### Example: Apply a style to a member

```
Dim lResult As String
```

```
lResult= Application.Run("SAPSetFormat", "Style1", "DS_1", "Green", "MEMBER",  
"OCALYEAR;2014")
```

The style Green is applied to member 2014 of the dimension OCALYEAR. If the member 2014 appears several times in the crosstab, the style is applied to each occurrence. The rule ID for the applied style is Style1.

### Example: Apply a style to the data cells of a member for one measure

```
Dim lResult As String
```

```
lResult= Application.Run("SAPSetFormat", "Style2", "DS_1", "Green", "TUPLE",  
"MEMBER;OCALYEAR;2014", "TUPLE", "MEMBER;[Measures];Sales")
```

The style Green is applied to the data cell for measure Sales of member 2014 of the dimension OCALYEAR. If the member 2014 appears several times in the crosstab with data cells for measures Sales, the style is applied to each corresponding data cell. The rule ID for the applied style is Style2.

### Example: Apply a style to a member and to the data cells of the member for several measures

```
Dim lResult As String
```

```
lResult= Application.Run("SAPSetFormat", "Style3", "DS_1", "Green", "TUPLE",  
"MEMBER;OCALYEAR;2014", "TUPLE", "MEMBER;[Measures];Sales", "Both")
```

The style Green is applied to member 2014 of the dimension OCALYEAR and to the corresponding data cell of measure Sales. If there are data cells for other measures in the same row, the style is also applied to these cells. If the member 2014 appears several times in the crosstab, the style is applied to each occurrence and the corresponding data cells. The rule ID for the applied style is Style3.

## 4.14.17.2 SAPInsertLine

With this API method, you can insert new lines in a crosstab.

To call the method, use **Application.Run** and specify the following input parameters:

- *RuleID*

If you define a rule ID, you can address it in other API methods, for example for deleting the rule with `SAPDeleteDesignRule`.

If you do not define an ID, the system generates one automatically.

- **Data Source Alias**  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- **Position**  
Enter one of the following positions: Before, After, BelowHeader or BesideHeader.
- **PositionBy**  
Enter one of the following crosstab elements: *Dimension*, *DimensionResult*, *DimensionGroup*, *DimensionMember*, *HierarchyNode* or *Tuple*.  
The constant TUPLE can be used to define a combination of members of various dimensions on an axis.
- **PositionBy parameters**  
If PositionBy is set to *Dimension*, *DimensionResult* or *DimensionGroup*, enter the technical name of the relative dimension.  
If PositionBy is set to *DimensionMember*, enter the technical name of the relative dimension and member.  
If PositionBy is set to *HierarchyNode*, enter the technical name of the relative dimension, member and node type.  
If PositionBy is set to *Tuple*, enter one of the following member types of the relative member with the corresponding attributes:
  - MEMBER with the technical names of the dimension and the member as attribute.
  - VIRTUALMEMBER with the rule ID of the inserted line as attribute.
  - RESULTMEMBER with the technical name of the dimension as attribute.
  - HIERARCHYNODE with the technical names of the dimension, the member and the node as attribute.If you define a tuple, you have to specify placeholders for the parameters dimension, member and node type (see example 2).

## Example: Inserting a line after a dimension

```
Dim lResult As String
```

```
lResult= Application.Run("SAPInsertLine", "NewLine1", "DS_1", "After", "Dimension",  
"OCALYEAR")
```

A new line is inserted after the dimension OCALYEAR in the crosstab of data source DS\_1. The rule ID for the new line is NewLine1.

## Example: Inserting a line after a tuple

```
Dim lResult As String
```

```
Dim tupleArray(1) As String
```

```
tupleArray(0)= "Member;OCALYEAR;2014"
```

```
tupleArray(1)= "Member;OCOUNTRY;FR"
```

```
lResult= Application.Run("SAPInsertLine", "NewLine2", "DS_1", "After", "Tuple",,,,  
tupleArray)
```

A new line is inserted after the combination of the members 2014 and France (FR) in the crosstab of data source DS\_1. The rule ID for the new line is NewLine2.

The three placeholders after the PositionBy element Tuple replace the attributes dimension, member and node type that are not relevant for the element Tuple.

## Example: Inserting a line after a hierarchy node

```
Dim lResult As String
```

```
lResult= Application.Run("SAPInsertLine", "NewLine3", "DS_1", "After",  
"HierarchyNode", "0Country", "FR", "State")
```

A new line is inserted after the member France (FR) in the crosstab of data source DS\_1. The node type is State. If the member France occurs several times in the crosstab, a new line is inserted after each occurrence. The rule ID for the new line is NewLine3.

## 4.14.17.3 SAPFixLineSize

With this API method, you can define the size for a inserted line.

To call the method, use **Application.Run** and specify the following input parameters:

- *RuleID*  
If you define a rule ID, you can address it in other API methods, for example for deleting the rule with `SAPDeleteDesignRule`.  
If you do not define an ID, the system generates one automatically.
- *Data Source Alias*  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- *ID of new line*  
Enter the rule ID of the inserted line whose size you want to fix.
- *Size*  
Enter an integer number. The measurement unit for the size is pixels.

## Example

```
Dim lResult As String
```

```
lResult= Application.Run("SAPFixLineSize", "SizeLine1", "DS_1", "NewLine1", "20")
```

The size of the new inserted line with ID NewLine1 is set to 20 pixels. The rule ID for the new line size is SizeLine1.

## 4.14.17.4 SAPListOfDesignRules

With this API method, you can list Table Design methods defined for a specific data source.

To call the method, use **Application.Run** and specify the following input parameters:

- *Data Source Alias*  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- *API Method Type*  
Enter one of the following types:
  - All to list all methods.
  - NewLine to list the inserted lines.  
You can specify the new lines in more detail by using NewDimensionLine, NewHeaderLine or NewMemberLine.
  - LineSize to list the defined sizes for new inserted lines.
  - Format to list the defined formats.

### Example

```
Dim lResult As Variant
```

```
lResult= Application.Run("SAPListOfDesignRules", "DS_1", "NewLine")
```

All inserted lines defined for data source DS\_1 with method SAPInsertLine are listed.

## 4.14.17.5 SAPDeleteDesignRule

With this API method, you can delete a rule that was applied with the Table Design API.

To call the method, use **Application.Run** and specify the following input parameters:

- *Data Source Alias*  
Enter the formula alias for the data source. You can set the alias when configuring the data source on the *Components* tab in the design panel.
- *RuleID*  
Enter the rule ID of the rule to be deleted. If you do not specify a rule ID, all rules defined for that data source will be deleted.

The system returns one of the following output parameters for each execution:

### Example

```
Dim lResult As Boolean
```

```
lResult= Application.Run("SAPDeleteDesignRule", "DS_1")
```

All rules defined for data source DS\_1 are deleted.

## 4.14.18 Using Analysis functions

You can use the Analysis functions in VBA macros. The return value of a formula in a macro corresponds to the return value of the formula used in a Microsoft Excel sheet. Depending on the function that you use, a formula can either return a single value (single string value) or a list (array). If the list contains only one line, the returning array is one-dimensional. If the list contains two or more lines the array is two-dimensional.

If you use a formula in a macro, the function is the first parameter followed by the arguments you use to create a formula with this function.

If a formula is invalid, Microsoft Excel returns an error.

### Example: Formula returning a single string value

```
Dim lResult As Variant
```

```
lResult = Application.Run("SAPGetVariable", "DS_2", "OBW_VAR", "Value")
```

This formula returns the current value of variable OBW\_VAR.

### Example: Formula returning an array

```
Dim lResult As Variant
```

```
lResult = Application.Run("SAPListOfDimensions", "DS_1")
```

This formula returns a list with the dimensions of data source DS\_1. If the data source contains only one dimension, the returned array is one-dimensional.

To make your programming easier, you can create an additional function to ensure that the array is always two-dimensional, for example the function GetAsTwoDimArray.

```
Function GetAsTwoDimArray(value As Variant) As Variant
    'value is error return the error
    If IsError(value) Then
        GetAsTwoDimArray = value
    'value is array
    ElseIf IsArray(value) Then
        'first check if the array is two-dimensional
        'by requesting the upper bound of the 2nd dimension.
        'if this is not the case an error occurs (Err.Number <> 0).
        'ignore errors, handled locally
        On Error Resume Next
        Dim lIndex As Integer
        Dim lErrorCode As Integer
        lIndex = UBound(value,2)
        lErrorCode = Err.Number
```

```
'set error handling back to default
On Error GoTo 0
If lErrorCode = 0 Then
'no error: array is two-dimensional
GetAsTwoDimArray = value
Else
'copy one-dimensional array into a two-dimensional one
Dim i As Integer
Dim lArray() As Variant
ReDim lArray(1 To 1, 1 To UBound(value))
For i = 1 To UBound(lArray, 2)
lArray(1, i) = value(i)
Next
GetAsTwoDimArray = lArray
End If
Else
'return Empty
GetAsTwoDimArray = Empty
End If
End Function
```

In the following example, a list of all dimensions is returned from function `SAPListOfDimensions`. All dimensions which are neither on the rows nor on the columns axis are added to a string that is displayed in the standard message dialog with the API method `SAPAddMessage`. Using the function `GetAsTwoDimArray`, you ensure that the returned array is always two-dimensional.

```
Public Sub ShowDimensionsNotOnRowsOrColumns()
Dim lList As String
Dim lResult As Variant
lResult = Application.Run("SAPListOfDimensions", "DS_1")
lResult = GetAsTwoDimArray(lResult)
For i = 1 To UBound(lResult, 1)
If lResult(i, 3) <> "ROWS" And lResult(i, 3) <> "COLUMNS" Then
lList = lList & " " & lResult(i, 2)
End If
Next i
Call Application.Run("SAPAddMessage", "Dimensions:" & lList, "INFORMATION")
End Sub
```

## Related Information

[Working with formulas \[page 70\]](#)

### 4.14.19 Hiding / Showing Analysis UI components

You can use the commands `Hide` and `Show` of the API method `SAPExecuteCommand` to hide and show Analysis UI components.

The changed visibility of UI components is valid for the active workbook in which the VBA macro is executed. If the Analysis tab is hidden in a workbook and the user switches to another workbook, the Analysis tab is visible again. When switching back to the first workbook, the Analysis tab is hidden again.

The following types and IDs are supported:

- Ribbon
  - All  
Use this ID to hide the Analysis tab in the ribbon.  
Example: `Application.Run("SAPExecuteCommand", "Hide", "Ribbon", "All")`
  - Default  
Use this ID to show the Analysis tab in the ribbon.  
Example: `Application.Run("SAPExecuteCommand", "Show", "Ribbon", "Default")`
- ContextMenu
  - All  
Use this ID to hide the Analysis context menu.  
`Application.Run("SAPExecuteCommand", "Hide", "ContextMenu", "All")` Default  
Default  
Example: Example:
  - Default Default  
Use this ID to show the complete context menu.  
Example: `Application.Run("SAPExecuteCommand", "Show", "ContextMenu", "Default")`  
`Application.Run("SAPExecuteCommand", "Show",`

## Related Information

[SAPExecuteCommand \[page 98\]](#)

[Enhancing the Analysis Ribbon \[page 125\]](#)

## 4.14.20 Enhancing the Analysis Ribbon

The Microsoft Office Ribbons can be enhanced and adjusted. You can learn more about the general ribbon enhancement in the Microsoft documentation.

If you build an application related to Analysis, you can visualize the dependencies to Analysis by locating the new ribbon tab beside (before/after) the Analysis tab or by adding new ribbon groups to the Analysis tab. The ribbon is described via an XML definition where an identifier is assigned to each item. The Analysis tab has a qualified identifier (idQ) with the name space SapExcelAddIn and the id com.sap.ip.bi.analysis.menu.

### Example: Adding a tab before the Analysis tab

```
<customUI xmlns="http://schemas.microsoft.com/office/2006/01/customui"
xmlns:x="SapExcelAddIn">
<ribbon>
<tabs>
<tab id="myCustomTab" label="Custom Tab"
insertBeforeQ="x:com.sap.ip.bi.analysis.menu" >
<group id="myCustomGroup" label="Custom Group" >
<button id="myButton" label="my Button" onAction="myAction" />
</group>
</tabs>
</ribbon>
</customUI>
```

```
</tab>
</tabs>
</ribbon>
</customUI>
```

With this example, the new tab 'Custom Tab' is added before the Analysis tab containing the group 'Custom Group' with the button 'my Button'.

## Example: Adding a ribbon group to the Analysis tab

```
<customUI xmlns="http://schemas.microsoft.com/office/2006/01/customui"
xmlns:x="SapExcelAddIn">
<ribbon>
<tabs>
<tab idQ="x:com.sap.ip.bi.analysis.menu">
<group id="myCustomGroup" label="Custom Group" >
<button id="myButton" label="my Button" onAction="myAction" />
</group>
</tab>
</tabs>
</ribbon>
</customUI>
```

With this example, the group 'Custom Group' with the button 'my Button' is added to the Analysis tab.

## Related Information

[Hiding / Showing Analysis UI components \[page 124\]](#)

### 4.14.21 Using BAdI information

Using macros in a workbook offers a great option to improve the user experience and to expose functionality in an easy consumable manner. As long as the macros only require information that is available in Analysis already (e.g. filter values), the implementation is straightforward.

For situations, where additional information from the BW back-end is required, Analysis offers the possibility to retrieve information from the BW system, and use it in the Analysis API. In the BW system, the information that can be retrieved, is provided by a Business Add-In (BAdI).

You can use the back-end information in variable scenarios, for example:

- Showing/hiding a ribbon or component based on the user configuration.
- Checking if new versions are available and as consequence, ask the user to update the software.
- Invoking a web page where the URL is system dependent.

The request to the BW system for retrieving the information, is sent once when a connection is established to the BW system. If a workbook is connected to several BW systems, each system can be asked for back-end

information. The transfer format is string. This means whatever information should be transferred, it requires an encoding to string format in the BW system and a decoding from string in Analysis.

You need to implement the BAdI `RSAO_GET_INITIAL_APPL_CONTEXT` in the BW system. To call the BAdI from Analysis, execute the following steps:

1. Define a functional area in BAdI `RSAO_GET_INITIAL_APPL_CONTEXT` in the BW system.  
The functional area is a BAdI filter of type `FUNCTIONAL_AREA`.
2. Add the functional area to the Analysis workbook.  
The functional area is added with API method `SAPExecuteCommand` and the command `SetFunctionalArea`.  
It must be set before Analysis is connected to the BW system. Therefore it is typically called in callback `Workbook_SAP_Initialize`.  
You can add one functional area per BW system to a workbook.
3. Use the information retrieved from the BAdI in an Analysis macro.  
The information is transferred to Analysis upon connecting to the BW system. The information can be accessed with the Analysis function `SAPListOf` and parameter `CUSTOMAPPLCONTEXT`. The formula returns a list with three columns (`SystemID`, `RowCount` and `Content`).

## Example: Hiding the Analysis ribbon

The following example shows the coding in Analysis.

1. Add the functional area to the Analysis workbook:

```
Public Sub Workbook_SAP_Initialize()  
    Dim result  
    result = Application.Run("SAPExecuteCommand", "SetFunctionalArea",  
        "FA_Ribbon")  
End Sub
```

The functional area `FA_Ribbon` is added to the workbook.

2. After connecting to the BW system, the result of the BAdI implementation that corresponds to the functional area `FA_Ribbon` is transferred to Analysis:  
The information is encoded to string format to be transferred.
3. You can retrieve the BAdI information with formula `SAPListOf` and use the result to hide the Analysis ribbon:

```
Dim result As String  
result = application.run("SAPListOf","CUSTOMAPPLCONTEXT")(3)  
'The formula returns in column 3 as result the content: "hideRibbon=X"  
  
Dim hideRibbon As Boolean  
Dim parameters() As String  
parameters() = Split(result, "=")  
If parameters(0) = "hideRibbon" and parameters(1) = "X" Then  
    hideRibbon = True  
Else  
    hideRibbon = False  
End If  
If hideRibbon = True  
    Application.run("SAPExecuteCommand","Hide","Ribbon")  
End If
```

The result `"hideRibbon=X"` is the encoded string provided by the BW system.

The BAdI information is decoded from string and used to hide the Analysis ribbon with API method `SAPExecuteCommand`.

## Related Information

[SAPListOf](#) [page 84]

[SAPExecuteCommand](#) [page 98]

## 4.14.22 Using Callbacks

Analysis offers different callbacks that are executed with certain events. Callback 'Workbook\_SAP\_Initialize' is always executed. The other callbacks listed below must be registered with the API method 'SAPExecuteCommand' before being used for the first time. To deregister the callbacks, you can also use 'SAPExecuteCommand'.

Every callback can be registered only once in a workbook. If you register one of the callbacks for a second time, the first registration is overwritten.

### i Note

Callback 'Workbook\_SAP\_Initialize' has to be defined in the 'ThisWorkbook' section of the VBA editor. The other callbacks should be defined in a module. If they are defined in the 'ThisWorkbook' or a sheet section, they have to be fully referenced during registration. In the 'ThisWorkbook' section, for example, you have to use `ThisWorkbook.<MacroName>` as a reference.

## Workbook\_SAP\_Initialize

Use this callback to define the initialization logic of the workbook. This callback is similar to the 'Workbook\_Open' event offered by Microsoft Excel, but it is executed after the Microsoft Excel event and after the Analysis standard initialization. If you open a workbook in Microsoft Excel and then activate Analysis, the Excel event is executed when opening the workbook, and the Analysis standard initialization and the callback are executed after the activation of Analysis.

The callback has to be defined as a subroutine without input parameters.

## Example

```
' defined in ThisWorkbook
Public Sub Workbook_SAP_Initialize()
' register callbacks
```

```

Call Application.Run("SAPExecuteCommand", "RegisterCallback", "AfterRedisplay",
"Callback_AfterRedisplay")
Call Application.Run("SAPExecuteCommand", "RegisterCallback",
"BeforePlanDataSave", "Callback_BeforePlanDataSave")
Call Application.Run("SAPExecuteCommand", "RegisterCallback",
"BeforePlanDataReset", "Callback_BeforePlanDataReset")
Call Application.Run("SAPExecuteCommand", "RegisterCallback",
"BeforeMessageDisplay", "Callback_BeforeMessageDisplay")
Call Application.Run("SAPExecuteCommand", "RegisterCallback",
"BeforeFirstPromptsDisplay", "Callback_BeforeFirstPromptsDisplay")
End Sub

```

The 'AfterRedisplay', 'BeforePlanDataSave' and 'BeforePlanDataReset' callbacks are registered during workbook initialization and can therefore be used in the VBA project.

## AfterRedisplay

Use this callback to define the logic that should take place every time the workbook is redisplayed. For example, you can automatically adapt the formatting in the workbook after every redisplay.

The callback has to be defined as subroutine without input parameters. The callback is called only when 'Pause Refresh' on workbook level is false or the user explicitly requested a 'Refresh All'. It is also called only when the active workbook has been changed.

## Example

```

' defined in a module
Public Sub Callback_AfterRedisplay()

ThisWorkbook.Worksheets("Sheet1").Cells(1, 1).Value = "Last redisplay: "
ThisWorkbook.Worksheets("Sheet1").Cells(1, 2).Value = Now()
End Sub

```

With this callback, the text "Last redisplay: " is displayed in cell A1 on sheet 1, and in cell A2, the date and time of the last redisplay are displayed. The information is updated after every redisplay.

## Example

With the following code, you can get the crosstabs and data sources that have been changed since the last user action and that therefore now have been updated in the sheets.

### i Note

These calls work only withing the 'Callback\_AfterRedisplay'. Otherwise they will return an error. Objects that have just been deleted will not be part of the returned array.

```
lResult= Application.Run("SAPGetProperty", "CHANGEDCROSSTABS")
```

An array is returned like for 'SAPListOf' with technical name, name and data source.

```
lResult= Application.Run("SAPGetProperty", "CHANGEDDATASOURCES")
```

An array is returned like for 'SAPListOf' with technical name and name.

## BeforePlanDataSave

Use this callback to define the logic that should be executed every time a user saves plan data. For example, you can execute a planning function (SAPExecutePlanningFunction).

The callback has to be defined as a function returning a Boolean value without input parameters. If the function returns false, the save will not be executed.

### Example

```
' defined in a module
Public Function Callback_BeforePlanDataSave() As Boolean
Dim lResult As Integer
lResult = Application.Run("SAPExecutePlanningFunction", "PF_1")
If lResult <> 1 Then
' planning function execution failed, cancel save
Call MsgBox("Planning Function (PF_1) execution failed. Data will not be
saved.", vbCritical, "Error")
Callback_BeforePlanDataSave = False
Else
Callback_BeforePlanDataSave = True
End If
End Function
```

Every time a user selects the Save button to save planning data, planning function PF\_1 is executed. If the planning function can be executed, the planning data is saved. If the execution fails, a message box with the text "Planning Function (PF\_1) execution failed. Data will not be saved." is displayed.

## BeforePlanDataReset

Use this callback to define the logic that should be executed every time a user resets plan data to the previously saved state. For example, you define that the user is asked if the data really should be reset.

The callback has to be defined as a function returning a Boolean value without input parameters. If the function returns 'false', the reset will not be executed.

### Example

```
' defined in a module
Public Function Callback_BeforePlanDataReset() As Boolean
Dim lAnswer As VbMsgBoxResult
```

```

lAnswer = MsgBox("Do you really want to reset planning data?", vbYesNo, "Reset")
If lAnswer = vbYes Then
    Callback_BeforePlanDataReset = True
Else
    Callback_BeforePlanDataReset = False
End If
End Function

```

Every time a user enters planning data and selects **Back > To previous saved state**, a message box with the text "Do you really want to reset planning data?" is displayed. If the user selects 'Yes', the entered data is reset to the previous saved state. If the user selects 'No', the reset is not executed.

## BeforeMessageDisplay

Use this callback to specify which messages are displayed in the message dialog.

This callback is called whenever the messages dialog is displayed. You can provide some actions that affect which messages will be displayed. Two typical actions are to suppress one or several messages with [SAPSuppressMessage](#) or to add one or several messages with [SAPAddMessage](#).

## Example

```

Public Sub Callback_BeforeMessageDisplay()
    Dim messageList As Variant
    Dim messages As Variant
    Dim lRet As Variant
    Dim messageCount As Variant
    Dim i As Integer

    messageList = Application.Run("SAPListOfMessages", , "True")
    messages = GetAsTwoDimArray(messageList) ' see "Using Analysis functions"
    messageCount = UBound(messages, 1)

    For i = 1 To messageCount
        If messages(i, 5) = "INFORMATION" Then
            lRet = Application.Run("SAPSuppressMessage", messages(i, 1))
        End If
    Next i
End Sub

```

The function `SAPListOfMessages` with parameter `True` lists the messages with detailed information. The details are always listed in a determined sequence.

At position 5, the message severity is listed. The example shows that all messages with severity `INFORMATION` will be suppressed in the message dialog.

## BeforeFirstPromptsDisplay

Use this callback to perform operations before the initial display of the prompting dialog. It is triggered when the prompting dialog is called for the first time and when changing the [Merge Variables](#) property in the

workbook. If the property *Force Prompt for Initial Refresh* is not selected, variable values are set without displaying the prompting dialog.

You can use this callback to overwrite default values before calling the prompting dialog for the first time, to define a workbook variant in the workbook that automatically fills the variable values or to use dynamic values (e.g. 'today' or 'last week') without defining dedicated variables.

## Example

```
' defined in a module
Public Sub Callback_BeforeFirstPromptsDisplay(dpNames As Variant)
    Dim dpName As Variant
    For Each dpName In dpNames
        If dpName = "DS_1" Then
            Call Application.Run("SAPSetVariable", "COUNTRY", "EN", "INPUT_STRING",
"DS_1")
        End If
    Next
End Sub
```

The variable COUNTRY is set to value EN before the prompting dialog is initially displayed. The input variable dpNames contains all data source aliases which are part of the current refresh processing. In this example, it is only asked for one data source: DS\_1.

## Related Information

[SAPExecuteCommand \[page 98\]](#)

[SAPExecutePlanningFunction \[page 104\]](#)

[SAPAddMessage \[page 95\]](#)

[SAPSuppressMessage \[page 116\]](#)

[SAPListOfMessages \[page 89\]](#)

## 4.14.23 Using LastError information

You can use the property LastError of the API method SAPGetProperty to get a more detailed error information when executing an Analysis API method or function. The error information contains a number and an English error text. You can enter as second parameter Text or Number if only this information should be returned. If you do not define the second parameter, both values will be returned.

In the table below, you find a list of all errors with number and text.

Number	Text
General critical errors	

Number	Text
1	A general error occurred.
2	A general error occurred in one of the data sources. This is most probably an error in the backend system or in the communication with the backend system.
3	An error from a previous call is still unhandled.
Errors on pre-execution checks	
10	The BI-Addin is disabled.
11	At least one worksheet is protected.
12	Variable submit is paused.
13	A callback is running.
14	The command is not enabled.
15	Action is currently not allowed.
Parameter errors	
20	A parameter is missing.
21	A parameter has an invalid value.
Analysis object reference errors	
30	No active workbook application.
31	The specified data source is invalid. The alias may be wrong or the data source is not yet refreshed.
32	The alias cannot be resolved. This applies to macros where an alias may be a data source or a planning object.
33	Variables are unmerged, data source alias needs to be specified.
34	No result set is available for the specified data source.
35	At least one data source must have input variables.
36	A data source was specified but variables are merged.
37	At least one data source must be refreshed.
Excel object reference errors	
40	Invalid sheet reference.
41	Invalid cell reference.
General uncritical errors	
50	A general error occurred.
51	Logon failed.
Errors dealing with selection state	
60	Single selection requested but multiple selection is applied.
61	Multiple selection requested but only single selection is allowed.

Number	Text
Errors dealing with interface restrictions	
70	Maximum string length is exceeded, texts are truncated.

## Example

Error Number 1

```
lResult= Application.Run("SAPGetProperty", "LastError", "Number")
```

If you execute this command, it returns '1'.

```
lResult= Application.Run("SAPGetProperty", "LastError", "Text")
```

If you execute this command, it returns the text 'A general error occurred'.

```
lResult= Application.Run("SAPGetProperty", "LastError")
```

If you execute this command, it returns an array with the members { "1", "A general error occurred"}.

## Related Information

[SAPGetProperty \[page 106\]](#)

## 4.14.24 Syntax for Entering Values

Follow these syntax rules when you enter members for filtering and values for prompting in macros.

For compounded dimensions, the complete compounded key of the member must be used.

Description	Example
Equal to	15
Exclude value	!22
Value range	1 - 5
Exclude value range	!6 - 9
Greater than	>8
Exclude values greater than <value limit>	!>8
Greater than or equal to	>=8
Exclude values greater than or equal to <value limit>	!>=8
Less than	<12

<b>Description</b>	<b>Example</b>
Exclude values less than <value limit>	!<12
Less than or equal to	<=12
Exclude values less than or equal to <value limit>	!<=12
Contains pattern (for example, all values that begin with A)*	A*
Exclude values that contain pattern (for example, exclude all values that begin with A)*	!A*
Dimension hierarchy node	+<Dimension Attribute>(<Technical Name of Dimension> For example: +ELEMENT1(WBS_ELEMENT)
Text hierarchy node	+<Technical Name of Hierarchy Node>(OHIER_NODE) For example: +EUROPE(OHIER_NODE)
Delimiter for multiple values; semicolon followed by a space	1-5; >12; !8
Multiple values may also be passed as an array of strings	1-5 >12 !8
Escape character; backslash without space	\  If you enter a string that contains one of the special characters used in these syntax rules, you can use a backslash as escape character.  For example: Enter the string CA-QU as CA\ -QU.

\*You cannot use these syntax rules for filtering.

# 5 Analyzing Data

On the Analysis tab in the ribbon, you have different options to analyze your data.

Using the design panel, you can analyze the data and change the view on the displayed data. You can add and remove dimensions and measures to be displayed with drag and drop. To avoid single refreshes after each step, you can pause the refresh to build a crosstab. After ending the pause, all changes are applied at once.

You can refine your analysis using filters, prompting, calculations and display hierarchies. You can also calculate new measures based on existing ones and define the display of members, measures and totals in the crosstab.

With Microsoft Excel functionality, you can add a note to a cell by inserting a comment. You can edit the text in comments and delete comments that you no longer need.

You can also plan business data based on the current data in your data source. You can enter the planning data manually and you can enter planning data automatically using planning functions and planning sequences of SAP BW Integrated Planning.

## Related Information

[Analysis tab \[page 13\]](#)

[Analyzing data with the design panel \[page 136\]](#)

[Prompting \[page 153\]](#)

[Filtering data \[page 162\]](#)

[Sorting data \[page 176\]](#)

[Working with hierarchies \[page 178\]](#)

[Calculating new measures \[page 184\]](#)

[Defining the display of members, measures and totals \[page 190\]](#)

[To jump to a target with Goto \[page 197\]](#)

[Commenting data cells \[page 198\]](#)

[Analyzing Data \[page 136\]](#)

[Planning Data \[page 202\]](#)

## 5.1 Analyzing data with the design panel

The Analysis design panel is an additional element on the user interface that you can use to create new views on your data, to find information on the used data sources and on the components of the workbook. The design panel consists of several tabs, for example *Analysis*, *Information* and *Components*.

You can show and hide the design panel by choosing **► Display ► Display Design Panel** in the design panel group. You can modify the size and position of the design panel in Microsoft Excel. Your modifications of the design panel will be preserved even if you close Analysis or hide the design panel and show it again.

You can also show and hide the technical names in the design panel by choosing **► Display ► Show Technical Names** in the design panel group. The technical names are displayed in squared brackets.

On the Analysis tab, you can show and hide the property view on the design panel by choosing **► Display ► Show Property View**. There, you can also decide to display dimensions as grouped dimensions by choosing **► Display ► Show Dimension Grouping**.

Working with the design panel, the crosstab is updated after each navigation step. You can choose *Pause Refresh* to deactivate the permanent refresh.

## Related Information

[The Analysis tab \[page 137\]](#)

[The Property View on the Analysis tab \[page 139\]](#)

[The Information tab \[page 141\]](#)

[The Components tab \[page 142\]](#)

[The Design Rules tab \[page 147\]](#)

[The Comments tab \[page 148\]](#)

[The Versions tab \[page 149\]](#)

[Pausing Refresh \[page 149\]](#)

[Managing components in the design panel \[page 150\]](#)

[Managing system connections in the design panel \[page 152\]](#)



## 5.1.1 The Analysis tab



On the *Analysis* tab, you can see the available fields for a single data source and the fields currently used to display the data in a crosstab in columns and rows. If you use multiple data sources in your analysis, select a crosstab cell of the required data source to specify which data source information should be displayed.

The Analysis tab contains the following sections:

- *Search* field  
You can search for a numeric or character string. The search results are highlighted.  
In the sections Columns, Rows and Background Filter, the relevant nodes are expanded automatically.  
Due to performance reasons, the nodes in the <Data source> section are not expanded automatically. If you expand them manually, you can see the highlighted search results. By expanding the nodes manually, the data has been loaded and the nodes will be expanded automatically in future searches.
- <Data source>  
The heading for this section is the name of the selected data source. You can check whether you have selected the right data source. The section displays all fields of the data source: measures, dimensions, attributes and hierarchies.

You can move a data source field to another section on the Analysis tab to add it to the crosstab. If you move a field to another section, its name is displayed in both sections: the data source section and the other crosstab section, for example *Rows*. Dimension, attribute and hierarchy fields that are added to another crosstab section are displayed in bold letters in the data source section.

In BEx Query Designer, a dimension can be defined as not to be displayed. These dimensions are displayed in the Analysis design panel in gray font. You can add them to your crosstab and filter them. This will affect the data and result in your crosstab, but the dimensions will not be visible in the crosstab. If you want a dimension to be displayed in an Analysis crosstab, you have to change the definition in BEx Query Designer. In SAP BW and SAP HANA systems, you can group dimensions in data sources, for example, grouping all dimensions containing geographical information. In Analysis, you can decide if the dimensions of a data source are displayed as a flat list or as grouped dimensions. You can change the display by choosing  *Display* > *Show Dimension Grouping*  in the ribbon. The selected display remains until you change it again, even if you leave Excel meanwhile.

- *Columns*  
This section contains all fields that are currently displayed in columns.
- *Rows*  
This section contains all fields that are currently displayed in rows.  
Next to the section header, you can select the Swap Axes icon to swap rows and columns.
- *Background Filter*  
This section contains all fields that are currently defined as background filters.
- *Properties*  
In this section, you can define properties for the data source, measures and dimensions. You can show or hide this section by choosing  *Display* > *Show Property View*  in the design panel group.

## Navigating through the data




Using the Analysis tab, you have the following options to analyze data and create new views on it:

- You can use the context menu to move fields in the four sections. You can add a new dimension from the <Data source> section to the Rows, for example.
- You can use drag and drop to move fields in the four sections.
- You can use drag and drop to add rows and columns to the crosstab. Drag a field directly to the crosstab and drop it. The field is added automatically to the corresponding section on the Analysis tab.
- You can also use drag and drop to remove a field from the crosstab. Select the border of a cell in the crosstab to drag the field to the Analysis tab and drop it there.
- You can use the context menu to create, change and remove filters. If a filter is defined for an object, you see a filter icon beside this object.
- You can swap the axis of the crosstab by selecting the icon next to the rows section.

## Related Information

[The Property View on the Analysis tab \[page 139\]](#)

## 5.1.2 The Property View on the Analysis tab

In the Property view on the Analysis tab, you can define properties for the data source, measures and dimensions in the workbook. You can show or hide this section by choosing  *Display*  *Show Property View*  in the design panel group.

The properties can also be changed using the ribbon. If you change a property on the Analysis tab, the changes are also reflected in the ribbon.

The available properties depend on the selected data source element. At the top of the Properties section, you will always find the description and technical name of the selected data source element. You can select the data source, measures, a single measures, dimensions, members and hierarchies.

For single measures, dimensions and attributes, you can enter a new description and thereby rename the selected data source element. The new description is directly available in the Analysis UI: in the design panel, in the crosstab and in the dialogs, for example the Filter dialog. You can also save new descriptions with a view in Analysis. Depending on the platform you use, this could be a query view, an Analysis view or a HANA view.

For hierarchies, only the properties of the assigned hierarchy can be changed.

### Data Source Properties

Property	Options
Zero Sup- pression	Zero Suppression in Rows
	Zero Suppression in Columns
Compact Display	Compact Display in Rows
	Compact Display in Columns
Totals	Total Rows Above Members
	Total Columns Left of Members
General Format	Display of Negative Values
	Display Zero as

### Measures Properties

Property	Options
Members	Display
	Use
Currency Conversion	You can open the currency conversion dialog.

### Single Measure Properties

Property	Options
Scaling Factor	You can select the scaling factor to be used.
Decimal Places	You can select the decimal places to be used.

#### Dimension Properties

Property	Options
Members	Display
	Use
Totals	Show Totals
	Hide Totals
	Hide Totals if Only One Member is Available
Sort	Sort Ascending
	Sort Descending
	You can also open the sorting dialog.

#### Member Properties

Property	Options
Members	Display
	Use
	Result Set Access Mode*
	*This option is only available if the setting AllowChangingAccessMode is set to true. You can do this in the Advanced options dialog or the Technical Configuration in the Analysis backstage area.
Totals	Show Totals
	Hide Totals
	Hide Totals if Only One Member is Available

#### Hierarchies Properties

Property	Options
Expand	Expand Downwards / Expand to the Right
	Expand Upwards / Expand to the Left
Show	Nodes with only One Lower-Level Node
	Values of Posted Nodes

## Related Information

[To show/hide zeros in rows and columns \[page 175\]](#)

[To display single dimensions as hierarchy \[page 182\]](#)

[To locate the totals display in the crosstab \[page 195\]](#)

[To define the display of zeros and negative values \[page 194\]](#)

[To define the number format \[page 192\]](#)

[To define currency translation \[page 193\]](#)

[To define the members display \[page 190\]](#)

[To show or hide totals in the crosstab \[page 195\]](#)

[To sort members \[page 178\]](#)

## 5.1.3 The Information tab

On the *Information* tab, you can see detailed information about a data source or the complete workbook. You can also find information on filters and variables on this tab. The general information is displayed as text elements.

In the *Information for* list, you can select the complete workbook or one of the inserted data sources. The information fields are displayed for the selected object. You can insert these fields in the analysis using drag and drop.

For a workbook, the following information is displayed:

- Workbook Name
- Created By
- Variables
- Logged On User
- Last Refreshed At
- Created At
- Last Changed At

For a data source, the following information is displayed:

- Data Source Name
- Key Date
- Last Data Update  
For MultiProviders, two dates are displayed: The date when all InfoProviders of the MultiProvider were updated successfully for the last time (LastDataUpdate) and the date when a single InfoProvider of the MultiProvider was updated for the last time (LastDataUpdateMaximum).
- Variables
- Filter
- BEx Conditions
- Query Technical Name
- InfoProvider Technical Name
- InfoProvider Name
- Created By
- Last Changed By
- Last Changed At
- System
- Logged On User

## Related Information

[To insert an info field \[page 43\]](#)

[To insert a filter \[page 43\]](#)

## 5.1.4 The Components tab

On the *Components* tab, you can see a list of all components used in the workbook together with the properties of these components. You can select if the components should be listed by data source or by sheet. In both cases, the highest node of the list is the workbook. Below this, the data sources or the sheets are displayed with their components.

You can use the context menu to manage the components. For more information, see [Managing components in the design panel \[page 150\]](#).

A set of properties is available for each component. To see and change the properties for a component, select the component in the structure.

The following sections list the available components and their properties.

### i Note

Depending on the platform and data source you use, you might see only a subset of these components.

## Workbook

If you select a workbook, you will find properties on two tabs: *General* and *Planning*.

The *General* tab contains the following properties:

Workbook Properties	Description
Name	Displays the name of the workbook. The workbook name is defined when the workbook is saved.
Refresh Workbook on Opening*	If you select this check box, the data sources in the workbook are refreshed every time the workbook is opened.  If this check box is not selected, the data in the workbook is not automatically refreshed on opening. You can refresh the data sources manually by choosing <i>Refresh All</i> in the menu.
Force Prompt for Initial Refresh*	If you select this check box, the prompting dialog is displayed on every refresh.

Workbook Properties	Description
Subsequent Refresh	<p>With this property, you can define the behavior for executing <i>Refresh All</i>. You have the following options:</p> <ul style="list-style-type: none"> <li>• <i>Transaction Data Only (default)</i> If the data sources in the workbook are offline and you execute <i>Refresh All</i>, the data sources get online. If you execute Refresh All again, only the transactional data is updated.</li> <li>• <i>Log Off and Reconnect</i> If the data sources in the workbook are offline and you execute <i>Refresh All</i>, the data sources get online. If you execute Refresh All again, the data sources are logged off and reconnected again. All data in the data sources (for example master data) are updated. It has the same behavior as using the API method SAPLogOff with parameter True: Application.Run("SAPLogOff", True). If some data sources in the workbook are online and some are offline and you execute <i>Refresh All</i>, only the connected data sources are logged off and reconnected.</li> </ul>
Store Prompts with Workbook*	<p>If you select this check box, the defined prompt values for characteristic value variables are saved with the workbook. These variables are defined in the BEx query designer in the <i>Default Values</i> area with refresh behavior <i>Refresh as Designed</i> or in the <i>Characteristic Restrictions</i> area.</p>
Remove Data Before Saving	<p>If you select this check box, the workbook is saved without the data. When you reopen the workbook, no data is displayed. To display the data, refresh the data sources manually by choosing <i>Refresh All</i> in the menu.</p>
Merge Variables*	<p>If you select this check box, the variables are merged for all data sources. If the check box is not selected, the variables can be defined for each data source separately.</p> <p>Every time you change your selection for this property in the workbook, the connected data sources are restarted. Therefore the prompting dialog will appear automatically so that you can define the values before the data sources are refreshed. If the data sources are not connected, the dialog will not appear and only the property will be changed.</p> <p>In the default setting, this check box is not selected when you create a new workbook. The default setting can be changed in the registry. For more information, contact your IT Administrator.</p>

### i Note

The following functions and API methods might need different parameters depending on the selection for merging variables:

- SAPListOfMembers
- SAPListOfVariables
- SAPCallMemberSelector
- SAPSetVariable

\* For more information on workbook properties, see [To select workbook properties for prompting \[page 160\]](#).

The *Planning* tab contains the following properties:

Workbook Properties	Description
System for Data Write Back	<p>Displays the currently selected planning system.</p> <p>For an empty workbook, this property is empty.</p> <p>When the first data source is inserted, the corresponding system is visible as the planning system.</p> <p>When you insert data sources from different systems, the system of the first inserted data source is the planning system. You can change the planning system by selecting another system in the list. The changed property value will only become effective when you close and reopen the workbook.</p> <p>When you change the planning system in a workbook before you are connected to a server, the changed property will become effective when you connect to a server. In this case, you do not need to close and reopen the workbook again.</p>
Cell Locking	<p>You can lock input-ready cells against manual changes in planning workbooks. A locked cell is a time-limited setting that only applies to the current user session. Locked cells are displayed in a different cell style.</p> <p>There are two implementations of the cell locking function in Analysis: Front-end cell locks and back-end cell locks (single query and cross query).</p> <p>You can select here which implementation should be applied in a workbook.</p> <p>For more information , see <a href="#">Locking Cells [page 209]</a>.</p>
Planning Model	
Switch All Data Sources to Change Mode for Initial Refresh	<p>If you select this check box when saving a workbook, the workbook (containing input-ready data sources) will be opened in display mode.</p> <p>With the initial refresh, the data sources will be switched to change mode automatically.</p> <p>By default, this check box is not selected when you create a new workbook.</p>
Refresh Planning Objects on Refresh All	<p>If you select this check box, planning objects in the workbook are refreshed when a refresh is executed with <i>Refresh All</i>.</p> <p>By default, this check box is selected when you create a new workbook.</p>

## Data Source

Data Source Properties	Description
Data Source Name	Displays the data source name. You can change it here.
Data Source Technical Name	Displays the technical name of the data source.
Formula Alias	Displays the formula alias. You can change it here.
System Alias	Displays the system alias.

Data Source Properties	Description
Use Data Source Formula-Optimized	You can use this check box to switch between the formula-optimized mode and the analysis mode.
Planning: Open Data Source Input-Enabled	<p>If you select this check box, the input-ready data source is opened in change mode when you open the workbook, and you can immediately start entering planning data.</p> <p>If you do not select this check box for an input-ready data source, you cannot enter and save planning data for this data source until you have selected the check box.</p>
Behaviour on Opening Workbook	<ul style="list-style-type: none"> <li> <b>Default Refresh Behaviour</b>            The data source is refreshed as defined on workbook level with the setting <a href="#">Refresh Workbook on Opening</a>.         </li> <li> <b>Reset</b>            The initial state of the data source defined in the back-end, will be displayed when you reopen the workbook. You cannot save a navigation state with that workbook as the navigation state will be replaced with the initial state.         </li> <li> <b>Reset and Clear Design Rules</b>            The initial state of the data source defined in the back-end, will be displayed when you reopen the workbook. The existing Table Design rules are deleted and will be no longer available when you reopen the workbook. You cannot save a navigation state with that workbook as the navigation state will be replaced with the initial state.         </li> </ul>
Document Store	<p>You can select the document store for a data source.</p> <p>Document stores for data sources are defined in the BW system.</p>
Document Filter	<p>You can create and/or select a document filter for a data source.</p> <p>Document filters can contain comments for one data source or for all data sources of a workbook.</p>

## Sheet

Sheet Properties	Description
Name	Displays the sheet name.
Sheet Type	<p>Displays the sheet type.</p> <p>The following sheet types are available: Analysis, Neutral and Non-COF.</p> <p>Empty sheets are defined as neutral. If you add a data source with the Analysis plug-in, the sheet type Analysis is assigned. To reset a sheet to type neutral, you have to remove all inserted data sources from the sheet.</p> <p>Sheets of type Non-COF are not checked from the Analysis Add-In. It might be useful to assign this type to sheets that do not contain Analysis content to improve the performance.</p>

## Crosstab

Crosstab Properties	Description
Name	Displays the crosstab name. You can change it here.
Formula Alias	<p>Displays the formula alias of the crosstab. You can change it here.</p> <p>The crosstab formula alias is the second part of the named range for a crosstab. The first part is always SAP. For example, if the crosstab formula alias is CrossTab1, the named range for this crosstab would be SAPCrossTab1. You can select a crosstab with its named range in the Name box in the formula bar.</p>
Data Source Name	Displays the data source name and the data source alias.
Range	Displays the cell range of the crosstab in the sheet. You can change the range to move the crosstab in the sheet.
Planning: Number of New Lines	You can define how many new lines should be added to the crosstab for planning data. The default number is 5 lines. The new lines are only visible in change mode.
Planning: New Line Validation	<p>You can select which data should be checked automatically:</p> <ul style="list-style-type: none"> <li>• None No automatic check is executed.</li> <li>• Member Automatic check runs whenever a member was changed.</li> <li>• Lines Automatic check runs whenever a line was changed and the derivation is done automatically.</li> </ul>
Apply Default Formats	If you select this check box, the default formats are applied after each navigation step.
Display Symbols for Parent Members	If you select this check box, a symbol is displayed for parent members in a hierarchy (+/-).
Optimum Cell Width/Height	If you select this check box, the table is displayed with optimum cell width and height.
Repeat Members	<p>If you select this check box, texts that are not displayed because they occur several times in a column or row are repeated in each cell.</p> <p>The setting made in BEx Query Designer to hide/show repeated key values is not evaluated in Analysis. To define the display for repeated members only this setting in the design panel is evaluated.</p>

## Planning Objects

Planning Objects Properties	Description
Name of Planning Function / Sequence	Displays the name of the planning object.
	Displays the technical name of the planning object. Technical name of Planning Function / Sequence

Planning Objects Properties	Description
Alias	Displays the alias of the planning object. You can change it here.
Process Changed Data	<p>If you select this check box, you can define that only data changed by the user in the current session since the last save will be processed.</p> <p>For planning functions, you can specify which aggregation level should be used as a filter to define the changed data. In the default setting, the aggregation level the planning function is built on is selected for a planning function.</p> <p>For planning sequences, it is not possible to select an aggregation level.</p>
Variables of Planning Function / Sequence	<p>You can define the variables values of the planning object:</p> <ul style="list-style-type: none"> <li>• For source <i>Member</i>, you can select the members with input help.</li> <li>• For source <i>Cell</i>, you can define the cell that contains the values.</li> </ul>
Filter	On the <i>Filter</i> tab, you can define the filter for a planning function.

For more information on planning objects, see [To plan data with planning objects \[page 206\]](#).

## Filter

Filter Properties	Description
Data Source Name	Displays data source name.
Dimension	Displays the name of the filtered dimension.
Filter applies to the following data sources	You can select which of the data sources in the workbook should be filtered.

## Chart

Chart Properties	Description
Name	Displays sheet and chart name.
Cross Tab Name	Displays the crosstab name.
Sheet	Displays the sheet name. You can change it here.

### 5.1.5 The Design Rules tab

On the *Design Rules* tab, you can display and maintain the rules created with Table Design. The rules are displayed for each data source and rule type.

You can select the data source in the *Rules For* field. If a workbook contains only one data source, then this data source is preselected.

You can copy the rules from one data source to another: Select the copy icon next to the data source. Then select another data source and the paste icon. The rules are added to the list. After copying the rules, they are no longer connected. If you edit a rule, the copied one will not be changed automatically.

In the *Rule Type* field, you can select a rule type. The following rule types are available:

- Format
- Formula
- Texts
- New Lines

The rules for the selected data source and rule type are displayed in the list. You see a name and a description for each rule. The information in the list is filled in automatically. If a rule is deactivated, the corresponding icon is displayed in front of the name.

Rules of type *Format* can be edited from the design panel. If you select a rule and *Edit*, a dialog box appears where you can change the rule. For rules of type *New Lines*, you can edit the Row Height / Column Width.

Rules of type *Formula* can be edited in the design panel. Only formulas added to data cells of an input-ready planning data source can not be edited in the design panel.

You can select a rule and *Copy Rule* to copy the rule. The new copied rule is added to the list with the same description as the original rule but with a different name.

You can select a rule and *Delete*, to delete it completely.

You can select a rule and *Deactivate* to deactivate it, if you don't want the rule to be applied at the moment. If you deactivate a rule, the icon is displayed in front of the rule name. You can activate the rule again.

You can use the arrows above the list to move the rules up and down in the list to change their priority.

If you select the button *Mark Rules Based on Crosstab Selection* and you select a cell in the crosstab containing a rule, the rule is highlighted in the list.

## Related Information

[Editing crosstabs with Table Design \[page 47\]](#)

## 5.1.6 The Comments tab

On the *Comments* tab, you can see a section that shows the comment history for a selected cell and a section that shows the context information for the cell.

In the *History* section, you can see different comment versions that were saved for a data cell. The comment history is only available for comments of SAP BW/4HANA data sources:

1. Enter a comment for a selected data cell and select *Update History* on the *Comments* tab.

The comment is displayed in the design panel. If you edit the comment, the text in the design panel is also changed.

2. Save the comment.
3. Edit the comment.

The new version of the comment is added in the design panel and the saved version is shown the earlier version.

Below the *History* section, the *Context* section shows the context information for a selected cell with the following information: data source name and context information (member and measure names). This information for all data sources.

## 5.1.7 The Versions tab

A version is a set of data from the database, viewed in a table. To control who can access a version, you make the versions public, private, or shared.

On the *Versions* tab in Analysis, you can see the versions for a SAP Analytics Cloud data source and properties and a history for each version. If you use multiple data sources in your analysis, you can select between the data sources and the corresponding versions will be displayed.

This tab is only visible in the design panel when a SAP Analytics Cloud model is inserted as data source.

The Versions tab contains the following sections:

- *Versions for*: field  
In this field, you see the name of the data source that is currently selected. If you use multiple data source in your analysis, you can use the drop down list to select another one.
- *Category*: field  
In this field, you can select by category which versions should be displayed on the Versions. If you select *All*, the versions for all categories are displayed. If you select a special category, for example *Planning*, the versions for this category are displayed.
- *Properties*  
In this section, you can find general and history information for a selected version. On the *General* tab, you can see general information such as the name, the category and the type of the selected version. On the *History* tab, you can see a list of the steps executed for this version. You can select a step and undo or redo it. You can also revert changes executed for a version. After reverting the changes, the steps on the History tab are deleted.

You can find more information in the SAP Analytics Cloud documentation at [About Version Management](#).

## 5.1.8 Pausing Refresh

You can choose *Pause Refresh* to deactivate the refresh after each navigation step when you work on the *Analysis* tab in the design panel. With the deactivated refresh, you can design a new view on your data with multiple navigation steps. All steps will be executed at once when you choose *Pause Refresh* again to activate the refresh. Note that most functions in the menu are not available when Pause Refresh is activated. You should only work on the Analysis tab when the automatic refresh is deactivated.

You can also activate *Pause Refresh* before inserting a data source. You can insert a data source and define your analysis on the *Analysis* tab in the design panel (columns, rows and filter). The crosstab with the data source and the defined analysis will only be inserted after *Pause Refresh* is deactivated again.

The following functions are available in the context menu of the design panel if *Pause Refresh* is selected:

- Define the members display  
More information: [To define the members display \[page 190\]](#)
- Define the number format  
More information: [To define the number format \[page 192\]](#)
- Show or hide totals in the crosstab  
More information: [To show or hide totals in the crosstab \[page 195\]](#)
- Define the calculation of totals  
More information: [To define the calculation of totals \[page 196\]](#)
- Calculate a new measure based on available measures  
More information: [To calculate a new measure based on available measures \[page 185\]](#)
- Add a new measure based on one available measure (dynamic calculation)  
More information: [To add a new measure based on one available measure \[page 186\]](#)

## 5.1.9 Managing components in the design panel

On the *Components* tab, all components included in a workbook are displayed in a tree-like structure. You can select if the components should be listed by data source or by sheet. In both cases, the highest node of the list is the workbook. Below this, the data sources or the sheets are displayed with their components.

You can use the context menu to manage the components. The following sections describe the functions of the context menu.

### To insert a component

On workbook level, you can insert the following components:

- data sources ('Use Data Source')
- planning functions ('Use Planning Function')
- planning sequences ('Use Planning Sequence')

On data source level, you can insert a crosstab for this data source ('Insert Crosstab'). A dialog box appears to enter the sheet and cell information where the copied crosstab should be inserted.

On the planning objects level, you can insert planning functions ('Use Planning Function') and planning sequences ('Use Planning Sequence')

### To refresh a component

You can refresh the following components with the context menu:

- workbooks
- crosstabs
- charts

## To copy a component

You can copy the following components with the context menu:

- data sources  
When you copy a data source, the copied data source is added to the tree-like structure with a new data source alias. No crosstab or other component of the original data source is added to the workbook.
- crosstabs  
When you copy a crosstab, a dialog box appears to enter the sheet and cell information where the copied crosstab should be inserted. The new crosstab is inserted on the defined position and in the structure, a entry with data source, data source alias and crosstab is added.
- sheets  
When you copy a sheet, the copied sheet is added to the tree-like structure.

## To move a component

You can move crosstabs and charts ('Move To'). A dialog box appears to enter the sheet and cell information where the crosstab or chart should be moved to.

## To jump to a component

You can use the context menu to jump to a crosstab or a filter ('Jump To'). You can use this function to jump to a crosstab on another sheet in the workbook, for example. The sheet with the selected crosstab will be displayed.

## To execute planning objects

You can use the context menu to execute planning functions and planning sequences.

## To delete a component

You can delete the following components with the context menu:

- data sources

- crosstabs
- charts
- filters
- planning functions
- planning sequences

## 5.1.10 Managing system connections in the design panel

On the *Components* tab, you can manage the system connections on workbook level. You can log off from all systems the workbook is connected to. You can also reconnect to the systems or replace a system with another system.

You can use the context menu to manage the system connections. The following sections describe the functions of the context menu.

### To log off a workbook

To disconnect a workbook from the server (BW and HANA), select the workbook and choose *Log Off* in the context menu.

The connections to all systems in the workbook are closed and the design panel is closed.

You can also disconnect a workbook using the API method `SAPLogOff`.

### To reconnect a workbook

To reconnect to the systems, display the design panel, select the workbook and choose *Refresh All* in the context menu.

All systems in the workbook are reconnected.

You can also reconnect a workbook using the API method `SAPLogOff`.

### To replace a system in a workbook

To replace a system, select *Replace System...* in the context menu. The option is only enabled if the workbook is not connected to any system.

In the dialog, you can select the system that you want to replace (Current System) and the system that you want replace the current system with (Replace By System).

For BW and HANA systems, you can replace local connections and connections from the BI platform. If a workbook contains BW and HANA connections, you will find two tabs in the dialog: BW and HANA.

To list the BIP connections in the *Replace By System* section, select the *Logon to BIP* icon and log on to the BI platform.

You can also open the dialog for replacing systems in the Platform options.

## Related Information

[SAPLogOff \[page 107\]](#)

[Platform Options \[page 303\]](#)

## 5.2 Prompting

In the prompting dialog, you can set values for prompts. Prompts are defined as variables in BEx Query Designer. The variables are parameters of a BW query and are filled with values when you insert a query as a data source in a worksheet. They serve as placeholders for members, hierarchies, hierarchy nodes, formula elements and texts. After defining the variable values, the crosstab is displayed according to the selected values.

To be editable in the prompting dialog, a variable must be defined as input-ready and set to Manual Input in Query Designer. A variable can be defined as mandatory or optional in Query Designer.

The prompting dialog appears automatically when you insert the query with variables in a worksheet. You can open the prompting dialog manually with the prompting icon in the menu to change variable values.

At workbook level, you can select properties to define how the prompting dialog and defined values behave in workbooks. For example, you can define that variable values are saved in a workbook. If you open this workbook again, the data is displayed according to the defined values. You can then open the prompting dialog and change the values.

Defined prompt values can be saved as variant for a query or for a document (workbook or presentation) using the variant dialog in the prompting dialog.

If there are no input-ready variables in the query that you use as your data source, the prompting dialog is disabled.

### i Note

Objects that are defined as input-ready variables in BEx Query Designer are called prompts in Analysis for Microsoft Office. In the following documentation the Analysis term *prompt* is used for these variables. A query can also contain other variable types. All variables and values of these variables that a query contains, are displayed in the *Variables* area on the *Information* tab in the design panel.

To find more information about the definition of variables in the BEx Query Designer, see the documentation on the SAP Help Portal at [Variables](#).

## Related Information

[To define prompt values \[page 154\]](#)

[Using Variants \[page 156\]](#)

[To select workbook properties for prompting \[page 160\]](#)

## 5.2.1 To define prompt values

### Procedure

1. Open the *Prompts* dialog.

In the *Prompts Summary* area, you see a list of all available prompts in the query and the selected values. In the *Specify Value for Prompts* area, you can expand single prompts to define values. The mandatory prompts are marked with an asterisk. If default values for the prompts are defined in BEx Query Designer, they are displayed as selected in the prompting dialog.

If you have selected to not merge the variables for this workbook on the *Components* tab in the design panel, the prompts are displayed separately for each data source and you can define different values for the same prompt for the corresponding data sources.

2. In the *Display* list box, select whether all prompts, no prompts or only the mandatory prompts are expanded in the *Specify Value for Prompts area*:
  - Select *Hide All* if all prompts should be displayed collapsed.
  - Select *Show All* if all prompts should be displayed expanded. This is the default option.
  - Select *Hide Optional Prompts* if only the mandatory prompts should be displayed expanded.
  - Select *Show Technical Names* if the technical names should be displayed additionally.
3. Define values for the expanded prompts.

You can define values for the following prompt types. Depending on the prompt definition in BEx Query Designer, you have various options:

Prompt Type	Value Definition
<b>Dimension*</b>	Single value You can select one member as the prompt value for this dimension.
<b>Dimension*</b>	Multiple value You can select multiple members as the prompt values for this dimension. Press the <b>+</b> button, to add a field for additional values. Press the <i>Filter</i> button if you want to select multiple values in a list at once. To remove a selected member, press the red <i>X</i> button.
<b>Dimension*</b>	Value range You can select a start and an end member to define a range of values for this dimension.

Prompt Type	Value Definition
	<p>You can also enter the start and end member manually in one field, separated with space-minus-space ( - ). When applying the values, Analysis moves the end member automatically to the second field. You can apply the values with selecting the tab key or selecting another field in the dialog.</p>
<b>Dimension*</b>	<p>Selection option</p> <p>You can select an operator and corresponding members to define a selection for this dimension. The following operators are available:</p> <ul style="list-style-type: none"> <li>○ = Equal To</li> <li>○ != Not Equal to</li> <li>○ &gt;= Greater Than or Equal To</li> <li>○ &gt; Greater Than</li> <li>○ &lt; Less Than</li> <li>○ &lt;= Less Than or Equal To</li> <li>○ [] Between</li> <li>○ ![ ] Not Between</li> <li>○ CP (Contains Pattern) / !CP (Excludes Pattern)</li> </ul> <p>The use of this operator must be enabled in the file system or in the Analysis backstage area (Technical Configuration). For more information, contact your IT administrator.</p> <p>You can add multiple selections for this dimension. Press the + button, to add a selection. To remove a selection, press the red X button.</p> <p>Definitions with other operators that are available in BEx Query Designer 3.5, can be displayed and deleted but not edited.</p>
<b>Hierarchy</b>	You can select a hierarchy from the list box to define a prompt value.
<b>Hierarchy node*</b>	<p>Single value</p> <p>You can select one hierarchy node as the prompt value for the selected hierarchy.</p>
<b>Hierarchy node*</b>	<p>Multiple value</p> <p>You can select multiple hierarchy nodes as the prompt values for the selected hierarchy.</p> <p>Press the + button, to add a field for additional value. Press the <i>Filter</i> button if you want to select multiple values in a list at once. To remove a selected member, press the red X button.</p>
<b>Formula</b>	<p>You can enter a numeric value.</p> <p>The numeric value is used for measure calculations defined in BEx Query Designer with prompts.</p>
<b>Text</b>	<p>You can enter a text.</p> <p>This prompt type can be used to assign text to columns or row headers or to change the description of a calculated measure.</p>

\* For these prompt types, you can enter the values manually or using input help. When you open input help, no values are selected. You can search for members or select members directly from the list. You can also define if the members in the list should be displayed with key, text, or both. For hierarchical dimensions, you can define if the members should be displayed in hierarchies, at leaf level, or on a specific level.

4. Press *OK* to apply the selected values to your data.

The system validates all prompt values. If the validation is successful, the crosstab is displayed according to your selection.

## Results

The data in the worksheet is displayed according to your selection.

If you select a data source on the *Information* tab in the design panel, you can see the defined values in the *Variables* area. If you select the workbook on the *Information* tab in the design panel, you can see the defined values in the *Variables* area if the variables in the workbook are merged. If the variables are not merged, the message 'Variables are not merged' is displayed in the *Variables* area.

The defined prompt values can be saved with the workbook and are then available to other users or in other sessions. To change the prompt values, open the prompting dialog again.

If you need to define new prompt values for a data source, you can select *Clear All* in the *Prompts* dialog to remove the defined values for all prompts and define new values.

## Related Information

[To select workbook properties for prompting \[page 160\]](#)

### 5.2.2 Using Variants

You can save defined prompt values as a variant for a query or for a document (workbook or presentation). The variants of a document can contain prompt values for different data sources. You can create a variant for your user (user-specific variant) or for all users (global variant). When you next open the workbook or query, you can apply your user-specific variants and all global variants. You can also change a variant by selecting different prompt values and saving the variant again. Variants can also be deleted.

You can create multiple variants, which allows you to choose different prompt values. Only one variant can be active at any one time however.

Variants are created and maintained in the prompting dialog. There are two different modes when working with variants. These are document mode for working with document variants and data source mode for working with query variants. An icon shows which mode is active. The currently active mode depends on the selection you made for workbook property *Merge Variables* on the Components tab in the design panel and on how you open the prompting dialog.

## Working in data source mode

In data source mode, you see the variants for this query in the prompting dialog. If you create a new variant, it is always saved to this query. The following prerequisites apply for working with query variants:

- Authorization object S\_RS\_PARAM must be maintained for the user in the SAP BW server so that the user can create a variant.
- You open the prompting dialog with the context menu on the crosstab. The context menu for prompts is only available if workbook property *Merge Variables* is not selected.
- Workbook property *Merge Variables* is not selected when a new query is inserted.

## Working in document mode

In document mode, you see the variants for this workbook or presentation in the prompting dialog. If you create a new variant, it is saved to this document. Depending on the platform where the document is saved, the following prerequisites apply for working with document variants:

Prerequisites for document variants using the SAP BW platform:

- Authorization object S\_RS\_PARAM must be maintained for the user in the SAP BW server so that the user can create a variant.
- The workbook is already saved on the SAP BW server.
- You open the prompting dialog with the icon in the ribbon.
- Workbook property *Merge Variables* is selected when a new query is inserted.

Prerequisites for document variants using the SAP BusinessObjects BI platform:

- You use at least version 4.2 SP 4 of SAP BusinessObjects BI platform.
- In the Central Management Console, properties of the Analysis runtime, the *Variants Support* is set to *User* or *User and Global*.
- *Web Intelligence Services* are installed on the SAP BusinessObjects BI platform.  
The Web Intelligence Services are part of the Servers instance in the SAP BusinessObjects BI platform features installation.
- The workbook is already saved on the SAP BusinessObjects BI platform.
- You open the prompting dialog with the icon in the ribbon.
- The workbook property *Merge Variables* is selected when a new query is inserted.

## Related Information

[To create a user-specific variant \[page 158\]](#)

[To create a global variant \[page 158\]](#)

[To apply a variant \[page 159\]](#)

[To change a variant \[page 159\]](#)

[To delete a variant \[page 160\]](#)

[To select workbook properties for prompting \[page 160\]](#)

## 5.2.2.1 To create a user-specific variant

### Procedure

1. Open the prompting dialog in the required mode.
2. Define the prompt values for the variant.
3. Enter a name for the variant in the *Use Variant* field.
4. Press the save icon.

### Results

The new variant can now be selected in the drop down list.

## 5.2.2.2 To create a global variant

### Prerequisites

When using document variants on the SAP BusinessObjects BI platform, global variants are available only if the *Variants Support* is set to *User and Global* in the Central Management Console, properties of the Analysis runtime.

### Context

You create a global variant by changing the type of a user-specific variant.

### Procedure

1. Open the prompting dialog in the required mode.
2. Select the *Show Variant Settings* icon.  
The available variants are listed in the *Variant Settings* dialog with name, type and technical name.
3. Select the change icon next to the user-specific variant you want to change.  
A user-specific variant has the type *User*.
4. Select type *Global* for the variant.
5. Enter a technical name.
6. Press the OK icon.

7. Press *Close* to close the variant settings dialog.

## Results

The variant is now saved as a global variant and can be used by all authorized users.

### 5.2.2.3 To apply a variant

#### Procedure

1. Open the prompting dialog in the required mode.
2. Select the variant in the drop-down list.

The variants are listed in alphabetical order: Firstly, the user-specific variants are listed. The global variants are then listed, separated by a bar.

After selecting the variant, the first prompt in the *Prompts Summary* area is selected and the corresponding entry in the *Specify Value for Prompts* area.

3. Press the *OK* button.

## Results

The variant is applied to the query or workbook, and the crosstab is displayed accordingly.

### 5.2.2.4 To change a variant

#### Procedure

1. Open the prompting dialog in the required mode.
2. Select the variant in the drop-down list.

The variants are listed in alphabetical order.

After selecting the variant, the first prompt in the *Prompts Summary* area is selected and the corresponding entry in the *Specify Value for Prompts* area.

3. Change the prompt values for the variant.
4. Press the save icon.

## Results

The variant is saved with the new defined prompt values. If you want to change the variant name, go to the variant settings dialog.

### 5.2.2.5 To delete a variant

#### Procedure

1. Open the prompting dialog in the required mode.
2. Press the *Show Variant Settings* icon.  
The available variants are listed in the *Variant Settings* dialog.
3. Press the delete icon next to the variant you want to delete.  
The variant is deleted.
4. Press *Close* to close the variant settings dialog.

## Results

The variant is deleted permanently.

### 5.2.3 To select workbook properties for prompting

#### Context

You can select various workbook properties in the design panel that affect the behavior of the prompting dialog and existing prompt values:

#### Procedure

1. Open the *Design Panel* in the menu.
2. Select the *Components* tab.
3. Select the required workbook properties in the *Properties* area. The following properties are available:
  - *Refresh Workbook on Opening*  
If you select this option, the data sources in the workbook are refreshed every time the workbook is opened. The behavior of the prompting dialog depends on the other properties you selected for this workbook.

If this check box is not selected, the data in the workbook is not automatically refreshed on opening. You can refresh the data sources manually by choosing [Refresh All](#) in the menu.

- [Force Prompt for Initial Refresh](#)

If you select this option, the prompting dialog is displayed every time you refresh. The dialog appears even if the query only contains optional prompts. If values are already saved with the workbook, these are displayed as predefined settings in the dialog. You can accept or change these values. A refresh happens when you open a workbook and select refresh, when you select the setting above and open a workbook or when you insert a new data source.

If this property is not selected, the system checks whether the workbook contains mandatory variables that no values are available for. If no values are available, the prompting dialog is displayed. If values are available, the workbook is displayed straight away.

- [Store Prompts with Workbook](#)

This is the default property. If it is used, the defined prompt values for characteristic value variables are saved with the workbook. These variables have to be defined in the [Characteristic Restrictions](#) area or in the [Default Values](#) area with refresh behavior [Refresh as Designed](#) in the BEx query designer. The stored values will be applied to the data when the workbook is refreshed during the current session or closed and opened again. The prompt values are also available to other users using the workbook. To change the values, you can open the prompting dialog manually.

If this property is not selected, the defined values will not be saved with the workbook.

Note: In the BEx query designer, you can define variables in the [Characteristic Restrictions](#) area or in the [Default Values](#) area. For variables defined as default values, you can also define the refresh behavior in the query designer. In the [Refresh Variables](#) area, the options are [Refresh Dynamically](#) or [Refresh as Designed](#):

- If [Refresh as Designed](#) is selected, the variable values defined in the [Default Values](#) area are applied according to the selection for property [Store Prompts with Workbook](#). If the property is selected, the values are applied. If it is not selected, the variable values are not applied.
- If [Refresh Dynamically](#) is selected, the variable values defined in the [Default Values](#) area are applied when a workbook is opened or refreshed. Therefore the values are also applied if the property [Store Prompts with Workbook](#) is not selected.

For more information, see the SAP BEx Query Designer documentation for [Filter](#).

- [Merge Variables](#)

If you select this option, all prompts in the workbook are listed in the [Prompts](#) dialog. You can define values for each prompt. These prompt values are valid for all data sources that contain the prompt. If this property is not selected, the data sources in the workbook are displayed in the [Prompt Summary](#) area of the [Prompts](#) dialog, and the corresponding prompts are listed in the [Specify Value for Prompts](#) area. You can define different values for the prompts for each data source.

Every time you change your selection for this property in the workbook, the connected data sources are restarted. The prompting dialog will therefore appear automatically so that you can define the values before the data sources are refreshed. If the data sources are not connected, the dialog will not appear, and only the property will be changed.

## Results

The behavior of the prompting dialog and saved prompt values will be in accordance with your workbook properties selection.

## 5.3 Filtering data

In Analysis, you can define criteria to restrict the data displayed in your analysis to a subset of data. This action is called filtering. By filtering, you define which subset of data appears in the analysis. This allows you to create new, more specific views of your data. Filters are additive, which means that each additional filter is based on the current filter and further reduces the subset of data.

There are two types of filters. With "Filtering members", you can specify which members to display and which not to display in your analysis. You can also define members to use as filters in the background of the analysis. With "Filtering measures", you can specify which measure values should be displayed in the analysis. For example, you can define that you want to see the Top 3 regions in sales volume for each product in your analysis. You can also filter out rows and columns that contain only zeros.

### Related Information

[Filtering members \[page 162\]](#)

[Filtering measures \[page 170\]](#)

[To show/hide zeros in rows and columns \[page 175\]](#)

### 5.3.1 Filtering members

By selecting and removing members of your analysis, you can create new views of your data. You can filter members of a flat dimension with the filter dialog or directly on the crosstab. You can also filter members of a hierarchical dimension (hierarchy). The measures in an analysis are combined to one dimension, the measure dimension. You can also define criteria that are used in the background of the analysis for filtering.

For example, if you analyse the sales volume of your company in the last three years, and you only want to see the values for the last three month of each year, you can remove the other months from your analysis.

Filtering members is a static action. Members that you have removed from your analysis remain excluded unless you change the filter criteria and include them again.

Member filters affect the totals and subtotals in your analysis. Only the values for the displayed members are included in the totals.

### Related Information

[To filter data by member \[page 163\]](#)

[To filter data by member within hierarchies \[page 165\]](#)

[To filter members directly on the crosstab \[page 166\]](#)

[To filter out members directly on the crosstab \[page 167\]](#)

[To create a background filter \[page 168\]](#)

[To remove a filter by member \[page 169\]](#)

## 5.3.1.1 To filter data by member

### Context

You can filter members of a dimension to create new views of your data. You can define the number of members displayed in the filter dialog in the *User* options.

If a dimension has more members than defined in the User options, the filter dialog for mass data is opened. In this dialog, you only see the selected members, but you can add all other members of the dimension using the input help.

The filter dialog for mass data can be disabled in the registry. For more information, contact your IT Administrator.

### Procedure

1. Select a cell in the crosstab that belongs to the dimension you want to use for filtering.
2. Choose **Filter > Filter by Member**.

The *Filter by Member* dialog box appears.

If you want to filter for a range of members, you can start the filter dialog directly in the range view by choosing if you want to filter individual members of the dimension. **Filter > Filter by Range**. When working with large master data tables, this avoids that all master data values have to be loaded.

#### i Note

You can also open this dialog box using the context menu on the crosstab or the design panel.

3. Select *Individual Selection*
  - a. If you want to search for members, clear the *Select All* checkbox in the members list and enter a search string in the entry field. In the dropdown list of the search button, you can select if you want to search for a key, a text or a member attribute.  
Members that match the search string are displayed in the members list and can be selected.
  - b. With the *Display* list, you can select how the members are displayed in the list. The following options are available: Key, Text, Text and key, Key and text and the member attributes.

#### i Note

If no text is defined for a member, the key is displayed as text. Members without text are displayed first in the filter dialog. Even if the key is displayed as text, the search for text is not possible if no text is defined. The result list will be empty.

You can also select the *Access Mode* and *Explicit Selection*.

The *Access Mode* option is available only if you select the check box *Show Access Mode for Member Display* in the Advanced options dialog.

The *Explicit Selection* option could be useful if you do not have the authorization to see all members of a dimension. If the option is not active and you select *Select All*, no crosstab will be rendered as you do not have the authorization to see all members. If the *Explicit Selection* option is active and you select *Select All*, the single members you are authorized to see, will be selected and displayed in a crosstab.

You can save an *Explicit Selection* with a workbook. Note that if your authorizations change and you are authorized to see additional members, they will not be selected automatically when you reopen the workbook. You need to select them manually.

- c. Choose the *Show Selected* button if only the selected members should be displayed. The button name changes to *Show All*. Choose this, if you want all members to be displayed again.
- d. Select the members that you want to use for filtering in the members list.

In the members list, all members are selected in the default setting. To deselect a member, clear the corresponding checkbox. You can also deselect all members by clearing the *Select All* checkbox and selecting individual members from the list.

4. Select *Range Selection* if you want to filter for a range of members.

- a. Select an operator.

The following operators are available: Between, Not Between, Greater Than, Greater Than Or Equal To, Less Than, Less Than Or Equal To, Equal To and Not Equal To.

- b. Select the member range.

For the range definition, the key of the members is relevant, not the text.

- c. Select *Add Range*.

The range is added to the *Range* area. You can add several ranges to the filter.

5. Paste members from a file or from clipboard.

You can paste members, exclude members and paste ranges:

Example for pasting a list of members: 1; 2; 3 (select the members 1, 2, 3).

Example for excluding a list of members: !1; !2; !3 (exclude the members 1, 2, 3).

Example for pasting a range: >=1 (select all members greater or equal to 1).

You can use the following operators for ranges: > (greater than), >= (greater than or equal to), < (less than), >= (less than or equal to), = (equal to), != (not equal to) and 1 - 3 (interval for member 1 to 3). The syntax for intervals requires a space character before and after the minus sign.

You can copy a single or a list of members to the clipboard and paste your selection to the filter dialog using the *Paste from Clipboard* icon at the left bottom of the filter dialog.

You can paste a single or a list of members using the *Paste from File* icon at the left bottom of the filter dialog. An open dialog appears and you can select a text file (e.g. a .txt file) with members in the Windows explorer. .

The members that you want to paste can be displayed as either text or key in the list. For pasting ranges or excluding members, the members must be displayed as key.

If you want to paste members that are displayed as key in the list, select *Key* or *Key and text* as *Display* option. If you want to paste members that are displayed as text in the list, select *Text* or *Text and key* as *Display* option. Members defined without text in BW cannot be pasted as text.

If you copy a list of members, you can use one of the following separators: \r\n (new line), \t (tab), ; (semicolon), | (pipe), : (colon). The sequence the separators are listed here, is relevant for their recognition as separator, that means if new lines are used as separator in the pasted list of members, a semicolon in

the list will not be recognized as separator. But if semicolons and pipes are used in the pasted list, the semicolon is the identified separator.

After pasting values from the clipboard or file, the copied members are selected in the member list.

6. Press **OK** to apply the filter to your data.

## Results

The data in the worksheet is filtered according to your selection. The filter is saved with the workbook and is available to other users or in other sessions. You can also open and edit existing filters again.

## Related Information

[User Options \[page 301\]](#)

### 5.3.1.2 To filter data by member within hierarchies

#### Prerequisites

The data source you are using contains hierarchies.

#### Procedure

1. Select a cell in the crosstab that belongs to the hierarchy you want to use for filtering.
2. Choose **Filter > Filter by member**.  
The *Filter by Member* dialog box for hierarchies appears.

#### **i** Note

You can also open this dialog box using the context menu on the crosstab or the design panel.

3. If you want to search for members, clear the *Select All* checkbox in the members list and enter a search string in the entry field. In the dropdown list of the search button, you can select if you want to search for a key or a text.  
Members that match the search string are displayed in the members list and can be selected.
4. With the *Display* list, you can select how the members are displayed. The following options are available: Key, Text, Text and key, Key and text and Show Attributes. You can also select the Access Mode.

## i Note

If no text is defined for a member, the key is displayed as text. Members without text are displayed first in the filter dialog. Even if the key is displayed as text, the search for text is not possible if no text is defined. The result list will be empty.

In the *Show Attributes* list, you can find the attributes that are available for the members in your hierarchy. For example, a product hierarchy could have 'size' and 'color' as attributes. You can select and add these attributes to the member list in the filter dialog. This additional information may help you to make the selections for filtering.

5. In the list box, choose which members should be displayed in the list for selection.
  - o Select *Hierarchy* in the list if you want the members to be listed with the hierarchy.
  - o Select *Leaves* if you want all members of all levels to be listed in a flat view.
6. Choose the *Show Selected* button if only the selected members should be displayed. The button name changes to *Show All*. Choose this if you want all members to be displayed.
7. Select the members that you want to use for filtering in the members list.

In the members list, all members are selected in the default setting. To deselect a member, clear the corresponding check box. You can also deselect all members by clearing the *Select All* checkbox and selecting individual members from the list.

If you deselect a node in a hierarchy, all leaf members of this node are deselected automatically. The parent members of the deselected node are still selected, but the background color of the checkbox changes to gray. That shows you that not all leaf members of this node are selected for filtering.

8. Press *OK* to apply the filter to your data.

## Results

The data in the worksheet is filtered according to your selection. The filter is saved with the workbook and is available to other users or in other sessions. You can also reopen and edit existing filters.

## Related Information

[Access mode for filtering \[page 169\]](#)

### 5.3.1.3 To filter members directly on the crosstab

#### Context

You can filter for one or multiple members directly using the context menu or double-clicking a member cell.

## Procedure

1. In the crosstab, select one cell for each member that you want to filter for.  
To select multiple members, hold down the Ctrl or Shift key as you select members.
2. Choose *Filter Members* in the context menu.

### i Note

To filter for one member only, you can also double-click the corresponding member cell.

In the design panel, the filter icon is added to the corresponding dimension.

## Results

The data in the worksheet is filtered according to your selection. You can apply this filter to dimensions with and without hierarchies.

### 5.3.1.4 To filter out members directly on the crosstab

## Context

You can filter out one or multiple members directly using the context menu or drag and drop.

## Procedure

1. In the crosstab, select one cell for each member that you want to filter out.  
To select multiple members, hold down the Ctrl or Shift key as you select members.
2. Choose *Filter Other Members* in the context menu.  
To filter out the selected members using drag and drop, select the border of the marked field(s) and drag them out of the crosstab.  
In the design panel, the filter icon is added to the corresponding dimension.

## Results

The data in the worksheet is filtered according to your selection. You can apply this filtering to dimensions with and without hierarchies.

## 5.3.1.5 To create a background filter

### Context

In Analysis, you can set filters for dimensions and hierarchies that you have chosen to be displayed in the crosstab. This allows you to narrow the scope of your analysis to the data you need. You can however also select members for filtering that should not be displayed in the crosstab. These non-visible filters are background filters.

For example, you have selected the dimensions Product and Region to be displayed in the crosstab. The data source contains the sales volume for the years, 2008, 2009 and 2010. If you are only interested in the sales volume for 2010, you can add the Year dimension as a background filter to your analysis and filter for 2010. The analysis will show the data for year 2010, but the Year dimension is not visible in the crosstab.

#### i Note

Background filters for a measure dimension can contain only one measure as a filter. The same is true for other dimensions that do not aggregate.

You have the following options to add a background filter to your analysis:

### Procedure

1. You can add a dimension to the *Background Filter* area and define a filter:
  - a. Add the dimension that you want to add to the *Background Filter* area in the design panel.  
You can add the dimension using drag and drop or you can choose *Move to Background Filter* in the context menu.
  - b. Choose *Filter By Member* in the context menu.  
The *Filter by Member* dialog box appears.
  - c. Make your selection and press *OK*.  
The background filter is created.
2. Or: You can add a dimension that already has a filter. The existing filter will be applied as a background filter in the crosstab.
3. Or: You can add a background filter using the context menu on the crosstab.
  - a. Select the member(s) that you want to define as background filter in the crosstab.
  - b. Choose *Filter Members and Swap With* in the context menu.  
The dimension with the selected filters is added automatically to the *Background Filter* area, and the dimension chosen for swapping with is added to the *Rows* area in the design panel.

### Results

The data in the crosstab is displayed according to your filter selection.

## 5.3.1.6 To remove a filter by member

### Procedure

1. Select a cell in the crosstab that belongs to the dimension you want to remove the filter for.
2. Choose ► *Filter* ► *Select All Members* ▾.

#### i Note

You can also choose *Select All Members* in the context menu on the crosstab or the design panel.

### Results

All members of the selected dimension are displayed in the crosstab.

## 5.3.1.7 Access mode for filtering

In the *Filter by Member* dialog, you can define the access mode that is used to display the members in the dialog. The access mode is available in the *Display* list. The available options vary for the different data sources: BW data sources, SAP HANA analytic views, SAP HANA calculated views and SAP Analytics Cloud models.

#### i Note

The Access Mode option is available only if you enable it. You have the following options to do so:

- Select the check box *Show Access Mode for Member Display* in the Advanced options dialog.
- Set the file setting *AllowChangingAccessMode* to true.  
You can do this in the *Technical Configuration* in the Analysis backstage area or in the file system.

### Access mode for BW data sources

For BW data sources, you have the following options:

- Choose ► *Access Mode* ► *Values in Master Data* ▾ to display all members for the dimension (with and without posted data).
- Choose ► *Access Mode* ► *Only Values in InfoProvider* ▾ to display the members contained in the InfoProvider (data source).
- Choose ► *Access Mode* ► *Only Values with Posted Data in Current Navigation State* ▾ to display the members with posted data that are selected in the current navigation state.
- Choose ► *Access Mode* ► *Characteristic Relationship* ▾ to display the members that are valid according to a defined characteristic relationship.

## Access mode for HANA analytic views

For SAP HANA analytic views, you have the following options:

- Choose ► *Access Mode* ► *Values in Master Data* ► to display all members for the dimension (with and without posted data).
- Choose ► *Access Mode* ► *Values in Fact Table* ► to display the members with posted data.
- Choose ► *Access Mode* ► *Only Values with Posted Data in Current Navigation State* ► to display the members with posted data that are selected in the current navigation state.
- Choose ► *Access Mode* ► *Only Values for Existing Attribute Combinations in Master Data* ► to display all members for existing attribute combinations.

## Access Mode for SAP HANA calculated views

For SAP HANA calculated views, you have the following options:

- Choose ► *Access Mode* ► *Values in Fact Table* ► to display the members with posted data.
- Choose ► *Access Mode* ► *Only Values with Posted Data in Current Navigation State* ► to display the members with posted data that are selected in the current navigation state.

## Access Mode for SAP Analytics Cloud models

For SAP Analytics Cloud models (analytic and planning), you have the following options:

- Choose ► *Access Mode* ► *Values in Master Data* ► to display all members for the dimension (with and without posted data).
- Choose ► *Access Mode* ► *Values in Fact Table* ► to display the members with posted data.
- Choose ► *Access Mode* ► *Only Values with Posted Data in Current Navigation State* ► to display the members with posted data that are selected in the current navigation state.
- Choose ► *Access Mode* ► *Only Values for Existing Attribute Combinations in Master Data* ► to display all members for existing attribute combinations.

## 5.3.2 Filtering measures

With filtering measures, you can define rules to filter the data of your analysis to your current scope of interest. You can apply one or multiple rules to a measure. Depending on where you open the *Filter by measure* dialog, the filter definition is applied to a selected dimension, to all dimensions or to the most detailed dimension in your analysis. The applied measure filters do not affect totals or subtotals in your analysis.

You can define rules to get a ranked list of data and a list of data matching a threshold value.

In a ranked list filtering, all entries of the currently displayed data are considered and their relationship to each other determines whether an entry is displayed. For example, you can define that the Top 3 entries for a measure should be displayed.

In a threshold list filtering, an entry is filtered independently of the other entries if its value matches the criteria in the defined rule. For example, you can define that all entries are displayed that are greater than a specific threshold value.

Filtering measures is a dynamic action. Whenever you change the view of your data, the filter is applied again. For example, if you add a Top 5 filter, five members are shown in your analysis. If you then add members that you previously removed from the analysis, some of these newly added members could match the Top 5 criteria and replace some of the previously displayed members.

## Related Information

[To filter data by measure \[page 171\]](#)

[To change a filter by measure \[page 173\]](#)

[To remove a filter by measure \[page 174\]](#)

### 5.3.2.1 To filter data by measure

#### Procedure

1. You have the following options to open the filter dialog:
  - Select a cell belonging to a dimension in the crosstab and choose **Filter > Filter by Measure > Edit**. If you select this option, the filter is applied to the members of the selected dimension.
  - Select a cell belonging to a measure in the crosstab and choose **Filter > Filter by Measure > All Dimensions Independantly > Edit**. If you select this option, the filter is applied to all dimensions in the analysis, from the outermost to the most detailed one. For example, in an analysis with three dimensions in columns A, B and C, the filter is first applied to the dimension in column A, then to the dimension in column B and finally to the dimension in column C.
  - Select a cell belonging to a measure in the crosstab and choose **Filter > Filter by Measure > Most Detailed Dimension in Rows > Edit**. If you select this option, the filter is applied to the most detailed dimension of the rows. For example, in an analysis with three dimensions in columns A, B and C, the filter is applied to the dimension in column C. This option is optimized for threshold value filtering.
  - Select a cell belonging to a measure in the crosstab and choose **Filter > Filter by Measure > Most Detailed Dimension in Columns > Edit**. If you select this option, the filter is applied to the most detailed dimension of the columns. For example, in an analysis with three dimensions in rows 2, 3 and 4, the filter is applied to the dimension in row 4. As with the option above, this option is also recommended for threshold value filtering.

The *Filter by Measure* dialog box appears.

### i Note

You can also open this dialog box using the context menu on the crosstab or the design panel.

2. In the *Based on measure*: list, select the measure you want to use.
3. In the *Definition* area, select the operator and value for a new rule.

The following operators are available: Between, Not Between, Greater Than, Greater Than or Equal to, Less Than, Less Than or Equal to, Equal to, Top N, Bottom N, Top%, Bottom %, Top Sum, Bottom Sum.

### i Note

For SAP HANA data sources the following operators are available: Top N and Bottom N.

4. Press *Add*.  
The new rule is added to the *Rules* area.
5. If you define multiple rules, select the logic for the rules.  
Select *AND*, if all rules should be applied simultaneously, or *OR*, if only one of the rules should be applied.
6. Press *OK*.

## Results

The data in the worksheet is filtered according to your selection. The filter is saved with the workbook and is available to other users or in other sessions. You can also open and edit existing filters again.

## Related Information

[Operators used for filtering measures \[page 172\]](#)

[To change a filter by measure \[page 173\]](#)

[To remove a filter by measure \[page 174\]](#)

### 5.3.2.2 Operators used for filtering measures

When defining filters by measure, you can create rules to get a list of data for a defined threshold value and rules to get ranked list of data.

To define threshold values, you can use the following operators:

- Between / Not Between  
For these operators, you define a range of values that should be included or excluded in the list.
- Greater Than / Less Than  
For these operators, you define a specific value to display members that are greater or less than this value.
- Greater Than or Equal to / Less Than or Equal to  
For these operators, you define a specific value to display members that are greater or equal to or less or equal to this value.

- Equal to  
For this operator, you define a specific value to display members that are equal to this value.

To define a ranked list of values, you can use the following operators:

- Top N / Bottom N  
For these operators, you must enter a whole, positive number as the value. The ranked list is arranged according to the defined number.  
For example, if you want a list of the three customers with the strongest sales volume and the five customers with with the weakest sales volume, you create a Top 3 and a Bottom 5 rule in the filter definition.
- Top% / Bottom %  
For these operators, you must enter a number between 0 and 100. The ranked list is arranged according to the defined percentage.  
For example, if you want a list of the strongest products in revenue until 25% of the total revenue is attained, you create a Top 25% rule for this measure with regard to products.
- Top Sum / Bottom Sum  
For these operators, you can also enter a floating point number (15.7 for example). The ranked list is arranged according to the defined total value.  
For example, you want a list of products with the highest sales volume, whose combined sales volume totals EUR 20,000. All sales volumes are first sorted in descending order. The totals are calculated until the threshold value of 20,000 is exceeded. If the last value that need to be included to achieve EUR 20,000, causes the total to exceed 20,000, it is also included in the list. All products with a lower sales volume than this product are no longer displayed. Products that exceed the EUR 20,000 threshold remain in the list.

#### i Note

Do not use these operators for ranked lists for filtering hierarchies.

## 5.3.2.3 To change a filter by measure

### Procedure

1. Select a cell in the crosstab.  
There are four options for defining a filter by measure. To delete an existing filter, you have to select a cell that belongs to the option you used for defining it.
2. Choose ► *Filter* ► *Filter by Measure* ► *[Selected Filter Option]* ► *Edit* ⌵.

#### i Note

You can also use the context menu on the crosstab.

3. In the *Rules* area, select the *Change* icon in the rule that you want to change.
4. Change the definition of the rule as required and press the *Confirm* icon to change the rule.
5. Press *OK* to apply the changed rule(s) to your data.

## Results

The data in the worksheet is displayed according to the changed filter rule(s).

### 5.3.2.4 To remove a filter by measure

#### Procedure

1. Select a cell in the crosstab.

There are four options for defining a filter by measure. To delete an existing filter, you have to select a cell that belongs to the option you used for defining the filter.

2. Choose ► *Filter* ► *Filter by measure* ► *[Selected Filter Option]* ► *Reset* ►.

#### **i** Note

You can also use the context menu on the crosstab.

## Results

The data in the crosstab is displayed without the removed filter.

### 5.3.3 To work with BEx conditions in Analysis

In Analysis, BEx conditions correspond to a filter by measure. In SAP Business Explorer, you can define a condition for a query, for a query view or for a workbook:

- Query conditions are created in BEx Query Designer.  
In the results area of the query, the data is filtered according to the conditions. You can define multiple conditions for a query.  
If you insert a query with conditions in Analysis, you can activate and deactivate the conditions as BEx conditions in the menu.
- Conditions that are attached to a workbook are created in the BEx Analyzer.  
In the BEx Analyzer, you can activate and deactivate conditions.  
In Analysis, you can activate and deactivate the conditions as BEx conditions in the menu after the workbook conversion.
- Conditions that are attached to a query view can be created with various BEx tools, for example BEx Web Analyzer.  
If you insert a query view with conditions in Analysis, some of the conditions can not be recognized as BEx conditions. These conditions are displayed as filter by measure in Analysis. Conditions that are not activated in the query view, are not displayed in Analysis.  
In Analysis, you can activate and deactivate the conditions as BEx conditions in the menu or with the Filter by Measure dialog.

## Using BEx conditions in Analysis

You can activate and deactivate BEx conditions:

- To activate a condition, choose ► *Filter* ► *BEx Condition* ▾ and select the condition you want to activate. The displayed data in the crosstab is changed according to the condition.
- To deactivate a condition, choose ► *Filter* ► *BEx Condition* ▾ and select the condition you want to deactivate. The displayed data in the crosstab is changed accordingly.

## Using BEx conditions as filter by measure in Analysis

The conditions are displayed as rules in the Filter by Measure dialog. You can activate and deactivate the conditions in Analysis:

- To activate a local condition, choose ► *Filter* ► *Filter by Measure* ► *Edit* ▾. The conditions are displayed in the rules section. Press *OK* to activate the conditions.
- To deactivate a local condition, choose ► *Filter* ► *Filter by Measure* ► *Edit* ▾ and remove all conditions in the rules section that you want to deactivate. Press *OK* to deactivate and delete the conditions. Deleted conditions cannot be activated again.

## Related Information

[Filtering measures \[page 170\]](#)

[To convert a BEx workbook \[page 37\]](#)

[The Information tab \[page 141\]](#)

[SAPListOfMeasureFilters \[page 87\]](#)

## 5.3.4 To show/hide zeros in rows and columns

### Procedure

1. Select a cell in the crosstab.
2. Choose *Filter* and one of the following options:
  - *Suppress Zeros in Rows*  
All rows that only contain zeros are removed.
  - *Suppress Zeros in Columns*  
All columns that only contain zeros are removed.

## Results

The zeros are suppressed in rows and/or columns. When zero suppression is active, a check mark is displayed beside the menu item.

### i Note

In the BEx Query Designer, a query can have 2 settings relevant for zero suppression.

The first one is available at the overall query level and configures whether zero suppression is active or not in general. The second one is available at the structure level and specifies the behavior of the structure with regards to zero suppression. Given the 2 settings, there are several configurations that can be set for each axis respectively (especially if there are 2 structures).

In order to simplify the UI, you can only switch on or off zero suppression for each axis in Analysis. Hence, there is only 1 setting with 2 options for each axis available.

Therefore, Analysis is not able to visualize all possible BEx Query configurations in the menu. Depending on the configuration in the Query Designer a second activation of zero suppression could be necessary. After this second activation, zero suppression will be active in Analysis.

For more information, see SAP Note [2366950](#).

To restore the hidden rows and columns, choose ► *Filter* ► *Suppress Zeros in Rows* ► or ► *Filter* ► *Suppress Zeros in Columns* ► again.

You can also define how zeros should be displayed in the crosstab if they are not hidden.

## Related Information

[To define the display of zeros and negative values \[page 194\]](#)

## 5.4 Sorting data

Sorting data is a fundamental part of data analysis. It helps you understand your data better as you can visualize and organize different views on the data in your crosstab. With sorting the data, you are able to quickly identify the data you need for your decisions.

You can sort data by values (smallest to largest and largest to smallest) and members (A to Z and Z to A). If you sort the values of one measure (for example sales volume) from smallest to largest, the other cells in your crosstab are also affected by the sorting. The metadata information therefore remains correct in the crosstab.

The following sorting options are available:

- Ascending  
If you select ascending sorting, the data is sorted from smallest to largest. Ascending sorting for members results in sorting from A to Z.

- Descending  
If you select descending sorting, the data is sorted from largest to smallest. Descending sorting for members results in sorting from Z to A.
- Break Hierarchies  
In the default setting, data is sorted in the hierarchy levels of the crosstab. Parent members are sorted in order, and child members are sorted below each parent member in their own order. If you want to sort data across the entire hierarchy, select the *Break Hierarchies* option.


## Related Information

[To sort values \[page 177\]](#)

[To sort members \[page 178\]](#)

### 5.4.1 To sort values

#### Procedure

1. Select a cell in the crosstab that belongs to the measure you want to use for sorting.
2. Choose **Sort** > *More Sort Options...* .  
The *Sort by* dialog box appears.
3. Select the sort options you want to use:
  - *Sort Ascending*  
Sorts values from smallest to largest.
  - *Sort Descending*  
Sorts values from largest to smallest.
  - *Break Hierarchies*  
If you select this option, the values are sorted across hierarchies. Existing hierarchies in the crosstab and their subtotals are not available any more after sorting. In the default setting, the option is not selected.

#### i Note

If you do not want to change the settings for *Break Hierarchies*, you can also select the sorting options *Sort Ascending* and *Sort Descending* directly in the menu or context menu.

4. Press *OK*.

## Results

The sorting is applied to the crosstab.

## 5.4.2 To sort members

### Procedure

1. Select a cell in the crosstab that belongs to the dimension you want to use for sorting.
  2. Choose **Sort > More Sort Options...**.  
The *Sort by* dialog box appears.
  3. Select the sort option you want to use:
    - *Sort Ascending*  
Sorts members from A to Z.
    - *Sort Descending*  
Sorts members from Z to A.
  4. In the *Sort by* list, select the object type for sorting:
    - Default (Dimension Members by First Display Type)  
The members are sorted by the first display type of the selected dimension.
    - Member Display Type  
You can select one of the member display types in the list below.
    - Measure  
If you select this object type, you can select one of the available measures in the list below. With this object type, the sorting is applied to the values of the selected measure and not to a dimension.
- i Note**  
If you do not want to change these settings, you can also select the sorting options *Sort Ascending* and *Sort Descending* directly in the menu or context menu.
5. Select *Break Hierarchies* if you want to sort the members across hierarchies. Existing hierarchies in the crosstab and their subtotals are not available any more after sorting.
  6. Press *OK*.

### Results

The sorting is applied to the crosstab.

## 5.5 Working with hierarchies

There are three options for working with hierarchical presentations in your analysis: dimensions with hierarchies, dimensions displayed as hierarchy and members grouped to a hierarchy.

## Including Dimensions with Hierarchies

Hierarchies for dimensions are created in the data modeling of SAP BW. In SAP BW, they are called *Characteristic Hierarchy*. A dimension can contain multiple hierarchies. In a hierarchy, dimension members are organized in a tree structure. For example, the hierarchy of cost centers that are assembled in cost center groups.

In BW, you can define different versions of a hierarchy for a dimension. In Analysis, these hierarchies are displayed several times in the design panel with the version number available in brackets after the hierarchy name.

You can also define different views of a hierarchy that are valid for a specific time period in BW. For that, you need to create the hierarchy structure time-dependently. In Analysis, you can specify in the Advanced options dialog whether all time-dependent hierarchies for a dimension should be available in the design panel, or only the hierarchy for a selected period. For time-dependent hierarchies, the validity date is displayed in brackets after the hierarchy name in the design panel.

Note that dimensions with hierarchies can behave differently during filtering and sorting, and that there are separate styles for displaying hierarchy levels. The behavior is described in the corresponding chapters.

To find more information about the definition of characteristic hierarchies in SAP BW, see the documentation on the SAP Help Portal at [Hierarchies](#).

## Displaying Single Dimensions as Hierarchy

You can display multiple dimensions in a crosstab as a flat presentation or as a hierarchy. By default, the flat presentation is displayed. You can change the display to a hierarchical presentation. You can also use dimensions with hierarchies as a single dimension and combine it with other dimensions to a hierarchical presentation.

## Grouping Members

If a dimension is displayed with its flat presentation, you can also group members together to be listed under a node. With these Custom Hierarchies, you can display the members as hierarchical presentation.

## Related Information

[To include dimensions with hierarchies in an analysis \[page 180\]](#)

[To display single dimensions as hierarchy \[page 182\]](#)

[To group members \[page 183\]](#)

[To filter data by member within hierarchies \[page 165\]](#)

[Sorting data \[page 176\]](#)

[SAP cell styles \[page 53\]](#)

## 5.5.1 To include dimensions with hierarchies in an analysis

### Context

By using a data source that contains a dimension with hierarchies, you can define different views on the hierarchies of the dimension. You can also display the members of the dimension in a flat presentation.

As of step two, the steps in the following procedure are optional. You can use the options you need to define your view on the data.

### Procedure

1. Move a dimension with hierarchies to the *Rows* section in the design panel.

#### i Note

Dimensions can also be moved to the *Columns* section. This procedure describes dimensions in the *Rows* section. Dimensions in columns behave accordingly.

The dimension is displayed in the crosstab according to the definition in the BEx Query Designer. If the dimension is defined with an active hierarchy and active attributes, the dimension is displayed in the crosstab accordingly. Nodes with a + contain subnodes. In the design panel, the active hierarchy and attributes are displayed in bold letters in the *<data source>* section.

If no hierarchy is active in the query definition, the data is displayed in a flat presentation.

2. Select a different hierarchy.

Select a different hierarchy of the dimension in the *<data source>* section in the design panel and move it to the dimension in the *Rows* section.

Only one hierarchy of a dimension can be active at the same time.

The data in the crosstab changes according to your selection. The selected hierarchy is now active and displayed in bold letters in the *<data source>* section.

3. Activate attributes

Select an attribute of the dimension in the *<data source>* section and move it to the dimension in the *Rows* section.

Multiple attributes can be active at the same time.

The data in the crosstab changes according to your selection. The selected attribute is now active and displayed in bold letters in the *<data source>* section.

4. Expand/Collapse all nodes.

Select the cell with the dimension name in the crosstab and choose **Hierarchy > Expand to Level > <Level>**. The number of available levels depends from the hierarchy.

The levels in the crosstab are expanded according to your selection. To collapse the levels, choose the first level.

5. Expand/Collapse a single node.

Select a node cell in the crosstab that you want to expand and choose **Hierarchy > Expand**.

The selected node is expanded to the next level. To collapse the subnodes of a selected node, choose **Hierarchy > Collapse**. You can also select the +/- symbols to expand and collapse a single node.

6. Define the available levels of the hierarchy.

Select a member cell of the dimension in the crosstab and choose **Hierarchy > Show Level...**. Select the levels that should be available in the analysis and press **OK**.

Only the selected levels can be displayed in the crosstab. The total is not affected by this selection because the leaf level is always available.

7. Expand the hierarchy in another direction.

By default, a hierarchy is expanded as defined in the BEx Query Designer. You can change the expansion direction for the hierarchy.

Select an element of the dimension and choose **Hierarchy > Expand Upwards (Rows)**. Alternatively, you can expand hierarchies to columns (left or right).

The hierarchy is expanded in the selected direction.

If you use several hierarchies in a crosstab, they can be expanded in different directions.

8. Display the dimension as flat presentation.

Select the **Flat Presentation** node in the <data source> section of the design panel and move it to the dimension in the **Rows** section.

The hierarchy is removed from the dimension and the members of the dimension are displayed in a flat presentation.

9. Switch between hierarchical and flat presentation.

You can activate and deactivate hierarchies in the design panel to switch between hierarchical and flat presentation.

In the **Rows** section, choose **Use Hierarchy <HierarchyName> For Filter Only** in the context menu for an hierarchy to deactivate the hierarchy and switch to the flat presentation.

The members of the dimension are displayed in a flat presentation in the crosstab. The hierarchy information is removed in the design panel. If you open the filter dialog for this dimension, the hierarchy is still active and can be used for filtering.

To activate the hierarchy again and switch to the hierarchical presentation, choose **Use Hierarchy <HierarchyName> For Filter Only** in the context menu for the dimension.

## Results

The dimension with hierarchies is displayed according to your definition.

## 5.5.2 To display single dimensions as hierarchy

### Context

If you use multiple dimensions in your analysis, you can display them as flat presentation or as hierarchy in the crosstab.

As of step three, the steps in the following procedure are optional. You can use the options you need to define your view on the data.

### Procedure

1. Move at least two dimensions to the *Rows* section in the design panel.

#### Note

Dimensions can also be moved to the *Columns* section. This procedure describes dimension in the *Rows* section. Dimensions in columns behave accordingly.

The dimensions are displayed side by side in the crosstab.

2. Change to hierarchical display.

Select a crosstab cell and choose **Hierarchy > Compact Display in Rows**.

The dimensions in the crosstab are displayed as hierarchy. The first dimension in the *Rows* section is the first level in the hierarchy. Nodes with a + contain subnodes.

3. Expand/Collapse all nodes.

Select the cell with the dimension name in the crosstab and choose **Hierarchy > Expand to Dimension > <Dimension Name>**. The number of available dimensions depends from the number of dimensions in the *Rows* section.

The hierarchy in the crosstab is expanded according to your selection. To collapse the hierarchy, choose the first dimension.

4. Expand/Collapse a single node.

Select a node cell in the crosstab that you want to expand and choose **Hierarchy > Expand Dimension > <Dimension Name>**.

The selected node is expanded to the next dimension below. To collapse the subnodes of a selected node, choose **Hierarchy > Collapse**. You can also select the +/- symbols to expand and collapse a single node.

5. Define the sequence of the dimensions in the hierarchy.

The sequence of the dimension in the *Rows* section determines their level in the hierarchy. The first dimension in the design panel is the first level in the hierarchy. By using drag and drop, you can move the dimensions up and down. The levels in the hierarchy are changed immediately.

The hierarchy is displayed from child level to parent level.

6. Display the dimensions again as flat presentation.

Select a crosstab cell and choose **Hierarchy > Compact Display in Rows**.

The dimensions are again displayed side by side in the crosstab.

## Results

The dimensions are displayed according to your definition.

### 5.5.3 To group members

#### Prerequisites

- You can group members of dimensions that are stored in an SAP BW system with version 7.50 SP6 or higher.  
Members in HANA data sources cannot be grouped.
- For the dimensions, the checkbox *With Hierarchies* needs to be selected in the BW system (transaction RSD1).

#### Context

Besides using pre-defined hierarchies for certain dimensions, you can group members together to be listed under a node interactively. If a dimension is displayed with its flat presentation, you can build a Custom Hierarchy to display the members as hierarchical presentation.

You can separate these members again by ungrouping them. You can ungroup a single node or the complete custom hierarchy.

#### Procedure

1. Select one or more members of a dimension in the crosstab.
2. Choose ► *Hierarchy* ► *Group Members* ►.  
The *Group Header* dialog appears.
3. Enter a name for the new node in the *Group Header* field.  
The name will be the header for the grouped members.
4. Press *OK*.

You have created a new hierarchy.

5. Optional step: Modify the new hierarchy.

You can add or remove members in the new hierarchy using drag and drop. You can drag and drop one or several members. You can drop new members at any position in the hierarchy. If you drop them to the node, the dropped members will be added at the end of the hierarchy.

You can also change the node name.

To do so, select the group header and choose *Rename Group* in the context menu. The *Group Header* dialog appears and you can enter a new name for the node.

6. Optional step: Expand the hierarchy in another direction.

You can change the expansion direction for the hierarchy per dimension.

Select an element of the dimension and choose ► *Hierarchy* ► *Expand Upwards (Rows)* ▾. Alternatively, you can also expand hierarchies to columns (left or right).

The hierarchies of the dimension are expanded in the selected direction.

## Results

This hierarchy is listed under the hierarchy nodes for the corresponding dimension in the Analysis tab of the Design Panel as it is done for predefined hierarchies. Moreover this hierarchy behaves like the pre-defined ones. You can build several nodes as well as group several nodes to higher nodes.

To ungroup a single node, select the node and choose ► *Hierarchy* ► *Ungroup Members* ▾. The created node is deleted and the former display is reestablished.

To ungroup your complete custom hierarchy, select one element of the dimension and choose ► *Hierarchy* ► *Ungroup All* ▾. The dimension is shown in flat presentation again. The custom hierarchy is deleted and is not shown in the Analysis tab of the Design Panel anymore.

## 5.6 Calculating new measures

In Analysis, you can calculate new measures based on measures that are available in the selected data source.

The measures used as basis for calculations must have a unique name in the data source. If several measures in a data source have the same name, these measures cannot be used for calculations. If one of these measures is selected, the *Calculation* option in the ribbon is disabled. Therefore, we recommend to use data sources containing measures with unique names.

For simple calculations, you use two or more available measures as operands and an operator to create a new measure. The new measure is the sum of two available measures, for example.

For dynamic calculations, you only use one available measure as operand and you create a new measure based on this operand. For example, if you use sales volume per region as operand, you can add a new measure that displays the rank of each region according to sales volume. If you now filter out or add new regions to your analysis, the rank numbers are changed dynamically. These calculations are therefore called dynamic calculations.

For advanced calculations, you can use the free-form editor. In this editor, you can create formulas with the available measures in the workbook and mathematical functions. The measures of the data source must have a unique name.

You can also add a new measure by restricting an existing one.

The newly created measures are added to the crosstab and to the design panel. In the design panel, you can switch the sequence of the measures, edit their name or delete them. Directly on the crosstab, you can also switch the sequence using drag and drop, and edit the name and delete measures with the context menus.

## Related Information

[To calculate a new measure based on available measures \[page 185\]](#)

[To add a new measure based on one available measure \[page 186\]](#)

[To add a new measure based on free-form calculation \[page 187\]](#)

[To add a restricted measure \[page 189\]](#)

[Calculating new measures based on SAP HANA views \[page 218\]](#)

## 5.6.1 To calculate a new measure based on available measures

### Procedure

1. Select the measure headings that you want to use.

The first measure that you select is the first operand. Use the **CTRL** key to select the next measure that is the second operand. You can also select more than two operands for your calculation.

2. Choose ► **Calculations** ► **Add Calculation** ► **[Operator]** ►.

The following operators are available:

- Add
- Subtract
- Multiply
- Divide
- Percentage Difference

The new measure calculates the difference between operand 1 and operand 2 as a percentage.

For example, operand 1 has value 80 and operand 2 has value 20. The new measure displays the value 300%, as the difference is 60 and 60 is 300% of 20.

If operand 2 has a higher value than operand 1, the result is negative. For example, operand 1 has value 20 and operand 2 has value 80. The new measure displays the value -75%, as the difference is -60 and -60 is -75% of 80.

- Percentage Share

The new measure calculates the share of operand 1 in comparison with operand 2 in percentage.

For example, operand 1 has value 20 and operand 2 has value 80. The new measure displays the value 25%, as 20 is 25% of 80.

## Results

The new calculated measure is added to the crosstab and the design panel.

A measure value that is displayed with sign reversal, is used in calculations as displayed.

### ❖ Example

A measure value is 200. If you have selected the sign reversal property in the BEx query definition, the value is displayed as -200 in Analysis. For calculations in Analysis, the value 200 is used. If you select this check box, the value -200 is used for calculations.

## 5.6.2 To add a new measure based on one available measure

### Procedure

1. Select the measure heading that you want to use.
2. Choose ► *Calculations* ► *Add Dynamic Calculation* ► *[Operator]* ►.

The following operators are available:

Option	Description
<b>Moving Minimum Value</b>	The new measure displays the smallest value available up to this point.  For example, there are ten rows with values in your crosstab. The new measure starts in the first row with the same value as the original measure. If the second row in the original measure has a higher value than the first, the value of the first row is repeated in the second row of the new measure and so on.
<b>Moving Maximum Value</b>	The new measure displays the highest value available up to this point.
<b>Accumulative Sum</b>	The new measure displays the sum of all values up to this point.  For example, the original measure has eight rows with values. In the fourth row of the new measure, the sum of first four rows is displayed.
<b>Accumulative Sum of Rounded Values</b>	The new measure displays the sum of all rounded values up to this point.
<b>Accumulative Count of All Detailed Values</b>	The values per measure are counted and numbered starting with 1 for the first value. If there are 8 rows with values, including zeros, that contribute to the total, the new measure displays the numbers 1 to 8 for the 8 values.
<b>Accumulative Count of All Detailed Values that are Not Zero, Null or Error</b>	The values per measure are counted and numbered starting with 1 for the first value, excluding values that are equal to zero.  For values that are equal to zero, the last number is repeated.
<b>Moving Average</b>	The new measure calculates the average of all values up to this point.  For example, if there are five rows with values in the crosstab, the new measure calculates in row 2 the average of the values in row one and two, and so on.

Option	Description
<b>Moving Average that is Not Zero, Null or Error</b>	The new measure calculates the average of all values up to this point, excluding values that are equal to zero.
<b>Rank Number</b>	<p>The new measure displays a rank number for each value of the original measure. The highest value in the original measure has the rank number 1.</p> <p>If a value occurs more than once, the values are assigned to the same rank number. If there are two identical values with rank number 4, the next smallest value has rank number 5.</p>
<b>Olympic Rank Number</b>	<p>The Olympic ranked list differs from the basic ranked list as follows: In the Olympic ranked list, when a value occurs more than once, the next smallest value is not assigned the rank incremented by one, but the rank that corresponds to the number of previous values (including the current value).</p> <p>For example, if the rank 4 occurs twice, the new measure displays number 6 for the next smallest value rank.</p>
<b>Percentage Contribution</b>	The new measure calculates the percentage contribution of a value in the original measure to the overall result of the original measure.

## Results

The new measure is added to the crosstab and the design panel.

### 5.6.3 To add a new measure based on free-form calculation

#### Context

You can add a new measure with the advanced calculation editor.

Note that the advanced calculation editor is available only if all measures of the data source, visible or hidden in the current navigation state, have a unique name.

#### Procedure

1. Select a measure heading in the crosstab.
2. Choose **► Calculations ► Add Advanced Calculation... ►**.

The *New Calculation* dialog appears.

3. Enter a name for the new calculation.

This will also be the name for the new measure.

4. Enter a formula for the new measure in the *Calculation* area.

With *Insert Member*, you can select the measures of the data source as operands.

With *Insert Function*, you can select the operator for the calculation. The following functions are selectable:

- Mathematical Functions:
  - Addition
  - Division
  - Multiplication
  - Subtraction
  - Power Function
  - Logarithm
  - Remainder from Integer Division
  - Root
  - Logarithm to the Base of 10
  - Absolute Value
  - Smallest integer value that is greater than operand
  - Integer Division
  - Largest integer value that is less than operand
  - Decimal part of operand
  - Maximum
  - Maximum of 0 and operand
  - Minimum
  - Minimum of 0 and operand
  - Integer value of operand
  - To the Power
- Data Functions:
  - Value without Dimension (No Unit)
  - Process Value as Date
  - Process Value as Time
  - NOERR(x)\: Equals 0 for undefined calculations; otherwise x
  - NDIVO(x)\: Equals 0 when Divided by 0; otherwise x
- Percentage Functions:
  - Percentage Deviation
  - Percentage Share
- Boolean Operators:
  - Logical AND
  - Logical OR
  - Logical EXCLUSIV OR
  - Logical NOT

You can also use CTRL+Space to enter operands and operators.

5. Select *OK* to save the new measure.

The button is enabled if the calculation is valid.

## Results

The new calculated measure is added to the crosstab and the design panel.

You can edit or delete the new measure.

## 5.6.4 To add a restricted measure

### Context

You can add new measures based on restrictions for existing measures. The existing measure can also be a restricted measure.

### Procedure

1. Select a measure heading or a data cell in the crosstab.
2. Choose ► *Calculations* ► *Add Restriction...* ►.

The *New Restriction* dialog appears.

3. Enter a name for the new restriction.

This will also be the name for the new measure. The name cannot be edited after the creation.

4. Select the measure that you want the new restricted measure to be *Based on*:
5. In the *Selections* area, select the dimension that you want to restrict and specify the members for the restriction.

You can select the *Open Value Help* icon to open the Filter by Member dialog for specifying the members.

If you opened the dialog from a data cell, this data cell is already defined as a restriction. You can change this restriction.

6. Select the *Add Line* icon to restrict an additional dimension.

This step is optional.

7. Select *OK* to create the new measure.

## Results

The new calculated measure is added to the crosstab and the design panel.

You can delete the new measure, but you cannot edit it.

## 5.7 Defining the display of members, measures and totals

When you insert a new crosstab in your analysis, members, measures and totals are displayed according to the default settings. You can change the display of these elements using the toolbar or context menu.

### Related Information

[To define the members display \[page 190\]](#)

[Defining the measures display \[page 191\]](#)

[Defining the totals display \[page 194\]](#)

### 5.7.1 To define the members display

#### Context

Members can be displayed as key, text or both. For texts, you can define which text should be displayed.

Key, text and short text of a dimension and its members is set during creation in SAP BW. In the query definition, you can define the member display that should be used as default. In Analysis, you can define the member display for each dimension.

The definition of the member display does not affect the totals and subtotals in your analysis.

#### Procedure

1. Select a member cell of a dimension to define the display for all members of this dimension.
2. Choose *Members* and the display option you want to use. You can select the following display options:
  - *Text*: The members are displayed with their text.
  - *Key*: The members are displayed with their key.
  - *Text and key*: The members are displayed with their text and key.
  - *Key and text*: The members are displayed with their key and text.
3. If you select a display option that contains text, you can choose between two display options for the text. Choose *Members* and one of the following options:
  - *Short Text*: The short text for the members is used for the text display.
  - *Text*: The text for the members is used for the text display.
4. Select the Access Mode for the dimension to define which members should be displayed in the crosstab.

Note that this step is not available for SAP HANA data sources.

## i Note

The Access Mode option is available only if you enable it. You have the following options to do so:

- Select the check box *Show Access Mode for Member Display* in the Advanced options dialog.
- Set the file setting *AllowChangingAccessMode* to true.  
You can do this in the *Technical Configuration* in the Analysis backstage area or in the file system.

You can select one of the following options:

- *Default*: The members are displayed in the crosstab according to the query definition (either members with posted data or all members).  
Note that this option is not available for SAP Analytics Cloud data sources.
- *Only Values with Posted Data*: The members with posted data are displayed.
- *Values in Master Data*: All members (with and without posted data) are displayed.

You can also apply an access mode to new lines added for planning.

## Results

### 5.7.2 Defining the measures display

To define the measures display, you can specify the number format for each measure in your analysis. This includes the scaling factor and the decimals used.

For all measures in your analysis, you can define in one single step whether the scaling factors should be displayed. If you use currency as unit, you can translate the current currency to the currency you want to use. You can also define how zeros and negative values are displayed in the crosstab.

## Related Information

[To define the number format \[page 192\]](#)

[To show or hide the scaling factors \[page 192\]](#)

[To define currency translation \[page 193\]](#)

[To define the display of zeros and negative values \[page 194\]](#)

## 5.7.2.1 To define the number format

### Procedure

1. Select a cell from the measure that you want to change.
2. Choose ► *Measures* ► *Number Format* ▾.
3. In the *Number Format* dialog box, select the *Scaling Factor* and the *Decimal Places* that you want to use for the measure display.
4. Select the *Display Scaling Factor and Units in the Header for All Measures* check box if the scaling factor should be displayed in the crosstab.
5. Press *OK*.

### Results

The data cells of the selected measure are displayed according to your definition.

## 5.7.2.2 To show or hide the scaling factors

### Procedure

1. Select a crosstab cell.
2. Choose ► *Measures* ► *Show Scaling Factors* ▾.  
When the option is active, a check mark is displayed beside the menu item. To deactivate the option, choose it again.

### Results

The scaling factors are displayed for all measures according to your definition.

The setting made in BEx Query Designer to display the scaling factors for key figures is not evaluated in Analysis. To define the display for scaling factors only this setting in Analysis is evaluated.

## 5.7.2.3 To define currency translation

### Context

You can translate the currency for all measures of an analysis.

### Procedure

1. Select a crosstab cell.
2. Choose ► *Measures* ► *Currency Translation* ►.
3. In the *Currency Conversion* dialog, select a *Target Currency*, a *Currency Translation Type* and press *OK*.  
The currency translation types are created in SAP BW. For more information, see SAP Help Portal at [Currency Translation](#).

#### i Note

The selection options in the Target Currency and Currency Translation Type lists are interdependent. Depending on the currency you selected, the system offers you all translation types that support this currency. To be able to display all translation types again, choose the standard setting *Display Original Currency*. Conversely, if you select a particular translation type, the system shows all currency options for this translation type.

4. Select the *Use Currency Translation from Query Definition* check box, if the currency translation should be based on the currency definition in the BEx Query Designer.  
The currency that you define in the query definition can be different from the currency in the original InfoProvider. For example, in the InfoProvider the currency USD is used and in the query definition, the currency is translated from USD to EUR. If you want to translate the currency in your analysis to GBP, the currency translation is based on EUR if you select this check box and it is based on USD, if you do not select it.

#### i Note

The check box is only available in the dialog when you select the options Advanced options dialog.

### Results

All measures that are scaled in a currency are translated to the selected currency.

### Related Information

[User Options \[page 301\]](#)

## 5.7.2.4 To define the display of zeros and negative values

### Procedure

1. Select a crosstab cell.
2. Choose **► Measures ► General Format ▾**.
3. Select an option from the *Display of Negative Values* list.  
The following options are available: -X, X-, (X).
4. Select an option from the *Display Zeros as:* list.  
The following options are available:
  - With Currency/Unit
  - Without Currency/Unit
  - Empty Cell
  - User DefinedIf you select this option, you can enter a text that should be displayed in the cells.
5. Press *OK*.

### Results

The values are displayed for all measures according to your definition.

You can also suppress zeros if there are rows or columns that only contain zeros.

### Related Information

[To show/hide zeros in rows and columns \[page 175\]](#)

## 5.7.3 Defining the totals display

For totals, you can define whether they should be displayed in the crosstab. If they are displayed, you can decide where they should be displayed in the crosstab. You can also define how the totals should be calculated.

### Related Information

[To locate the totals display in the crosstab \[page 195\]](#)

[To show or hide totals in the crosstab \[page 195\]](#)

[To define the calculation of totals \[page 196\]](#)

## 5.7.3.1 To locate the totals display in the crosstab

### Context

In the default display, totals are displayed below members and to the right of the members. You have the following options to change the display:

### Procedure

1. Select a crosstab cell.
2. Choose ► *Totals* ► *Total Rows Above Members* ►.  
The totals are displayed above the members.
3. Choose ► *Totals* ► *Total Columns Left of Members* ►.  
The totals are displayed left of the members.

### Results

The totals are displayed according to your definition. To change the location again, deselect the option.

## 5.7.3.2 To show or hide totals in the crosstab

### Procedure

1. Select a member cell in the crosstab.
2. To show the totals in the crosstab, choose ► *Totals* ► *Show Totals* ►.
3. To hide totals that only have one member, choose ► *Totals* ► *Hide Totals if Only One Member is Available* ►.

### Results

The totals are shown / hidden according to your definition.

## 5.7.3.3 To define the calculation of totals

### Context

The default setting is to display totals according to the definition in the query. You can calculate with these values to display another result, for example the highest value for an area.

### Procedure

1. Select a measure heading in the crosstab.
2. Choose ► **Totals** ► **Calculate Totals As** ► **[Option]** ►.

The following options are available:

Option	Description
<b>Minimum</b>	The smallest value for the corresponding area is displayed as the result.
<b>Maximum</b>	The highest value for the corresponding area is displayed as the result.
<b>Sum</b>	The total of all values for the corresponding area is displayed as the result.
<b>Sum of Rounded Values</b>	If you have set a scaling factor, it is useful to calculate the totals of rounded values locally since the totals can differ considerably (in particular with high scaling factors).
<b>Total Number of Detailed Values</b>	The number of values for the corresponding area is displayed as the result.
<b>Total Number of Detailed Values that are Not Zero, Null or Error</b>	The number of values for the corresponding area, excluding zeros, is displayed as the result.
<b>Standard Deviation</b>	<p>The deviation of the values from its mean is displayed as the result. The more spread out the data is, the higher the deviation. Standard deviation is calculated as the square root of variance.</p> <p>For example, the mean of the following two is the same: 15, 15, 15, 14, 16 and 2, 7, 14, 22, 30. However, the second is clearly more spread out.</p>
<b>Average</b>	The average (total divided by total number of values) of all values for the corresponding area is displayed as the result.
<b>Average of All Detailed Values that are Not Zero, Null or Error</b>	The average (total divided by number) of all values for the corresponding area is displayed as the result, with the exception of zeros, meaning that values equal to zero are not counted.
<b>Median</b>	<p>The system sorts the values internally by size and displays the mean value from the list as the result.</p> <p>For example, if there are five values in the list, the third value is displayed as the result. If the list has an even number of values (for example six), the system adds the two values in the middle of the list together (in this case, the third and fourth values) and divides the result by two. The resulting value is displayed as the result.</p>

Option	Description
<b>Median of Detailed Values that are Not Zero, Null or Error</b>	Excluding the values that are equal to zero, the system sorts the values internally by size and displays the mean value from the list as the result.
<b>Variance</b>	The variance is calculated as the square of the standard deviation and displayed as the result.
<b>Hide</b>	The result is not displayed.
<b>Default</b>	With this setting, the results provided by the query are displayed.

## Results

The totals are calculated according to your definition.

## 5.8 To jump to a target with Goto

The report-report interface (RRI) allows you to call a jump target from a BEx query executed in Analysis. You can use the RRI to jump from the executed query (sender) to another report (receiver) containing more information. You can call targets in BW systems and beyond.

You can use the report-report interface to call targets that you have defined in your query. To be able to call the targets from executed queries, you must define the targets for the query with the sender/receiver assignment. The sender/receiver assignment is defined in transaction RSBBS in the BW system.

In Analysis, you can assign the following receivers:

- Query  
If you call a query with Goto, a new workbook is opened with the query.
- Web address  
http://www.google.com for example.
- The following targets are opened in the web browser:
  - ABAP transaction. The transaction must be classified for using SAP GUI for HTML.
  - ABAP Report
  - BEx Web Application (3.x and 7.0)
  - Crystal Report
  - Xcelsuis dashboard

After making the assignment, you can choose [Goto](#) in the context menu to call all the targets for cells in the crosstab of the executed sender query.

## More Information

You can find more information about the report-report interface and the sender/receiver assignment in the SAP NetWeaver documentation in the SAP Help Portal under [Using the Report-Report Interface](#).

## 5.9 Commenting data cells

With Microsoft Excel functionality, you can add a note to a cell by inserting a comment. You can edit the text in comments and delete comments that you no longer need. You can also choose if the comment text box is shown or hidden in your analysis.

You can add comments to the data cells in the crosstab. The data source must have been refreshed before you add a comment to a data cell. A comment is available internally as long as it is not explicitly deleted. For example, you add a comment to a cell and then change the view on your data. In the new navigation state, the cell with the comment is not visible. As soon as you change the navigation state so that the cell with the comment is again visible in the crosstab, the comment is also available.

You can use the options in the Comments group in the ribbon or the context menu to create, edit or delete comments. Please note that Microsoft Excel uses the term Note for comments. You see this term if you use the context menu to work with comments.

The comments are associated with the selection of the cell (context). The selection includes all selected values for the dimension, for example prompt, filter and background filter values. Analysis stores for each dimension the single member selection if the selection is a single member selection, for example a single value in the prompting definition. For other selections, Analysis stores that there has not been a single member selection for this dimension. Note that a comment that is placed on the total cell of a dimension also is considered not to have a single member selection associated for this dimension. A comment is visible in a cell if the cell has for all dimensions the matching single member selection or the comment and the cell both do not have associated a single member selection.

Any hidden selections of a restricted key figure is not taken into account. Instead, the key figure id is considered as single member selection.

If you add a comment to a member or header cell in the crosstab, the comment is not available after changing the navigation state.

### **i** Note

If you create a Microsoft Powerpoint presentation with a crosstab that contains comments, the comments will not be available in the presentation.

You can save comments in SAP BW/4HANA, on the BI platform and locally.

## 5.9.1 To comment a data cell

### Context

The options for commenting a data cell are available in the *Comments* group in the ribbon.

### Procedure

1. Make sure that the data source has already been refreshed.
2. Select the cell where you want to add a comment.

The context of the selected cell is displayed in the *Context* section of the *Comments* tab in the design panel.

Note: If the crosstab contains data cells with exactly the same context, for example after copying a column, the comment will be added to all data cells with this context.

3. Choose *New Comment* in the ribbon.

You can also use the context menu to create, edit or delete comments. Please note that Microsoft Excel uses the term Note for comments. You see this term if you use the context menu to work with comments.

4. In the comment text box, type the comment text.  
A small red triangle in the corner of the cell indicates that the cell has a comment attached to it.
5. To edit an existing comment, select the cell and choose *Edit Comment* in the ribbon.
6. To keep a comment visible with the cell, select the cell and choose *Show/Hide Comment* in the ribbon.  
The comment text box is shown permanently in your analysis. You can move the text box using drag and drop.
7. To delete an existing comment, select the cell and choose *Delete Comment* in the ribbon.

### Results

You can save the comments in the workbook.

## 5.9.2 To save comments in SAP BW/4HANA

### Prerequisites

For saving comments in SAP BW/4HANA the following prerequisites must be met:

- You use SAP BW/4HANA 1.0 SPS8 or a higher version as BW system.
- You have selected *Business Warehouse* as preferred document storage for comments in the *Platform Options* .
- In the BW system, you have created a document store and you have to defined which InfoProviders it will store.  
In Analysis, you can use queries that are based on these InfoProviders to add comments and use this document storage.  
For more information about document stores, see the SAP BW/4HANA documentation at [Document in the Document Store](#).
- In the BW system, you have defined the document support for the query:  
Set the query property *Document Support* to value `YES`.  
Enter the name of the document store in the *Default Document Application ID* field.

## Context

You can save comments in SAP BW/4HANA. The comments are saved per workbook and data source.

Comments from an SAP BW system are loaded automatically when a workbook is opened or the data source is inserted in a workbook.

## Procedure

1. To save comments in SAP BW/4HANA, choose *Save Comments* in the Comments group.
2. Optional step: Change the *Document Store* for the data source.

On the Components tab in the design panel, you can select another document store of the data source.

3. Optional step: Create or select a *Document Filter*.

To create a document filter, enter a name in the document filter field on the Components tab in the design panel and save the document filter. You have two options to save a document filter:

- *Save for This Workbook*  
The document filter will contain all comments entered in the workbook.
- *Save for This Data Source*  
The document filter will contain the comments of the currently selected data source.

Alternatively, you can also select an existing document filter and save it for the workbook or the selected data source.

All comments that you add after saving a document filter, are assigned to this document filter.

The selected document filter is saved with the workbook. When you or one of your team members open the workbook, the document filter is already applied.

## Results

The comments are saved in SAP BW/4HANA.

You can see the context (data source and cell context) and history of each comment on the [Comments](#) tab in the design panel.

In SAP BW/4HANA, you can use the report RSDOC\_DELETE\_TAGGED\_DOCS to delete saved comments and document filters.

You can find more information about the report RSDOC\_DELETE\_TAGGED\_DOCS in the SAP BW/4HANA documentation at [Creating a Document Store](#).

## Related Information

[Platform Options \[page 303\]](#)

[The Components tab \[page 142\]](#)

[The Comments tab \[page 148\]](#)

## 5.9.3 To save comments on the BI platform

### Prerequisites

You have selected [Business Intelligence Platform](#) as preferred document storage for comments in the [Platform Options](#).

### Context

You can save comments on the BI platform. The comments are saved per workbook.

Comments from the BI platform are loaded automatically when a workbook is opened.

### Procedure

To save comments to the BI platform, choose [Save Comments](#) in the Comments group.

## Results

The comments are saved on the BI platform.

You can see the context (data source and cell context) of a selected cell on the *Comments* tab in the design panel.

## 5.9.4 To save comments locally in the workbook

### Context

The comments are saved within a workbook if it is saved locally or on a fileshare.

## 5.10 Planning Data

In Analysis, you can plan business data based on the current data in your data source. You can enter the planning data manually and you can enter planning data automatically using planning functions and planning sequences of SAP BW Integrated Planning.

While planning, you can recalculate and save your data, lock cells, set a work status and switch between display and change mode.

### Related Information

[To plan data manually \[page 203\]](#)

[To plan data with planning objects \[page 206\]](#)

[To save planning data \[page 209\]](#)

[Locking Cells \[page 209\]](#)

[Setting the Work Status \[page 212\]](#)

[To undo changes while planning data \[page 214\]](#)

[To switch between display and change mode \[page 214\]](#)

[Editing short texts in queries \[page 215\]](#)

## 5.10.1 To plan data manually

### Context

You can enter the planning data manually in a crosstab in your analysis. You can either change the values in existing data cells in the crosstab or you can add new lines to crosstabs in change mode.

To enter planning data, the data source must be available in change mode. A data source can only be available in change mode for one user and for one workbook at the same time.

### Procedure

1. Enter an input-ready data source or open a workbook with an input-ready data source.

The crosstab is visible in change mode. In change mode, read-only cells of the data source are displayed with different background colors. If you switch to display mode, the distinct visualization of input-ready and read-only cells is not visible.

#### **i** Note

Microsoft Excel cells are always input enabled. That means you can also enter values in the read-only cells of the data source. These values will not affect your planning though. If the read-only cells are displayed with a different background color, you can avoid entering data in read-only cells.

If a data source is not displayed in change mode, you can check the following settings:

- Check on the *Components* tab in the design panel if the correct planning system is connected. In the default setting, the planning system is the first system you connect to after opening a workbook. If a workbook contains objects from different systems, you can select one of these systems as the planning system in the list. The changed planning system will only be effective after closing and reopening the workbook.
- Check on the *Components* tab in the design panel if the check box *Switch All Data Sources to Change Mode for Initial Refresh* is selected. If the check box is selected, the input-ready data sources in the workbook are opened in display mode. With the initial refresh, they will be switched to change mode automatically.
- The data source is already used in change mode by another user or by the current user in another workbook.

You can also use the property *IsDataSourceEditable* of the API method *SAPGetProperty* to check if a data source is input-ready.

2. You can enter planning data by changing data in existing data cells of the crosstab.

The format of an entered value is immediately validated. If the format is not correct, a message is displayed.

To check the consistency of the new values and apply it to the crosstab, choose *Recalculate*.

3. You can enter planning data in new lines added to the crosstab.

The following section describes the process for BW systems of type SAP BW/4HANA SP8 or higher and SAP BW 7.50 SP12 or higher. For former BW versions, the legacy mode will be applied automatically (see step below).

Note that new lines are not supported if the data source has two structures and one structure is used in the rows and one in the columns.

You can define the number of new lines on the *Components* tab in the design panel. The default number of lines is 5. New lines for planning are added to the bottom of the crosstab immediately. You can add up to 1000 lines. In display mode, the new lines are not visible.

To enter planning data in the data cells, you need to enter a member of the member combination for the data cell. For example, the dimensions Country, Region and City are in the rows of your crosstab. If you want to enter a planning value for Paris, you enter the member for Paris and the planning value in the data cell for the key figure. With choosing *Check New Lines* in the context menu or *Recalculate* in the ribbon, the members for Country and Region will be derived automatically.

You can also activate a check of the entered data on the Components tab in the design panel. You can select property *Planning: New Line Validation* and define which data should be checked automatically:

- None  
No automatic check is executed.
- Member  
Automatic check runs whenever a member was changed.
- Lines  
Automatic check runs whenever a line was changed and the derivation is done automatically.

The derivations in this mode can only work if the characteristic relationship is modeled in the BW system. Values that can't be derived will be replaced with character #, or any other character defined in RSADMIN.

Entered members that do not exist, are indicated with red frames after the check.

You can enter a high amount of new lines using the clipboard. You can paste more new line rows than configured in the design panel. Analysis will automatically increase the number of new lines temporarily. The performance critical extension of the crosstab with a high number of new lines is only done if needed after pasting values.

This allows to paste and transfer a high number of new lines once without the need to change the defined number of new lines in the design panel.

If the data source contains dimensions that are created in BW as characteristics without master data, you can enter master data in the new lines to create new members in Analysis. The new members will also be saved to the BW system.

Note that you can only create new members if the RSADMIN parameter `PLAN_ALLOW_NOT_EXIST_MEM` is active (value X) in the BW system. For more information, contact your IT administrator.

Using SAP BW/4HANA SP8 or higher, you can create new master data and attributes in new lines. You can also change attributes of existing master data. The master data is automatically enabled for a query on the characteristic. To update added or changed master data in planning queries in Analysis, you can choose *Reset Data Source* in the ribbon, use API `SAPExecuteCommand` with parameter `Restart` or log off and log on. With the last two options, you can keep the navigation status.

To be able to create and change master data in SAP BW/4HANA, the following properties must be selected for the characteristic in the BW system: *Usable as InfoProvider* and *Planning Mode*.

To check the consistency of the new values and apply them to the crosstab, choose *Recalculate*.

#### 4. Optional step: Legacy mode for entering planning data in new lines added to the crosstab.

The legacy mode is used if your BW system does not support the current version for entering planning data (see step above). The legacy mode can also be enabled for all BW systems with file setting `UseNewLinesLegacyMode`. For more information contact your IT administrator.

You can define the number of new lines on the *Components* tab in the design panel. The default number of lines is 5. New lines for planning are added to the bottom of the crosstab after a navigation step, for example *Recalculate*. In display mode, the new lines are not visible.

To enter data in the data cells, you have to enter the complete member combination for the data cell. For example, the dimensions Country, Region and City are in the rows of your crosstab. If you want to enter a planning value for Paris, you also have to enter the region and country member in the corresponding line. You can enter the member manually or select values with input help. The entered values are immediately validated for format and master data. If a value is not correct, a message is displayed.

You can enter a high amount of new lines using the clipboard. You can paste more new line rows than configured in the design panel. Analysis will automatically increase the number of new lines temporarily. The performance critical extension of the crosstab with a high number of new lines is only done if needed after pasting values.

This allows to paste and transfer a high number of new lines once without the need to change the defined number of new lines in the design panel.

If the data source contains dimensions that are created in BW as characteristics without master data, you can enter master data in the new lines to create new members in Analysis. The new members will also be saved to the BW system.

Note that you can only create new members if the RSADMIN parameter `PLAN_ALLOW_NOT_EXIST_MEM` is active (value X) in the BW system. For more information, contact your IT administrator.

To check the consistency of the new values and apply them to the crosstab, choose *Recalculate*.

5. You can undo all changes that are not recalculated or undo recalculated changes that are not saved.

Select [▶ Back ▶ To Previous Consistent Client State / Saved State ▶](#).

### **i** Note

Planning data that you entered in a data cell in the crosstab, will be written to the planning buffer with the next action executed in Analysis. This could be swapping axis or adding or removing a dimension, for example. Even if the data cell where you entered the planning data is no longer displayed in the crosstab, the data is still available in the buffer.

If data is no longer needed, you can delete it manually in the data cell or delete the complete buffer with

[▶ Back ▶ To Previous Consistent Client State ▶](#).

6. Choose *Save* to save the planning data to the system.

## Results

The changed data is saved in the data source.

## Related Information

[To recalculate planning data \[page 208\]](#)

[To save planning data \[page 209\]](#)

[To undo changes while planning data \[page 214\]](#)

[To switch between display and change mode \[page 214\]](#)

[The Components tab \[page 142\]](#)

[SAP cell styles \[page 53\]](#)

## 5.10.2 To plan data with planning objects

### Context

You can use two planning objects: planning functions and planning sequences. Planning functions allow system-based processing or generation of data. A planning sequence is a sequence of planning functions. Both objects are created with Planning Modeler in SAP BW.

The following procedure explains the steps to be done in *Analysis* when working with planning objects. For more information about SAP BW Integrated Planning, Planning Modeler and objects like planning function or aggregation level, see the corresponding documentation on the SAP Help Portal.

### Procedure

1. Insert an input-ready data source or open a workbook with an input-ready data source.

#### i Note

This step is optional. You can also use planning objects without visualization in a crosstab.

2. Add a planning object to the workbook.

On the *Components* tab in the design panel, choose *Use Planning Function* or *Use Planning Sequence* in the context menu of the workbook. In the *Open* dialog box, search for the planning object, select an object and press *OK*.

The node Planning Objects is added to the workbook on the *Components* tab. All planning objects are assigned to this node.

3. Check on the *Components* tab in the design panel if the correct planning system is connected.

In the default setting, the planning system is the first system you connect to after opening a workbook. If a workbook contains objects from different systems, you can select one of these systems as the planning system in the list. The changed planning system will only be effective after closing and reopening the workbook.

4. Select the *Process Changed Data* check box if you want to use process changed data.

If you select this check box, you can define that only data changed by the user in the current session since the last save will be processed. You can specify for planning functions which aggregation level should be used as a filter to define the changed data. In the default setting, the aggregation level the planning function is built on is selected for a planning function.

Note that an aggregation level can only be set for planning functions. Therefore, it is not possible to select an aggregation level for planning sequences.

5. Define the variables values of the planning object.

On the *Variables of Planning Sequence / Functions* tab, select the source and define the values:

- For source *Member*, you can select the members or hierarchies with input help.
- For source *Cell*, you can define the cell that contains the values.
- For source *Display*, you cannot define values in Analysis. Values for these variables can be defined in customer exits in the backend system and cannot be changed in Analysis.

Variables act as placeholders for members, hierarchies and numbers. They can be defined in BEx Query Designer or in Planning Modeler. All variable values are valid locally in the planning object. To plan data for two different years for example, you can insert the same planning function twice in a workbook and enter different values for the dimension year.

6. Define the filter for the planning function.

On the *Filter* tab, you get a list of all dimensions of the aggregation level.

a. Define the filter values for the dimensions as required.

For source *Member*, you can select the members with input help.

For source *Cell*, you can define the cell that contains the values.

For source *Link*, the current filter values in the data source are used.

#### **i** Note

For dimension with source *Link*, you can also use the API method `SAPSetFilter`, to define the filter values.

b. You can also select the filter the planning function is linked to:

- Press the *Open Filter* button to insert a filter object.  
The filter object can be created in the BEx Query Designer and in the Planning Modeler.
- Press the list button to select the data source you want to use for planning. The data source should already be used in the workbook.

When you insert a filter object or data source, the source type changes from *Member* to *Link* for all dimensions that have not previously been defined, and the current filter values in the data source or filter object are used.

c. Redefine filter values.


You can change the filter definition for a planning function at any time. After inserting a data source, for example, you can change the source from *Link* to *Cell* and define a cell that contains the values that should be used as filter.

7. Choose *Execute Planning Funtion* in the context menu of a planning function on the *Components* tab to execute the object.

When working with planning sequences, choose *Execute Planning Sequence* in the context menu of a planning sequence on the *Components* tab to execute the object.

You can also upload the values for executing planning objects from a file. This has to be configured in the BW system. When you then choose *Execute Planning Funtion* or *Execute Planning Sequence* in Analysis, a dialog opens and you can select the file with the values.

To upload the values from a file, the following prerequisites must be met:

- As BW system, you use a BW/4HANA SP8 or BW 7.50 SP12.
- You have defined a planning function of type `ORSPL_FILE_UPLOAD_AO`. For more information, see SAP Note [2621959](#) 

### **i** Note

The changed data is recalculated automatically when you execute a planning object. To undo all changes since the last save, choose **► Back ► To Previous Saved State ►**.

8. Optional step: Delete a planning object from the workbook.  
Choose *Delete* in the context menu of the planning object on the *Components* tab.
9. Choose *Save* to save the planning data to the system.

## **Results**

The changed data is saved in the data source.

## **5.10.3 To recalculate planning data**

### **Procedure**

1. Enter or change planning data.
2. Choose *Recalculate*.

## **Results**

All changed values are checked for consistency. If the changed data entries are correct, the recalculated data is available, the totals are updated and the new values are visible in all other components of the workbook. The recalculation is executed for all data sources in the workbook that are available in the selected planning system.

You can also use the `PlanDataTransfer` command from API method `SAPExecuteCommand` to recalculate planning data.

## **Related Information**

[SAPExecuteCommand \[page 98\]](#)

## 5.10.4 To save planning data

### Procedure

1. Enter or change planning data.
2. Choose [Save](#).

### Results

All changed values are checked for consistency. If the changed data entries are correct, the data is saved in the data sources and a message is displayed that the data was saved. The saving is executed for all data sources in the workbook that are available in the selected planning system.

You can also use the PlanDataSave command from API method SAPExecuteCommand to save planning data.

### Related Information

[SAPExecuteCommand \[page 98\]](#)

## 5.10.5 Locking Cells

You can lock input-ready cells against manual changes in planning workbooks. A locked cell is a time-limited setting that only applies to the current user session. Locked cells are displayed in a different cell style. You can also undo cell locks.

There are two implementations of the cell lock function in Analysis. These are

- Front-end cell locks  
Cell locks are managed in the runtime for the current result set. Cell locks remain in effect so long as no significant changes are made to the result set.
- Back-end cell locks (Single Query and Cross Query)  
Cell locks are managed exclusively in the back-end system for one single or all input-ready queries in the planning workbook. Locking cells in all queries of the workbook makes it possible for a cell that is locked in one query to be displayed as locked in other queries and to be treated as such.

You can select on the Components tab in the design panel which implementation should be applied in a workbook.

For more information about cell locking in SAP BW, see the BW documentation on the SAP Help Portal at [Lock Concept and Lock Management](#).

## 5.10.5.1 Front-End Cell Locks

Front-end cell locks in a query remain in effect until the user makes significant changes to the crosstab.

The cell locks remain in effect if the user performs any of the following activities:

- Sorting the crosstab
- Expanding or collapsing nodes in BW hierarchies
- Changing display settings for dimensions

If the user performs one of the following activities though, the cells locks are undone: :

- Adding/removing dimensions or measures to the crosstab
- Calling the prompting dialog to change variable values
- Swapping row and column axes
- Hierarchical axis display
- Zero suppression
- Setting a filter
- Executing a planning function or sequence

## 5.10.5.2 Back-End Cell Locks

Back-end cell locks in a planning application allow more navigation steps in a query while retaining the cells locks as front-end cell locks. In Analysis, you can decide if the cells are locked for a a single query or multiple queries in a planning application.

Cross query cell locks could be particularly suitable if a planning workbook contains multiple sheets, and its queries are closely linked but are used for different aspects of the planning application.

The back-end cell locks remain in effect if the user performs any of the following activities:

- Displaying the properties of the dimensions (like Key or Text )
- Displaying attributes of the dimensions
- Sorting the result set by dimension members, texts or key figures
- Swapping row and column axes
- Expanding or collapsing nodes in hierarchies
- Hiding/showing structure elements (filter structure elements for example)
- Adding a dimension to the crosstab: At the lowest point on the right in the rows and the furthest point inside in the columns
- Removing a dimension from the crosstab: At the lowest point on the right in the rows and the furthest point inside in the columns

Note that some of the operations listed above hide locked cells. This does not undo the cell locks however. These "hidden" cell locks also remain active and are taken into account by the system when calculating inverse formulas and during disaggregation.

If the user performs any of the following activities, all cell locks are undone :

- Changing the order of the dimensions included in a cell lock

- Change the axis of the dimension included in a cell lock
- Changing the settings for BW hierarchies (on/off) for dimensions included in a cell lock,
- Change the query's dynamic filter (by restricting the variable values without restarting the query for example)
- Changing the filter using the prompting dialog
- Performing a planning function or planning sequence

## 5.10.5.3 To lock cells

### Context

You can lock input-ready crosstab and total cells in Analysis. A locked cell can not be changed manually when the crosstab is recalculated.

#### i Note

Microsoft Excel cells are always input enabled. That means you can also enter values in the locked cells of the data source. These values will not affect your planning though.

The cells are locked for the current user session only. You can also unlock the cells during the session.

### Procedure

1. Select the crosstab cells you want to lock.  
The selected cells have to be input enabled and have not been changed since the last recalculation.  
If you select a mixture of input-ready and already locked cells, all selected cells will be locked.
2. Choose [Lock Cells](#) in the planning group.  
You can also use the context menu to lock and unlock the cells.

### Results

The selected cell are locked for changes. The locked cells are displayed in a different cell style.

To unlock the cells, select all cells you want to unlock and choose [Unlock Cells](#).

## 5.10.6 Setting the Work Status

With the work status function, you can define who can enter planning data in a workbook and who can set a work status. Work states can also be used to apply a label to a current view and lock its data for review or approval.

For example, your month-end close business process requires that a specific set of data is locked down so that accurate month-end reports can be created. After a data submission, the owner sets the work state to Submitted. This locks the data intersection from subsequent submissions.

Another use case is that users have the authorization to enter planning data for defined members of a dimension. For example, user A can enter the planning data for France and user B for Spain.

A work status that will be used in Analysis, has to be defined with SAP Business Planning and Consolidation (BPC), version for SAP NetWeaver embedded. The work status is configured in the BPC Web Admin Tool and is based on a BPC model. A model can include one or several InfoProviders. All InfoProviders in the model must contain the owner dimension

In the work status configuration, a list of work states can be defined with any name. Typical work states are Unlocked, In Process and Approved. In the section [Locking Dimensions](#), an [Owner Dimension](#), a [Hierarchy](#) and an [Owner](#) (and Team) must be defined. The hierarchy is a BW hierarchy based on the owner dimension. Other dimensions can be added to the work status.

The ownership for single members is defined in the master data maintenance of the SAP BW system. In transaction rsd1, an owner can be defined for each member of a selected dimension. As the ownership is defined on member level, several users can plan data or set a work status in a workbook at the same time.

### i Note

Check the licensing conditions for SAP BusinessObjects Planning and Consolidation, version for SAP NetWeaver embedded.

For more information about setting up work states, see [Work Status Setup](#)

## Related Information

[To set a work status \[page 212\]](#)

### 5.10.6.1 To set a work status

## Context

You can change the planning data and set a work status for data cells of defined members. You can also lock data cells for updates with a work status.

To be able to set a new work status at least one work status is defined in the BPC Web Admin Tool and you have the authority to set a work status for one or several members. The ownership is defined in the master data maintenance of the SAP BW system.

## Procedure

1. Open a workbook containing a planning model and a corresponding data source.

You can also open an empty workbook and insert a planning model and a corresponding data source. Planning models are available on the *Environment* tab in the open workbook dialog.

2. Select a data cell.

3. Choose ► *Work Status* ► *Work Status Criteria* ▾ to check the work status criteria.

You can only set a work status if all defined criteria is fulfilled.

4. Choose ► *Work Status* ► *Set Work Status Hierarchy* ▾ to apply the hierarchy defined for the owner dimension in the BPC Web Admin Tool.

This step is optional. We recommend it to fulfill the criteria, but you can also create a navigation state in Analysis that fulfills all criteria without setting the hierarchy.

5. Choose ► *Work Status* ► *Include All Descendants* ▾ to apply the work status to the selected member and all descendant members.

This step is optional.

The entry is available only if a data cell of a member is selected that has descendant members in the hierarchy.

The setting is valid for the workbook and not only for the selected data cell.

6. Choose ► *Work Status* ► *<Defined Work Status>* ▾ to set a new work status.

You can only see and select a work status if you are defined as owner for this member in the master data maintenance of the SAP BW system.

The available work states are defined in the BPC Web Admin Tool.

7. Choose *Save Data* in the planning group to save your changes.

## Results

The workbook is saved with the changed data and work status.

## 5.10.7 To undo changes while planning data

### Procedure

1. Enter or change planning data.
2. To undo your changes, you have the following options:
  - Choose **► Back ► To Previous Saved State ►**.  
With this option, you undo all changes since the last save of your data. The undo is executed for all data sources in the workbook that are available in the selected planning system.  
You can also use the PlanDataReset command from API method SAPExecuteCommand to revert to the last saved state of your planning data.
  - Choose **► Back ► To Previous Consistent Client State ►**.  
With this option, you undo all changes since the last successful recalculation of your data. The undo is executed for all data sources in the workbook that are available in the selected planning system.  
You can also use the PlanDataClientReset command from API method SAPExecuteCommand to revert to the last recalculated state of your planning data.

### Related Information

[SAPExecuteCommand \[page 98\]](#)

## 5.10.8 To switch between display and change mode

When you work with input-ready data sources, Analysis offers two different modes: display mode and change mode. You use change mode to enter planning data and to recalculate or save the changed data to the backend system. If a user is working with an input-ready data source in change mode, this data source is locked for the other users. If the user switches to display mode, another user can work with this data source in change mode.

When you open a workbook containing input-ready data sources, the crosstabs of these data sources are displayed in change mode. If you select the check box *Switch All Data Sources to Change Mode for Initial Refresh* on the components tab in the design panel when saving a workbook, the workbook will be opened in display mode. The data sources will be switched to change mode with the initial refresh automatically.

In change mode, new lines for planning are added to the bottom of the crosstab, and read-only cells of the data source are displayed with different background colors. The Plug-in includes SAP standard styles for the visualization (SAPEditableDataCell, SAPEditableDataTotalCell, SAPReadonlyDataCell and SAPReadonlyDataTotalCell). You can change the cell styles as required. In display mode, the distinct visualization of input-ready and read-only cells is not visible.

### Switch to display mode

You work in a workbook that includes input-ready data sources.

1. Enter or change planning data.
2. Choose *Display* to switch to display mode.
3. A dialog box asks if you want to save your changes before switching to display mode. Press *Yes* to save your changes or *No* to discard them.

The data sources are switched to display mode. The distinct visualization of input-ready and read-only cells is not visible any more and the new lines at the bottom of the crosstab are deleted. The data sources can now be used in change mode by another user.

You can also use the `PlanDataToDisplayMode` command from API method `SAPExecuteCommand` to switch to display mode.

## Switch to change mode

You work in a workbook that includes input-ready data sources. You are currently working in display mode.

1. Check on the *Components* tab in the design panel that the correct planning system is connected.
2. Check on the *Components* tab in the design panel that the check box *Planning: Open Data Source Input-Enabled* is selected for the data sources you want to change.
3. Choose *Change* to switch to change mode.

All data sources in the workbook that are available in the selected planning system and are selected as input-enabled are switched to change mode. The distinct visualization of input-ready and read-only cells is applied, and the new lines are added at the bottom of the corresponding crosstabs.

### **i** Note

If a data source is locked by another user, a message is displayed.

You can also use the `PlanDataToChangeMode` command from API method `SAPExecuteCommand` to switch to change mode.

## Related Information

[The Components tab \[page 142\]](#)

[SAP cell styles \[page 53\]](#)

## 5.10.9 Editing short texts in queries

With release SAP BW 7.40 SPS08, you can edit short texts in input-ready queries. This enables you to use characteristics of a suitable InfoProvider as key figures in the query definition, in order to change characteristic values (for example, classifications) in queries or to write comments (short texts) on key figure values in queries.

In Analysis, you can select or modify characteristic values for classification, depending on the definition in the query designer. You can also enter or change short texts as comments in a query.

You can find more information about editing short texts in queries and the definition in the query designer, in the SAP BW documentation in the SAP Help Portal at [Editing Short Texts in Queries](#).

# 6 Analyzing SAP HANA Data

## 6.1 Analyzing SAP HANA data sources

After configuring a connection to the SAP HANA appliance software, you can analyze the following information models developed with SAP HANA as data sources in Analysis: analytic views and calculation views (type CUBE). You can also analyze data sources from a SAP BW, powered by SAP HANA.

### SAP HANA data sources

With SAP HANA data sources, you can use the following Analysis functions to analyze your data:

- Inserting a crosstab with SAP HANA data  
You can also insert more than one data source in a workbook.
- Saving/Opening a workbook to/from the business intelligence platform server
- Undo/Redo
- Defining style sets for crosstabs
- Swapping axes
- Analyzing data with the design panel
- Prompting
- Filtering members
- Sorting data by members and by measures
- Calculating new measures (Rank Number and Olympic Rank Number)
- Inserting charts
- Inserting filter components
- Converting crosstab cells to formula
- Creating Presentations
- Pause Refresh

These functions are described in the corresponding chapters in this guide.

### SAP BW data sources

If you use SAP Business Warehouse, powered by SAP HANA, you can analyze BW queries, query views and InfoProvider with Analysis. You can use the BW data sources with Analysis like data sources of other BW systems.

## 6.1.1 Calculating new measures based on SAP HANA views

Using native SAP HANA data sources, you can use the following methods to calculate new measures:

- Adding a measure based on one available measure.  
You can use the operators *Rank Number* and *Olympic Rank Number*.  
This method is available as of SAP HANA 1.0 SP 12.
- Adding a measure based on free-form calculation.
- Adding a restricted measure.

### Exception Aggregation

As of version SAP HANA 2.0 SP1, you can define an exception aggregation for calculated measures based on free-form calculation.

1. Add a new measure based on free-form calculation to the crosstab.
2. Select the new calculated measure in the crosstab and choose ► *Calculations* ► *Edit* ⌵.
3. Select the *Aggregation* tab.  
The selected default aggregation is *Use Standard Aggregation*.
4. Select an aggregation from the list.  
The following aggregations are available: Summation, Maximum, Minimum, Counter for All Detailed Values, Distinct Counter for All Detailed Values, Average, Variance, Standard Deviation, First Value and Last Value.
5. Press *OK* to apply the changed aggregation for the calculated measure.

### Related Information

[To add a new measure based on one available measure \[page 186\]](#)

[To add a new measure based on free-form calculation \[page 187\]](#)

[To add a restricted measure \[page 189\]](#)

# 7 Analyzing SAP Analytics Cloud Data

## 7.1 Analyzing SAP Analytics Cloud models

In Analysis, you can use live data connections and import data connections to connect with SAP Analytics Cloud data sources.

Live Data connection means that you're directly connected to an SAP BW or SAP HANA system via your SAP Analytics Cloud. When using live connections, you can use all options that Analysis offers for these systems. The following chapter specifies the options you can use when working with import data connections.

Import data connection means that you can import (copy) data and save it in SAP Analytics Cloud. Changes made to the data in the source system don't affect the imported data until and unless the data is imported again, either manually or as a scheduled job.

You can insert and analyze the following model types developed in the SAP Analytics Cloud in Analysis: [analytic models](#) and [planning models](#).

You can open a cloud model using the [Insert Data Source](#) dialog in the Analysis ribbon. In the [Select Data Source](#) dialog, you can search for a model on the [Search](#) tab or you can select a model in the list on the [Area](#) tab.

After inserting a cloud model, you can analyze the data with Analysis. You can save Analysis workbooks containing SAP Analytics Cloud models as data sources locally.

Note that the following functions are not supported in Analysis:

- You cannot create exception aggregation on the fly (but you can use exception aggregation defined in the model).
- You cannot create dynamic calculations rank and rank olympic.
- You cannot add new lines for planning to the crosstab to enter planning data.  
For planning, you can change data in data cells of the crosstab or you can enter planning data in empty cells (unbooked data cells) of the crosstab.

The following restrictions apply for the SAP Analytics Cloud models:

- Analysis only supports the usage of the default currency (similar to cloud chart behavior). You cannot access non-converted currencies (similar to a cloud table).
- You cannot consume defined thresholds from the model definition. In Analysis, you can use conditional formatting and Table Design formatting.
- Analysis can't visualize SAP Analytics Cloud data locking. Therefore you can change the data in locked cells in Analysis, but you cannot recalculate and save the changed data.  
For more information about SAP Analytics Cloud data locking, see [Configure Data Locking](#).
- Comments added to SAP Analytics Cloud models are not displayed in Analysis.  
For more information about SAP Analytics Cloud comments, see [Adding Comments to a Data Cell](#).

## Related Information

[Planning Data with SAP Analytics Cloud models \[page 220\]](#)

## 7.2 Planning Data with SAP Analytics Cloud models

In Analysis, you can use live data connections and import data connections to connect with SAP Analytics Cloud data sources.

Live Data connection means that you're directly connected to an SAP BW or SAP HANA system via your SAP Analytics Cloud. When using live connections, you can use all options that Analysis offers for planning for these systems. The following chapter specifies the options you can use when planning with import data connections.

Import data connection means that you import (copy) data and save it in SAP Analytics Cloud. Changes made to the data in the source system don't affect the imported data until and unless the data is imported again, either manually or as a scheduled job.

You can enter the planning data manually in a crosstab in your analysis. To enter planning data, you can change the values in existing data cells in the crosstab and you enter planning data in empty cells (unbooked data cells) of the crosstab.

For entering planning data in unbooked data cells, be aware of the following behavior:

- If the dimension has hierarchies defined, but no hierarchy has been selected as display hierarchy (flat presentation), the unbooked cells are not input enabled.  
As hierarchy nodes are shown in the same way as leaves in a flat presentation, this check supports that you do not accidentally book values on hierarchy nodes.
- The data entry on an account with a specific account type is spread to only the accounts with a compatible account type.  
In SAP Analytics Cloud, you can select between the following account types for account dimensions: INC (Income), EXP (Expense), AST (Asset), LEQ (Liabilities and Equity) and NFIN (Non-Financial).  
Example for entering planning data in Analysis: The account dimension REVENUE is of type INC and has two children, INCOME of type INC and EXPENSES of type EXP. A data entry on an unbooked cell on REVENUE will only be spread to INCOME. This leads to the limitation that a parent account will not be input enabled if all the leaf accounts of its descendants have an incompatible account type.  
In theory, the following two groups of account types are compatible among themselves: (INC, AST) and (EXP, LEQ, NFIN). However, semantically meaningful combinations in a hierarchy will only combine INC and EXP or AST and LEQ accounts. So in a semantically meaningful model the value is spread only to leaf accounts of the same account type.

When inserting a cloud model for planning, the *System for Data Write back* field on the Components tab shows the SAP Analytics Cloud system as planning system. Cell locking for cloud model is always set to *Front End Cell Locking*.

When working with public versions, you can recalculate the planning data and you can save planning data to save it in the model.

On the *Versions* tab in the design panel, you can also copy the data of a version to create a new private version. The new private version is added to the crosstab and you can enter the planning data for the new version. When

working with private versions, you can recalculate the planning data to save the data automatically in the model.

On the Versions tab you can also:

- *Save* a private version to publish the data and merge it with the corresponding public version.
- Select *Save as* for a private version and save it as a new public version.
- *Undo* and *Redo* steps executed in private and public versions.
- *Share* private versions with other users and grant them read or read and write access.
- *Revert* changes for a version to delete all changes executed for a version since the version was last saved.
- *Delete* a private or public version.

## Related Information

[The Versions tab \[page 149\]](#)

## 8 Creating Presentations

You can insert a data source in Analysis for Microsoft PowerPoint. The data is displayed in a crosstab.

If a crosstab includes more rows than can be displayed on a slide, the *Fit Table* dialog box appears. In this dialog, you can define if the crosstab should be splitted to multiple slides or displayed abbreviated in one slide.

You can change the view on the data with the following options: Prompting, Filtering, Sorting and Hierachies.

You can change the formatting of a crosstab with *Table Tools*. In the *Table Style Options* group, you can define the display of rows and columns. If you select *Header Row* and *First Column*, the formatting is only applied to the first header row and first column even if your Analysis crosstab contains more than one header rows or columns with member information. The formatting will be applied to these rows and columns after the next Analysis refresh.

### i Note

You should not use the other options in the Table Tools ribbon to format an Analysis crosstab. If you use them, your changes will be overwritten with the next Analysis refresh.

You can select a crosstab to insert the data as a *Chart* or an *Info Field*. You can edit the charts with the Microsoft PowerPoint Chart Tools. The option is available if a chart is selected.

With *Move to*, you can move any object in the presentation to another slide.

You can also paste a copied object from Analysis for Microsoft Exel to Analysis for Microsoft PowerPoint (Smart Copy).

In the properties dialog, you can define specific properties for Analysis in Microsoft PowerPoint. To open this dialog, choose *Properties* in the *Presentation* group in the ribbon.

Corresponding to workbooks, you can save a presentation to a platform server and open it from there. Choose **► Analysis ► Open Presentation ►** or *Save Presentation*. Presentations without charts can also be saved as 1.x version.

### Related Information

[Working with Analysis in Microsoft PowerPoint \[page 20\]](#)

[To open a workbook \[page 30\]](#)

[To save a workbook \[page 32\]](#)

[Presentation properties \[page 224\]](#)

[To smart paste a data source \[page 223\]](#)

## 8.1 To smart paste a data source

### Prerequisites

You have inserted a data source in Analysis for Microsoft Excel.

### Context

In Analysis for Microsoft Excel, you can copy a crosstab or chart that is currently displayed in the workbook. In Analysis for Microsoft PowerPoint, you can then insert the copied object.

In Microsoft PowerPoint, the current navigation state of the objects in Excel is displayed, but the objects exist independently in Excel and PowerPoint. After the creation of the slide, navigation steps in Excel do not affect the objects in PowerPoint and vice versa.

### Procedure

1. In Analysis for Microsoft Excel, select the crosstab or chart that should be copied.
2. Choose *Smart Copy*.
3. Open Analysis for Microsoft PowerPoint and select a slide.
4. Choose ► *Smart Paste* ► *Smart Paste as Table* ◄ or ► *Smart Paste* ► *Smart Paste as Chart* ◄.

The copied object is pasted as table or chart into the slide.

5. Optional step: Change the view on the data.

You can change the view on the data in Microsoft PowerPoint using the functions of the Analysis ribbon.

You can use the undo/redo in the Analysis ribbon to undo and redo steps that you executed with functions in the Analysis ribbon. To undo/redo steps that you executed with Microsoft PowerPoint, for example the deletion of a crosstab, you should use the undo/redo of Microsoft PowerPoint.

6. Optional step: Change the formatting of a crosstab or chart.

### Results

The copied object is displayed on a Microsoft PowerPoint slide.

## 8.2 Presentation properties

In the *Presentation Properties* dialog, you can define specific properties for Analysis in Microsoft PowerPoint. In Microsoft Excel, these settings are available on the *Components* tab in the design panel.

To open this dialog in Microsoft PowerPoint, choose *Properties* in the *Presentation* group in the ribbon.

### Refresh Presentation on Opening

If you select this check box, the data sources in the presentation are refreshed every time the presentation is opened.

If this check box is not selected, the data in the presentation is not automatically refreshed on opening. You can refresh the data sources manually with choosing *Refresh All* in the menu.

### Force Prompt for Initial Refresh

If you select this check box, the prompting dialog is displayed on every refresh.

### Store Prompts with Presentation

If you select this check box, the defined prompt values are saved with the presentation.

### Remove Data Before Saving

If you select this check box, the presentation is saved without the data. When you reopen the presentation, no data is displayed. To display the data, refresh the data sources manually with choosing *Refresh All* in the menu.

### Merge Variables

If you select this check box, the variables are merged for all data sources in the presentation. If the check box is not selected, the variables can be defined for each data source separately.

# 9 Using Analysis in other SAP Applications

After performing an analysis, you might want to use other SAP applications to communicate your findings and share the analysis with others.

To use an analysis in other applications, you save it as an analysis view. An analysis view is a saved navigation state of an analysis including applied filters, hierarchies, and available metadata.

Analysis views can be saved with SAP Analysis for Microsoft Office and with SAP BusinessObjects Analysis, edition for OLAP. They can be inserted in both editions of SAP Analysis.

Like workbooks, analysis views are saved to the business intelligence platform server. While a workbook can comprise several analyses, an analysis view comprises just one.

If analysis views created with a 1.x Analysis version and containing SAP HANA data sources are available, you can use the [Analysis View Migration](#) dialog on the [Conversion](#) tab in the options dialog to migrate them to 2.x analysis views.

You can also export data sources from Analysis to the SAP BusinessObjects Design Studio and use it there for creating applications and dashboards.

## Related Information

[To save an analysis view \[page 225\]](#)

[To insert an analysis view \[page 226\]](#)

[Conversion Options \[page 305\]](#)

## 9.1 To save an analysis view

### Prerequisites

You have to be connected to a business intelligence platform server.

### Procedure

1. In the worksheet, select the crosstab component that you want to save as an analysis view.
2. Choose [Save View](#).

### i Note

If you are not connected to a BI platform server, the Save Query View dialog box appears to save a query view in the BW system.

3. Select the location to save your analysis view to.
4. Type a filename for the analysis view and press [Save](#).

## Results

The analysis view is saved to the repository.

## 9.2 To insert an analysis view

### Procedure

1. In your worksheet, select the cell where you want to insert the analysis view.
2. Choose ► [Insert Data Source](#) ► [Insert Analysis View](#) ►.

If you are not already connected to a business intelligence platform server, connect to the server where the analysis view you want to open is saved.

### i Note

If you want to insert analysis views created with a 1.x Analysis version and containing SAP HANA data sources, you can use the [Analysis View Migration](#) dialog on the [Conversion](#) tab in the options dialog to migrate them to 2.x analysis views.

3. In the folder list, select an analysis view and press [OK](#).

## Results

The selected analysis view is added to the worksheet. You can change the displayed data set and create new views on the data as required.

## Related Information

[Conversion Options \[page 305\]](#)

## 9.3 To export a data source to SAP Lumira / SAP BusinessObjects Design Studio

### Prerequisites

To export a data source and use it in SAP Lumira / SAP BusinessObjects Design Studio, the following prerequisites have to be fulfilled:

- Make sure that you have installed SAP Analysis for Microsoft Office and SAP Lumira / SAP BusinessObjects Design Studio on your machine.
- Make sure that you use both tools in either local mode or with a BI platform. Otherwise you can not insert an exported data source.

### Context

SAP Lumira / Design Studio enables application designers to create analysis applications and dashboards for browsers and mobile devices on top of BW and SAP HANA data sources. In Analysis for Microsoft Office, you can use analysis methods that are not available in Lumira / design studio. For example, you can define calculations, conditional formatting and filters to change a data source. With exporting a changed data source to Lumira / design studio, you can use these changes in the data source also for the application design.

### Procedure

1. To export a data source to the design studio , select the crosstab of the corresponding data source.
2. Choose *Smart Copy* in the context menu.
3. Insert the data source in Lumira / Design Studio.

For more information, see the corresponding Application Designer guide on the SAP Help Portal:

- [SAP Lumira](#)
- [SAP BusinessObjects Design Studio](#)

# 10 Scheduling and Publishing

Using BI Launch Pad and the Central Management Console (CMC), you can schedule Analysis workbooks to automatically run at specified times. The flexible time-based scheduling system allows you to process large reports during off hours and to avoid unnecessary database hits.

You can also create a publication to publish to documents to different users or recipients, schedule documents to run at intervals, and send the documents to multiple destinations, including BI Inboxes and email addresses. Publications can help send information through your organization more efficiently. For example, they enable you to easily distribute information to users or groups of users and to personalize the information each user or group receives.

You can schedule and publish workbooks with SAP BW and SAP HANA data sources. In order to use HANA data sources, you must create the connection from the BI platform to the SAP HANA platform as an HTTP connection.

When a scheduled object runs successfully, an instance is created. An instance is a version of the object that contains data from the time the object ran.

With scheduling, you can precalculate and distribute Analysis workbooks stored on the BI platform repository. To open a distributed workbook, Analysis does not have to be installed. This allows you to make the workbooks available to a wide spectrum of users.

To schedule and publish an Analysis workbook, the connection to BI platform and BW / HANA system must be configured as an SSO and the user needs *Schedule* rights for Analysis workbooks.

## i Note

To enable scheduling, the Analysis BI platform Add-On must be installed. For more information, contact your IT Administrator.

In the BI Launch Pad, you have the following options to schedule an Analysis workbook:

- Scheduling an object.
- Labeling a scheduled instance to name the instance.
- Specifying the recurrence pattern for the object to be run, for example weekly.
- Selecting a destination, for example BI Inbox or Email.
- Selecting a scheduling server group for the object.
- Scheduling an object with parameters (prompts).  
You can enter the data source, the technical name of the variable and the input string as value. The data source is not necessary if the variables are merged in a workbook. For more information on the syntax, see the section Syntax for Entering Values in this guide.
- Scheduling an object with events.  
When you schedule an object with events, it runs after the specified event occurs in the BI platform.
- Rescheduling an object.  
You can reschedule a previously run object or you can make changes to an existing periodic object. For rescheduling an object, you have to use a BI platform 4.2 SP5 or a higher version.
- Publishing

You can create a new publication and select a report bursting method to determine how source documents are personalized, processed, and delivered in a publication.

In the Central Management Console, you have the following additional options to manage and monitor an instance:

- In the Inbox, you can manage and delete instances.
- In the Instance Manager, you can filter for Analysis workbook instances and check their status.
- For instances with status Failed, you can check the error description.
- Publishing  
You can create a new publication and select a report bursting method to determine how source documents are personalized, processed, and delivered in a publication.

For more information about scheduling, see the BI Launch Pad User Guide on SAP Help Portal at <http://help.sap.com/bobip> and the SAP BusinessObjects BI platform CMC Online Help.

## Related Information

[Syntax for Entering Values \[page 134\]](#)

## 10.1 Precalculation Restrictions

There are precalculation restrictions for the following workbook settings and elements:

- The setting *Refresh Workbook on Opening* is ignored.
- The setting *Force Prompt for Initial Refresh* is ignored.
- If the setting *Remove Data Before Saving* is selected, the workbook is not precalculated.
- If the workbook is in protected mode, it is not precalculated.
- SAP formulas concatenating other SAP formulas are not precalculated.
- VBA macros are not executed during precalculation.
- Icons are ignored, for example arrows in conditional formats.
- Comments are ignored.

# 11 Customizing Analysis

In the *Customizing* area in the Analysis backstage menu, you have different options to customize Analysis according to your needs:

- *Customizing the User Interface*  
Here you can define the commands that you want to see on the ribbon tabs. You can hide commands that you don't need. And you can also define new groups and commands whenever you need them.
- *Maintaining settings in Analysis*  
You can specify settings for the Analysis add-in, the Analysis plug-in and the BPC plug-in here.
- *Options*  
Here you can change user options according to your personal preferences. Depending on the selected platform, you might see only a subset of the described options.

## Related Information

[Analysis in the File tab \[page 12\]](#)

## 11.1 Customizing the User Interface

### 11.1.1 Introduction to Customizing the User Interface

#### Why customize the user interface?

To meet your business profile needs, you want to display only the commands that you use the most, and hide those commands you use less often.

For example, you can customize the *Analysis* tab of the Ribbon by hiding the commands you do not need. You can add the groups and commands you use the most.

You customize the user interface by using profiles.

## Which areas of the user interface can you customize?

- In the Microsoft Excel version of Analysis, you can customize:
  - The tabs of the Ribbon for the plug-ins installed (such as Analysis and BPC),
  - The context menu for the plug-ins.
  - The entries to the installed plug-ins in the File Menu.
  - The tabs of the design panel for the Analysis plug-in.
- In the Microsoft PowerPoint version of Analysis, you can customize:
  - The tabs of the Ribbon for the plug-ins installed (such as Analysis and BPC),
  - The context menu (called from tables) for the plug-ins.
  - The entries to the installed plug-ins in the File Menu.

## 11.1.2 Profiles

### What's a profile?

A profile is a set of user interface customizations.

Here are the several types of profiles that exist.

### Profile (or user profile)

When you create a profile using the dedicated dialog box, it is a user profile.

The profile is stored in your user information on your computer at the following location: C:\Users\[user]\AppData\Roaming\SAP\Cof\User Interface

- *Standard Profile (read-only)*. If you haven't made any customizations yet, the standard profile is applied by default. It is tagged with (read-only) and you cannot modify or delete it.
- *Default profile*. This is the profile you set as the profile to be applied the next time you open Analysis. As said above, as long as you haven't made any customizations, the standard profile is applied by default. See [Setting a profile as a default profile \[page 240\]](#).
- *User profile tagged with (read-only)*. So that users can share user profiles without having to make a copy of them, you can store user profiles in a specific dedicated folder: all the profiles are read-only. [Using user profiles as read-only profiles \[page 242\]](#).

## Company Profile

If you are an administrator, you can define a specific profile for your company and provide it to your end-users. This profile is tagged with “Company” and cannot be deleted or modified by end-users.

**Company default profile:** you can set a default company profile to make sure that the end-users have at least one company profile applied.

Besides, you can impose the default company profile and not allow the end-users to create their own profiles. Therefore, they cannot access the User Interface Customization feature.

For further details, see [Creating a Company Profile \[page 241\]](#).

## Embedded Profile

An embedded profile is a user profile that is embedded in a workbook. When you provide the workbook to other users, the embedded profile applies by default. See [Embedding a profile in a workbook \[page 240\]](#).

### 11.1.3 Creating a profile

If you haven't made any customizations yet, the standard profile is applied by default. You can't modify or delete it. The standard profile is named Standard Profile (read-only).

But you can create other profiles, using the dedicated dialog box [Customize User Interface](#).

#### Where do you access the Customize User Interface dialog box?

Select  [File](#) > [Analysis](#) > [Customize Analysis](#) > [Customize User Interface](#) .

#### How do you create a profile?

Using the dialog box [Customize User Interface](#), you can create a profile by clicking [Save As](#) and then entering a name for your profile in the dialog box that appears.

##### Note

if you want this profile to be applied the next time you open Analysis, select the option [Set as Default Profile](#). But you can always set later a profile as the default profile.

The name of profile you have just created is selected in the [Profile](#) dropdown list. You can now start to define the customizations you want: in the tabs of the Ribbon, in the context menus or in the Backstage view.

### i Note

the profile that is currently applied is displayed in the bottom-left part of the *Customize User Interface* dialog box.

## 11.1.3.1 How do you customize a tab in the Ribbon?

You can arrange, add or hide groups or commands as needed in the tabs of the Ribbon.

### i Note

To see how you can customize the Ribbon, check out this how-to video (produced for version 2.4): <https://youtu.be/X1jXGF8NiFs> ➔

### → Tip

You can quickly search for every occurrence of a specific word or phrase, by entering the text that you want to find in the upper-right part of the window. Any instances of the found text are highlighted in yellow.

## Arrange

To arrange the groups or commands of a tab, you can do the following:

- Use the buttons *Move Up*, *Move Down* and *Move to Group*.

### i Note

Using the *Move Up* and *Move Down* buttons, you can move the elements only inside a node element. Using the *Move to Group* button, you can move an element only into a first-level element.

- Use the drag-and-drop to move an element inside or outside a node element, and into any level.

## Show or hide

- To hide or display all the groups and elements of a tab, use the *Show / Hide all Elements*.
- To hide or display all the sub-elements of a group or an element, select the group or element, then right click and use the commands *Show all Elements of* and *Hide all Elements of*.
- Select or deselect the checkboxes in the tree view.

## Add

You can create new tabs.

You can create the following in a tab:

- new groups
- inside a group you have created, you can create the following types of elements: menu, separator, button, split button, or toggle button.

### How do you do it?

To create or delete new tabs, use the [New Tab](#) button (or the tab with the + icon) and the [Delete Tab](#) button. You can enter a name for a new tab and specify where to position it, before or after an existing tab (amongst the tabs that are natively delivered with Microsoft Excel).

To create or delete the groups or elements, use the [New Group](#), [New Sub-Element](#), [Delete](#) buttons or the same commands that you access from the context menu in the tree view of the tab.

### What elements can you create?

Here is the list of elements you can create, with their properties (that are explained below).

- Menu: label, size of the icon, icon.
- Separator: no properties.
- Button: identifier, label, macro, image size, image.
- Split button: identifier, label, macro, image size, image.
- Toggle button: identifier, label, macro (for the action), macro for the state of the button (pressed or not), image size, image.

### What are the possible properties of the elements you create?

- Label: enter the name of the element as it will appear on the interface.
- Image: choose the image that you want to be displayed next to the label. The images are stored within the profile xml.
- Image size: choose the size [Normal](#) or [Large](#), depending on the type of element and available user interface space. If you are creating a group, choose [Large](#). If you are creating another element, choose [Normal](#).
- Get Button State: macro that sets the state of the toggle button, pressed or not.
- Macro: see below the Macro section.
- Display ID: see below the ID section.

## IDs

For buttons, split buttons, and toggle buttons, you can display and modify the automatically generated ID.

### Caution

Once you have changed an ID, the original ID is overwritten, which means that you cannot reset it to the automatically generated ID.

To display an ID, select the option [Display ID](#).

To modify an ID, select the option, then modify it in the row *ID*.

Here are the syntax rules for an ID : must be unique, must not contain colons, must start with either a letter or underscore (\_) and may contain only letters, digits, underscores (\_), hyphens (-), and periods (.).

### **i** Note

You can use the IDs to create VBA root macros.

### **⚠** Caution

As of version 2.5, macros with IRibbonControl are available. If you have defined macros in an earlier version of Analysis, you need to add the control of type IRibbonControl as the first parameter in each macro so that the macros function in version 2.5.

## Macros

For buttons, split buttons, and toggle buttons, you call VBA macros to perform the actions.

A macro can be defined in either one of the following:

- In an .xlam file that you need to enable as an add-in to Microsoft Excel. In this case, in the *Macro* area, enter the name of the file, followed by ! and then by the name of the macro that you want to call. For example: functions.xlam!opendocument
- Directly in the current workbook. In this case, you simply enter the name of the macro in the *Macro* area. For example: opendocument

Even if you do not use it, the first parameter must be the control of type IRibbonControl.

### Example of macro for a button action:

#### **≡** Sample Code

```
Sub MacroName(control As IRibbonControl)
    'You can check control.ID
    'Do whatever you need depending on the control Identifier
End Sub
```

### Example of macro for a toggle button action:

### **i** Note

A macro for a toggle button action has a boolean parameter as the second parameter (the first parameter being IRibbonControl), which corresponds to the state of the button (pressed or not pressed).

#### **≡** Sample Code

```
Sub MacroName(control As IRibbonControl, pressed As Boolean)
    'You can check control.ID
    'Do whatever you need depending on the control Identifier
End Sub
```

**Example of macro for the state of a toggle button:**

#### Sample Code

```
Function MacroName(control As IRibbonControl) As Boolean
    'You can check control.ID
    'Evaluate pressed state as boolean
    MacroName = <pressed state as boolean>
End Function
```

## 11.1.3.2 How do you customize context menus?

You can arrange, add or hide the elements of the context menu for a plug-in.

### Arrange

To arrange the elements of the context menu for a plug-in, you can do the following:

- Use the buttons *Move Up* and *Move Down*.

#### i Note

Using the *Move Up* and *Move Down* buttons, you can move the elements only inside a node element.

- Use the drag-and-drop to move an element inside or outside a node element.

### Show or hide

To hide or display all the elements of a the context menu for a plug-in, use the *Show / Hide all Elements*.

To hide or display all the elements of a the context menu or a specific element, select the context menu or element, then right click and use the commands *Show all Elements of* and *Hide all Elements of*.

### Add

You can create the following:

- new first-level elements in a context menu,
- inside a first-level element you have created, you can create the following types of sub-elements: menu, menu separator, button, split button, or toggle button.

**How do you do it?**

To create or delete elements, use the [New Element](#), [New Sub-Element](#) and [Delete](#) buttons or the same commands that you access from the context menu in the tree view of the tab.

### What elements can you create?

Here is the list of elements you can create, with their properties (that are explained below).

- Menu: label, image, [Activate only on](#), [Hide if not active](#).
- Menu separator: no properties.
- Button: identifier, label, macro, image, [Activate only on](#), [Hide if not active](#).
- Split button: identifier, label, macro, image, [Activate only on](#), [Hide if not active](#).
- Toggle button: identifier, label, macro (for the action), macro for the state of the button (pressed or not), image, [Activate only on](#), [Hide if not active](#).

### What are the possible properties of the elements you create?

- Label: enter the name of the element as it will appear on the interface.
- Image: choose the image that you want to be displayed next to the label. The images are stored within the profile xml.
- Get Button State: macro that sets the state of the toggle button, pressed or not.
- Macro: see the Macro section here: [How do you customize a tab in the Ribbon? \[page 233\]](#).
- Display ID: see the ID section here: [How do you customize a tab in the Ribbon? \[page 233\]](#).
- [Activate only on](#): see the [Activate only on](#) section below.
- [Hide if not active](#): see the [Hide if not active](#) section below.

## Activate only on

By default, the element you are creating is active and visible in the context menu where and whenever, meaning in all the situations listed below. You can restrict this. So that the element be active only in some of situations, you need to select the corresponding options, depending on the plug-in you are using:

- When using the Analysis plug-in, you can activate an element only:
  - on an Analysis worksheet
  - on a data source (whether it is active or not)
  - on an active data source
  - on an inactive data source

### i Note

When selecting several options, all the criteria need to be met so that the element be activated in the context menu.

### i Note

Using the Microsoft PowerPoint version of Analysis, when you create an element in a context menu, the following options are available in the Activate only on area:

- When in the Analysis context menu: [Active Data Source](#).

## Hide if not active

By default, this option is selected. This means that when an element is not active, it is hidden by default.

However, you may want to always show an element, whether it is active or not. In such a case, you can deselect the option *Hide if not active*.

## 11.1.3.3 How do you customize the File menu?

You can arrange, add or hide menu groups or commands as needed for the plug-ins in the File menu.

### Arrange

To arrange the columns, the groups or commands of a tab, you can do the following:

- Use the buttons *Move Up* and *Move Down*.

#### i Note

Using the *Move Up* and *Move Down* buttons, you can move the elements only inside a node element.

- Use the drag-and-drop to move an element inside or outside a node element.

### Renaming columns

To rename the columns, the groups or commands of a tab, click the column and enter a text in the *Label* area.

### Show or hide

- To hide or display all the groups and commands of a tab, use the *Show / Hide all Elements*.
- To hide or display all the elements of a group or an element, select the group or element, then right click and use the commands *Show all Elements of* and *Hide all Elements of*.
- Select or deselect the checkboxes in the tree view.

### Add

You can create new tabs.

You can create the following in a tab: new sub-elements.

You can create the following types of sub-elements: menus under one of the columns, menu groups under menus, then buttons or toggle buttons under menu groups.

### How do you do it?

To create or delete new tabs, use the [New Tab](#) button (or the tab with the + icon) and the [Delete Tab](#) button. You can enter a name for a new tab and specify where to position it, before or after an existing tab (amongst the tabs that are natively delivered with Microsoft Excel).

To create or delete the groups or elements, use the [New Sub-Element](#) and [Delete](#) buttons or the same commands that you access from the context menu in the tree view of the tab.

### What elements can you create?

Here is the list of elements you can create, with their properties (that are explained below).

- Menu: label, description, image.
- Menu group: size.
- Button: identifier, label, description, macro, image, [Leave Backstage View](#).
- Toggle button: identifier, label, description, macro (for the action), macro for the state of the button (pressed or not), image.

### What are the possible properties of the elements you create?

- Label: enter the name of the element as it will appear on the interface.
- Description: enter a text describing what is the aim of the element.
- Image: choose the image that you want to be displayed next to the label. The images are stored within the profile xml.
- Get Button State: macro that sets the state of the toggle button, pressed or not.
- Macro: see the Macro section here: [How do you customize a tab in the Ribbon? \[page 233\]](#).
- Display ID: see the ID section here: [How do you customize a tab in the Ribbon? \[page 233\]](#).
- [Leave Backstage View](#): see the [Leave Backstage View](#) section below.

## Leave Backstage View

When you select this option, once you click the button you are currently creating, you leave the Backstage view.

### 11.1.3.4 How do you customize the Design Panel

You can show and hide tabs of the design panel as needed for the Analysis plug-in. By default, all tabs are shown in Analysis.

To show or hide a tab in the design panel, select or deselect the corresponding checkbox.

The following tabs are available for the design panel in the Analysis plug-in:

- Analysis
- Information

- Components
- Design Rules
- SAC Versions  
This tab is only displayed if a SAC data source is used.
- Comments

## 11.1.4 Renaming a profile

Select the profile you want to rename, then enter the new name and click [Save](#).

## 11.1.5 Setting a profile as a default profile

To set a profile as the default profile:

- When creating a new profile, select the option [Set as Default Profile](#). See [Creating a profile \[page 232\]](#).
- At any time, you can set a default profile by selecting the profile and clicking the appropriate icon (next to cross icon) in the upper part of the [Customize User Interface](#) dialog box.

The profile is applied the next time you open Analysis.

### i Note

As long as you haven't made any customizations, the standard profile is applied by default.

## 11.1.6 Embedding a profile in a workbook

You can embed a user profile in one or more workbooks. When you provide the workbook to other users, the embedded profile applies by default. You can also modify embedded profiles.

Note that you can only apply an embedded profile to workbooks that are saved locally.

When you open a workbook that has an embedded profile, a message appears in the [Customize User Interface](#) dialog.

### i Note

You can embed a profile only in workbooks with an Open XML format: xlsx and xlsm. This concerns all workbooks saved with Excel versions before version 97-2003.

## Why would you embed a profile in a workbook?

For example, for a viewer type of profile, you want all the greyed out buttons to be hidden.

When using a workbook in which a profile is embedded:

- The profile is applied and overrides all the other profiles that you can see in the profile dropdown list of the *Customize User Interface* dialog.
- You can still modify and create other user profiles, but you cannot apply them to this workbook.

## How do you embed a profile in one or more workbooks?

1. Make sure that the workbook(s) that is to get an embedded profile is saved locally.
2. Open an empty workbook and open *Customize User Interface* dialog.
3. Select the profile in the dedicated dropdown list, then do one of the following:
  - To embed the profile in the current workbook, save the current workbook if not already, then select the icon *Embed Profile*, then *Embed [profile name] in the Current Workbook [workbook name]*.
  - To embed the profile in one or more workbooks, then select the icon *Embed Profile*, then *Embed [profile name] in one or more Workbooks*. Select the workbooks, then click *Embed Profile*.
4. You can cancel the process by clicking the *Cancel* button that appears instead of the *Close* button during the embedding process.
5. You can upload the workbook(s) with the embedded profile to the server.

The embedded profile is applied to the selected workbook(s).

## How do you modify an embedded profile?

1. Make sure that the workbook(s) that contains the embedded profile is saved locally.
2. Open an empty workbook and open the *Customize User Interface* dialog.
3. Select the embedded profile in the dedicated dropdown list and change the user interface according to your needs.
4. Select *Save* to save your changes.
5. Select *Embed Profile* and apply the changed profile as embedded profile to the local workbook(s).
6. You can upload the workbook(s) to the server and overwrite the existing version with the changed embedded profile.

## 11.1.7 Creating a Company Profile

If you are an administrator, you can define a specific profile for your company and provide it to your end-users.

When you create a profile, it is a user profile that is stored locally at the following location: C:\Users\[user]\AppData\Roaming\SAP\Cof\User Interface.

## Making a profile a company profile

As an administrator, to make your profile a company profile, you take your profile from the above location and move it to the location for company profiles: C:\ProgramData\SAP\COF\User Interface. This is the default location.

When you open your application, the profile is tagged with "Company" and cannot be deleted or modified by end-users.

## Setting a company profile as the default company profile

You can set a default company profile to make sure that the end-users have at least one company profile applied.

If there is a default company profile, the following rules apply:

- If you have set a user profile as the default profile, this user profile is applied,
- If not, the default company profile is applied,
- If there is no default company profile, the standard profile is applied.

To set a company profile as the default company profile, enter the path to the default company profile in the file Cof\_app.config located in C:\ProgramData\SAP\COF, using the following parameters:

- `<section name="AppBuilderExcelDefaultCompanyProfilePath" type="String" configurationLevel="PerMachine" />`
- `<section name="AppBuilderPptDefaultCompanyProfilePath" type="String" configurationLevel="PerMachine" />`
- `<section name="AppBuilderWrdDefaultCompanyProfilePath" type="String" configurationLevel="PerMachine" />`

If needed, you can impose the default company profile and not allow the end-users to create their own profiles, so they cannot access the User Interface Customization feature.

To impose the default company profile and not give access to the User Interface Customization feature, set this parameter to False:

```
<section name="CanCustomizeUserInterface" type="Boolean" configurationLevel="PerMachine" />
```

## 11.1.8 Using user profiles as read-only profiles

Read-only profiles behave just like user profiles, except that they cannot be modified.

Typically, you store read-only profiles in a shared folder so that other users can use the profiles.

To create a read-only profile, go to this configuration file: "C:\Users\[User1]\AppData\Roaming\SAP\Cof\Cof\_user\_roaming.config"

- You enter the path to the folders that contain the user profiles you want to use as read-only profiles.
- If you enter several paths, enter ; between each path.

- Here is an example where I is a shared folder:

```
<AppBuilderReadOnlyProfileDirectories value="I:\...
\ReadOnlyProfilesDirectory1;I:\...\ReadOnlyProfilesDirectory2 "/>
```

- If you use images in the read-only profiles, the images are stored within the profile xml.

You do not need to have administrator rights to manage read-only profiles.

## 11.2 Maintaining settings in Analysis

Besides maintaining settings directly in the file system, you can modify settings in the Analysis backstage area.

1. Open the *Technical Configuration* dialog.

Select **File > Analysis > Adapt Analysis > Technical Configuration**.

While the dialog with all settings is loaded, you can already start the search for a specific setting and enter the setting name in the search field.

2. Display the settings you want to modify.

You have the following options:

- Enter a setting name in the search field.  
The search is executed immediately and the settings are displayed accordingly.
- Select the configuration file prefix:  
*Ao* for the Analysis plug-in, *Bpc* for the BPC plug-in and *Cof* for the Analysis Add-in.  
The settings are displayed according to their assignment in the file system.

3. Specify the settings to be displayed.

You have the following options:

- *All Settings*  
All settings are displayed.
- *All Settings Deviating from Default Value*  
The settings that have already been modified are displayed.
- *All Settings Containing a Value in File*  
All settings that contain a value in the corresponding file (e.g. *Ao\_app.config*) are displayed.

4. Modify a setting.

Change the default value for the setting. Depending on the type, you can select the check box for boolean values, enter an integer value or enter a text.

In the details view, you get the following information per setting:

- *Type:*  
The type can be *Bool* for boolean values, *Int32* for integer value or *String* for texts.  
For boolean values, the value is *True* if the check box is selected.
- *Default:*  
The default value for the setting is displayed.
- *Publish:*  
The value is always *True*. This means that the setting is public.
- *Loaded From:*  
The path to the configuration file in the file system is displayed.
- *Allowed Configuration Level:*  
The configuration level can be *UserRoaming* or *PerMachine*.

A user can change settings with configuration level *UserRoaming*. The changed settings are stored in the file system under `Users\<UserID>\AppData\Roaming\SAP\Cof`. The file names for the changed settings are *cof\_user\_roaming.config*, *ao\_user\_roaming.config* and *bpc\_user\_roaming.config*. These files are created automatically if you change a setting in the *Technical Configuration* dialog.

An administrator can also change settings with configuration level *PerMachine*. The changed settings are stored in the file system under `C:\ProgramData\SAP\Cof`. The file names are *Cof\_app.config*, *Ao\_app.config* and *Bpc\_app.config*.

As an administrator, you can also change the configuration level from *UserRoaming* to *PerMachine* for any setting. Then, these settings can no longer be changed by a user.

- *Config Prefix:*

The prefix of the corresponding configuration file is displayed. The prefix can be *Ao* for the Analysis plug-in, *Bpc* for the BPC plug-in and *Cof* for the Analysis Add-in.

5. Press *OK*.

The changed setting values are executed.

The Analysis settings are modified according to your changes. The changed values are available in the *Technical Configuration* dialog and in the corresponding files in the file system.

## 11.2.1 Analysis Settings

Analysis Settings

Setting and Setting Value	Setting Description	Available for
AbapTrace= 1 or 0 (default value)	<p>You use this setting to activate the trace tool environment of SAP BW.</p> <p>The default value is 0, meaning that it is deactivated. With a value of 1, you can activate this setting.</p> <p>You can also activate the setting in the Support Settings dialog with setting <i>Enable BW Server Tracing</i>.</p> <p>For more information, see <a href="#">Troubleshooting in Analysis [page 307]</a>.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
ActivateFormulaRecognitionOnExistingCells= true (default value) or false	<p>You use this setting to specify whether formulas in input-enabled cells are saved as table design formulas.</p> <p>The default value is <code>True</code>. This means that the formulas added to input-enabled cells are saved as table design formulas. They are added to the formula section on the design rules tab in the design panel and can be changed there.</p> <p>If you change the value to <code>False</code>, the formulas are not saved in the design panel.</p>	Analysis plug-in
AllowChangingAccessMode= true or false (default value)	<p>You use this setting to specify whether the access mode for member display should be enabled.</p> <p>After installation, the default value is <code>False</code>. This means that the access mode is not enabled for the member display definition in the crosstab and for the <i>Filter by Member</i> dialog box.</p> <p>If you change the parameter value to <code>True</code>, the access mode option is displayed in the menu.</p>	Analysis plug-in
AllowLeavesViewInHierarchicalSelector= true (default value) or false	<p>Changing the display mode from hierarchy to leaves can take a very long time depending on the number of levels and leaves the hierarchy contains. Therefore you can use this setting to specify whether the leaves view should be enabled for hierarchies in the filter and prompts dialog.</p> <p>The default value is <code>True</code>. This means that the leaves view is enabled.</p> <p>If you change the parameter value to <code>False</code>, the leaves view will be disabled.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
<p>AllowLiteralMemberForNodesInHana= true or false (default value)</p>	<p>You open a workbook containing a HANA data source that has a dimension filtered on hierarchy elements. You observe a high amount of HTTP calls to HANA for retrieving the members of the hierarchy.</p> <p>You use this setting to specify whether the number of calls should be reduced.</p> <p>The default value is <code>False</code>. This means that the number of calls is not reduced.</p> <p>If you set the parameter value to <code>True</code>, the number of calls will be reduced.</p>	Analysis plug-in
<p>AllowLiteralMemberSelectionsInVariablesHana= true (default value) or false</p>	<p>You use this setting to specify whether the values you enter for a variable, are validated against the HANA system or not.</p> <p>The default value is <code>True</code>. This means that the values are not validated against the HANA system, but Analysis still checks if the format and data type are correct. Not validating the values may improve the performance in Analysis.</p> <p>If you set the parameter value to <code>False</code>, Analysis validates the entered values against the HANA system.</p>	Analysis plug-in
<p>AllowOverwritingOfDimensionNames= true (default value) or false</p>	<p>You use this setting to specify whether users are allowed to overwrite and change the name of dimensions in the property view in the design panel.</p> <p>The default value is <code>True</code>. This means that users can overwrite the name of dimensions.</p> <p>If you change the parameter value to <code>False</code>, users can no longer overwrite dimension names in the property view.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
AllowOverwritingOfStructureMemberNames= true (default value) or false	<p>You use this setting to specify whether users are allowed to overwrite and change the name of structure members in the property view in the design panel. Structure members are single measures and attributes.</p> <p>The default value is <code>True</code>. This means that users can overwrite the name of members.</p> <p>If you change the parameter value to <code>False</code>, users can no longer overwrite member names in the property view.</p>	Analysis plug-in
AlwaysDoApplicationSteps= true or false (default value)	<p>You use this setting to specify when the statistics are written in table RSDDSTAT_OLAP.</p> <p>The default value is <code>False</code>. This means that the statistics are written when Analysis closed. If Analysis is terminated unexpectedly, e.g. by time out on a citrix server, the entries are never written to RSDDSTAT_OLAP.</p> <p>If you change the parameter value to <code>True</code>, the entries are written after each action executed in Analysis without the need to close Analysis. Note that this might have a negative impact on the performance.</p>	Analysis plug-in
AoTenantSite= "" (default value)	<p>This setting is used for Usage Tracking.</p> <p>After installation, the default value is <code>""</code>. We recommend not changing the value.</p> <p>You can deactivate Usage Tracking for Analysis with setting <code>TrackUsage</code>.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
AppBuilderCompanyProfileDirectory="C:\ProgramData\SAP\Cof\User Interface" (default value)	<p>You use this setting to specify the path to the directory which contains the company profiles.</p> <p>The default value is C:\ProgramData\SAP\Cof\User Interface.</p> <p>The company profiles are owned by an administrator. The current user is not the owner of these profiles and he won't be able to modify them.</p> <p>This setting can only be maintained by an administrator in the file system under C:\ProgramData\SAP\Cof.</p>	Analysis add-in (COF)
AppBuilderDefaultInterfaceArea="" (default value), ContextMenu, Backstage or SidePanel	<p>You use this setting to specify which user interface area is selected when you open the Customer User Interface dialog.</p> <p>After the installation, the default value is empty (""). This means that the user interface area Ribbon is selected when you open the dialog.</p> <p>You can enter ContextMenu to enable the user interface area Context Menu when you open the dialog, Backstage to enable the user interface area File Menu, and SidePanel to enable the user interface area Side-Panel View.</p> <p>Note that when you close the dialog, the currently active user interface area is saved as value in the setting. Therefore, the next time you open the dialog again, the same interface area is enabled.</p>	Analysis add-in (COF)

Setting and Setting Value	Setting Description	Available for
AppBuilderDefaultProfilePath= "" (default value)	<p>You use this setting to specify the path to the default profile that will be applied when Microsoft Excel is started.</p> <p>The path is set automatically as soon as a user defines a profile in the <i>Customize User Interface</i> dialog and then selects the <i>Default Profile</i> button.</p> <p>After the installation, no path is defined. Therefore the default value is empty ("").</p> <p>We do not recommend to change the path manually.</p>	Analysis add-in (COF)
AppBuilderExcelDefaultCompanyProfilePath= "" (default value)	<p>You can use this setting to specify the path to the company profile for Microsoft Excel.</p> <p>After the installation no path is defined. Therefore the default value is empty ("").</p> <p>As an administrator, you can enter the path to the default company profile and provide it to your end-users.</p>	Analysis add-in (COF)
AppBuilderPptDefaultCompanyProfilePath= "" (default value)	<p>You can use this setting to specify the path to the company profile for Microsoft PowerPoint.</p> <p>After the installation no path is defined. Therefore the default value is empty ("").</p> <p>As an administrator, you can enter the path to the default company profile and provide it to your end-users.</p>	Analysis add-in (COF)
AppBuilderPptDefaultProfilePath= "" (default value)	<p>You use this setting to specify the path to the default profile that will be applied when Microsoft PowerPoint is started.</p> <p>The path is set automatically as soon as a user defines a profile in the <i>Customize User Interface</i> dialog and then selects the <i>Default Profile</i> button.</p> <p>After the installation no path is defined. Therefore the default value is empty ("").</p> <p>We do not recommend to change the path manually.</p>	Analysis add-in (COF)

Setting and Setting Value	Setting Description	Available for
AppBuilderReadOnlyProfileDirectories="path1;path2"	<p>You use this setting to specify a directory or a list of directories where users can share profiles without having to make a copy.</p> <p>You can enter a single path to a directory or paths to several directories. The paths should then be entered with a semicolon-separated list.</p> <p>The current user is not the owner of these profiles and he won't be able to modify them.</p>	Analysis add-in (COF)
AppBuilderUserProfileDirectory="" (default value)	<p>You use this setting to specify the path to the directory which contains the current user profiles.</p> <p>The current user is the owner of these profiles and can modify them. Each time, the user creates a new profile in the <i>Customize User Interface</i> dialog, it will be stored in this directory.</p> <p>After the installation no directory is defined. Therefore the default value is empty ("").</p> <p>As soon as a user saves the first profile, the path to the directory is set automatically- The path will be something like: C:\Users\<user id="">\AppData\Roaming\SAP\Cof\User Interface.</user></p> <p>A user can change the path manually to use another specific folder.</p>	Analysis add-in (COF)

Setting and Setting Value	Setting Description	Available for
<code>AutoCompleteFetchMemberLimit= 30</code> (default value) or any integer value	<p>You can use an autocomplete input help for entering members in cells of new lines for planning. To use this help, enter a string in the cell, for example the first letters of a member, then press Ctrl+Space. A dialog is opened containing all members matching the string.</p> <p>With this setting, you can define a limit for the number of members available as input help.</p> <p>The default value is 30. This means that up to 30 members are offered in the input help.</p> <p>You can enter any integer value for this setting.</p>	Analysis plug-in
<code>BexAdvancedMode= true</code> or <code>false</code> (default value)	<p>You use this setting to specify whether the 'Use Currency Translation from Query Definition' checkbox should be available in the currency translation dialog for measures.</p> <p>After installation, the default value is <code>False</code>. This means that the checkbox is not available.</p> <p>If you change the parameter value to <code>True</code>, the checkbox is displayed in the dialog.</p>	Analysis plug-in
<code>BicsNewLineFillInitial= true</code> (default value) or <code>false</code>	<p>You use settings to define how Analysis handles empty cells in new lines that are used for planning with new member combinations.</p> <p>The default value is <code>True</code>. This means that Analysis tries to derive missing member values. If a member value cannot be derived, Analysis fills in Not Assigned (Key: #).</p> <p>If you change the value to <code>False</code>, this behavior is stopped and Analysis does not try to derive missing member values.</p> <p>For more information on related SAP BW parameters, see SAP Note <a href="#">2508938</a>.</p> 	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
<code>BipSequentialDataTransferBufferSize= 262144</code> bytes (default value) or any integer value for bytes	<p>You use this setting to specify the number of bytes used for the UploadFile web service call when saving a workbook on the BI platform.</p> <p>The default value is 262144 bytes per call.</p> <p>When working with larger workbooks, increasing the number of bytes per call could improve the performance, for example, a value of 524288 bytes per call.</p> <p>If you change the value for this setting, you need to change accordingly the <code>maximumChunkSize</code> for the BI platform at <code>&lt;InstallDir&gt;\SAP BusinessObjects Enterprise XI 4.0\warfiles\webapps\dswsbobje\WEB-INF\classes</code>.</p>	Analysis plug-in
<code>BOESessionRefreshTime= -1</code> (default value) or any integer value.	<p>You use this setting to specify the time in seconds Analysis is waiting to make a dummy Http request to keep the BI platform session active.</p> <p>The default value is -1. This means that the time taken is the session timeout retrieved from the BI platform minus one minute: Analysis will send the dummy Http request to the BI platform before the timeout value so that Analysis remains active.</p> <p>You can enter any integer value for this setting to define the time for Analysis.</p>	Analysis plug-in
<code>BOESystems</code>	This setting contains the configuration to the Business Intelligence Platform. For more information, see .	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
CancelPopupDelay= 5 seconds (default value) or any integer value	<p>You use this setting to specify after how many seconds the cancel dialog should be displayed when a data update is requested from the server (BW and HANA) and the crosstab is redrawn. This could be inserting a data source or navigating through the data, for example filtering data or adding dimensions to the crosstab.</p> <p>The default value is 5 seconds. This means that the cancel dialog will appear after 5 seconds. You can enter any integer value for this setting.</p> <p>If the cancel dialog (Fetching data from server) is displayed, you have two options:</p> <ul style="list-style-type: none"> <li>You press <i>Cancel</i> to cancel the server request. In the following <i>Messages</i> dialog, you can select <i>Restart</i> to go back to the workbook and the data source is active. Or you select <i>Close</i> to go back to the workbook and the data source is offline.</li> <li>You do not cancel the server request. The dialog will disappear automatically when the server request is completed.</li> </ul>	Analysis plug-in
CanCustomizeUserInterface= true(default value) or false	<p>You use this setting to specify whether users are authorized to customize the user interface and define their own profiles.</p> <p>After installation, the default value is <code>True</code>. This means that are authorized to customize the user interface.</p> <p>If you change the parameter value to <code>False</code>, are no longer able to customize the user interface.</p>	Analysis add-in (COF)

Setting and Setting Value	Setting Description	Available for
<code>CheckIfWorkbookIsInplaceWhenWindowIsActivated= true</code> (default value) or <code>false</code>	<p>This setting might only be relevant if you use the add-in with Microsoft Excel 2010 and at least two workbooks are used at the same time.</p> <p>Please see SAP Note <a href="#">2463180</a> for more information.</p>	Analysis plug-in
<code>CheckInfoAreasAuthorization= true</code> or <code>false</code> (default value)	<p>In BW systems, you can define with authorization object S_RS_FOLD if InfoAreas should be displayed in dialogs of the BEx tools. If it is defined in a BW system that the InfoAreas are not displayed, they are still displayed in the Analysis Open Data Source dialog.</p> <p>You use this setting to specify whether the areas should be displayed in Analysis.</p> <p>The default value is <code>False</code>. This means that the areas are displayed in Analysis.</p> <p>If you change the parameter value to <code>True</code>, the areas are no longer displayed in the Analysis Open Data Source dialog.</p>	Analysis plug-in
<code>ClientLanguage= EN</code> or any Analysis language value	<p>You use this setting to specify the language for all Analysis UI texts.</p> <p>The default value is EN. This means that the UI texts will be displayed in English.</p> <p>You can enter any Analysis language value for this setting.</p> <p>For more information on available languages and their values, please see <a href="#">Supported Languages [page 307]</a></p> <p>Note that the setting <code>DetermineClientLanguageFromOfficeLanguage</code> overrules this setting as long as it has the value <code>true</code>.</p>	Analysis add-in (COF)

Setting and Setting Value	Setting Description	Available for
ClientProfiling= true or false (default value)	<p>You use this setting to activate client profiling.</p> <p>After installation, the default value is False. With a value of True, you can activate this setting.</p> <p>You can also activate the setting in the Support Settings dialog with setting <a href="#">Enable Client Profiling</a>.</p> <p>For more information, see <a href="#">Troubleshooting in Analysis [page 307]</a>.</p>	Analysis add-in (COF)
ColumnsOfData= 12 (default value)	<p>You use this setting to define the default number of columns.</p> <p>After installation, the default value is 12.</p>	Analysis plug-in
ContinuousUsageDataUploadIntervalInDays= 20 (default value)	<p>This setting is used for Usage Tracking.</p> <p>After installation, the default value is 20. We recommend not changing the value.</p> <p>You can deactivate Usage Tracking for Analysis with setting <code>TrackUsage</code>.</p>	Analysis plug-in
ConversionType= 0 (default value), 1 or 2	<p>You use this setting to define which objects of a BEx workbook should be converted. The default value for this parameter is 0. This means that all objects are converted.</p> <p>If you set the parameter value to 1, data sources and crosstabs are converted. If you set the value to 2, only data sources are converted.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
CpicTraceLevel= 0 (default value), 1, 2 or 3	<p>This setting can be used for SAP error handling. Common Programming Interface - Communication (CPIC) is the communication layer under JRFC (or JCo).</p> <p>You use this setting to specify the desired level for tracing.</p> <p>The default value is 0. This means no tracing takes place.</p> <p>You can change the parameter value to 1, 2 or 3, where 3 is the highest and most detailed level of tracing.</p> <p>If you now work with Analysis, log files will be created in the %temp% folder of windows. There you can find a nco_cplic_XXX.trc file that you can attach to the message.</p>	Analysis add-in (COF)
CurrentCodePage= Encoding for the system's current code page	<p>In some cases, the Open dialog shows system entries with unreadable characters when using saplogon.ini. This can be caused by the fact that the encoding of the saplogon.ini content is different from Windows default encoding.</p> <p>You use this setting to overrule the default code page by applying the appropriate code page number here.</p>	Analysis add-in (COF)
CurrentCodePage= Encoding for the system's current code page	<p>In some cases, the Open dialog shows system entries with unreadable characters when using saplogon.ini. This can be caused by the fact that the encoding of the saplogon.ini content is different from Windows default encoding.</p> <p>You use this setting to overrule the default code page by applying the appropriate code page number here.</p>	Analysis plug-in


Setting and Setting Value	Setting Description	Available for
DefaultBWQueryDesigner= 0 (default value) or 1	<p>You can access the BEx Query Designer or the BW Modeling Tools directly from Analysis with the ribbon option <i>Call Query Designer</i>. If both designers are available, you can open a drop down list to select one.</p> <p>You use this setting to specify which designer is opened if the <i>Call Query Designer</i> option is selected in the ribbon without opening the drop down list.</p> <p>After installation, the default value is 0. This <i>Call Query Designer</i> is selected directly.</p> <p>If you change the parameter value to 1, the BEx Query Designer will be opened if you select <i>Call Query Designer</i> directly.</p>	Analysis plug-in
DefaultConnectionLimit= 10 or any integer value	<p>You use a workbook with several SAP HANA data sources. When refreshing the workbook, a thread is created for each data source for retrieving the result set via HTTP (Note that the number of parallel threads might be limited by the Analysis setting MaxNumberOfParallelThreads).</p> <p>Due to a limitation in the underlying .net framework, the SAP HANA server gets at most two HTTP requests in parallel.</p> <p>You use this setting to overwrite the .net default value and specify</p> <p>The default value is 10. This means that Analysis can get up to 10 HTTP requests in parallel.</p> <p>You can enter any integer value for this setting.</p>	Analysis add-in (COF)

Setting and Setting Value	Setting Description	Available for
<code>DefaultLauncherScheme=""</code> (default value) or <code>https</code>	<p>When logged on to BI Launchpad using HTTPS and trying to view an Analysis workbook or presentation, an error message "Failed to launch with specified file!" is displayed.</p> <p>You can implement a BI platform with version 4.1 SPO3 (or higher) or 4.1 Patch 2.2 (or higher) to solve the issue.</p> <p>Alternatively, you can also set this setting to value <code>https</code>. All viewed Analysis workbooks/presentations will then launch via HTTPS. Thus remove this registry value as soon as you have implemented one of the requested BI platform versions.</p>	Analysis plug-in
<code>DefaultWorkbookPath= path to default workbook in directory.</code>	Use this setting to define the path to the default workbook in the directory.	Analysis plug-in
<code>DetermineClientLanguageFromOfficeLanguage= true</code> (default value) or <code>false</code>	<p>You use this setting to specify whether the Microsoft Office language should determine the language for Analysis. This setting overrules the setting <code>ClientLanguage</code>.</p> <p>The default value is <code>True</code>. This means that the Microsoft Office language determines the language used for Analysis. If you enter a different language value for the setting <code>ClientLanguage</code>, the Analysis texts are still displayed in your selected Microsoft Office language.</p> <p>If you change the parameter value to <code>FALSE</code>, the Microsoft Office language no longer determines the language used for Analysis texts. If you enter a different language value for the setting <code>ClientLanguage</code>, the Analysis texts are displayed in the language specified in setting <code>ClientLanguage</code>.</p>	Analysis add-in (COF)

Setting and Setting Value	Setting Description	Available for
<code>DisplayCompoundAsKeys= true</code> or <code>false</code> (default value)	<p>You use this setting to define the display of variables in the summary view of the prompting dialog box.</p> <p>The default value is <code>False</code>. This means that the variables are displayed as display strings.</p> <p>For compound characteristics, the display string may not be unique. To display compound characteristics with the key instead of the display string, you can set the parameter value to <code>True</code>.</p>	Analysis plug-in
<code>DocumentCacheFolderPath= ""</code> (default value)	<p>You use this setting to overwrite the default cache directory path.</p> <p>Analysis workbooks are saved to the directory <code>sapaocache</code> which is located beneath the users <code>Temp</code> directory. It is not possible to add the specific workbooks path to Excel's <code>Trusted Location</code> section. This is due to the fact that Microsoft does not allow adding paths to <code>Trusted Location</code> which points to a directory beneath the users <code>Temp</code> directory whenever working with MS Office 2010 or higher.</p> <p>If you want to overwrite the default cache directory path, you can enter a path here.</p> <p>Note that the path must not be set to a OneDrive or SharePoint folder.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
DoSelectorHierarchyExplicitSelection=true or false (default value)	<p>You use this setting to specify the behavior of the <i>Filter By Member</i> dialog for hierarchies. For hierarchies, the selection of a node means also the selection of its children in the filter dialog box. And vice versa, the selection of all children means also the section of the corresponding node.</p> <p>This is the behavior for the default value <code>False</code>.</p> <p>If you change the value to <code>True</code>, the hierarchy selection behaves different. The selection of a node still means the selection of its children. But you can select all children without having selected automatically the corresponding node.</p> <p>If new children are available in the hierarchy, they are not selected automatically as long as the node is not selected.</p>	Analysis plug-in
DPPPopup_enforce= 1 (default value) or any integer value	<p>You use this setting to specify whether the data protection and privacy pop-up should be displayed when Analysis is started.</p> <p>The data protection pop-up will be displayed when starting Analysis as long as the value for this setting is higher than the value for setting <code>DPPPopup_shown</code>. Therefore, you can also re-enforce that the pop-up is displayed with setting the value for this setting to a higher value than <code>DPPPopup_shown</code>.</p> <p>1. This means that the pop-up will be displayed as the default value for <code>DPPPopup_shown</code> is 0.</p> <p>For more information on Data Protection and Pricacy, see .</p>	Analysis add-in (COF)

Setting and Setting Value	Setting Description	Available for
DPPPopup_shown= 0 (default value) or any integer value	<p>You use this setting to specify whether the data protection and privacy pop-up should be displayed again when Analysis is started.</p> <p>The data protection pop-up will be displayed again when starting Analysis as long as the value for this setting is lower than the value for setting <code>DPPPopup_enforce</code>. Therefore, you can also re-enforce that the pop-up is displayed with setting the value for this setting to a lower value than <code>DPPPopup_enforce</code>.</p> <p>The default value is 0. This means that the pop-up will be displayed as the default value for <code>DPPPopup_enforce</code> is 1.</p> <p>If you select the check box <i>Do not show again</i> in the pop-up, the value is set to 1 and the pop-up will not be displayed when starting Analysis the next time as both settings have the same value.</p>	Analysis add-in (COF)
EnableAnalysisViewConversion= true or false (default value)	<p>You use this setting to define whether the <i>Conversion</i> tab with the <i>Analysis View Migration</i> setting in the settings dialog is displayed.</p> <p>After installation, the default value is <code>False</code>. Nevertheless, the conversion tab with the <i>Analysis View Migration</i> is visible as long as the setting <code>EnableWorkbookConversion</code> is set to <code>true</code>.</p> <p>If you change the parameter value to <code>True</code>, the <i>Conversion</i> tab is enabled and the <i>Analysis View Migration</i> is available even if the setting <code>EnableWorkbookConversion</code> is set to <code>false</code>.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
<code>EnableDoubleClick=true</code> (default value) or <code>false</code>	<p>You use this setting to specify whether filtering a member with a double click should be enabled.</p> <p>After installation, the default value is <code>True</code>. This means you can filter for one member with a double click on the member cell.</p> <p>If you change the parameter value to <code>False</code>, the function is disabled.</p>	Analysis plug-in
<code>EnableImprovedAddInCompatibility=true</code> or <code>false</code> (default value)	<p>If you use different Microsoft Office add-ins at the same time, you might encounter Microsoft Office crashes. The reason for this is that many of the add-ins use the Window Procedure Subclassing technique. You can improve the compatibility with other add-ins with this setting. It reduces the subclassing in Analysis. Note that you cannot use drag and drop to add or delete a filter in the crosstab when the setting is active.</p> <p>The default value is <code>false</code>. This means that the subclassing is not reduced and you can use drag and drop in the crosstab.</p> <p>If you change the parameter value to <code>True</code>, the subclassing is reduced to improve the compatibility with other add-ins. As a consequence, you cannot use drag and drop in the crosstab.</p> <p>For more information, see SAP Note <a href="#">2750514</a> </p>	Analysis plug-in
<code>EnableMassDataSelector=true</code> (default value) or <code>false</code>	<p>You use this setting to specify whether the filter dialog box for mass data should be enabled.</p> <p>After installation, the default value is <code>True</code>. This means that the filter dialog box for mass data is opened if the maximum number of members defined in the User settings is reached.</p> <p>If you change the parameter value to <code>False</code>, the function is disabled and the filter dialog box for mass data is not opened.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
<code>EnableNativeFormatting=true</code> or <code>false</code> (default value)	<p>Analysis renders date, date time and time span values with format <i>Text</i>. It does not use the native Microsoft Excel formats <i>Date</i> and <i>Time</i>.</p> <p>You use this setting to enable native Microsoft Excel formats for rendering in Analysis.</p> <p>The default value is <code>False</code>. This means that native Excel formats are not used for rendering..</p> <p>If you change the value to <code>True</code>, Analysis renders date, date time and time span values as native Microsoft Excel <i>Date</i> and <i>Time</i> formats.</p>	Analysis plug-in
<code>EnablePreferredDocumentStorage=true</code> (default value) or <code>false</code>	<p>You use this setting to specify whether a user should be enabled to select a preferred comments storage in the platform options dialog box.</p> <p>After installation, the default value is <code>True</code>. This means that the Comments Storage section is visible in the platform options dialog box and that the user can change the selection.</p> <p>If you change the parameter value to <code>False</code>, this section is hidden in the platform options dialog box, and the user cannot change the preferred comments storage. The user is therefore unable to change the comments storage defined in setting <code>PreferredDocumentStorage</code>.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
EnablePreferredPlatform=true (default value) or false	<p>You use this setting to specify whether a user should be enabled to select a preferred platform in the platform settings dialog box.</p> <p>After installation, the default value is <code>True</code>. This means that the preferred platform section is visible in the platform settings dialog box and that the user can change the selection.</p> <p>If you change the parameter value to <code>False</code>, this section is hidden in the platform settings dialog box, and the user cannot change the preferred platform. The user is therefore unable to change the preferred platform defined in the <i>Preferred Platform</i> setting.</p>	Analysis plug-in
EnableResetDataSource=true (default value) or false	<p>This setting is used in Analysis with minor versions <math>\leq 3</math> to specify whether the <i>Call Query Designer</i> icon should be displayed in the ribbon.</p> <p>As of release 2.4, you customize the ribbon with the <i>Customize User Interface</i> dialog. Therefore, we recommend to use the <i>Customize User Interface</i> dialog to define the options displayed in the ribbon.</p> <div data-bbox="603 1339 991 1736" style="border-left: 2px solid #0070C0; padding-left: 10px; margin-top: 10px;"> <p><b>i Note</b></p> <p>When switching to a version greater than 3, for example Analysis 2.4, Analysis takes your definition from this setting. If the parameter value for this setting was <code>True</code>, the <i>Call Query Designer</i> icon is displayed in the standard profile for the ribbon. If it was <code>False</code>, it is not displayed.</p> </div>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
EnableWorkbookConversion= true (default value) or false	<p>You use this setting to define whether the conversion of BEx workbooks is enabled. The settings for the BEx workbooks conversion are available on the <a href="#">Conversion</a> tab in the settings dialog.</p> <p>After installation, the default value is <code>True</code>. This means that conversion tab with the conversion settings and the menu entry for conversion are visible in Analysis.</p> <p>If you change the parameter value to <code>False</code>, the menu entry and conversion tab are hidden, and the user is not able to convert BEx workbooks. If you enable the setting <code>EnableAnalysisViewConversion</code>, the conversion tab with the conversion settings is visible in Analysis, but the menu entry for BEx workbook conversion is hidden.</p>	Analysis plug-in
EnableWorkspaces= true (default value) or false	<p>You use this setting to define whether the workspace options should be enabled.</p> <p>After installation, the default value is <code>True</code>. This means that the workspace options are enabled and the menu entries are visible in the ribbon.</p> <p>If you change the parameter value to <code>False</code>, the menu entries are not displayed in the ribbon, and the user is not able to use the workspace options.</p>	Analysis plug-in


Setting and Setting Value	Setting Description	Available for
EnforceDatePickerForCalendarDayVariable= true or false (default value)	<p>You use this setting to specify whether the date picker for any variable of value type calendar / day should be enabled.</p> <p>After installation, the default value is <code>False</code>. This means that the date picker is not enabled and the dates are displayed in a flat list. Dates that are displayed in a flat list, are fetched from back-end and validated in Analysis. This could take longer than using the date picker.</p> <p>If you change the parameter value to <code>True</code>, the date picker is enabled. It is shown directly without fetching the date values from back-end. A validation does not take place and it is assumed that the selected date is valid.</p>	Analysis plug-in
FailedPingCounter= 0 (default value)	<p>This setting is used for Usage Tracking.</p> <p>After installation, the default value is 0. We recommend not changing the value.</p> <p>You can deactivate Usage Tracking for Analysis with setting <code>TrackUsage</code>.</p>	Analysis plug-in
FetchMemberLimit= 1000 (default value) or any integer value	<p>You use this setting to define the maximum number of members displayed in the <i>Filter by Member</i> dialog box for selection. If you filter on a dimension that contains more members than defined here, you only see the currently selected members (but you can search for all members).</p> <p>The default value is 1000. This means that up to 1000 members will be displayed. You can enter any integer value for this setting.</p> <p>You can also define the maximum number of members in the user settings in Analysis.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
ForceRefreshConnectionInfo = true or false (default value)	<p>You use this setting to specify whether the SNC (Secure Network Communications) information is read from the launcher file or from the local SAPUI-Landscape.xml installation.</p> <p>The default value is <code>False</code>. This means that the SNC information is read from the launcher file.</p> <p>If you change the parameter value to <code>True</code>, only the SNC information in the local SAPUILandscape.xml is used.</p> <p>Therefore we do not recommend changing this setting to value <code>True</code>.</p>	Analysis plug-in
ForceWriteOlapConnectionInformationDuringWorkbookSave = true or false (default value)	<p>You can use this setting to fix workbooks with OLAP connections saved on the BI platform using an Analysis version 2.5 SP1 or lower.</p> <p>With these workbooks, it is possible that precalculation does not work and you get a message from the server like 'Document does not contain OLAP connections'.</p> <p>To fix the workbook, set the value to <code>True</code>.</p> <p>After setting the value to true, restart Analysis and open the workbook from the BI platform to overwrite the existing one.</p>	Analysis plug-in
FormulaExpressionValidationInterval = 2 (default value)	<p>You can add a new measure based on a free-form calculation to a crosstab. The new measures are defined in the <a href="#">New Calculation</a> dialog box. The formula that you enter in the dialog box is checked on a regular basis.</p> <p>You use this setting to specify the number of seconds. The default value is 2. This means that the formula is checked two seconds after your last change in the formula editor.</p> <p>You can enter any integer value for this setting.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
HanaHttpConnectionTimeout= -1 (default value) or any integer value.	<p>You use this setting to specify the time Analysis is waiting to get a connection to a HANA HTTP server.</p> <p>The default value is -1. This means that there is no timeout for connecting to a HANA HTTP server.</p> <p>You can enter any integer value for this setting to define the time Analysis is waiting to get a connection. The unit is millisecond. If you specify value 60000, for example, Analysis is waiting 60000 milliseconds to establish the connection. If it is not possible to connect to a HANA HTTP server in the defined timeout period, a message is displayed in Analysis.</p>	Analysis plug-in
HanaHttpSessionPingInterval	<p>You use this setting to specify the time Analysis is waiting to send a regular HTTP ping request to HANA XS to keep the session alive.</p> <p>The default value is 180. This means that Analysis is sending the request every 180 seconds.</p> <p>You can enter any integer value for this setting to define the time for the next ping request.</p> <p>There is a correlation of the session timeout defined in HANA XS. This session timeout value can be configured in the HANA XS configuration under xsengine.ini -&gt; httpserver -&gt; sessiontimeout (value in seconds).</p>	Analysis plug-in
HierarchyCollapsePaddingCols= " " (default value)	<p>You use this setting to specify the indent in columns when the node is expanded with space characters.</p> <p>The default value is " " (five spaces). You can enter any number of spaces here to define the indent.</p> <p>The value for this setting is only applied in Analysis if setting UseAdvancedHierarchySettings is activated with value True.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
<code>HierarchyCollapsePaddingRows= " " (default value)</code>	<p>You use this setting to specify the indent in rows when the node is expanded with space characters.</p> <p>The default value is " " (five spaces). You can enter any number of spaces here to define the indent.</p> <p>The value for this setting is only applied in Analysis if setting <code>UseAdvancedHierarchySettings</code> is activated with value <code>True</code>.</p>	Analysis plug-in
<code>HierarchyCollapseSymbolColumns= [-] (default value)</code>	<p>You use this setting to specify the character to be used as a symbol for collapsing hierarchy nodes in columns.</p> <p>The default value is [-]. You can use Unicode codepoints to specify other symbols, e.g. <code>\u25bc</code> for a black down-pointing triangle.</p> <p>The value for this setting is only applied in Analysis if setting <code>UseAdvancedHierarchySettings</code> is activated with value <code>True</code>.</p>	Analysis plug-in
<code>HierarchyCollapseSymbolRows= [-] (default value)</code>	<p>You use this setting to specify the character to be used as a symbol for collapsing hierarchy nodes in the rows.</p> <p>The default value is [-]. You can use Unicode codepoints to specify other symbols, e.g. <code>\u25bc</code> for a black down-pointing triangle.</p> <p>The value for this setting is only applied in Analysis if setting <code>UseAdvancedHierarchySettings</code> is activated with value <code>True</code>.</p>	Analysis plug-in
<code>HierarchyExpandPaddingColumns= " " (default value)</code>	<p>You use this setting to specify the indent in columns when the node is expanded with space characters.</p> <p>The default value is " " (five spaces). You can enter any number of spaces here to define the indent.</p> <p>The value for this setting is only applied in Analysis if setting <code>UseAdvancedHierarchySettings</code> is activated with value <code>True</code>.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
<code>HierarchyExpandPaddingRows</code> = " " (default value)	<p>You use this setting to specify the indent in rows when the node is collapsed with space characters.</p> <p>The default value is " " (five spaces). You can enter any number of spaces here to define the indent.</p> <p>The value for this setting is only applied in Analysis if setting <code>UseAdvancedHierarchySettings</code> is activated with value <code>True</code>.</p>	Analysis plug-in
<code>HierarchyExpandSymbolCols</code> [+] (default value)	<p>You use this setting to specify the character to be used as a symbol for expanding hierarchy nodes in the columns.</p> <p>The default value is [+]. You can use Unicode codepoints to specify other symbols, e.g. <code>\u25bc</code> for a black down-pointing triangle.</p> <p>The value for this setting is only applied in Analysis if setting <code>UseAdvancedHierarchySettings</code> is activated with value <code>True</code>.</p>	Analysis plug-in
<code>HierarchyExpandSymbolRows</code> [+] (default value)	<p>You use this setting to specify the character to be used as a symbol for expanding hierarchy nodes in the rows (e.g. <code>\u25ba</code>).</p> <p>The default value is [+]. You can use Unicode codepoints to specify other symbols, e.g. <code>\u25bc</code> for a black down-pointing triangle.</p> <p>The value for this setting is only applied in Analysis if setting <code>UseAdvancedHierarchySettings</code> is activated with value <code>True</code>.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
IgnoreCaseForNewLineCharacteristics=true or false (default value)	<p>You use settings to specify whether the text for a characteristic member that you enter in a new line cell should be treated as case sensitive data.</p> <p>The default value is <code>False</code>. This means that the text will be treated as case sensitive and a member value which only is different in an upper or lower case from the expected value on the server is not accepted.</p> <p>If you change the value to <code>True</code>, the entered text will be treated as upper case text. Also here, a member value which only is different in an upper or lower case from the expected value on the server is not accepted.</p>	Analysis plug-in
IgnoreQueryDesignerVersionCheck=true or false (default value)	<p>You use this setting to define if the designer version on the machine should be checked.</p> <p>The default value for this setting is <code>False</code>. This means that the designer version will be checked. The Setting <i>Show 'Launch Designer' in Tools Group</i> in the user settings is only selectable if the correct designer version is installed.</p> <p>If you change the parameter value to <code>True</code>, the designer version will not be checked and the setting <i>Show 'Launch Designer' in Tools Group</i> is always selectable.</p>	Analysis plug-in
InputReadyCellsValueHelpMemberAccessMode=P	<p>You use this setting to specify the member access mode (value help) for input-ready cells.</p> <p>The default value is <code>P</code> (Planning).</p> <p>For more information on the existing modes, see <a href="#">2180059</a> .</p>	Analysis plug-in
InsertCopiedProfilingResultsToNewSheet	Coming soon....	Analysis add-in (COF)

Setting and Setting Value	Setting Description	Available for
<code>IsCachingDocuments=true</code> (default value) or <code>false</code>	<p>You use this setting to specify whether caching should be enabled.</p> <p>After installation, the default value is <code>True</code>. This means that caching is active.</p> <p>If you change the parameter value to <code>False</code>, the function is disabled and caching cannot be used.</p>	Analysis plug-in
<code>KeepAliveTimeoutInSeconds=60</code> (default value)	<p>You use this setting to set the keep-alive value. You set the value to the time interval (in seconds) between each request, so that the session stays alive. We recommend that you enter a value below the value of the server timeout. For example, if the server timeout is set to 1200 seconds, enter the value 900 in this setting.</p> <p>The default value is 60. This means that the time interval between each request is 60 seconds.</p>	BPC plug-in
<code>LastContinuousUsageDataUpload=01.01.0001</code> (default value)	<p>This setting is used for Usage Tracking.</p> <p>After installation, the default value is <code>01.01.0001</code>. We recommend not changing the value.</p> <p>You can deactivate Usage Tracking for Analysis with setting <code>TrackUsage</code>.</p>	Analysis plug-in
<code>LastPeriodicUsageDataUpload=200000</code> (default value)	<p>This setting is used for Usage Tracking.</p> <p>After installation, the default value is 200000. We recommend not changing the value.</p> <p>You can deactivate Usage Tracking for Analysis with setting <code>TrackUsage</code>.</p>	Analysis plug-in
<code>LastUsedBpfConnections</code>	<p>This setting lists all the latest connections you have used. This setting is filled automatically and you should not change its value.</p>	BPC plug-in
<code>LinesOfData=12</code> (default value)	<p>You use this setting to define the default number of rows.</p> <p>After installation, the default value is 12.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
LogType= 0 (default value), 1 or 2	<p>You use this setting to define whether a log should be created during conversion. The default value for this parameter is 0. This means that no log will be created.</p> <p>If you set the parameter value to 1, a log is created and displayed on a workbook sheet. If you set the value to 2, a log is created and stored on a hidden workbook sheet.</p>	Analysis plug-in
MaxNumberOfParallelThreads = 10 (default value) or any integer value	<p>You use this setting to define the maximum number of parallel threads that Analysis can use to open the SAP HANA data sources of a workbook.</p> <p>The default value is 10. This means that up to 10 SAP HANA data sources can be opened with parallel threads. You can enter any integer value for this setting.</p> <p>If you set the value to 1 or lower, no parallel threads will be used. The data sources will be opened sequentially.</p>	Analysis plug-in
MergeVariables= true or false (default value)	<p>You use this setting to specify whether or not the <i>Merge Variables</i> check box in the <i>Components</i> tab in the design panel is selected when you create a new workbook</p> <p>After installation, the default value is <code>False</code>. This means that the check box is not selected when you create a new workbook.</p> <p>You can change this manually by selecting the check box for single workbooks or setting the parameter value to <code>True</code>. The check box is then always selected when you create a new workbook.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
<p><code>MessagePopupSeverity= ""</code> (default value), <code>Error</code>, <code>Warning</code> or <code>Success</code></p>	<p>You use this setting to specify whether messages should be displayed in a pop-in or in a dialog box, depending on the message severity.</p> <p><code>Critical</code> is the most severe category, <code>Success</code> is the least severe. The severity <code>Success</code> corresponds to severity <code>Information</code> in Analysis.</p> <p>The default value is <code>""</code>. This means that after the installation no value is defined and only messages with severity <code>Critical</code> are displayed in a dialog box. Messages with lower severities (<code>Error</code>, <code>Warning</code> or <code>Success</code>) are displayed in a pop-in dialog.</p> <p>You can change the parameter value to <code>Error</code>, <code>Warning</code> or <code>Success</code>. If you select a severity, all messages are displayed which have this severity or higher. If you enter <code>Warning</code>, for example, all messages with severity <code>Warning</code>, <code>Error</code> and <code>Critical</code> are displayed in a dialog box. Messages with severity <code>Success</code> are displayed in a pop-in dialog.</p>	Analysis plug-in
<p><code>MinorVersion= &lt;integer value of the installed version&gt;</code> (default value) or any integer value lower than the installed version</p>	<p>You use this setting to specify the minor version that should be used.</p> <p>The default value is the integer value of the installed version, for example value 3 for version 2.3.</p> <p>If you change the value to 1, version 2.1 will be used.</p> <p>This setting can only be maintained by an administrator in the file system under <code>C:\ProgramData\SAP\Cof</code>.</p>	Analysis add-in (COF)

Setting and Setting Value	Setting Description	Available for
NcoTraceLevel= 0 (default value), 1, 2, 3 or 4	<p>This setting can be used for SAP error handling. Analysis uses the .Net connector (NCO) for calling ABAP RFCs from client. NCO supports logging of RFC traces.</p> <p>You use this setting to specify the desired level for tracing.</p> <p>The default value is 0. This means no tracing takes place.</p> <p>You can change the parameter value to 1, 2, 3 or 4.</p> <p>If you now work with Analysis, log files will be created according to the selected level in the %temp% folder of windows. There you can find a dev_nco_rfc.log file and a number of files "nco_rfc_XXXX_Y.trc". Additionally, there is the Analysis log file "SAPAdvancedAnalysisXLS.log". You can zip all of them to attach them to the message.</p>	Analysis add-in (COF)
NewLineCheckMode= NONE (default value), MEMBERS or LINES	<p>You use this setting to specify whether a check for new or changed data in new lines is done.</p> <p>The default value is NONE. This means that no automatic check is executed.</p> <p>You can enter MEMBERS to run a check whenever a member was changed or added in new lines.</p> <p>Or you use the value LINES to run an automatic check whenever a line or a cell in a line was changed, and the derivation is done automatically. The derivations can only work if the characteristic relationship is modeled in the BW system.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
NoSystemMessages=true or false (default value)	<p>You use this setting to specify whether back-end system messages should be displayed or not.</p> <p>The default value for this setting is <code>false</code>. This means that back-end system messages will be displayed.</p> <p>If you change the value to <code>true</code>, back-end system messages will not be displayed.</p>	Analysis plug-in
NrOfSingleCellsInContext= greater or equal 1, default value = 100	<p>This setting defines the maximum number of crosstab cells that can be selected to execute analysis options, for example filtering. If more cells are selected in a crosstab, the analysis options are disabled.</p> <p>The default value is 100.</p>	Analysis plug-in
NumberOfNewLines= 5 (default value)	<p>You use this setting to define the default number of new lines.</p> <p>The default value is 5. This means that 5 new lines will be added to the crosstab.</p> <p>You can enter any integer value for this setting.</p>	Analysis plug-in
NwbcticketIssuerPath	Coming soon....	Analysis add-in (COF)
OpenDSTreeView=true or false (default value)	<p>You want to insert a new data source in Analysis and open the Select Data Source dialog. The selectable data sources are displayed in a list. This setting can be used to define the view for the selectable data sources. The options are list view and tree view.</p> <p>The default value is <code>false</code>. This means that the data sources are displayed in the list view.</p> <p>We recommend not changing the value.</p> <p>You can switch to the tree view in the Select Data Source dialog at any time.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
OperatorContainsPattern= true or false (default)	<p>You use this setting to define whether the operators Contains Pattern (CP) and Excludes Pattern (!CP) should be enabled for prompting.</p> <p>The default value is <code>False</code>. This means that the contains pattern operators cannot be used for prompting.</p> <p>To enable the contains pattern operators for prompting, you can set the parameter value to <code>True</code>.</p>	Analysis plug-in
PasteMemberParsingOptions= All (default value) or SingleMembers	<p>You use this setting to specify how signs like the minus sign (-) are interpreted when using the options 'Paste from Clipboard' and 'Paste from File' in the filter by member dialog.</p> <p>The default value is <code>All</code>. This means that values with a minus sign, for example 'abc-xyz', are interpreted as ranges.</p> <p>If you change the parameter value to <code>SingleMember</code>, the example 'abc-xyz' will be interpreted as one single member with the name 'abc-xyz'. If you want to define a range with this parameter value, you can use spaces: abc - xyz.</p> <p>For more information about using signs, see SAP Note <a href="#">2422924</a>.</p>	Analysis plug-in
PreferLeavesOverNodesInInputString= true (default value) or false	<p>You use this setting to define whether an input string should select a leaf or node if both have the same string.</p> <p>The default value is <code>True</code>. This means that the leaf will be selected.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
PreferredDocumentStorage= 0, 1 or 2 (default value)	<p>You use this setting to define the preferred document storage for comments in Analysis.</p> <p>After installation of Analysis, the default value of this parameter is set to 2. This means that the preferred document storage is SAP Business Warehouse.</p> <p>If you set the parameter value to 1, the SAP BusinessObjects Business Intelligence Platform is used as document storage for comments. If you set the value to 0, comments are not stored on platform.</p>	Analysis plug-in
PreferredPlatform= 0 (default value), 1, 2 or 3	<p>You use this setting to define the preferred platform for your Analysis installation. After installation of Analysis, the default value of this parameter is set to 0. This means that all platforms are enabled.</p> <p>If you set the parameter value to 1, the SAP BusinessObjects Business Intelligence Platform is enabled. If you set the value to 2, SAP BW is enabled as platform.</p> <p>With parameter value 3, the SAP BusinessObjects Business Intelligence Platform with compatibility mode is enabled. This means that the workbook is saved as Analysis Workbook (Compatibility Mode). The Analysis Workbook (Compatibility Mode) object corresponds to the Microsoft Excel object used with former BI platform releases.</p>	Analysis plug-in
Profiling= true or false (default value)	<p>You use this setting to activate the query runtime statistics of SAP BW.</p> <p>For more information, see .</p> <p>You can also activate the setting in the Support Settings dialog with setting <a href="#">Enable Workbook Profiling</a>.</p> <p>For more information, see <a href="#">Troubleshooting in Analysis [page 307]</a>.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
PromptForCertificate=true or false (default value)	<p>You use this setting to specify whether the certificate should be searched for automatically or entered manually.</p> <p>The default value for this setting is <code>False</code>. This means that the certificate will be searched for automatically. If no certificate can be found, you have to enter a user ID and a password.</p> <p>If you change the parameter value to <code>True</code>, you have to enter the certificate manually.</p>	Analysis plug-in
PromptWhenInsertingDataSource=true (default value) or false	<p>You use this setting to define the behavior of the prompting dialog box when inserting a new data source.</p> <p>After installation, the default value is <code>True</code>. This means that the prompting dialog box always appears automatically when you insert a data source containing variables.</p> <p>If you set this parameter to <code>False</code>, the prompts dialog only appears when the data source contains mandatory variables.</p>	Analysis plug-in
PropagateSelectionInStructures=true or false (default value)	<p>You use this setting to define the behavior of hierarchical structures in the filter dialog box.</p> <p>In characteristic hierarchies the selection of a node leads to the selection of all its children and vice versa.</p> <p>For hierarchical structures, selection of a node is independent to the selection of its children in the filter dialog box.</p> <p>This is the behavior for the default value <code>False</code>.</p> <p>If you change the value to <code>True</code>, hierarchical structures will behave like characteristic hierarchies in the filter dialog box.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
RecentListSize= greater or equal 1, default value = 10	<p>This setting defines the number of entries in the list of last opened data sources in the <i>Insert Data Source</i> dialog box.</p> <p>You can also define this number in the <i>User Settings</i> in the Settings dialog box. By pressing the <i>Delete Recently Used List</i> button, you can delete the history of the recently used data sources.</p>	Analysis plug-in
RefreshType= 0 (default value), 1 or 2	<p>You use this setting to define whether the workbook should be refreshed after conversion. The default value of this parameter is set to 0. This means that the workbook is always refreshed.</p> <p>If you set the parameter value to 1, the workbook is not refreshed. If you set the value to 2, the workbook is refreshed after conversion if the corresponding properties are selected on the components tab in the design panel.</p>	Analysis plug-in
RemoveDataBeforeSaving=true or false (default value)	<p>You use this setting to define whether the check box <i>Remove Data Before Saving</i> on the Components tab in the design panel is selected as default for new workbooks.</p> <p>If the check box is selected, it is saved without data. When you reopen the workbook, no data is displayed. To display the data, you have to refresh the data sources manually by choosing <i>Refresh All</i> in the menu.</p> <p>The default value is <code>False</code>. This means that the check box on the Components tab is not selected as default.</p> <p>If you change the parameter value to <code>True</code>, the check box is selected as default when you open a new workbook.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
<p><code>ResultSetSizeLimit = -1, n</code> or empty (default)</p>	<p>This setting defines the maximum number of crosstab cells that are loaded from the server for one data source. If a data source contains data for more cells than defined here, a message displays.</p> <p>The standard value for this setting is empty and the maximum number of cells is 500000.</p> <p>If you set the parameter to a specific number greater than or equal to 0, you define the maximum number of cells with this value.</p> <p>If you set the parameter to -1, the setting uses the values defined in the BW system. In a BW system, the parameter is set in the RSADMIN table for object BICS_DA_RESULT_SET_LIMIT_MAX.</p>	Analysis plug-in
<p><code>RetrieveMultilingualTexts = true</code> (default value) or <code>false</code></p>	<p>On the BI platform, the name and description of documents and folders can be translated. You use this setting to specify if the translated texts should be available in Analysis.</p> <p>The default value is <code>True</code>. This means that the translated names are available in Analysis. Depending on the selected language, you see the original version or a translated version.</p> <p>If you set the value to <code>False</code>, only the original version is available in Analysis.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
ReverseVariableProcessingOrderForHanaFromOldXml	<p>This setting is relevant for workbooks saved with Analysis 2.7 SPO or lower.</p> <p>You use this setting to specify whether the variable order will be reversed during the initial processing for all workbooks that are saved with an Analysis version 2.7 SPO or lower.</p> <p>The default value is <code>False</code>. This means that the variable order will not be reversed. This might have a negative effect on performance.</p> <p>If you change the parameter value to <code>True</code>, the variable order will be reversed. Independent of the setting value when saving the workbook, it will be serialized in the correct order and with a new document version. The new document version will ensure that the variable order will not be reversed again when you open the workbook the next time.</p>	Analysis plug-in
RfcBundling= true (default value) or false	<p>This setting is recommended when working in a WAN environment to reduce network traffic.</p> <p>After installation, the default value is <code>True</code>.</p>	Analysis plug-in
RGBValueColumns= #808080 (default value)	<p>You use this setting to define the color for the columns that represent the start and end values.</p> <p>The default value is <code>#808080</code>. This means that the columns display in grey.</p>	Analysis plug-in
RGBValueNegative= #FF000C (default value)	<p>You use this setting to define the color for the negative delta values.</p> <p>The default value is <code>#FF000C</code>. This means that the negative values display in red.</p>	Analysis plug-in
RGBValuePositive= #90CE00 (default value)	<p>You use this setting to define the color for the positive delta values.</p> <p>The default value is <code>#90CE00</code>. This means that the positive values display in green.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
SacHttpSessionPingInterval = 180 (default value) or any integer value.	<p>You use this setting to specify the time Analysis is waiting to send a regular HTTP ping request to SAP Analytics Cloud to keep the session alive.</p> <p>The default value is 180. This means that Analysis is sending the request every 180 seconds.</p> <p>You can enter any integer value for this setting to define the time for the next ping request.</p>	Analysis plug-in
SapWebGuiPath	Coming soon....	Analysis plug-in
SaveAs1xByDefault=true or false (default value)	<p>You use this setting to define whether a workbook should be saved by default with 1.x format in Analysis 2.x versions.</p> <p>The default value is <code>False</code>. This means that the checkbox <i>Save as 1.x format</i> is not selected by default in the saving dialog in Analysis.</p> <p>If you change the parameter value to <code>True</code>, the checkbox <i>Save as 1.x format</i> is selected by default in the saving dialog in Analysis.</p>	Analysis plug-in
SearchFetchMemberLimit	<p>You use this setting to define the maximum number of members displayed in the <i>Filter by Member</i> dialog as result for a search.</p> <p>The default value is 10000. This means that up to 10000 members will be displayed. You can enter any integer value for this setting.</p> <p>If the result contains more members than defined here, you get an information in the dialog.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
SetEmptiedDoubleDataCellsToValue0= true (default value) or false	<p><b>i Note</b></p> <p>This setting is deprecated and is planned to be removed with next minor Analysis release &gt; 2.5 as <code>True</code> seems to be the only relevant value.</p> <p>Please contact the Analysis team, if you need this setting in future.</p> <p>You use this setting to specify whether empty planning data cells are saved as 0(zero) or with their old value.</p> <p>The default value is <code>True</code>. This means that empty planning data cells are saved as 0.</p> <p>If you change the value to <code>False</code>, the empty planning data cells are not saved as 0. The old value remains.</p>	Analysis plug-in
SetFilterComponentApplyToAllDataSources= true(default value) or false	<p>In Analysis, you can insert a filter component using the ribbon. If your workbook contains more than one data source, the <i>Select Data Source</i> dialog box appears where you can define the data sources for the filter.</p> <p>You use this setting to specify if the check box <i>Apply filter to all data sources</i> should be selected by default.</p> <p>After installation, the default value is <code>True</code>. This means that the filter is applied to all data sources.</p> <p>If you change the parameter value to <code>False</code>, the filter will only be applied to the selected data source.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
ShowAllHierarchies= true(default value) or false	<p>You use this setting to specify whether all time-dependent hierarchies for a dimension should be available in the design panel.</p> <p>After installation, the default value is <code>True</code>. This means that all hierarchies are available in the design panel.</p> <p>If you change the parameter value to <code>False</code>, the hierarchies are no longer available in the design panel.</p>	Analysis plug-in
ShowBracketsForVirtualEntriesOfNonCumulativeKeyfigures= true (default value) or false	<p>You use this setting to specify whether Analysis shows brackets for non-cumulative measures that don't have a booked value in the corresponding time interval.</p> <p>In SAP BW brackets for non-cumulative key figures are always shown.</p> <p>After installation, the default value is <code>True</code>. This means that Analysis shows brackets for non-cumulative measures.</p> <p>If you change the parameter value to <code>False</code>, Analysis will not show brackets for non-cumulative measures.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
<code>ShowConvertToFormulaInToolsGroup= true</code> (default value) or <code>false</code>	<p>This setting is used in Analysis with minor versions <math>\leq 3</math> to specify whether the <i>Convert To Formula</i> icon should be displayed in the ribbon.</p> <p>As of release 2.4, you customize the ribbon with the <i>Customize User Interface</i> dialog. Therefore, we recommend to use the <i>Customize User Interface</i> dialog to define the options displayed in the ribbon.</p> <div data-bbox="603 741 986 1144" style="background-color: #f0f0f0; padding: 10px; border-left: 2px solid #0070c0;"> <p><b>i Note</b></p> <p>When switching to a version greater than 3, for example Analysis 2.4, Analysis takes your definition from this setting. If the parameter value for this setting was <code>True</code>, the <i>Convert To Formula</i> icon is displayed in the standard profile for the ribbon. If it was <code>False</code>, it is not displayed.</p> </div>	Analysis plug-in
<code>ShowCreateWebApplicationInToolsGroup= true</code> or <code>false</code> (default value)	<p>This setting is used in Analysis with minor versions <math>\leq 3</math> to specify whether the <i>Create Web Application</i> icon should be displayed in the ribbon.</p> <p>As of release 2.4, you customize the ribbon with the <i>Customize User Interface</i> dialog. Therefore, we recommend to use the <i>Customize User Interface</i> dialog to define the options displayed in the ribbon.</p> <div data-bbox="603 1547 986 1951" style="background-color: #f0f0f0; padding: 10px; border-left: 2px solid #0070c0;"> <p><b>i Note</b></p> <p>When switching to a version greater than 3, for example Analysis 2.4, Analysis takes your definition from this setting. If the parameter value for this setting was <code>True</code>, the <i>Create Web Application</i> icon is displayed in the standard profile for the ribbon. If it was <code>False</code>, it is not displayed.</p> </div>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
ShowCutCopyPasteInGridContextMenu	<p>In Microsoft Excel, the context menu for each cell contains entries to cut, copy and paste cell content. You can use this setting to specify whether these entries should be available for Analysis cross-tab cells.</p> <p>The default value is <code>false</code>. This means that the context menu on Analysis crosstab cells does not contain the Microsoft Excel Cut, Copy and Paste options.</p> <p>If you change the parameter value to <code>True</code>, the Cut, Copy and Paste options are available in the context menu for Analysis crosstab cells.</p>	Analysis plug-in
ShowDSVariantsForWorkbooksWithOneDS= <code>true</code> or <code>false</code> (default value)	<p>You use this setting to specify the mode for the prompting dialog when you open a document from the BW server that contains exactly one data source.</p> <p>The default value is <code>False</code>. This means that the prompting dialog is opened in document mode.</p> <p>If you change the parameter value to <code>True</code>, the prompting dialog is opened in data source mode..</p>	Analysis plug-in
ShowGroupedDimensions= <code>true</code> or <code>false</code> (default value)	<p>You use this setting to define whether dimensions are displayed grouped or as a flat list on the Analysis tab in the design panel.</p> <p>The default value is <code>False</code>. This means that the dimensions are displayed as a flat list. You can change the view in the ribbon. The selected view remains in the design panel until you deselect it again, even if you close Excel meanwhile.</p> <p>If you change the parameter value to <code>True</code>, the dimensions are displayed in groups by default.</p> <p>Note that the dimension grouping is done in the corresponding backend system, SAP BW or SAP HANA.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
ShowNewLinesOnTop= true or false (default value)	<p>You use this setting to specify whether the new lines should be added to the bottom or to the top of the crosstab.</p> <p>The default value is <code>False</code>. This means that new lines are added to the bottom of the crosstab.</p> <p>If you change the value to <code>True</code>, the new lines are added to the top of the crosstab.</p>	Analysis plug-in
ShowPlanningToolbar= true or false (default value)	<p>This setting is used in Analysis with minor versions <math>\leq 3</math> to specify whether the planning group should be displayed in the ribbon.</p> <p>As of release 2.4, you customize the ribbon with the <a href="#">Customize User Interface</a> dialog. Therefore, we recommend to use the <a href="#">Customize User Interface</a> dialog to define the options displayed in the ribbon.</p>	Analysis plug-in

**i Note**

When switching to a version greater than 3, for example Analysis 2.4, Analysis takes your definition from this setting. If the parameter value for this setting was `True`, the planning group is displayed in the standard profile for the ribbon. If it was `False`, it is not displayed.

Setting and Setting Value	Setting Description	Available for
ShowProperties=true or false (default value)	<p>You use this setting to define whether the <i>Property View</i> in the design panel is displayed by default.</p> <p>The default value is <code>False</code>. This means that the Property View is not displayed when you open the design panel. You can open the property view in the ribbon. The property view remains in the design panel until you deselect it again, even if you close Excel.</p> <p>If you change the parameter value to <code>True</code>, the Property View is displayed in the design panel by default.</p>	Analysis plug-in
ShowSaveDialog=true or false (default value)	<p>You use this setting to specify whether the save dialog box should be displayed after a workbook conversion.</p> <p>The default value is <code>False</code>, meaning that the save dialog box will not display after conversion.</p> <p>If you change the parameter value to <code>True</code>, the save dialog box will be displayed after conversion.</p>	Analysis plug-in
ShowSsoLogonDialog=true or false (default value)	<p>You use this setting to specify whether the <i>Logon</i> dialog box should be displayed when using SSO with the SAP BW platform.</p> <p>The default value is <code>False</code>. This means that the logon dialog box is not displayed.</p> <p>If you change the parameter value to <code>True</code>, the logon dialog box is displayed and the user can change the client and the logon language.</p> <p>You can also enable this setting in the Advanced Settings dialog in Analysis.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
ShowSsoLogonDialogBip= true or false (default value)	<p>You use this setting to specify whether the <i>Logon</i> dialog box should be displayed when using SSO with the BI platform.</p> <p>The default value is <code>False</code>. This means that the logon dialog box is not displayed.</p> <p>If you change the parameter value to <code>True</code>, the logon dialog box is displayed and the user can select one of the available BI platforms and change the logon language.</p> <p>You can also enable this setting in the Advanced Settings dialog in Analysis.</p>	Analysis plug-in
ShowSuppressedMessages= true or false (default value)	<p>You use this setting to specify whether messages that are suppressed with API method <i>SAPSuppressMessage</i> should be displayed.</p> <p>The default value is <code>False</code>. This means that the suppressed messages are not displayed.</p> <p>If you change the parameter value to <code>True</code>, the messages that are suppressed with API method <i>SAPSuppressMessage</i> will be displayed.</p>	Analysis plug-in
ShowUtcTimeStampsInDataCells= true or false (default value)	<p>You use this setting to specify whether the UTC timestamps or localized timestamps are displayed in the data cells.</p> <p>The default value is <code>False</code>. This means that localized timestamps are displayed in the data cells.</p> <p>If you change the value to <code>True</code>, the UTC timestamps are displayed.</p>	Analysis plug-in
SidePanelWidth	Coming soon....	BPC plug-in

Setting and Setting Value	Setting Description	Available for
<p><code>SkipPreferringSystemIpAddressOverHostName=true</code> or <code>false</code> (default value)</p>	<p>You use this setting to specify whether the launcher file for the BEx Query Designer is generated using the IP address of the relevant BW system or its host name.</p> <p>The default value is <code>False</code>. This means that the IP address is used for generating the launcher file. When the IP address of the BW system is not accessible from the client where Analysis is installed, you might run into an RFC error.</p> <p>If you change the parameter value to <code>True</code>, the launcher file for the BEx Query Designer will be generated using the host name of the relevant BW system.</p>	Analysis plug-in
<p><code>StylesCountPerformanceThreshold= 2000</code> (default value) or any integer value.</p>	<p>The number of cell styles used in a workbook can decrease the performance in Analysis. You use this setting to specify the number of styles that can be used in workbook before a warning message is displayed.</p> <p>The default value is <code>2000</code>. This means that a warning message will appear if more than 2000 cell styles are used in a workbook.</p> <p>You can enter any integer value for this setting. With value <code>0</code>, the warning is disabled and no message will be displayed.</p>	Analysis plug-in
<p><code>SubsequentRefreshDefault=TransactionDataOnly</code> (default value) or <code>LogOffReconnect</code></p>	<p>You use this setting to specify the default behavior for executing <a href="#">Refresh All</a>.</p> <p>After the installation, the default value is <code>TransactionDataOnly</code>. This means that for all data sources that are online in a workbook, the transactional data is updated when <a href="#">Refresh All</a> is executed.</p> <p>If you change the parameter value to <code>LogOffReconnect</code>, executing <a href="#">Refresh All</a> logs off and reconnects all data sources that are online.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
SupportAutomatedOffice= true or false (default value)	<p>You use this setting to specify whether the Analysis Plug-in should be supported if the Microsoft Office tools are running in embedded mode (also called automated mode).</p> <p>The default value is <code>False</code>. This means that the Analysis Plug-in is not supported.</p> <p>If you change the parameter value to <code>True</code>, the Analysis Plug-in will be supported.</p> <p>This setting is only evaluated if you set <code>SupportEmbeddedMode</code> in the <code>Cof_app.config</code> file to <code>True</code>.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
SupportEmbeddedMode= True or False (default value)	<p>You use this setting to specify whether the Analysis Add-in should be supported when the Microsoft Office tools are running in embedded mode (also called automated mode).</p> <p>The default value is <code>False</code>. This means that the Analysis Add-in is not supported.</p> <p>If you change the parameter value to <code>True</code>, the Analysis Add-in will be supported.</p> <p>If you set the setting to <code>True</code>, you can disable the Analysis Plug-in using the setting <code>SupportAutomatedOfficein</code> in the <code>Ao_app.config</code> file.</p> <div style="background-color: #f0f0f0; padding: 10px; border-left: 2px solid #0070c0;"> <p><b>i Note</b></p> <p>Microsoft does not support Add-ins if the MS Office tool is running in embedded mode. This is the case if the tool is embedded into a hosting window, for example in another MS Office tool or ABAP GUI, or when the MS Office tool is started with <code>excel.exe -Embedding</code> by Windows.</p> <p>We do not recommend changing this setting because in many scenarios, for example if the tool is embedded in a hosting window, some issues can occur which cannot be resolved.</p> </div>	Analysis add-in (COF)
SupportsSaveAs1x=true or false (default value)	<p>You use this setting to define whether a workbook could be saved with 1.x format in Analysis 2.x versions.</p> <p>The default value is <code>False</code>. This means that it is not possible to save a workbook with 1.x format.</p> <p>If you change the parameter value to <code>True</code>, the checkbox <i>Save as 1.x format</i> is available in the saving dialog in Analysis.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
<code>SynchronizeRuleEditorWithCrosstab=true</code> (default value) or <code>false</code>	<p>You use this setting to specify whether a table design rule is highlighted in the design panel when you select the cell or line in the crosstab containing the rule.</p> <p>The default value is <code>True</code>. This means that the rule is highlighted in the design panel.</p> <p>If you change the parameter value to <code>False</code>, the rules will not be highlighted in the design panel.</p>	Analysis plug-in
<code>TaskPaneDockPosition=1</code> (default value)	<p>You use this setting to define where the design panel should be inserted.</p> <p>The default value is 1. This means that the design panel is inserted on the right.</p> <p>You can change the parameter to 2 to insert it on the left, to 3 to insert it on the top, or to 4 to insert it at the bottom.</p> <p>If you change the parameter to 0, the design panel is free-floating.</p>	Analysis plug-in
<code>TaskPaneHeight=975</code> (default value)	<p>You use this setting to define the height of the design panel. The height is only relevant if the design panel is inserted at the top or bottom.</p> <p>The default value is 975 points.</p>	Analysis plug-in
<code>TaskPaneWidth=498</code> (default value)	<p>You use this setting to define the width of the design panel. The width is only relevant if the design panel is inserted on the left or right.</p> <p>The default value is 498 points.</p>	Analysis plug-in
<code>TextKeyDisplay=Default</code> (default value), <code>TextKey</code> , <code>KeyText</code> , <code>Key</code> or <code>Text</code>	<p>You use this setting to define the member display in the design panel.</p> <p>The default value is <code>Default</code>. This means that the selection made in the query designer defines the member display.</p> <p>You can change the parameter value to one of the listed values, for example <code>Key</code>.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
<code>TextKeyDisplayForPrompts</code>	Coming soon....	Analysis plug-in
<code>TrackUsage=true</code> (default value) or <code>false</code>	<p>You use this setting to define whether Usage Tracking for Analysis should be enabled.</p> <p>After installation, the default value is <code>True</code>. This means that Usage Tracking is active.</p> <p>If you change the parameter value to <code>False</code>, Usage Tracking for Analysis is deactivated.</p>	Analysis plug-in
<code>TransferFilterValuesWhileAssigningHierarchy=true</code> or <code>false</code> (default value)	<p>You use this setting to define whether filter values of a dimension are transferred when a different hierarchy or a flat presentation is assigned to the dimension.</p> <p>The default value is <code>False</code>. This means that filter values will not be transferred.</p> <p>If you change the parameter value to <code>True</code>, the filter values will be transferred as far as possible when flat presentation or a different hierarchy is assigned to a dimension.</p>	Analysis plug-in
<code>TwoFactorAuthenticationCertificate = &lt;certificate&gt;</code>	After the first usage of the two-factor authentication, you can see here the selected certificate.	Analysis plug-in
<code>TwoFactorAuthenticationDisabled = true</code> or <code>false</code> (default value)	<p>You can use this setting to disable the two factor authentication..</p> <p>The default value is <code>False</code>. This means that the two-factor authentication is enabled as soon as a URL is available in setting <code>TwoFactorAuthenticationUrl</code>.</p> <p>If you set the value to <code>True</code>, the two-factor authentication is disabled and you can log on to another BI platform. You have to enter the BI platform and your credentials in the log on dialog.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
TwoFactorAuthenticationUrl = <URL>	<p>You can enter the URL of the BI platform that should be used for two-factor authentication (single-sign on).</p> <p>If you enter a URL of a BI platform, two-factor authentication is used for logon and no logon dialog is displayed after the first usage. For the first usage, the user has to select a client certificate.</p> <p>If you need to log on to another BI platform, you can disable the two-factor authentication with setting <code>TwoFactorAuthenticationDisabled</code>.</p>	Analysis plug-in
UndoStackSize= 10 (default value)	<p>You use this setting to specify the number of steps that can be undone or redone with the Analysis <i>Undo/Redo</i> function.</p> <p>The default value is 10.</p>	Analysis plug-in
UsageDataPauseDate= 01.01.0001 (default value)	<p>This setting is used for Usage Tracking.</p> <p>After installation, the default value is 01.01.0001. We recommend not changing the value.</p> <p>You can deactivate Usage Tracking for Analysis with setting <code>TrackUsage</code>.</p>	Analysis plug-in
UseAdvancedHierarchySettings	Coming soon....	Analysis plug-in
UseDataSourceDeltaUpdate= true (default value) or false	<p>You use this setting to specify whether only the delta data or the complete data of a data source should be reloaded and updated in Analysis.</p> <p>After installation, the default value is <code>True</code>. This means that only the delta data of the data source is updated in Analysis.</p> <p>If you change the parameter value to <code>False</code>, the complete data of the data source is reloaded to Analysis.</p>	Analysis plug-in
UseDefaultRQueryInsteadOfCube	Coming soon....	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
UseDocumentDescriptionToConstructBIPDocumentFileName= =true or false (default value)	<p>You use this setting to specify whether the description or the technical name of a workbook is used to construct the file name for workbooks saved on the BI platform. The file name is shown in the Microsoft Excel title bar, for example.</p> <p>The default value for this setting is <code>False</code>. This means that the technical name is used to construct the file name.</p> <p>If you change the parameter value to <code>True</code>, the description will be used to construct the file name.</p>	Analysis plug-in
UseDocumentDescriptionToConstructBWDocumentFileName= true (default value) or false	<p>You use this setting to specify whether the description or the technical name of a workbook is used to construct the file name for workbooks saved on an SAP BW platform. The file name is shown in the Microsoft Excel title bar, for example.</p> <p>The default value for this setting is <code>True</code>. This means that the description is used to construct the file name.</p> <p>If you change the parameter value to <code>False</code>, the technical name will be used to construct the file name.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
UseKeyForHierarchySelection	<p>You use this setting to specify whether the selection of hierarchy variables is text-based or key-based in the prompts dialog.</p> <p>The default value is <code>False</code>. This means that selection of hierarchy variables in the dialog is text-based:</p> <ul style="list-style-type: none"> <li>• The texts are placed in the first column, the keys in the second column.</li> <li>• The list of available hierarchies is sorted by text.</li> <li>• The autocompletion of the textbox within the dropdown uses texts.</li> </ul> <p>If you change the parameter value to <code>True</code>, the selection of hierarchy variables in the dialog is key-based:</p> <ul style="list-style-type: none"> <li>• The keys are placed in the first column, the texts in the second column.</li> <li>• The list of available hierarchies is sorted by key.</li> <li>• The autocompletion of the textbox within the dropdown uses keys.</li> </ul>	Analysis plug-in
UseNewLinesLegacyMode= true or false (default value)	<p>You use this setting to specify which mode for entering planning data in new lines is used.</p> <p>After installation, the default value is <code>False</code>. This means that current mode will be used for entering planning data in new lines. This mode is valid for BW systems of type SAP BW/4HANA SP8 or higher and SAP BW 7.50 SP12 or higher. For former BW versions, the legacy mode will be applied automatically.</p> <p>If you change the parameter value to <code>True</code>, the legacy mode will be used for data sources of all BW systems.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
UseQueryVariantsForViews= true or false (default value)	<p>You use this setting to specify whether you see the variants of the view or the variants of the query used for the view in the prompts dialog.</p> <p>The default value is <code>False</code>. This means that you will see the variant of the view in the prompts dialog.</p> <p>If you change the parameter value to <code>True</code>, you will see the variants of the query used for the view the prompts dialog.</p>	Analysis plug-in
UseSapUILandscapeComApi=true or false (default value)	<p>You use this setting to define whether Analysis uses the SAP GUI SAPUI-LandscapeLib to access information about system data or its own parsing logic.</p> <p>The default value is <code>False</code>. This means that Analysis uses its own parsing logic. This makes it independent from the SAP GUI version but could also lead to restricted possibilities if files defined on the server side are used.</p> <p>If you change the parameter value to <code>True</code>, Analysis uses the COM API SAPUI-LandscapeLib if SAP GUI 7.40 SP8 or higher is installed. If the usage of the COM library fails for any reason, Analysis falls back to its own parsing logic.</p>	Analysis plug-in

Setting and Setting Value	Setting Description	Available for
<p>UseSeparateDisplaySettings ForFilterDialog= true or false (default value)</p>	<p>In Analysis, the filter dialog uses the same display settings as the crosstab when you open the filter dialog for the first time. For example, if the members of a dimension are shown as Key and Text in the crosstab and the filter dialog is opened on that dimension, the filter dialog will display the members also with Key and Text. You can change the display settings for the members in the filter dialog and if the dialog is closed and opened again later in the session it will show the last used display settings independent from what is currently set in the crosstab. But if the workbook has been saved and closed and is opened again, the filter dialog will again take the display settings from the crosstab.</p> <p>You use this setting to specify whether the display settings defined in the filter dialog should be saved with the workbook.</p> <p>After installation, the default value is <code>False</code>. This means that the display settings for the filter dialog are not saved with the workbook.</p> <p>If you change the parameter value to <code>True</code>, the display settings for the filter dialog are saved with the workbook.</p>	Analysis plug-in
<p>UseUnicodeCodepageInNco= True or False (default value)</p>	<p>You use this setting to specify whether a password for an ABAP system may contain special characters such as €.</p> <p>The default value is <code>False</code>. This means that special character ares not supported.</p> <p>If you change the parameter value to <code>True</code>, special characters are supported.</p> <p>You should only set the setting to <code>True</code> if all your ABAP systems use UniCode.</p>	Analysis add-in (COF)

## 11.3 Options

In Analysis for Microsoft Excel, you can change user options according to your personal preferences. The *User Options* dialog box is divided into the following tab pages, each providing different settings:

- *User Options*
- *Advanced Options*
- *Platform Options*
- *Conversion Options*

Depending on the selected platform, you might see only a subset of these tab pages or options.

To open the options dialog, select ► *File* ► *Analysis* ► *Customize Analysis* ► *Options* ► in the ribbon tab.

### Related Information

[User Options \[page 301\]](#)

[Advanced Options \[page 302\]](#)

[Platform Options \[page 303\]](#)

[Conversion Options \[page 305\]](#)

### 11.3.1 User Options

On the *User* tab, you can define the following options.

In Analysis for Microsoft PowerPoint, additional options are available in the *Presentation Options* dialog.

### Recently Used List

When you choose *Insert* in the menu to insert a new data source into the workbook, you get a list of recently used data sources for quick selection. To define the number of listed data sources, enter the required number in the *Maximum Number of Entries in Recent List* field.

If you choose *Delete Recently Used List*, the history of recently used data sources is deleted.

### Number of Members Displayed in Filter Dialog

Here, you can enter the maximum number of members displayed in the *Filter by Member* dialog for selection. To define the number of members displayed, enter the required number in the *Maximum Number of Members*

*Displayed in Filter Dialog* field. If you filter on a dimension that contains more members than defined in the user options, you see only the currently selected members, but you can search for all members.

#### **i Note**

You can also define options at workbook level. These workbook properties are part of the *Component* tab in the design panel.

## **Default Workbook Location**

You can set the location of your default workbook. The default workbook is used when you open a data source by choosing **► Open ► Data Source ►**. The data source is opened with the styles, formatting, and the workbook options of the default workbook.

## **Related Information**

[The Components tab \[page 142\]](#)

[To insert a data source into a default workbook \[page 29\]](#)

[Presentation properties \[page 224\]](#)

## **11.3.2 Advanced Options**

On the *Advanced* tab, you can define the following options.

### **Enhance Currency Conversion by Query Definition**

Select this checkbox if you want the *Use Currency Translation from Query Definition* checkbox to be available in the currency translation dialog for measures.

### **Show All Time-Dependent Hierarchies in Analysis Tab**

Select this checkbox if all time-dependent hierarchies for a dimension should be available in the design panel. If this option is not selected, only the hierarchy for a selected period is available in the design panel.

## Transfer Filter Values while Assigning a Hierarchy

Select this checkbox to transfer filter values of a dimension as far as possible when a different hierarchy or a flat presentation is assigned to the dimension.

## Show Access Mode for Member Display

Select this checkbox to enable the access mode for the member display definition in the crosstab and for the 'Filter by Member' dialog. If this option is not selected, the access mode option is not displayed in the menu.

## Allow Client and Language Selection for SSO Logon

Select this checkbox if the *Logon* dialog box should be displayed when using SSO with BW systems (locally or on a SAP BW platform). In this logon dialog box, you can change the client and the logon language.

## Force Logon Dialog for SSO Logon on BIP

Select this checkbox if the *Logon* dialog box should be displayed when using SSO with the BI platform.

The last used BI platform is selected as default. In this logon dialog box, you can select one of the available BI platforms.

## 11.3.3 Platform Options

On the *Platform* tab, you can define the following options.

### Preferred Platform

You can select the following options:

- **Selectable**  
This selection enables all platforms. You can save workbooks to all platforms and open workbooks from there.
- **Business Intelligence Platform**  
This selection enables the SAP BusinessObjects Business Intelligence Platform. You can save workbooks to this platform and open workbooks from this platform only.  
If you use SAP BusinessObjects Business Intelligence platform 4.0, the workbook is saved as a Microsoft Excel object.

If you use SAP BusinessObjects Business Intelligence platform 4.1, the workbook is saved as an Analysis Workbook object.

- Business Warehouse  
This selection comprises SAP BW and SAP BW4/HANA. You can save workbooks to these platforms and open workbooks from these platforms only.

## Comments Storage

You can select one of the following options to define where comments are stored:

- No Storage
- Business Intelligence Platform
- Business Warehouse

## Workbook Caching

Caching allows you to store copies of frequently used documents, Analysis workbooks and presentations, on a local client machine for quick access. In Analysis, caching can be used for documents that are stored on a SAP BW server. The documents will be cached for each server (BW system) independently.

To enable caching, the option [Enable Workbook Caching](#) must be selected. After the installation, the option is selected by default.

In Analysis for Microsoft PowerPoint, the name of the option is [Enable Presentation Caching](#).

The default location for cached documents is the directory sapaocache which is located beneath the users Temp directory. You can change the [Cache Location](#) here.

To delete the cached documents, press [Clear Cache](#).

You can also enable the option and change the folder path with the file system settings `IsCachingDocuments` and `DocumentCacheFolderPath` in the `Ao_app.config` file. For more information, contact your IT administrator.

## Replace System

With this option, you can replace one or multiple systems in a workbook. The [System Replacement...](#) button to open the dialog is enabled if a workbook with a data source is opened but not connected to any of the systems.

For BW and HANA systems, you can replace local connections and connections from the BI platform. If a workbook contains BW and HANA connections, you will find two tabs in the dialog: BW and HANA.

In the dialog, select the system that should be replaced (Current System) and the system that should replace the former system (Replace By System).

To list the BIP connections in the [Replace By System](#) section, select the [Logon to BIP](#) icon and log on to the BI platform.

You can also open the dialog for replacing systems with the context menu on workbook level in the design panel.

## Related Information

[Commenting data cells \[page 198\]](#)

### 11.3.4 Conversion Options

On the *Conversion* tab, you can define the following settings.

#### BEx Workbook Conversion

You can convert workbooks created with SAP BEx Analyzer 3.5 and SAP BEx Analyzer 7.0 to an Analysis workbook. You can specify which BEx objects should be converted and you can also specify whether a conversion log should be created.

Option: *Conversion Settings*

You can select the following settings:

- Convert All  
All objects in the workbook are converted.
- Convert Data Sources and Crosstabs  
Data sources and crosstabs are converted.
- Convert Data Sources Only  
Only the data sources are converted.

Option: *Refresh Workbook*

You can select the following options:

- Always  
The workbook is always refreshed after conversion.
- Never  
The workbook is not refreshed after conversion.
- Use Workbook Settings  
The workbook is refreshed after conversion if the corresponding property is active in BEx Analyzer.

Setting: *Conversion Log*

You can select the following options:

- Disabled  
No conversion log is created.
- Create on New Visible Worksheet

A log is created and displayed on a new worksheet.

- Create on New Hidden Worksheet

A log is created and stored on a hidden sheet in the workbook.

Option: *Show Save Dialog After Conversion*

If you select this option, the Save dialog is displayed after conversion.

## Analysis View Migration

You can use the *Analysis View Migration* dialog to migrate analysis views (which were created with Analysis version 1.x and which contain SAP HANA data sources) to 2.x analysis views.

SAP HANA ODBC connections that were used in previous Analysis releases (1.x) are no longer supported. Therefore, you cannot open an analysis view created with a 1.x version in Analysis 2.x. For Analysis 2.x versions, HTTP connections are used for SAP HANA data sources. With Analysis 2.3 and higher, you can migrate the analysis views.

After creating a SAP HANA HTTP connection, you can open the *Analysis View Migration* dialog and migrate the analysis views:

1. Log on to the BI platform.
2. Select a HANA ODBC connection.
3. Select a HANA HTTP connection.
4. Select the analysis view to be migrated.  
You can migrate one analysis view per migration.
5. Press *Migrate*.  
You can select another analysis view and start the migration again, or press *Close* to close the dialog.

The migrated analysis views can now be opened in Analysis.

### **i** Note

Migrated Analysis Views can no longer be used in Analysis 1.4. Furthermore, they cannot be used with other clients that use JDBC-based OLAP connections, such as Design Studio and Analysis, edition for OLAP.

# 12 Troubleshooting

## 12.1 Troubleshooting in Analysis

You can run Analysis in different troubleshooting modes. After the first Analysis installation, no troubleshooting mode is active.

You can specify the troubleshooting options in the Analysis backstage area: Select **File > Analysis > Troubleshoot > <troubleshooting mode>**.

The following troubleshooting modes are available:

- **Support Mode**  
In the Support Mode, the system stores exceptions, error messages and traces. BW Server Tracing for the Analysis plug-in is also active.
- **Profiling Mode**  
In the Profiling Mode, the client profiling is active. BW Server Tracing and Workbook Profiling for the Analysis plug-in are active. The system also stores exceptions and error messages.
- **Advanced Mode**  
You can specify the options for troubleshooting in the *Advanced Support and Profiling Mode* dialog. These tabs are available in the dialog: *General* (for all plug-ins) and *Analysis* (for the Analysis plug-in). The options on the three tabs are described later in this chapter.

The Support Mode and the Profiling Mode can be active at the same time.

You can check the statistics under **File > Analysis > Troubleshoot > Show Profiling Statistics** or **Show Workbook Profiling Statistics**.

Note that writing profiling statistics may have an influence on the performance. You can select *Deactivate* in the Troubleshoot area to deactivate all troubleshooting modes if they are no longer needed.

### Advanced Mode: General Tab

#### Log Severity

The Log Severity defines the amount of log and trace information that is stored. In this area, you can select:

- **Support**  
By default, the system stores exceptions and error messages. If the check box *Support* is selected, the system stores exceptions, error messages and traces.  
The information is stored as .glf files under `C:\Users\<user>\AppData\Local\Temp\Sap\Cof`.  
The system can also store messages and traces that are defined in the log.config file under `C:\Users\<user>\AppData\Roaming\SAP\Cof`. For more information, contact your system administrator.
- **Profiling**

To activate the Analysis client profiling, select the *Profiling* check box. If the setting is activated, Analysis will capture every interaction.

You can choose between *Standard Profiling* and *Modified Profiling*.

You can also activate the client profiling with the file system setting *clientProfiling*.

In the *Client Profiling Statistics* dialog, the captured interactions are displayed as steps in a tree view and you can navigate to each interaction. You can see the overall processing time per step (in ms), the time the user spent in dialogs, the number of RFC calls per step and the time that was spent for RFC. In section Update UI, you can see if a navigation step has caused (unexpected) RFC calls.

At the bottom of the *Client Profiling Statistics* dialog, you find a summary containing the time that was used on client side for processing (in ms), the number of RFC calls (count) and the time that was used for that (in ms) and the time the user spent in dialogs (in ms).

## Advanced Mode: Analysis Tab

### Enable BW Server Tracing

To activate the SAP BW trace tool environment, select the *Enable BW Server Tracing* check box.

You can also activate the BW server tracing with the file system setting *AbapTrace* in the `Ao_app.config` file.

The trace tool environment (transaction code **RSTT** in the connected BW system) has special tools to log and play back traces and process automatic regression tests.

For more information about the trace tool environment, see the SAP BW documentation on the SAP Help Portal.

### Enable Workbook Profiling

To activate the SAP BW query runtime statistics, select the *Enable Workbook Profiling* check box.

You can also activate the workbook profiling with the file system setting *Profiling* in the `Ao_app.config` file.

Using the query runtime statistics, you can find out how much time it takes to execute certain user actions in Analysis and the BW analytic engine. The system records the performance-critical parts of the processing (statistics events). It calculates the net times by calculating the runtime of an event using the difference between the start and end times (minus the times for other events called from within the event).

For more information about the query runtime statistics, see section *Query runtime statistics* in the Analysis Administrator guide.

### Enable NCO Tracing

This setting may be used for SAP error handling.

Analysis uses the .Net connector (NCO) for calling ABAP RFCs from client. NCO supports logging of RFC traces. You can activate the tracing by selecting *Enable NCO Tracing* and choosing the desired level (usually 4).

If you now work with Analysis, log files will be created in the %temp% folder of windows. There you can find a `dev_nco_rfc.log` file and a number of files "nco\_rfc\_XXXX\_Y.trc". Additionally, there are the Analysis log files of type .glf, for example `AO_Log_<processID>.glf`. You can zip all of them to attach them to the message.

### Enable CPIC Tracing

This setting may be used for SAP error handling. Common Programming Interface - Communication (CPIC) is the communication layer under JRFC (or JCo).

You can activate the tracing by selecting *Enable CPIC Tracing* and choosing the desired level. You can choose a trace level from 1 to 3, where 3 is the highest and most detailed level of tracing.

If you now work with Analysis, log files will be created in the %temp% folder of windows. There you can find a nco\_cplic\_XXX.trc file that you can attach to the message.

### Show Suppressed Messages

Select this check box if you want the suppressed messages to be shown.

## 12.2 To enable the Analysis Add-In after system crash

### Context

If Microsoft Excel or Microsoft PowerPoint crashes, and you have to close the application, the Analysis Add-In might be disabled by the Microsoft application. If this happens, you have to re-enable the Analysis Add-In in Microsoft Excel or Microsoft PowerPoint.

### Procedure

1. Open Analysis for Microsoft Excel or Microsoft PowerPoint.  
After the system crash the Analysis Add-In is not visible in the menu.
2. Choose File in Microsoft Excel or Microsoft PowerPoint.
3. In Microsoft Excel, press *Excel Options*. In Microsoft PowerPoint, press *PowerPoint Options*.
4. In the *Excel Options* dialog box and in the *PowerPoint Options* dialog box in the categories pane, select *Add-Ins*.
5. In the *Manage* box, select *Disabled Items*.
6. Press *Go...*
7. In the *Disabled Items* dialog box, select the Analysis Add-In.
8. Press *Enable*.
9. In the *Manage* box, select *COM Add-Ins*.
10. Press *Go...*
11. In the *COM Add-Ins* dialog box, make sure that *Analysis* option is activated.
12. Press *OK*.

### Results



The *Analysis* ribbon is available again.

# Important Disclaimers and Legal Information

## Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.

About the icons:

- Links with the icon : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon : You are leaving the documentation for that particular SAP product or service and are entering a SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

## Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.

The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

## Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

## Gender-Related Language

We try not to use gender-specific word forms and formulations. As appropriate for context and readability, SAP may use masculine word forms to refer to all genders.

## Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.



© 2019 SAP SE or an SAP affiliate company. All rights reserved.

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

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company 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.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

Please see <https://www.sap.com/about/legal/trademark.html> for additional trademark information and notices.