

SAP BusinessObjects Design Studio  
Document Version: 1.6 SP01 – 2016-02-23

# Application Designer Guide: Designing Analysis Applications



---

# Content

<b>1</b>	<b>About This Guide.</b>	<b>24</b>
1.1	About the Documentation Set.	24
1.2	Who Should Read This Guide?.	25
<b>2</b>	<b>Getting Started.</b>	<b>26</b>
2.1	What is SAP BusinessObjects Design Studio?.	26
2.2	Basic Concepts.	27
2.3	Feature Dependencies on Platform Deployment.	28
2.4	Feature Dependencies on Data Source.	30
2.5	Launching SAP BusinessObjects Design Studio.	31
2.6	Starting with the Welcome Page.	33
2.7	Working with the Design Tool.	33
	Components View.	34
	Outline View.	36
	Properties View.	36
	Error Log View.	37
	Problems View.	37
	Layout Editor.	37
	Maintaining Settings in the Design Tool.	38
	Storage of Applications and Images.	42
<b>3</b>	<b>Creating New Analysis Applications.</b>	<b>44</b>
3.1	Using the SAPUI5 m Library.	45
3.2	Adding Components to an Application.	47
	Specifying the Properties of a Component.	48
	Deleting Components in an Application.	48
	Selecting Multiple Components in the Editor.	49
<b>4</b>	<b>Creating Analysis Applications for Mobile Devices.</b>	<b>50</b>
<b>5</b>	<b>Creating the Layout of an Analysis Application.</b>	<b>51</b>
5.1	Changing the Layout Properties and Docking Behavior.	51
5.2	Using Container Components.	53
<b>6</b>	<b>Working with the Outline View.</b>	<b>54</b>
6.1	Using the Context Menu in the Outline View.	56
6.2	Using the Context Menu (Technical Component).	60
	Calculating New Measures at Runtime.	62

	Creating Filters by Measure. . . . .	65
<b>7</b>	<b>Adding a Data Source. . . . .</b>	<b>68</b>
7.1	Selecting a Connection. . . . .	69
7.2	Selecting a Data Source. . . . .	71
	Creating a Query. . . . .	73
	About the Member Selector. . . . .	74
	Filtering Data in the Query Panel. . . . .	83
7.3	Universe Access in Design Studio. . . . .	88
7.4	Selecting Data from a Data Source for Charts and SDK Extensions. . . . .	90
7.5	Deleting a Data Source. . . . .	92
<b>8</b>	<b>Inserting a Copied Data source from SAP BusinessObjects Analysis, Edition for Microsoft Office. . . . .</b>	<b>93</b>
<b>9</b>	<b>Working with the Initial View Dialog Box for Data Source Aliases. . . . .</b>	<b>94</b>
<b>10</b>	<b>Assigning a Data Source to a Component. . . . .</b>	<b>97</b>
<b>11</b>	<b>Binding the Properties of Standard Components to Data Sources. . . . .</b>	<b>98</b>
11.1	Displaying Cell Values. . . . .	99
11.2	Configuring a List Box for Filtering Data . . . . .	100
11.3	Configuring an SDK Component to Use Multiple Data Sources. . . . .	101
<b>12</b>	<b>Saving an Application . . . . .</b>	<b>102</b>
<b>13</b>	<b>Working with Templates. . . . .</b>	<b>103</b>
13.1	Using the Basic Analysis Layout Template. . . . .	104
13.2	Using the Generic Analysis Ready-to-Run Template. . . . .	106
13.3	Using the Online Composition Template. . . . .	109
13.4	Using the Planning Layout Template. . . . .	111
13.5	Using the Data Discovery and Visualization Template. . . . .	112
<b>14</b>	<b>Executing an Application. . . . .</b>	<b>119</b>
<b>15</b>	<b>Executing an Application on a Mobile Device. . . . .</b>	<b>120</b>
<b>16</b>	<b>Recording Applications for Offline Use. . . . .</b>	<b>121</b>
<b>17</b>	<b>Printing an Analysis Application to a Browser. . . . .</b>	<b>123</b>
<b>18</b>	<b>Exporting to PDF. . . . .</b>	<b>125</b>
18.1	Enabling Export to PDF in Analysis Applications. . . . .	126
18.2	Export to PDF Properties. . . . .	127
18.3	WYSIWYG Export to PDF. . . . .	129
	Scripting WYSIWYG Export to PDF. . . . .	131
18.4	Application Export to PDF . . . . .	132

	Scripting Application Export to PDF Settings. . . . .	133
<b>19</b>	<b>Exporting to SAP Lumira Desktop. . . . .</b>	<b>134</b>
<b>20</b>	<b>Changing an Analysis Application. . . . .</b>	<b>135</b>
20.1	Saving an Application Using a Different Name. . . . .	135
20.2	Maximum Number of Steps Back or Resetting within an Application. . . . .	136
<b>21</b>	<b>Deleting an Analysis Application. . . . .</b>	<b>139</b>
<b>22</b>	<b>Searching in Applications. . . . .</b>	<b>140</b>
<b>23</b>	<b>Assigning Analysis Applications to the Mobile Category. . . . .</b>	<b>141</b>
<b>24</b>	<b>Creating Planning Applications. . . . .</b>	<b>142</b>
24.1	Selecting a Planning Connection. . . . .	143
24.2	Entering Data in the Crosstab (Manual Planning). . . . .	144
24.3	Editing Short Texts in Queries. . . . .	144
24.4	Using Planning Functions and Sequences (Automated Planning). . . . .	145
24.5	Cell Locking. . . . .	146
	Front-End Cell Locks. . . . .	147
	Back-End Cell Locks. . . . .	147
	Locking Cells. . . . .	148
<b>25</b>	<b>Working with Online Composition. . . . .</b>	<b>150</b>
25.1	Creating a Bookmark Based on a Query. . . . .	153
25.2	Working with Fragment Gallery Components. . . . .	153
25.3	Working with Split Cell Containers. . . . .	155
25.4	Applying Global Filters. . . . .	156
	Sample Scripting for Global Filters. . . . .	157
<b>26</b>	<b>Working with Bookmarks. . . . .</b>	<b>159</b>
26.1	Standard Bookmarks. . . . .	160
	Clearing Prompts on Standard Bookmark Load. . . . .	161
26.2	Fragment Bookmarks. . . . .	162
26.3	Portable Fragment Bookmarks. . . . .	163
26.4	Scripting for All Bookmark Types. . . . .	165
	Deleting Bookmarks. . . . .	166
	Listing Bookmarks. . . . .	167
	Loading Bookmarks Using Scripting. . . . .	169
	Loading Shared Bookmarks Using a Url. . . . .	170
	Saving a Bookmark. . . . .	170
	Sharing a Bookmark. . . . .	172
	Assigning Bookmarks to Folders. . . . .	173
26.5	Obsolete Bookmarks. . . . .	175



26.6	Personalization. . . . .	176
<b>27</b>	<b>Working with Info Charts. . . . .</b>	<b>177</b>
27.1	Info Chart Settings. . . . .	177
27.2	Configuring Initial Settings for Info Charts. . . . .	177
27.3	Using the Chart Configuration Dialog. . . . .	178
27.4	Info Chart Types. . . . .	179
<b>28</b>	<b>Working with Charts. . . . .</b>	<b>182</b>
28.1	Chart Settings. . . . .	182
28.2	Configuring Initial Settings for Charts. . . . .	182
	100% Stacked Bar. . . . .	183
	100% Stacked Column. . . . .	185
	Area. . . . .	186
	Bar. . . . .	187
	Bar Combination. . . . .	188
	Bubble. . . . .	190
	Column. . . . .	191
	Column Combination. . . . .	192
	Dual Axis. . . . .	193
	Horizontal Area. . . . .	194
	Horizontal Line. . . . .	195
	Horizontal Waterfall. . . . .	196
	Line. . . . .	198
	Multiple Pie. . . . .	199
	Multiple Radar. . . . .	200
	Pie. . . . .	201
	Radar. . . . .	202
	Scatter. . . . .	203
	Stacked Bar. . . . .	204
	Stacked Column. . . . .	205
	Stacked Waterfall. . . . .	206
	Waterfall. . . . .	209
28.3	Configuring Additional Chart Types. . . . .	211
28.4	Conditional Formatting for Charts. . . . .	212
<b>29</b>	<b>Working with Geo Maps. . . . .</b>	<b>215</b>
29.1	Geo Map Settings. . . . .	216
29.2	Configuring Initial Geo Map Settings. . . . .	216
29.3	Configuring Geo Map Layers. . . . .	217
	Configuring a Shapes Layer. . . . .	218
	Configuring a Points Layer. . . . .	219

	Configuring a Bubble or Pie Chart Layer. . . . .	221
29.4	Interacting with Geo Maps. . . . .	222
<b>30</b>	<b>Working with Drag and Drop in Applications and Crosstabs. . . . .</b>	<b>224</b>
30.1	Dragging a Dimension or Dimension Member. . . . .	225
30.2	Removing a Dimension or Dimension Member. . . . .	228
30.3	Dropping Dimensions or Dimension Members. . . . .	229
30.4	Drag and Drop Constraints. . . . .	232
30.5	Dragging and Dropping Between Crosstab and Navigation Panel. . . . .	233
30.6	Drag and Drop in Crosstabs with Property "Display Repeated Texts". . . . .	234
<b>31</b>	<b>Working with Scorecards. . . . .</b>	<b>236</b>
31.1	Scorecard Configuration. . . . .	237
	Defining the Row Scope. . . . .	238
	Generating the Initial Scorecard. . . . .	238
31.2	Basic Scorecard Concepts . . . . .	239
31.3	Configuring Initial Scorecard Settings. . . . .	239
	Configuring Scorecard Columns. . . . .	239
	Configuring Scorecard Headers and Group Headers. . . . .	240
	Configuring the Scorecard Cell Content. . . . .	240
	Defining the Scorecard Column Content. . . . .	240
	Defining the Textual Content. . . . .	241
	Configuring Trend Charts. . . . .	241
	Configuring Comparison Charts. . . . .	241
31.4	Interacting with Scorecards. . . . .	242
31.5	Binding the Properties of Scorecards to Data Sources . . . . .	242
	Properties with Single Values. . . . .	243
	Properties with Multiple Values. . . . .	243
	Binding Type Settings. . . . .	244
<b>32</b>	<b>Using Header Scrolling in Crosstabs. . . . .</b>	<b>248</b>
32.1	"Horizontal Scrolling Enabled" Set to "False". . . . .	249
32.2	"Horizontal Scrolling Enabled" Set to "True". . . . .	250
	General Strategy if "Maximum Width of Header Area" Set to "Auto". . . . .	250
	Behavior after Manual Changing of Header With. . . . .	254
	Crosstabs With a Dimension Area or a Measure Area Only. . . . .	259
	"Maximum Width of Header Area" Set To A Pixel Value. . . . .	260
<b>33</b>	<b>Working with Real-Time Dashboards. . . . .</b>	<b>262</b>
33.1	How to Create Real-Time Dashboards with Streaming Data Sources. . . . .	263
33.2	How to Create Real-Time Dashboards with a Single Pull-Based Data Source. . . . .	264
33.3	How to Create Real-Time Dashboards with Multiple Pull-Based Data Sources. . . . .	265
33.4	Streaming Data Source Additional Properties. . . . .	267

<b>34</b>	<b>Advanced Design Tasks.</b>	<b>270</b>
34.1	Scripting for User Interaction Enablement.	270
	Using the Script Editor.	277
	Using the Statement Wizard.	281
	Selecting Members of a Dimension.	282
	Script Processing Flow in Applications.	284
	Process Flow at Application Start.	286
	Business Cases.	288
34.2	Working with Global Scripts Objects and Global Script Functions.	297
34.3	Enabling Text Translation in Analysis Applications.	298
	Using Scripts for Translatable Dynamic Texts.	299
	Working With Translatable Texts in Analysis Applications.	300
34.4	Deploying SDK Extensions.	301
	Installing Design Studio SDK Extensions to SAP BusinessObjects Design Studio.	302
	Installing CVOM Chart Extensions for SAP BusinessObjects Design Studio.	303
	Removing Extensions from SAP BusinessObjects Design Studio.	304
34.5	Creating a Generic Analysis Template for RRI Jump Targets.	304
34.6	Exporting Analysis Applications.	305
34.7	Importing Analysis Applications.	307
34.8	Exporting Applications As Templates.	309
34.9	Using the Backend Connection Component for Data Source Browsing.	311
34.10	Using Processing Groups for Parallel Query Execution.	313
34.11	Unmerging Prompts (Variables).	316
<b>35</b>	<b>Troubleshooting.</b>	<b>318</b>
35.1	Auto-Recovering Applications.	318
35.2	Using the Script Problems View.	318
35.3	Working with Large eval() Statements.	319
35.4	Setting Network Connections If Logon Problems Occur.	319
35.5	Activating Runtime Traces.	320
35.6	Activating SAP JCo Traces.	321
35.7	Viewing And Collecting Statistics Data At Runtime.	322
35.8	Collecting Support Information.	323
35.9	Managing Logs in the Design Tool.	323
35.10	Best Practices.	324
	Improving Performance of Analysis Applications.	324
<b>36</b>	<b>User Interface Reference.</b>	<b>325</b>
36.1	Properties of the Application.	325
36.2	Properties of the Data Source Alias.	332
36.3	General Properties for All Components.	335
36.4	Display Properties for All Components.	336

36.5	Layout Properties for All Components. . . . .	336
36.6	Analytic Components. . . . .	338
	Chart. . . . .	338
	Info Chart. . . . .	352
	Crosstab. . . . .	372
	Dimension Filter. . . . .	383
	Filter Line. . . . .	386
	Filter Panel. . . . .	387
	Geo Map. . . . .	390
	Navigation Panel. . . . .	394
	Spreadsheet. . . . .	394
36.7	Basic Components. . . . .	397
	Button. . . . .	397
	Chart Type Picker. . . . .	399
	Info Chart Feeding Panel. . . . .	400
	Checkbox. . . . .	401
	Checkbox Group. . . . .	401
	Date Field. . . . .	402
	Dropdown Box. . . . .	403
	Formatted Text View. . . . .	404
	Fragment Gallery. . . . .	406
	Icon. . . . .	407
	Image. . . . .	409
	Input Field. . . . .	411
	List Box. . . . .	411
	Radio Button Group. . . . .	413
	Text. . . . .	414
	Tree. . . . .	415
36.8	Container Components. . . . .	416
	Grid Layout. . . . .	416
	Pagebook. . . . .	417
	Panel. . . . .	419
	Popup. . . . .	420
	Tabstrip. . . . .	421
	Split Cell. . . . .	421
<b>37</b>	<b>API Reference. . . . .</b>	<b>423</b>
37.1	Action Sheet. . . . .	430
	Add Item (addItem). . . . .	430
	Get Selected Text (getSelectedText). . . . .	430
	Get Selected Value (getSelectedValue). . . . .	431
	Open (open). . . . .	431

	Remove All Items (removeAllItems).	432
	Remove Item (removeItem).	432
37.2	Application (Application).	432
	Alert (alert).	432
	Create Error Message (createErrorMessage).	433
	Create Info Message (createInfoMessage).	433
	Create Warning Message (createWarningMessage).	434
	Do Background Processing (doBackgroundProcessing).	434
	Export (export).	435
	Get Info (getInfo).	435
	Get Resource String (getResourceString).	436
	Get Tick Count (getTickCount).	436
	Get User Agent (getUserAgent).	437
	Load Data Sources (loadDataSources).	437
	Log (log).	437
	Open New Window (openNewWindow).	438
	Open Prompt Dialog Box (openPromptDialog).	439
	Set Variable Value (setVariableValue).	439
	Print (print).	441
	Search Data Sources (searchDataSources).	441
	Set Variable Value Ext (setVariableValueExt).	441
37.3	ApplicationInfo.	443
37.4	Array.	444
	Accessing Array Elements.	444
	For Each (forEach).	445
	Length (length).	445
	Pop (pop).	445
	Push (push).	446
37.5	AttributeMember.	446
37.6	Bookmark.	447
	Assign To Folder (assignToFolder).	447
	Bookmark With Title Exists (bookmarkWithTitleExists).	448
	Delete All Bookmarks (deleteAllBookmarks).	448
	Delete Bookmark (deleteBookmark).	449
	Get All Bookmarks (getAllBookmarks).	450
	Get All Bookmarks By Folder (getAllBookmarksByFolder).	450
	Get Bookmark Folders (getBookmarkFolders).	451
	Get Bookmark Folders Tree Model (getBookmarkFoldersTreeModel).	451
	Get Bookmark Info (getBookmarkInfo).	452
	Get Bookmark Url (getBookmarkUrl).	452
	Load Bookmark (loadBookmark).	453



	Save Bookmark (saveBookmark).	453
	Share Bookmark (shareBookmark).	454
37.7	Fragment Bookmark.	455
	Assign To Folder (assignToFolder).	455
	Delete All Bookmarks (deleteAllBookmarks).	455
	Delete Bookmark (deleteBookmark).	456
	Get All Bookmark Infos (getAllBookmarkInfos).	456
	Get All Bookmarks By Folder (getAllBookmarksByFolder).	457
	Get Bookmark Info (getBookmarkInfo).	458
	Get Bookmark Url (getBookmarkUrl).	458
	Load Bookmark (loadBookmark).	459
	Save Bookmark (saveBookmark).	459
	Share Bookmark (shareBookmark).	460
37.8	Portable Fragment Bookmark.	461
	Assign To Folder (assignToFolder).	461
	Delete All Bookmarks (deleteAllBookmarks).	461
	Delete Bookmark (deleteBookmark).	462
	Get All Bookmark Infos (getAllBookmarkInfos).	462
	Get All Bookmark Infos For Application (getAllBookmarkInfosForApplication).	463
	Get All Bookmarks By Folder (getAllBookmarksByFolder).	464
	Get BookmarkInfo (getBookmarkInfo).	464
	Get Bookmark Url (getBookmarkUrl).	465
	Load Bookmark (loadBookmark).	465
	Save Bookmark (saveBookmark).	466
	Share Bookmark (shareBookmark).	467
37.9	Button.	468
	Get Icon (getIcon).	468
	Get Text (getText).	468
	Get Tooltip (getTooltip).	469
	Is Enabled (isEnabled).	469
	On Click (onClick).	469
	Set Enabled (setEnabled).	470
	Set Icon (setIcon).	470
	Set Text (setText).	471
	Set Tooltip (setTooltip).	471
37.10	Chart.	471
	Get Chart Type (getChartType).	471
	Get Axis Scaling Max (getAxisScalingMax).	472
	Remove Axis Scaling (removeAxisScaling).	472
	Get Axis Scaling Min (getAxisScalingMin).	473
	Set Axis Scaling (setAxisScaling).	473

	Clear Selection (clearSelection).	474
	Get Legend Position (getLegendPosition).	474
	Get Selected Member (getSelectedMember).	474
	Get Selected Members (getSelectedMembers).	475
	Set Chart Type (setChartType).	475
	Get Style (getStyle).	476
	Is Visible (isVisible).	476
	On Select (onSelect).	476
	Set Legend Position (setLegendPosition).	477
	Set Style.	477
	Set Visible (setVisible).	478
	Show Data Labels (showDataLabels).	478
	Show Scaling Factors (showScalingFactors).	479
	Show Totals (showTotals).	479
	Swap Axes (swapAxes).	480
	Set Data Selection (setDataSelection).	480
	Set Data Source (setDataSource).	481
37.11	Chart Type Picker.	481
	Get Selected Value (getSelectedValue).	481
	Set Selected Value (setSelectedValue).	482
37.12	Info Chart.	482
	Clear Selection (clearSelection).	482
	Get Axis Scaling Max (getAxisScalingMax).	483
	Get Axis Scaling Min (getAxisScalingMin).	483
	Get Bottom Margin (getBottomMargin).	484
	Get Chart Type (getChartType).	484
	Get CSS Class (getCSSClass).	484
	Get Data Source (getDataSource).	485
	Get Height (getHeight).	485
	Get Left Margin (getLeftMargin).	485
	Get Legend Position (getLegendPosition).	486
	Get Right Margin (getRightMargin).	486
	Get Selected Member (getSelectedMember).	487
	Get Selected Members (getSelectedMembers).	487
	Get Top Margin (getTopMargin).	487
	Get Width (getWidth).	488
	Hide Loading State (hideLoadingState).	488
	Is Visible (isVisible).	489
	Remove Axis Scaling (removeAxisScaling).	489
	Set Axis Scaling (setAxisScaling).	489
	Set Bottom Margin (setBottomMargin).	490

Set CSS Class (setCSSClass).	490
Set Chart Type (setChartType).	491
Show Data Labels (showDataLabels).	491
Set Data Selection (setDataSelection).	492
Set Data Source (setDataSource).	492
Set Height (setHeight).	493
Set Left Margin (setLeftMargin).	493
Set Legend Position (setLegendPosition).	494
Show Loading State (showLoadingState).	494
Set Right Margin (setRightMargin).	495
Set Top Margin (setTopMargin).	495
Show Totals (showTotals).	496
Set Visible (setVisible).	496
Set Width (setWidth).	497
37.13 Info Chart Feeding Panel.	497
Get Bottom Margin (getBottomMargin).	497
Get CSS Class (getCSSClass).	498
Get Height (getHeight).	498
Get Left Margin (getLeftMargin).	499
Get Right Margin (getRightMargin).	499
Get Top Margin (getTopMargin).	499
Get Width (getWidth).	500
Hide Loading State (hideLoadingState).	500
Is Visible (isVisible).	500
Set Bottom Margin (setBottomMargin).	501
Set CSS Class (setCSSClass).	501
Set Chart Reference (setChartReference).	502
Set Height (setHeight).	502
Set Left Margin (setLeftMargin).	503
Set Right Margin (setRightMargin).	503
Set Top Margin (setTopMargin).	504
Set Visible (setVisible).	504
Set Width (setWidth).	505
Show Loading State (showLoadingState).	505
37.14 Checkbox.	506
Get Text (getText).	506
Get Tooltip (getTooltip).	506
Is Checked (isChecked).	507
Is Enabled (isEnabled).	507
Is Visible (isVisible).	507
On Click (onClick).	508

	Set Checked (setChecked).	508
	Set Enabled (setEnabled).	509
	Set Text (setText).	509
	Set Tooltip (setTooltip).	510
	Set Visible (setVisible).	510
37.15	Checkbox Group.	511
	Add Item (addItem).	511
	Get Selected Texts (getSelectedTexts).	511
	Get Selected Values (getSelectedValues).	512
	Is Enabled (isEnabled).	512
	Remove All Items (removeAllItems).	513
	Remove Item (removeItem).	513
	Set Enabled (setEnabled).	513
	Set Items (setItems).	514
	Set Selected Values (setSelectedValues).	515
	Sort (sort).	515
37.16	Constants.	516
	Axis.	516
	ChartType.	516
	ClickArea.	520
	ContextMenuItemId.	520
	DataSourceBrowseType.	524
	DataSourceConnectionType.	524
	DataSourceExportType.	525
	DataSourceType.	525
	IconBackgroundShape.	526
	InfoChartType.	526
	Layout.	532
	LegendPosition.	532
	MemberDisplay.	532
	MemberPresentation.	533
	NegativeNumberDisplay.	534
	Scaling.	534
	TotalsDisplay.	536
	TotalsPosition.	536
	UnitsAndScalingFactorsDisplay.	537
	ZeroDisplay.	537
37.17	Component.	538
	Get Bottom Margin (getBottomMargin).	538
	Get CSS Class (getCSSClass).	539
	Get Height (getHeight).	539

Get Left Margin (getLeftMargin).	539
Get Right Margin (getRightMargin).	540
Get Top Margin (getTopMargin).	540
Get Width (getWidth).	541
Hide Loading State (hideLoadingState).	541
Is Visible (isVisible).	541
Set Bottom Margin (setBottomMargin).	542
Set CSS Class (setCSSClass).	542
Set Height (setHeight).	543
Set Left Margin (setLeftMargin).	543
Set Right Margin (setRightMargin).	544
Set Top Margin (setTopMargin).	544
Set Visible (setVisible).	545
Set Width (setWidth).	545
Show Loading State (showLoadingState).	546
37.18 Connection.	546
Get Children (getChildren).	546
Get Connections (getConnections).	547
Get Root Folders (getRootFolders).	547
Get Selected Data Source (getSelectedDataSource).	548
Get System (getSystem).	548
Search Data Sources (searchDataSources).	549
Set System (setSystem).	549
Show Data Source Browser (showDataSourceBrowser).	550
37.19 Context Menu (ContextMenu).	551
Get Click Area (getClickArea).	551
Get Component (getComponent).	551
Get Context (getContext).	551
Get Data Source (getDataSource).	552
Is Item Visible (isItemVisible).	552
Set Item Visible (setItemVisible).	553
37.20 Convert.	553
Float to String (floatToString).	553
Float to String Using Locale (floatToStringUsingLocale).	554
Index Of (indexOf).	555
Replace All (replaceAll).	556
String Length (stringLength).	556
String to Float (stringToFloat).	557
String to Float Using Locale (stringToFloatUsingLocale).	558
String to Int (stringToInt).	558
Substring (substring).	559



	URL Encode (urlEncode).	560
37.21	Crosstab.	560
	Get Selected Member (getSelectedMember).	560
	Get Selected Members (getSelectedMembers).	561
	Get Selection (getSelection).	561
	Is Column Resizing Enabled (isColumnResizingEnabled).	562
	Is Conditional Formatting Visible (isConditionalFormattingVisible).	562
	Is Hierarchy Navigation Enabled (isHierarchyNavigationEnabled).	563
	Is Sorting Enabled (isSortingEnabled).	563
	On Select (onSelect).	563
	Remove Selection (removeSelection).	564
	Reset All Column Widths (resetAllColumnWidths).	564
	Reset Column Width (resetColumnWidth).	565
	Set Column Resizing Enabled (setColumnResizingEnabled).	565
	Set Column Width (setColumnWidth).	566
	SetConditionalFormattingVisible (setConditionalFormattingVisible).	566
	Set Default Column Width (setDefaultColumnWidth).	567
	Set Hierarchy Navigation Enabled (setHierarchyNavigationEnabled).	567
	Set Sorting Enabled (setSortingEnabled).	568
	Set Units and Scaling Factors Display (setUnitsAndScalingFactorsDisplay).	568
37.22	DataBoundComponent.	569
	Get Data Source (getDataSource).	569
	Set Data Source (setDataSource).	569
37.23	Data Cell (dataCell).	570
37.24	Data Source Alias (DataSourceAlias).	571
	Activate Hierarchy (activateHierarchy).	571
	Assign Data Source (assignDataSource).	571
	Assign Hierarchy (assignHierarchy).	572
	Clear All Filters (clearAllFilters).	572
	Clear Filter (clearFilter).	573
	Collapse Node (collapseNode).	573
	Configure Input Readiness (configureInputReadiness).	574
	Copy Filter (copyFilter).	574
	Copy Filters (copyFilters).	575
	Deactivate Hierarchy (deactivateHierarchy).	576
	Expand Node (expandNode).	576
	Export (export).	577
	Get Assigned Hierarchy (getAssignedHierarchy).	577
	Get Conditional Formats (getConditionalFormats).	578
	Get Conditional Format Name (getConditionalFormatName).	578
	Get Conditional Format Value (getConditionalFormatValue).	579

Get Conditional Format Value Ext (getConditionalFormatValueExt).	579
Get Data (getData).	580
Get Data as String (getDataAsString).	581
Get Data as String Ext (getDataAsStringExt).	582
Get Decimal Places (getDecimalPlaces).	583
Get Dimension Text (getDimensionText).	584
Get Dimensions (getDimensions).	584
Get Filter Ext (getFilterExt).	585
Get Filter Text (getFilterText).	585
Get Hierarchies (getHierarchies).	586
Get Info (getInfo).	586
Get Measures Dimension (getMeasuresDimension).	586
Get Measure Filters (getMeasureFilters).	587
Get Measure Filter Name (getMeasureFilterName).	587
Get Members (getMembers).	588
Get Member Display (getMemberDisplay).	588
Get Member List (getMemberList).	589
Get Negative Number Display (getNegativeNumberDisplay).	590
Get Scaling Factor (getScalingFactor).	590
Get Static Filter Ext (getStaticFilterExt).	591
Get Static Filter Text (getStaticFilterText).	591
Get Text (getText).	592
Get Totals Display (getTotalsDisplay).	592
Get Totals Position (getTotalsPosition).	593
Get Variables (getVariables).	593
Get Variable Value (getVariableValue).	593
Get Variable Value Ext (getVariableValueExt).	594
Get Variable Value Text (getVariableValueText).	594
Get Zero Display (getZeroDisplay).	595
Get Zero Display Custom Text (getZeroDisplayCustomText).	595
Is Conditional Format Active (isConditionalFormatActive).	596
Is Hierarchy Active (isHierarchyActive).	596
Is Initialized (isInitialized).	597
Is Input Ready (isInputReady).	597
Is Measure Filter Active (isMeasureFilterActive).	597
Is Result Set Empty (isResultSetEmpty).	598
Load Data Source (loadDataSource).	598
Move Dimension After (moveDimensionAfter).	599
Move Dimension Before (moveDimensionBefore).	600
Move Dimension to Columns (moveDimensionToColumns).	600
Move Dimension to Rows (moveDimensionToRows).	601

Open Prompt Dialog (openPromptDialog).	601
Reload Data (reloadData).	602
Remove Dimension (removeDimension).	603
Set Conditional Format Active (setConditionalFormatActive).	603
Set Decimal Places (setDecimalPlaces).	604
Set Drill Level (setDrillLevel).	604
Set Filter (setFilter).	605
Set Filter Ext (setFilterExt).	605
Set Measure Filter Active (setMeasureFilterActive).	607
Set Member Display (setMemberDisplay).	607
Set Negative Number Display (setNegativeNumberDisplay).	608
Set Scaling Factor (setScalingFactor).	608
Set Totals Display (setTotalsDisplay).	609
Set Totals Position (setTotalsPosition).	609
Set Variable Value (setVariableValue).	610
Set Variable Value Ext (setVariableValueExt).	611
Set Zero Display (setZeroDisplay).	612
Sort by Attribute (sortByAttribute).	613
Sort by Hierarchy (sortByHierarchy).	613
Sort by Measure (sortByMeasure).	614
Sort By Member (sortByMember).	614
Swap Dimensions (swapDimensions).	615
Unassign Hierarchy (unassignHierarchy).	615
37.25 Data Source Info (dataSourceInfo).	616
37.26 DataSourceConnection.	618
37.27 DataSourceDescriptor.	618
37.28 DataSourceName.	619
37.29 Date Field.	619
Get Date (getDate).	619
Is Enabled (isEnabled).	619
On Select (onSelect).	620
Set Date (setDate).	620
Set Enabled (setEnabled).	621
37.30 Dimension.	621
37.31 Dimension Filter.	622
Cancel (cancel).	622
Get Dimension Key (getDimensionKey).	622
Get Dimension Name (getDimensionName).	623
Set Dimension (setDimension).	623
Show Filter Dialog (showFilterDialog).	623
Submit (submit).	624

37.32	Dropdown Box, List Box, Radio Button Group. . . . .	624
	Add Item (addItem). . . . .	624
	Get Selected Text (getSelectedText). . . . .	625
	Get Selected Texts (getSelectedTexts). . . . .	625
	Get Selected Value (getSelectedValue). . . . .	626
	Get Selected Values (getSelectedValues). . . . .	626
	Dropdown Box: Get Tooltip (getTooltip). . . . .	627
	Is Enabled (isEnabled). . . . .	627
	On select (onSelect). . . . .	628
	Remove All Items (removeAllItems). . . . .	628
	Remove Item (removeItem). . . . .	628
	Set Enabled (setEnabled). . . . .	629
	Set Items (setItems). . . . .	629
	Set Selected Value (setSelectedValue). . . . .	630
	Set Selected Values (setSelectedValues). . . . .	631
	Dropdown Box: Set Tooltip (setTooltip). . . . .	631
	Sort (sort). . . . .	632
37.33	Fragment Gallery. . . . .	632
	Add Item (addItem). . . . .	632
	Add Items (addItems). . . . .	633
37.34	Filter Panel. . . . .	633
	Cancel (cancel). . . . .	633
	Set Dimensions (setDimensions). . . . .	633
	Submit (submit). . . . .	634
37.35	Formatted Text View. . . . .	634
	Get HTML Text (getHTMLText). . . . .	634
	Set HTML Text (setHTMLText). . . . .	635
37.36	FragmentBookmarkInfo. . . . .	635
37.37	Geo Map. . . . .	635
	Center Map (centerMap). . . . .	635
	Get Copyright Text (getCopyrightText). . . . .	636
	Get Selected Layer (getSelectedLayer). . . . .	636
	Is Layer Visible (isLayerVisible). . . . .	637
	Set Copyright Text (setCopyrightText). . . . .	637
	Set Layer Visible (setLayerVisible). . . . .	637
	Set Map Url (setMapUrl). . . . .	638
	Get Selected Member (getSelectedMember). . . . .	638
37.38	Hierarchy. . . . .	639
37.39	Icon. . . . .	639
	Get Background Color (getBackgroundColor). . . . .	639
	Get Background Shape (getBackgroundShape). . . . .	640

Get Color (getColor).	640
Get Icon Uri (getIconUri).	640
Get Size Factor (getSizeFactor).	641
Get Tooltip (getTooltip).	641
Set Background Color (setBackgroundColor).	642
Set Background Shape (setBackgroundShape).	642
Set Color (setColor).	643
Set Icon Uri (setIconUri).	643
Set Size Factor (setSizeFactor).	643
Set Tooltip (setTooltip).	644
37.40 Image.	644
Get Click Image (getClickImage).	644
Get Hover Image (getHoverImage).	645
Get Image (getImage).	645
Get Opacity (getOpacity).	646
Get Tooltip (getTooltip).	646
On Click (onClick).	646
Set Click Image (setClickImage).	647
Set Hover Image (setHoverImage).	647
Set Image (setImage).	648
Set Opacity (setOpacity).	648
Set Tooltip (setTooltip).	649
37.41 Input Field.	649
Get Value (getValue).	649
Get Tooltip (getTooltip).	650
Is Editable (isEditable).	650
Is Enabled (isEnabled).	651
Set Editable (setEditable).	651
Set Enabled (setEnabled).	651
Set Tooltip (setTooltip).	652
Set Value (setValue).	652
37.42 KeyValuePair.	653
37.43 JSON.	653
For Each (forEach).	653
37.44 Math.	654
Abs (abs).	655
Acos (acos).	656
Asin (asin).	656
Atan (atan).	656
Atan2 (atan2).	657
Ceil (ceil).	657



	Cos (cos).	658
	Exp (exp).	658
	Floor (floor).	659
	Log (log).	659
	Max (max).	660
	Min (min).	661
	Pow (pow).	661
	Random (random).	662
	Round (round).	662
	Sin (sin).	663
	Sqrt (sqrt).	663
	Tan (tan).	663
37.45	Member.	664
37.46	Navigation Panel.	665
	Set Dimensions (setDimensions).	665
37.47	Pagebook.	666
	Get Page Count (getPageCount).	666
	Get Selected Page (getSelectedPage).	666
	Get Selected Page Index (getSelectedPageIndex).	666
	Get Selected Page by Name (getSelectedPageByName).	667
	On Select (onSelect).	667
	Set Selected Page by Name (setSelectedPageByName).	668
	Set Selected Page Index (setSelectedPageIndex).	668
37.48	Panel.	669
	On Click (onClick).	669
37.49	PDF.	669
	Export Application (exportApplication).	669
	Export Panel Screen (exportPanelScreen(panelArray)).	670
	Export Application Screen (exportApplicationScreen).	670
37.50	Planning.	671
	Client Reset (clientReset).	671
	Has Client Changes (hasClientChanges).	671
	Has Unsaved Changes (hasUnsavedChanges).	672
	Recalculate (recalculate).	672
	Reset (reset).	672
	Save (save).	673
37.51	PlanningFunction.	673
	Clear All Filters (clearAllFilters).	673
	Clear Filter (clearFilter).	674
	Copy Filters (copyFilters).	674
	Execute (execute).	675

Get Dimension Text (getDimensionText).	675
Get Dimensions (getDimensions).	676
Get Filter Ext (getFilterExt).	676
Get Filter Text (getFilterText).	677
Get Member List (getMemberList).	677
Set Filter (setFilter).	678
Set Filter Ext (setFilterExt).	678
37.52 PlanningObjectWithVariables.	679
Copy Variable Value (copyVariableValue).	679
Get Variable Value Ext (getVariableValueExt).	680
Set Variable Value Text (setVariableValueText).	681
Set Variable Value (setVariableValue).	681
Set Variable Value Ext (setVariableValueExt).	682
Set Variable Value Range (setVariableValueRange).	683
37.53 PlanningSequence.	684
Execute (execute).	684
37.54 Popup.	684
Hide (hide).	684
Is Showing (isShowing).	685
Show (show).	685
37.55 SdkDataSource.	685
37.56 SearchExpression.	686
37.57 SingleMemberFilter.	686
37.58 Splitcell Container.	686
Get Data Sources (getDataSources).	686
Is Display Mode (isDisplayMode).	687
On Delete (onDelete).	687
On Drop (onDrop).	687
Set Display Mode (setDisplayMode).	688
37.59 State.	688
Back One Step (backOneStep).	688
Back To Start (backToStart).	689
Is Back To Start Available (isBackToStartAvailable).	689
Is Back One Step Available (isBackOneStepAvailable).	690
Set Personalization (setPersonalization).	690
Delete Personalization (deletePersonalization).	691
37.60 String.	691
Index Of (indexOf).	691
Length (length).	692
Split (split).	692
Substring (substring).	693

37.61	Tab	693
	Get Text (getText)	693
	Is Enabled (isEnabled)	694
	Set Enabled (setEnabled)	694
	Set Text (setText)	695
37.62	Tabstrip	695
	Get Selected Tab (getSelectedTabIndex)	695
	Get Selected Tab Index (getSelectedTabIndex)	696
	Get Tab (getTab)	696
	On Select (onSelect)	696
	Set Selected Tab (setSelectedTab)	697
	Set Selected Tab Index	697
37.63	Text	698
	Get Text (getText)	698
	Get Tooltip (getTooltip)	698
	Set Text (setText)	698
	Set Tooltip (setTooltip)	699
37.64	Timer	699
	Start (start)	699
	Stop (stop)	700
	Is Running (isRunning)	700
37.65	Tree	701
	Get Bottom Margin (getBottomMargin)	701
	Get CSS Class (getCSSClass)	701
	Get Height (getHeight)	701
	Get Left Margin (getLeftMargin)	702
	Get Right Margin (getRightMargin)	702
	Get Selected Member (getSelectedMember)	703
	Get Selected Members (getSelectedMembers)	703
	Get Selected Text (getSelectedText)	703
	Get Selected Texts (getSelectedTexts)	704
	Get Selected Value (getSelectedValue)	704
	Get Selected Values (getSelectedValues)	705
	Get Top Margin (getTopMargin)	705
	Get Width (getWidth)	706
	Hide Loading State (hideLoadingState)	706
	Is Visible (isVisible)	706
	Remove Model (removeModel)	707
	Set Bottom Margin (setBottomMargin)	707
	Set CSS Class (setCSSClass)	708
	Set Height (setHeight)	708

Set Left Margin (setLeftMargin).	709
Set Model (setModel).	709
Set Right Margin (setRightMargin).	710
Set Top Margin (setTopMargin).	710
Set Visible (setVisible).	711
Set Width (setWidth).	711
Show Loading State (showLoadingState).	712
37.66 Variable.	712
<b>38 Working with the Local Mode of the Design Tool.</b>	<b>714</b>
38.1 Launching SAP BusinessObjects Design Studio.	714
38.2 Creating New Analysis Applications.	715
38.3 Maintaining Settings in the Design Tool in Local Mode.	716
38.4 Storage of Applications and Images.	720
38.5 Selecting a Connection.	721
Defining Connections to BI Backend Systems.	722
38.6 Saving an Application Using a Different Name.	723
38.7 Executing an Application.	724
38.8 Executing an Application on a Mobile Device.	725
38.9 Collecting Support Information in Local Mode.	725
38.10 Coordinating the Translation of Translatable Texts in Local Mode.	726
38.11 Configuring the Report-Report Interface for Analysis Applications in Local Mode.	727
<b>39 Terminology Essentials.</b>	<b>729</b>

---

# 1 About This Guide

## 1.1 About the Documentation Set

The documentation set for SAP BusinessObjects Design Studio 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 on a regular basis.

### Administrator Guide

The Administrator Guide contains detailed information, which users need in order to install, configure and manage SAP BusinessObjects Design Studio. The guide is available on the SAP Help Portal at <http://help.sap.com/boad>.

### Application Designer Guide

The Application Designer Guide contains the conceptual information, procedures and reference material, which application designers need in order to create analysis applications. The guide is available on the SAP Help Portal at <http://help.sap.com/boad>.

### End User Guide

The End User Guide contains procedures and background information for users working with analysis applications. This guide is available on the SAP Help Portal at <http://help.sap.com/boad>.

### Developer Guide

The *Developer Guide: Design Studio SDK* contains procedures, reference material and background information, which enables developers to create third-party components with the Design Studio SDK and enhance the analysis applications with custom components. The guide is available on the SAP Help Portal at <http://help.sap.com/boad>.



---

## What's New Guide

The What's New Guide lists new and modified features in SAP BusinessObjects Design Studio that have been implemented since the previous release. The guide is available on the SAP Help Portal at <http://help.sap.com/boad>.

## Online Help

The Online Help contains the same information as the *Application Designer Guide* and the *Developer Guide: Design Studio SDK*, plus an additional chart properties guide called *Charts: Additional Information*. Choose ► [Help](#) ► [Help Contents](#) ▾ to open the guides in the design tool. You can also set the online help as a view within the design tool. Choose ► [View](#) ► [Help](#) ▾.

## 1.2 Who Should Read This Guide?

This guide is intended for application designers and users interested in building analysis applications using SAP BusinessObjects Design Studio.

## 2 Getting Started

### 2.1 What is SAP BusinessObjects Design Studio?

SAP BusinessObjects Design Studio enables application designers to create analysis applications and dashboards - based on BW, SAP HANA and universe data sources - for browsers and mobile devices (iPads, for example). It is the product of choice when full support for BW and HANA data models and engine capabilities is required. The product offers a design tool that allows you to create applications easily and intuitively without the need for native HTML and iPad UI programming skills.

SAP BusinessObjects Design Studio can be used locally and integrated in the following platforms:

- SAP BusinessObjects Business Intelligence (BI platform)
- SAP NetWeaver
- SAP HANA

#### Note

In local mode, you can create demo analysis applications for evaluating SAP BusinessObjects Design Studio with users at customer locations.

Besides analysis applications, application designers can also create planning applications that support both manual and automated data entry and changes to data. The application user can enter the planning data manually in the crosstab (in cells or rows) or use planning functions and planning sequences (also known as planning objects) in SAP BW Integrated Planning to enter data automatically. For planning data, you have to use a BW backend system as the planning system..

SAP BusinessObjects Design Studio also allows application designers to create visualizations with streaming data (push based), and to create visualizations, which have a near real-time connection to SAP HANA or SAP BW (pull based). The application designer simply connects to a streaming data source such as HANA SDS (Streaming Data Services) or SAP ESP (Event Stream Processor) from within the [Outline](#) view in the designer and then connects their streaming data to any OOTB [Chart](#). Real-time dashboards can be useful in the area of operational systems, where data may only be relevant as it occurs and where it may not be necessary to persist the data - for example, utility companies monitoring smart buildings and the financial capital markets.

In addition to the standard palette of components in Design Studio, which are used to visualize data and enable user interaction, you can develop 3rd party components with the Design Studio SDK and enhance your analysis applications with custom components. Besides SDK components that visualize data from a data source, you can also create SDK components that act as data sources for SDK components themselves. This enables SDK components to access a broad range of data sources such as local files, Web services, or new types of backend system. You can store and provide access to the analysis applications containing the 3rd party components on any of the supported platforms.

---

## Related Information

[Working with the Local Mode of the Design Tool \[page 714\]](#)

[Creating Planning Applications \[page 142\]](#)

## 2.2 Basic Concepts

### Entities

The design tool of SAP BusinessObjects Design Studio enables application designers to create and edit applications. These applications are saved in an XML file format. Applications generally consist of user interface components (such as charts, crosstabs, buttons) and data source aliases.

### Data source and data source alias

A data source alias represents an instance of a data source (for example, a BW query, or a SAP HANA data source) at runtime of the application. A single application can contain multiple instances of one data source. Every instance, for example, can apply its own filters on the same data source and thus represents its own subset of data. You can see the list of components and data source aliases used in an application in the outline view of the design tool.

### Data binding

To visualize data from a data source alias in a crosstab or chart, define a data binding in the design tool for these components. The data binding is simply a reference to the data source alias which provides the data. When the data of a data source alias changes (for example, a filter is applied at runtime), the system automatically updates all components that have a data binding to that data source alias. You can also see the data binding relationships between components and data source aliases in the outline view

### User interaction

Typically you create applications that not only visualize data, but which also provide possibilities for the application user to interact with the data; for example, changing filters, selecting drill-down dimensions or branching into a details view for a selected set of aggregated data. To enable user interaction with the data, add the relevant UI components to the application. For example, you could provide a row of buttons above a crosstab and chart component to filter the visualized data by different criteria.

## Eventing and script API

UI components provide a set of specific events that the application user can execute on the component. For example, every button provides an on click event. You can see a list of all available events of the component in the properties view of the design tool. Here, you can open the script editor and specify which actions should take place when the application user triggers the event. The script language is a subset of JavaScript and allows a sequence of script API method calls to be defined. The script API provides access at runtime to the application itself as well as to its components and data source aliases. Therefore, the application designer has flexible control of the application behavior by using event scripts.

## Setting properties in design time and run time

Besides the events, components also offer a variety of other properties in the properties view of the design tool. Some properties are common for all or almost all components; the component name identifies the component within the application, for example, and the layout properties define the size and position of the component within the application. Other properties are specific and depend on the component type. In all cases, the property values displayed in the properties view define the initial state of the components at application startup time. At application runtime, you can enable modification of properties by executing event scripts, as almost all component properties are accessible through the script API.

## Different Rendering Modes for Components

As with release 1.6 Design Studio supports two modes for rendering the components - the SAPUI5 m Mode and the SAPUI5 commons Mode - the components palette can vary in terms of available components for the respective rendering mode and in terms of the available properties. For further information, see [Using the SAPUI5 m Library \[page 45\]](#).

## 2.3 Feature Dependencies on Platform Deployment

The following table shows you an overview of the main Design Studio features and their availability in different deployments of the design tool.

Table 1: Features delivered with Release 1.6

Feature	Local Mode	BI Platform	SAP NetWeaver	SAP HANA
Basic Analysis Layout Template	yes	yes	yes	no
Chart Feeding Configurator	yes	yes	yes	no

Feature	Local Mode	BI Platform	SAP NetWeaver	SAP HANA
CSV Data Source / Rapid Prototyping with CSV	yes	no	no	no
Custom Measures, Top N filters	yes	yes	yes	no
Drag and Drop of Dimensions within/onto Crosstab	yes	yes	yes	no
Embedded CSS Editor	yes	yes	yes	no
Global Filters in Online Composition	yes	yes	no	no
Icon Component	yes	yes	yes	no
Info Chart	yes	yes	yes	no
SAPUI5 m Library	yes	yes	yes	no
Scorecard	yes	yes	yes	no
Spreadsheet	yes	yes	yes	no
Right-to-Left-Support	yes	yes	yes	no
Tree, Timer	yes	yes	yes	no

Table 2: Features prior to Release 1.6

Feature	Local Mode	BI Platform	SAP NetWeaver	SAP HANA
Attribute Planning Based on SAP BW integrated planning	yes	yes	yes	no
Bookmarking	yes	yes	yes	no
Bookmark Administration on BI Platform	-	yes	no	no
Chart Type Picker	yes	yes	yes	no
Context Menu	yes	yes	yes	yes (limited)
Databound Component Properties	yes	yes	yes	yes

Feature	Local Mode	BI Platform	SAP NetWeaver	SAP HANA
Data Discovery and Visualization	yes	yes	no	no
Enhanced Data Connectivity (UNIX)	no	yes	no	no
Export and Import Wizard	yes	yes	yes	yes
Export to SAP Lumira	yes	yes (but not for BI Mobile)	yes	no
Export to PDF and Printing	yes	yes	yes	no
Filter Line	yes	yes	yes	no
Geo Maps	yes	yes	yes	no
Multiple Language Support	no	yes (BIP >= 4.1 SP6)	yes	no
Navigation Panel	yes	yes	yes	yes
Offline Clickthrough	yes	yes	yes	no
Online Composition	yes	yes	no	no
Parallel Query Execution	yes	yes	no	(yes)
Standard Analysis Template	yes	yes	yes	no
Unmerge Variables	yes	yes	yes	no

## 2.4 Feature Dependencies on Data Source

The following table shows you an overview of Design Studio features and their availability in different data sources.

Table 3: Features delivered with Release 1.6

Feature	SAP BW Data Source	SAP HANA Data Source	Universe Data Sources	CSV Data Source
Cascading Filters	yes	yes	no	no

Feature	SAP BW Data Source	SAP HANA Data Source	Universe Data Sources	CSV Data Source
Calculations at Runtime	yes	yes	no	no
Data Discovery and Visualization Template	yes	yes	no	no
Dynamic Calculations at Runtime	yes	no	yes	no
Generic Analysis Ready-to-Run Template	yes	yes	no	no
Info Chart Conditional Formatting	yes	no	no	no
Online Composition Template	yes	yes	no	no
Scorecard Conditional Formatting	yes	yes (only available for single values)	yes (only available for single values)	yes (only available for single values)
Spreadsheet	yes (data entry/copy and paste only available for planning applications with SAP Integrated Planning / BPC 10.1))	yes (data entry/copy and paste only available for planning applications with SAP Integrated Planning / BPC 10.1))	yes (data entry/copy and paste only available for planning applications with SAP Integrated Planning / BPC 10.1))	yes (data entry/copy and paste only available for planning applications with SAP Integrated Planning / BPC 10.1))
Measure Based Filters (Top N Filters)	yes	yes (supported with new, HTTP-based SAP HANA connector (singleTop N filter on flat dimensions))	no	no

## 2.5 Launching SAP BusinessObjects Design Studio

### Procedure

1. To launch SAP BusinessObjects Design Studio, choose **Start** > **All Programs** > **SAP Business Intelligence** > **SAP BusinessObjects Design Studio** > **Design Studio**.
2. Depending on the mode you log on (BI Platform, SAP NetWeaver, SAP HANA or local), perform the following steps.

Option	Description
<b>BI Plat- form</b>	<ol style="list-style-type: none"> <li>1. In the <a href="#">Logon to SAP BusinessObjects BI Platform</a> dialog box, type in your user name for the BI Platform.</li> <li>2. Type in your password for the BI Platform.</li> <li>3. When you launch the Design Studio for the first time, type in the host and port into the <a href="#">Web Service URL</a>. For further information, contact your system administration.</li> <li>4. Select the authentication.</li> <li>5. Click <a href="#">OK</a>.</li> </ol> <div> <p><b>i Note</b></p> <p>Once you have entered the host and port in the Web Service URL, this information is stored by the system and you do not have to type it again.</p> </div>
<b>SAP Net- Weaver</b>	<ol style="list-style-type: none"> <li>1. In the <a href="#">Logon to SAP NetWeaver</a> dialog box. When you launch the Design Studio for the first time, select the BW system from the dropdown box.</li> </ol> <div> <p><b>i Note</b></p> <p>You can only select a system from the dropdown box, if you have added the system in your <a href="#">SAP Logon</a> dialog box.</p> </div> <ol style="list-style-type: none"> <li>2. Type in the client for the system.</li> <li>3. Type in your user for the system.</li> <li>4. Type in your password for the system.</li> <li>5. Type in the language you prefer.</li> <li>6. Click <a href="#">OK</a>.</li> </ol> <div> <p><b>i Note</b></p> <p>Once you have entered the system and, this information is stored by the system and you do not have to type it again.</p> </div>
<b>SAP HANA</b>	<ol style="list-style-type: none"> <li>1. In the <a href="#">Logon to SAP HANA</a> dialog box, type in your user name for the SAP HANA system.</li> <li>2. Type in your password for the SAP HANA system.</li> <li>3. When you launch the Design Studio for the first time, type in the XSEngine URL which consist of <a href="#">http:// &lt;server&gt;:&lt;port&gt;</a>. This URL describes the connection to SAP HANA.</li> </ol> <p>For further information, contact your system administration.</p>
<b>Local</b>	The Design Studio is launched immediately, and you do not have to enter any user credentials.

## Results

The design tool is launched with the welcome page.

### **i Note**

If you want to switch from a platform mode to the local mode (for example, from the BI platform to the local mode) during logon, click [Skip](#) on the logon dialog box of the BI platform.



---

## Related Information

[Starting with the Welcome Page \[page 33\]](#)

## 2.6 Starting with the Welcome Page

When you start SAP BusinessObjects Design Studio for the very first time, the welcome page is displayed. The welcome page consists of different sections described below:

- **Getting Started:** Enables you to learn how to use the design tool of SAP BusinessObjects Design Studio by providing access to tutorials. These are videos and How To documents that guide you through essential and basic tasks and concepts of the solution and help you to familiarize yourself with the design tool and its features.
- **Create New:** Enables you to create a new analysis application. You can select different templates that are optimized for desktop Web browser applications or mobile applications. You can choose between blank templates or predefined templates which correspond to various design and business needs. If you choose a predefined template, the system automatically creates a copy of this template. You can change your copy according to your needs.
- **Recently Used Analysis Applications:** Enables you to open recently-used applications.
- **Useful Links:** Enables you to find further useful information about the solution, for example, SAP Community Network and SAP Help Portal.

### Note

When you want the welcome page to be displayed every time the design tool is started, select the [Always show the Welcome page on startup](#) checkbox.

On the right bottom corner of the welcome page you will find buttons for social media. Clicking a button launches the Web browser and opens the relevant social media website.

## Related Information

[Creating New Analysis Applications \[page 715\]](#)

## 2.7 Working with the Design Tool

Before you start creating applications, take a moment to familiarize yourself with the interface. The design tool includes a layout editor, the Components, Outline and Properties views, a menu and a toolbar. You can show or hide the views or move them to other screen positions.

## Related Information

[Components View \[page 34\]](#)

[Outline View \[page 36\]](#)

[Properties View \[page 36\]](#)

[Layout Editor \[page 37\]](#)

[Managing Logs in the Design Tool \[page 323\]](#)

[Maintaining Settings in the Design Tool \[page 38\]](#)

[Storage of Applications and Images \[page 720\]](#)

[Activating Runtime Traces \[page 320\]](#)

### 2.7.1 Components View

The *Components* view contains all components that you can use for creating analysis applications. Components are user interface elements that you can drag and drop into the layout editor and thus create the content of the application. Once you have inserted a component in your application, you can change its layout and behavior by editing its properties in the *Properties* view. Most of the components offer a set of specific events that the application user can execute on the component.

As with release 1.6 Design Studio supports two modes for rendering the components - the SAPUI5 Main (m) Mode and the SAPUI5 Commons Mode - the components palette can vary in terms of available components for the respective rendering mode and in terms of the available properties for these components. For further information, see [Using the SAPUI5 m Library \[page 45\]](#).

#### Note

You cannot use components from both library parts (commons and m) in one application and mix them. Following this principle, you have to decide which mode to choose in the *Create New Application* dialog box when creating a new application. This decision is visualized in the application property *SAPUI5 m Mode* with the values *true* or *false*; this property is read-only and cannot be changed.

In case you have chosen the m mode, you have also to set the application property *Compact Form Factor true* or *false*. The Compact Form Factor is the rendering mode for desktop style applications in the SAPUI5 m mode.

As components have different functions, they are grouped in different folders:

### Analytic components

You use analytic components like crosstabs and charts to visualize your data. After you have dragged and dropped a crosstab or a chart into the layout editor, the component initially displays dummy data. As soon as you assign a data source to the crosstab or chart, it displays the data of the data source. This group contains the following components:

- chart

- 
- crosstab  
The crosstab displays multi-dimensional data in a table with analytic functions.
  - dimension filter
  - filter line
  - filter panel
  - geo map
  - info chart
  - navigation panel
  - scorecard
  - spreadsheet

## Basic components

The Basic Components folder contains a variety of components with different functions. Components like [Dropdown Box](#), [Radio Button](#) and [Checkbox](#) are used to select or filter data and thus enable user interaction at runtime. Components like [Button](#) and [Image](#) are used to enable navigation, whereas components like [Text](#) and [Date Field](#) are used to enhance the design and layout of the application. This group contains the following components:

- button
- chart type picker
- checkbox
- checkbox group
- date field
- dropdown box
- formatted text view
- fragment gallery
- icon
- image
- info chart feeding panel
- input field
- list box
- radio button group
- text
- tree

## Container components

Container components are used to group and structure the content of an application. They can also be used to optimize an application for a mobile device and thus enable specific motion gestures in mobile applications. This group contains the following components:

- grid layout

- pagebook
- panel
- popup
- split cell
- tabstrip

## 2.7.2 Outline View

The *Outline* view gives a hierarchical overview of the currently active application (the application whose editor is in focus). The overview contains all data sources and all UI components.

There are several ways of using drag and drop in the outline view :

- You can drag and drop components from the components view onto another container UI element in the outline view.

### ➔ Remember

The Layout node also serves as a container.

If a container component (tabstrip, pagebook, grid layout) only supports certain child types, drag and drop might not be allowed. For example, you cannot drop anything onto a grid layout node or a tabstrip node. Instead you need to drop onto grid cells and tabs.

- You can move/drag and drop components within the outline view (within the same container to change the order of UI components, or between different containers). For example, drag a button from a grid cell to a tab on a tabstrip. You can even drag a tab from one tabstrip to another tabstrip. Grid cells, however, are treated in a special way and cannot be dragged at all.
- You can assign data source aliases to data-binding sensitive components like crosstabs or charts by using drag and drop.

You can use various functions in the context menu of the respective elements (components or data sources).

## Related Information

[Working with the Outline View \[page 54\]](#)

[Using the Context Menu in the Outline View \[page 56\]](#)

## 2.7.3 Properties View

The *Properties* view is the standard Eclipse properties view. It displays the properties of the currently selected object (component or data source), which can be a tree node in the outline view or a component in the editor.

### i Note

Some components do not expose properties, for example the cells of a grid layout.

---

When you click on the *Value* column of a property, a cell editor specific to the property is displayed. The cell editor might be a text input field, a dropdown box, or contain a button that opens a dialog.

If you select multiple components (currently only possible in the outline view), a property sheet is displayed. Only those properties that are common to all selected components are displayed. These properties can now be edited for all selected components simultaneously.

#### Restriction

Only simple properties support this multi-editing function.

## 2.7.4 Error Log View

The *Error Log* view can be used to indicate general errors like system problems and messages that will be displayed when using some special methods for script validation.

### Related Information

[Error Analysis in Scripting \[page 279\]](#)

## 2.7.5 Problems View

The *Problems* view can be used to display errors in scripting during design time and after saving the application. To resolve and examine the problems double-click the errors listed in the view.

## 2.7.6 Layout Editor

For each application, an editor is opened in the editor area. The editor contains an SWT browser control. Under Windows, Internet Explorer is embedded into this SWT browser control. With other operating systems, a browser such as Mozilla or Safari is embedded into the SWT browser control.

There is a two-way interaction between the browser control and the various parts of the design tool:

- Dragging and dropping from the components view to the editor area: This creates new components.
- Dragging and dropping a data source alias from the outline view onto a data-binding aware component in the editor: This assigns the data source alias to the target component.
- Selecting a component in the outline view: This updates the selection in the editor and the reverse is also true.
- Moving, resizing or deleting components in the editor: This updates the outline and properties view.

## 2.7.7 Maintaining Settings in the Design Tool

You can define settings for the design tool in the [Preferences](#) dialog box. To start the [Preferences](#) dialog box, choose **Tools > Preferences > Application Design**.

Change the default values as required and choose [Apply](#) to make the entered values effective. If you want to reset the default values, choose [Restore Defaults](#). To close the [Preferences](#) dialog box, choose [OK](#).

The following settings are available:

### General

You can switch the startup mode. In the default setting, the design tool starts in local mode after installation. If you want to use SAP BusinessObjects Design Studio integrated into one of the supported platforms, select the required platform:

- [SAP BusinessObjects BI Platform](#)
- [SAP HANA](#)
- [SAP NetWeaver](#)

After restarting the design tool, you need to log on to the selected platform.

#### Note

In the [Logon to <selected platform>](#) dialog box, you can still decide to log on locally by clicking [Skip](#), even if the startup mode is set to one of the supported platforms. The design tool starts in local mode.

In the [Undo History Size](#) field, you can change the default value 50 and enter a number between 20 and 999. This value defines how often application designers can undo their changes when working in the design tool.

### Embedded Web Server

You can define the network port for the embedded Web server of the design tool. Enter a number between 1024 and 65535. When an application is executed, the port number can be seen in the URL. If the port is set to 5555 for example, the URL is `http://localhost:5555/aad/web.do?APPLICATION=MYAPP`. If you leave the default value at 0, the system automatically assigns a network port.

#### Tip

If users create bookmarks for their applications, the network port has to be set to a fixed number. Auto-assigning the network port does not guarantee that the bookmarks will work properly.

## Application Preview

The language of message texts and tooltips is defined in the language setting in the BI launch pad or the Web browser. The format for numbers, dates and times varies according to the defined language setting. To preview the analysis applications in the required language and formatting, application designers can switch between the preferred viewing locale of the BI launch pad (*BI Platform User Settings* radio button) or the Web browser language (*Web Browser* radio button).

## Application Recovery

In the default setting, the system automatically saves unsaved applications every minute. You can configure the auto-save time interval as required. Select the *Save application recovery information every <1> minute* checkbox and enter the required number for the auto-save time interval.

### ➔ Tip

The design tool needs to be restarted for the new interval to become active.

There is a background job that searches for unsaved applications during the specified interval. If unsaved applications are found, the system will extract their XML code and store it under `<user home directory>\Analysis-workspace\metadata\plugins\com.sap.ip.bi.zen`. The content of this file is encrypted using Eclipse secure store technology.

### i Note

The auto-save function does not replace saving an application. Saving an application or closing it and answering the *Save changes?* question with either *Yes* or *No* will delete the XML from the `autosaves` file.

In the event of a system crash, the auto-saved XML persists. When restarting the design tool and opening the affected application again, the designer is informed that an auto-saved version of the application exists.

- If the designer decides to restore the auto-saved version, the system takes the stored XML, saves the application and opens the application in the layout editor. The auto-saved XML is removed.
- If the designer decides to discard the auto-saved version, the auto-saved XML is removed as well.

## Member Selection

Application designers need to pick single members of a dimension when using statements like `setFilter` for a dropdown box, for example. They can pick the members in the content assistance of the *Script Editor* dialog box or in the *Select Member* dialog box. The *Select Member* dialog box can list a small or large number of members, depending on the maximum threshold number of members. You can set the default threshold in the *Preferences* dialog box.

- You can define the maximum number of members that are displayed in the *Select Member* dialog box. Enter the required number in the *Maximum number of members to fetch from backend in dialog* checkbox. The default threshold is 1000.

- You can define the maximum number of members that are displayed in the content assistance of the *Script Editor* dialog box. Enter the required number in the *Maximum number of members to fetch from backend in content assistance* checkbox. The default threshold is 20. If the number of available members exceeds this threshold number, the content assistance will not list single members. Instead it will offer the *Select Member...* entry, which opens the *Select Member* dialog box.

#### ➔ Tip

Designers can access the content assistance in the script editor by pressing `CTRL` + `SPACE`.

You can also decide whether the system should display warnings in the script editor whenever designers manually enter non-existent values. To activate the warnings, select the *Display warnings for manually entered invalid values* checkbox.

## Prompt Handling

In the *Prompts* dialog box, application designers and application users set values for prompts. For SAP BW data sources, prompts are defined as variables. In SAP HANA, prompts are defined as input parameters or variables. When working with data sources with defined variables, application designers might be prompted to set the required values before continuing their work in the design tool (if there are mandatory variables without default values, or if variables have invalid default values, for example). SAP BusinessObjects Design Studio stores all valid prompt values of each analysis application in the user's cache file in `<user home directory>\Analysis-workspace\.metadata\.plugins\com.sap.ip.bi.zen\cache`. This provides application designers with a smooth workflow in the design tool. If this was not the case, the *Prompts* dialog would appear when designers reload an analysis application or modify the initial state of the data source in the *Initial View* dialog box, for example.

In the *Preferences* dialog box, you can specify whether the *Prompts* dialog box should appear when an analysis application is executed locally:

- If you want to simulate how an application user opens the analysis application, leave the checkbox deselected (default setting). When you execute an analysis application locally, the *Prompts* dialog box appears and you can set the prompt values in the same way an end user would do.
- If you want to use the prompt values from the cache file, select the *Use cached prompt values for local execution* checkbox. The *Prompts* dialog box does not appear and application designers can test their analysis applications quickly.

#### i Note

If the *Prompts* dialog box still appears even if this setting is activated, check if the *Force Prompts On Startup* property of the analysis application is set to true. This property always forces the *Prompts* dialog box to appear, regardless of whether the *Use cached prompt values for local execution* checkbox is selected.

If you want to clear the prompt values of an analysis application in the cache file, press *Clear Prompt Value Cache...* You can select the required analysis application(s) for this cache deletion. When you reload the application, the *Prompts* dialog box will appear and you can set new values, for example.



---

## Application Templates

SAP BusinessObjects Design Studio includes a set of templates that offer designers an easy way to get started. When creating new applications, designers can choose between different templates that are optimized for desktop Web browser applications or mobile applications in the [New Application](#) dialog box (▮ [Application](#) ▸ [New...](#) ▮). For more information, see “Creating a new analysis application” in the *Application Designer Guide: Designing Analysis Applications* under ▮ [Help](#) ▸ [Help Contents](#) ▮ in the design tool.

Application designers can also create analysis applications and provide them as templates for other application designers. Under ▮ [Tools](#) ▸ [Preferences](#) ▸ [Application Design](#) ▸ [Application Templates](#) ▮, you can add the path to the folder where these templates are stored. Here you can also define your own template categories that indicate the target device types recommended for a specific template. The template categories are then listed in the [New Application](#) dialog box (▮ [Application](#) ▸ [New...](#) ▮), and also in the [Export Application As Template](#) dialog box (▮ [Application](#) ▸ [Export as Template...](#) ▮). For more information, see “Exporting Applications As Templates” in the *Application Designer Guide: Designing Analysis Applications* under ▮ [Help](#) ▸ [Help Contents](#) ▮ in the design tool.

## Backend Connections

Under ▮ [Tools](#) ▸ [Preferences](#) ▸ [Application Design](#) ▸ [Backend Connections](#) ▮, you can reload new connections or universes in the design tool. This function allows application designers to continue their work in the design tool without having to close and open it again.

## Network Connections

Under ▮ [Tools](#) ▸ [Preferences](#) ▸ [Application Design](#) ▸ [Network Connections](#) ▮, you can specify the proxy settings to be used when opening connections. For more information, see “Network Connection Preferences” in the online help. You can access this chapter by pressing F1 or the [Help](#) button in the [Preferences](#) dialog box.

## Support Settings

Under ▮ [Tools](#) ▸ [Preferences](#) ▸ [Application Design](#) ▸ [Support Settings](#) ▮, you can specify the amount of information stored in a log file and activate functions to record traces or collect statistics data. For more information, see the links in the [Related Information](#) section below.

Under [SAP HANA Data Sources \(HTTP\)](#), you can specify that you use an HTTP proxy for communication with the SAP HANA system. If you have configured your SAP HANA system so that it can only be accessed using a proxy, select the [Use HTTP Proxy](#) checkbox.

- Under [Proxy Host](#), enter the name of the system hosting the proxy service used by the HTTP destination.
- Under [Proxy Port](#), enter the port to connect on the system hosting the proxy service.

You can also use this setting for support purposes: Using Fiddler as a proxy, you can record Fiddler traces, which helps to SAP diagnose your issues with SAP HANA HTTP data sources. To set Fiddler as a proxy, select the [Use HTTP Proxy](#) checkbox.

- Under [Proxy Host](#), enter **localhost**.
- Under [Proxy Port](#), enter **8888**.

For more information, see SAP Note [2166049](#).

In the status bar of the design tool, the indicator [HTTP Proxy: Off/On](#) shows you if this setting is switched on or off. By double-clicking on the indicator, you can quickly access the [Preferences](#) page to change your current setting.

#### **i** Note

This setting only applies to SAP HANA data sources that are connected using HTTP as the protocol for communication (not JDBC). It is available for all platforms except SAP NetWeaver.

## **Related Information**

[Setting Network Connections If Logon Problems Occur \[page 319\]](#)

[Activating Runtime Traces \[page 320\]](#)

[Activating SAP JCo Traces \[page 321\]](#)

[Viewing And Collecting Statistics Data At Runtime \[page 322\]](#)

## **2.7.8 Storage of Applications and Images**

To work efficiently with the design tool on the BI platform, you need to know where your applications are stored and where to store the images and icons that you want to insert in your applications.

### **Analysis applications, images and icons**

You store the analysis application files in folders on the BI platform. You can also store images and icons you use in your analysis applications on the BI platform.

You can also use images and icons in your analysis applications from the Internet or intranet.

#### **➔ Tip**

We recommend that you use one folder for all images and icons. This makes it easier to transport analysis applications and the referred images or icons between different BI platform systems.

---

## Analysis applications history

The analysis applications history in the Application menu is persisted in your cache file <home directory>\.sap\com.sap.ip.bi\cache.

## Data source history



The data source history in the [Add Data Source](#) dialog box is persisted in <home directory>\.sap\com.sap.ip.bi\cache.

## 3 Creating New Analysis Applications

### Context

When creating new analysis applications, you can select different templates. You can choose between standard templates (*Blank* or predefined ones like *Basic Analysis Layout*, *Basic Layout*) or ready-to-run predefined templates (such as *Data Discovery and Visualization*, *Generic Analysis* and *Online Composition*) that correspond to various design and business needs. When you choose a template, the system automatically creates a copy of it. You can change this copy according to your needs.

### Procedure

1. Click  *Application* > *New...*  in the menu of the design tool or click *Create Analysis Application* on the Welcome page. The *New Application* dialog box is displayed.
2. In the *New Application* dialog box, perform the following steps:
  - a. Enter a unique name for your application in the field under *Application Name*.
  - b. If your Design Studio is deployed on a BI or SAP HANA platform, the system suggests under *Folder* a folder on the platform to store your application. However, you can also browse for a different folder.
  - c. Choose the rendering type for the components in your application: *SAPUI5* or *SAPUI5 m*. The SAP UI5m rendering type creates applications using the SAPUI5 main control set. For more information about the use of the rendering types, see [Using the SAPUI5 m Library \[page 45\]](#). Depending on your choice, the templates available for the rendering type are displayed automatically.
  - d. Choose the template you want to use for your application. A short description of the template is displayed at the right side of the dialog box.
  - e. Choose *Create*. The template is opened in the editor where you can change and edit it.

### Results

You have created a new application. Now you can define the content of your application by adding components and data sources to the blank template, or by adjusting the predefined or ready-to-run templates according to your needs.

### Related Information

[Working with Templates \[page 103\]](#)

## 3.1 Using the SAPUI5 m Library

For SAP BusinessObjects Design Studio 1.6 the rendering capabilities have been extended.

In previous releases Design Studio used the so-called **Commons** part of the SAP UI5 library (these are controls in name space sap.ui.commons of SAPUI5) in order to create the Web user interface (UI) for all supported Design Studio components like List Box, Filter Panel, Button, etc. The Commons library was developed for creating basic common controls (components), mainly intended for desktop applications.

Design Studio 1.6 now also supports the so-called **Main** part of the SAPUI5 library (sometimes just referred as “m”, these are controls in name space sap.m of SAPUI5). The Main (m) part was developed having the mobile use case in focus; those components are therefore specialized for mobile devices.

However, the usage of the SAP UI5 m library is not restricted to mobile scenarios; it also supports desktop applications. In order to adjust the visualization accordingly, there are two form factors for the m mode: the **compact** for the desktop and the **cozy** form factor with more spacing and padding for mobile use cases.

Another reason for introducing components based on the SAPUI5 library is that SAP also uses the SAPUI5 Main library to create SAP Fiori UIs. That is why this library part is the go-to solution for the future.

### Basic principles, limitations and recommendations

Take the following aspects in account when deciding on which rendering mode to choose for an application:

- You cannot use controls from both library parts (commons and m) in one application and mix them. Following this principle, you have to decide which mode to choose when creating a new applications in the [Create New Application](#) dialog box. This decision is visualized in the application property [SAPUI5 m Mode](#) with the values [true](#) or [false](#), this property is read-only and cannot be changed.  
In case you have chosen the m mode, you also have to set the application property [Compact Form Factor](#) [true](#) or [false](#).
- The Design Studio component model has also been extended in a way that provided components can offer both rendering modes. If your application is set to one of the rendering modes, only those components are displayed in the components palette, which offer the corresponding rendering. SAP tries to ship all components with support for both rendering modes – in rare cases there might be restrictions (see below).
- Once you have selected the rendering mode, you cannot change this setting for the application anymore. However, Design Studio offers a function to migrate SAPUI5 Commons-based applications to the Main mode by choosing the [Migrate to SAPUI5 m Mode](#) in the [Tools](#) menu. The migration script sets the new mode and adjusts also component-specific settings, if needed. Most of the components have identical properties and methods, but there are also cases where you find differences in the component properties and the according API (see below).
- If a component does not support the SAPUI5 m mode rendering, it will be deleted by the migration script. SAP recommends copying the application first and then starting the migration.
- The right to left (RTL-) compliant rendering for RTL languages is only available in the SAPUI5 m rendering mode.

## Available components depending on the rendering type

Most of the components from the components palette are available in both rendering modes. However, there are also some components which are only available in the SAPUI5 commons or the m mode:

Table 4:

Components available only in the SAPUI5 Commons mode	Components available only in the SAPUI5 Main mode
<ul style="list-style-type: none"><li>• Popup</li><li>• Filter Line</li><li>• Split Cell Container</li><li>• Fragment Gallery</li><li>• Chart</li></ul>	<ul style="list-style-type: none"><li>• Action Sheet (Technical Component)</li></ul>

Due to the different rendering in SAPUI5 m mode, some components and script elements have a different properties sheet in the m mode:

- application
  - Property *Position of Message Window* has been removed.  
The message popup will now always open where the message button is located.
  - Property *Compact Mode* has been added.  
Renders the application in compact mode. The components will take up less space and be more compact in general. The compact mode is more useful for desktop applications.
- dimension filter
  - Property *Display Mode* has been removed.
  - Property *Popup Width* has been removed.
  - Property *Popup Height* has been removed.
  - Property *Popup Position* has been removed.
  - Property *Popup is Modal* has been removed.
  - Property *Auto Apply* has been removed.
- filter panel
  - Property *Display Mode* has been removed.
  - Property *Drag and Drop* has been removed.
  - Property *Auto Apply* has been removed.
  - Property *Direct Input for Filter* has been removed.
  - Property *Desktop Style for Popup* has been removed.
  - Event *On Cancel* has been removed.
- list box
  - Property *Multi Selection* has been removed.
  - Property *Selection Mode* has been added.  
This property specifies the behavior as well as the visualizations for the list box. This includes a mode which supports multiple selection.
- pagebook
  - Property *Transition Effect* has been removed.
  - Property *Transition Direction* has been removed.
  - Property *Enable Swiping* has been removed.

- text
  - Property *Style* has been removed.

## 3.2 Adding Components to an Application

### Prerequisites

You have opened a new or existing application.

### Context

You add components to an application to make the data visible, enable user interaction and create the layout of the application. Usually you use crosstabs or charts to visualize the data. Buttons or dropdown boxes enable the application user to interact with the data.

### Procedure

In the *Components* view, click a component:

- Drag and drop the component of your choice into the editor area.
- Drag and drop the component of your choice into the *Layout* folder of the *Outline* view.
- Enter a text (no wildcards) in the *filter text* box at the top of the *Components* view, if you want to filter for a certain component. The component groups (for example, *Analytic Components* or *Basic Components*) are automatically hidden if no match was found inside the group, or expanded if they are currently collapsed and contain a match with the filtering string.

### Results

You have created the general user interface of your application. You can now specify and change the properties of the components you have added to the application.

### Related Information

[Specifying the Properties of a Component \[page 48\]](#)

[Deleting Components in an Application \[page 48\]](#)

## 3.2.1 Specifying the Properties of a Component

### Context

Once you have dragged and dropped one or more components into the editor area, you can adjust the layout and behavior of the components by changing their properties

### Procedure

1. Choose the component you want to adjust:
  - Click on a component in the editor area.
  - Click on a component in the *Layout* folder of the *Outline* view.

The properties of the component are now ready for editing in the *Properties* view.

2. In the *Properties* view, click on the property you want to change.
3. Enter the corresponding property value on the right side.

There are several ways to set the value of the property (depending on the property type):

Option	Description
<b>numeric entry</b>	Enter a number and click <input type="text" value="Enter"/> . This number represents either a numeric pixel description (such as for the properties <i>Top Margin</i> , <i>Left Margin</i> , <i>Bottom Margin</i> , <i>Right Margin</i> , <i>Width</i> , <i>Height</i> ), or an absolute number (such as for the properties <i>Grid Row</i> and <i>Grid Column</i> ).
<b>string entry</b>	Enter a text and click <input type="text" value="Enter"/> . Properties of this type include <i>Caption</i> and <i>Tooltip</i> .
<b>Boolean choice</b>	Click the dropdown box in the <i>Value</i> column of the relevant property. Click <i>false</i> or <i>true</i> and then <input type="text" value="Enter"/> . Properties of this type include <i>Enabled</i> and <i>Style</i> .
<b>interaction using dialog boxes</b>	With more complex properties you open and work in special dialogs. Click the <i>Value</i> column of the relevant property. The value field is now ready for editing. Click on the button next to the field. A dialog box opens. You can now edit the items for the dropdown-box component, add a data source or use the script editor for creating interactive components.

## 3.2.2 Deleting Components in an Application

### Prerequisites

You are in the editing mode of an application.

### Context

You want to change an application by removing existing components.



---

## Procedure

You can either:

- Click the component you want to delete in the editor and press `Del` on your keyboard.
- Right-click the component you want to delete in the *Layout* folder of the *Outline* view and click *Delete*.

## Results

You have deleted the selected component from your application.

### 3.2.3 Selecting Multiple Components in the Editor

When working on your application in the editor, you can select (and move) multiple components by holding the CTRL key while clicking.

When selecting multiple components in the editor, take the following hints into account:


- You can move multiple components once they are selected.
- You can only move components that are in the same container.
- It is not possible to resize the components once they are selected.
- When changing a container, the position of the mouse is used to specify the container which all selected components are moved to.
- You can select every component in the application but you can only move components that are in the same container.
- Container components do not change to show multiple selected components (unlike single selection).
- You can only move container components into or out of container components as described above (and not by using the arrow keys).

## 4 Creating Analysis Applications for Mobile Devices


When creating mobile analysis applications for mobile devices, you should take note of the following SAP recommendations:

- Keep the applications simple and do not use too many components.
- Do not use crosstab components for smartphone applications.
- To improve performance and user experience for tablet applications that contain a crosstab component, SAP recommends the following:
  - limit to 500 the total number of cells in your crosstab.
  - your crosstab should contain, for example, a maximum of 50 rows and 10 columns for pixel-based scrolling.
- When running an application in the SAP BusinessObjects Mobile application, it is recommended that you design smartphone applications in portrait format and design tablet applications in landscape format.

### Note

- Design Studio 1.6 now supports the so-called **Main** part of the SAPUI5 library. The Main (m) part was developed with the mobile use case in mind and those components are therefore specialized for mobile devices. When you select to create an analysis application in "m" mode, you also have to set the application property *Compact Form Factor* *true* or *false*. This allows you to adjust your visualization between two form factors: the **compact** form factor for the desktop and the **cozy** form factor with more spacing and padding for mobile use cases.
- For more information about Android version support, see the SAP Product Availability Matrix: <https://support.sap.com/release-upgrade-maintenance/pam.html> 

You should also refer to the following SAP Note:

SAP Note Number	Description
<a href="#">2240553</a> 	Considerations when viewing Design Studio 1.6 applications on SAP BI Mobile client application (MOBI).

## Related Information

[Crosstab \[page 372\]](#)

[Using the SAPUI5 m Library \[page 45\]](#)

## 5 Creating the Layout of an Analysis Application

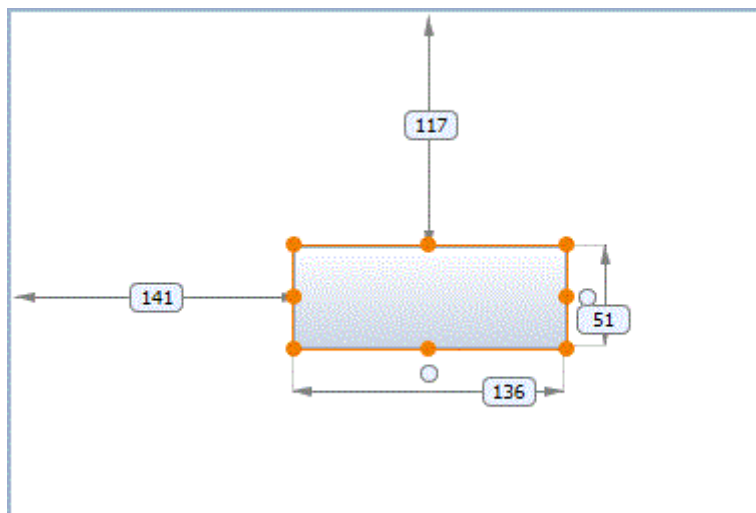
You create the layout of an application by inserting components in the editor, changing the properties of the components and arranging the components within an application (by using container components). Most of the work is performed in the layout editor of the design tool. The editor provides a What-You-See-Is-What-You-Get display. This enables you to view the working application during the creation process and get immediate feedback. Some of the steps you perform in the layout editor can also be performed in the [Outline](#) and [Properties](#) view of the design tool.

The following aspects are relevant during the layout creation process:

- changing the (layout) properties
- working with the layout structure of the Outline view
- using container components

### 5.1 Changing the Layout Properties and Docking Behavior

You create the layout of an application by inserting components in the editor and changing the layout properties of the components. You can change the layout properties of components either in the [Properties](#) view (where you can edit all properties) by entering the values manually or in the layout editor by dragging the borders of a component. The following figure shows a button component in the editor:



The red rectangle shows the size of the component. It contains eight red handles for resizing. In this figure you see two (out of a possible four) docking visualizers that show the distance between the component and the top border and left border of the application. Docking visualizers are displayed as an arrow with a number if the component is bound to a border (number is displayed by the relevant property). A docking visualizer displayed as an empty circle means that the component is not bound to a border ("auto" is displayed by the relevant property). The

number and the empty circle are interactive. Clicking the number changes the display to "auto" and clicking on the empty circle changes the display to a number that represents the distance to the container border.

When changing the layout properties, keep the following points in mind:

There are three properties that specify the position and size of a component in an application:

- width: distance to left margin and distance to right margin along the horizontal axis
- height: distance to top margin and distance to bottom margin along the vertical axis

To enable dynamic/relative resizing of the components with changing Web browser windows, one of these three properties of each axis is always set to `auto`. It is not possible to set all three properties to `auto`.

### Example

#### Crosstab with fixed margins

Top margin = 60  
Left margin = 120  
Bottom margin = 116  
Right margin = 455

The values for width and height are set to `auto`. This means that the distances between the margins of the crosstab and the margins of the application are fixed, while the height and width of the crosstab are variable. The height and width of the application vary according to different screen or window sizes. In this case the height and width of the crosstab vary accordingly.

### Example

#### Crosstab with fixed width and height

If you set the height and width of the crosstab to fixed values, one property of each axis is set to `auto`.

Top margin = 60  
Left margin = 120  
Bottom margin = `auto`  
Right margin = `auto`  
Width = 600  
Height = 400

In this case, the left margin and the width of the crosstab are fixed, while the third property on the horizontal axis (the right margin) is variable. The top margin and the height of the crosstab are fixed, while the third property on the vertical axis (the bottom margin) is variable. With different screen or window sizes, the height and width of the application vary. In this case, the bottom margin and the right margin vary accordingly.

## Related Information

[Working with the Outline View \[page 54\]](#)

---

## 5.2 Using Container Components

You can use container components to arrange and organize the structure and layout of your application. After inserting a container component (tabstrip, pagebook, grid layout) in the editor, you can place analytic components, basic components and other container components into the (first) container component.

## 6 Working with the Outline View

In the [Outline](#) view of the Design Studio, you see all available components and elements of an application like data sources and components. They are listed in a hierarchical folder structure with each folder representing one type of application element. You can use the [filter text](#) box to type in the names of application elements or parts of these names. As soon as you have typed in a filter string, the box is marked in orange bold and the findings in the structure are displayed in bold as well. In addition, you can use the various context menu functions to create your application (copy and paste, for example). Currently the following folders are available:

- In the [Layout](#) folder, you will find all components used in the application. You can easily change the order and position of the components within the folder or within container components. Keep the following principle in mind when designing an application and working within the [Layout](#) folder: When you place components in the [Layout](#) folder at the same level as siblings, the one at the lowest position of this level is in foreground of the application, whereas the sibling component above is in the background of the application. This principle is also valid for components that are listed as lower levels (children) or higher levels (parents) in the folder hierarchy. The children are in the foreground of the application, the parents in the background. This principle is important when you add transparent components to the front and thus hide the components behind it. In this case the user cannot interact with the background components at runtime and you cannot click on them in the editor at design time. To be able to work with a background component, click on it in the [Layout](#) folder.
- The [Data Source](#) folder lists all data sources used in the application. You can assign a data source to a component by simply dragging and dropping it to the component in the editor or in the layout folder.
- In the [Planning Objects](#) folder, you can add planning functions and sequences of SAP BW Integrated Planning, which enable you to enter data automatically.
- In the [Technical Components](#) folder, you will find the following elements:
  - action sheet  
The [Action Sheet](#) is a new technical component, which can be scripted to contain an action list that opens next to another component. To activate the [Action Sheet](#), you must call a scripting method, for example, from within the [onClick](#) event of a button. This function takes as a non-optional parameter, the component to which it is rendered beside. The [Action Sheet](#) is modal, as it is closed when the application user clicks away from it. You can configure multiple instances within an application of the [Action Sheet](#) technical component. The [Action Sheet Items](#) property allows you to set the Value, Text and Icon to appear in your [Action Sheet](#). The [Placement](#) property allows you to position the [Action Sheet](#) relative to another component. The options include the following:
    - auto
    - left
    - right
    - top
    - bottom
    - horizontal
    - vertical
  - backend connection  
The [Backend Connection](#) enables the application user to select a generic data source at runtime by means of a data source selection dialog box. You can either use the predefined data source selection dialog box for runtime and configure it in the properties of the Backend Connection, or you can use the

---

API to create your own user interface for browsing data sources. Before you can use the Backend Connection, you have to assign a system at design time. This can be done in the [Property](#) view of the design tool or using the API.

- context menu  
The technical component [Context Menu](#) enables the application user to navigate and analyze data at runtime. At design time, the context menu is automatically added to every new application, but can be removed if not required. Each application can have only one context menu.
- global scripts object  
The [Global Scripts Objects](#) is a technical component type, which provides a grouping of global script functions. On each global scripts object, you can create any number of script functions. Each script function has a configurable return type and can have any number of typed input parameters.
- pdf  
The [PDF](#) is a technical component type, which enables the export to PDF functionality. Depending on the scripting methods applied, you can export to PDF a WYSIWYG version of your application or panel within your application. You can also export to PDF a report style version of your application, with one component per page.
- text pool  
The [Text Pool](#) enables text translation in analysis applications. In addition to the texts of the data from the back-end system ( SAP BW), analysis applications can contain translatable texts, like labels on buttons or messages, which are created by you, the application designer. If you want to provide your analysis applications in different languages, you need to make your application translatable by adding a [Text Pool](#) component to your application. The system collects all translation-relevant texts that you enter as property values in the [Properties](#) view in the [Text Pool](#) component and saves them for translation.
- timer  
The [Timer](#) allows you to refresh or change a data source at regular intervals to update a data bound component. It can also be used with any OOTB [Chart](#) component to simulate real-time dashboards with single or multiple SAP HANA or SAP BW data sources. The time interval of the [Timer](#) is set with the [Interval in Milliseconds](#) property within the component. The default setting is [1000](#). The [On Timer](#) event allows you to set the Design Studio script that is executed periodically, each time the time interval elapses.

## Related Information

[Using the Context Menu in the Outline View \[page 56\]](#)

[Using Planning Functions and Sequences \(Automated Planning\) \[page 145\]](#)

[Working with Global Scripts Objects and Global Script Functions \[page 297\]](#)

[Enabling Text Translation in Analysis Applications \[page 298\]](#)

[Exporting to PDF \[page 125\]](#)

[Enabling Export to PDF in Analysis Applications \[page 126\]](#)

[Using the Backend Connection Component for Data Source Browsing \[page 311\]](#)

[Working with Real-Time Dashboards \[page 262\]](#)

## 6.1 Using the Context Menu in the Outline View

The context menu for the respective elements in the [Outline](#) view offers you a variety of functions that help you create your application and work with it efficiently. The scope of functions depends on the element (data source or component) in the structure.

### Context menu functions for data sources

For single data sources in the [Outline](#) view, you have the following functions:

- copy  
Use [Copy](#) to copy the selected data source.
- paste  
Use [Paste](#) to paste a copied data source. The system displays the pasted data source immediately in the [Data Sources](#) folder and automatically generates a new data source alias (for example DS\_2).
- rename  
Use [Rename](#), if you want to rename an existing data source alias.
- delete  
Use [Delete](#) to delete an existing data source.
- edit initial view  
Use [Edit Initial View...](#), if you want to change the initial view of a data source. The [Edit Initial View of <your selected data source>](#) dialog box opens.
- reset initial view  
Use [Reset Initial View...](#), if you want to reset a previously changed view of a data source. Therefore this context menu function is only activated if the initial view of the data source has been changed.
- smart paste  
This context menu function is only displayed if you have created a data source in SAP BusinessObjects Analysis, edition for Microsoft Office and you have used the smart copy function to copy it.
- find references  
Use [Find References](#), if you want to know to which components the selected data source is assigned to or in which scripts of the application the selected data source is used. The results are displayed in the [Search Results](#) view.

### Context menu functions for components

For single components in the [Outline](#) view you find the following functions:

- copy  
Use [Copy](#) to copy the selected component.
- paste  
Use [Paste](#) to paste a copied component. The system displays the pasted component immediately in the [Layout](#) folder and automatically generates a new name (for example CHART\_2).
- rename  
Use [Rename](#) if you want to rename an existing component.



- arrange

Use ► [Arrange](#) ► [Align Left](#) / [Align Right](#) / [Align Top](#) / [Align Bottom](#) ►, if you want to arrange components in a special way. To do this, select at least two components in a container (for example, in the application container). Use the context menu or the toolbar of the Design Studio to arrange them as required in the application.

#### **i** Note

Depending on the docking behaviour of the selected components and the selected alignment, the docking behavior might be different.

- arrange

Use ► [Arrange](#) ► [Distribute Horizontally](#) / [Distribute Vertically](#) ►, if you want to distribute at least three components vertically or horizontally in the application. The selected outermost components keep their positions, whereas the selected inner components are arranged evenly and are the same distance to each other. All selected components keep their docking behaviour.

- delete

Use [Delete](#) to delete an existing component.

- hide

Use [Hide](#) to hide the selected component in the editor. This function is especially useful, if your application has many container components that contain other components. The hidden component is grayed out in the [Layout](#) folder and the total number of all hidden components in the application is displayed in parentheses next to the [Layout](#) folder.

#### **i** Note

If you do not want hidden components to be displayed (grayed out) in the structure of the [Outline](#) view, press the white arrow in the upper right corner of the [Outline](#) view and deselect [Display Hidden Components](#).

If you want to display all hidden components in the application, choose [Show All Hidden Components](#) in the context menu of the [Layout](#) folder.

- show

Use [Show](#) if you want to display the hidden component in the editor. This function is only displayed for hidden components.

- find references

Use [Find References](#) if you want to know to which applications the selected component is referenced to. The results are displayed in the [Search Results](#) view.

## Context menu functions for planning objects

The following functions are available for single planning functions and sequences in the [Outline](#) view:

- copy

Use [Copy](#) to copy the selected planning sequence or planning function.

- paste

Use [Paste](#) to paste a copied planning function or sequence. The system displays the pasted planning object or sequence immediately in the [Planning Objects](#) folder and automatically generates a new name (for example PF\_2).

- rename  
Use [Rename](#) if you want to rename an existing planning function or sequence.
- delete  
Use [Delete](#) to delete an existing planning function or sequence.
- find references  
Use [Find References](#) if you want to know which applications reference the selected planning function or sequence. The results are displayed in the [Search Results](#) view.

## Context menu functions for backend connection

- copy  
Use Copy to copy the selected backend connection.
- paste  
Use [Paste](#) to paste a copied backend connection. The system displays the pasted backend connection immediately in the [Technical Components](#) folder and automatically generates a new name (for example Connection\_2).
- rename
- delete  
Use [Rename](#) if you want to rename an existing backend connection.
- find references  
Use [Find References](#) if you want to know which applications reference the selected backend connection. The results are displayed in the [Search Results](#) view.

## Context menu functions for context menu

The following functions are available for the technical component [Context Menu](#):

- delete  
Use [Delete](#) to delete the context menu from the application.
- find references  
Use [Find References](#) if you want to know which elements of the application reference the context menu. The results are displayed in the [Search Results](#) view.

## Context menu functions for global scripts objects

The following functions are available for the technical component [Global Scripts Object](#):

- copy  
Use [Copy](#) to copy the selected global scripts object.
- paste  
Use [Paste](#) to paste a copied global scripts object with its global script functions. The system displays the pasted global scripts object immediately in the [Technical Components](#) folder and automatically generates a new name (for example, GLOBAL\_SCRIPTS\_2).

- rename  
Use [Rename](#) if you want to rename an existing global scripts object.
- delete  
Use [Delete](#) to delete an existing global scripts object.
- create script function  
Use [Create Script Function](#) to create a new script function.
- find references  
Use [Find References](#) if you want to know which applications reference the selected global scripts object. The results are displayed in the [Search Results](#) view.

### **i** Note

If you want to change an existing global script function, right-click this function and choose [Edit](#). The [Edit Script Function](#) dialog box opens where you can make your changes.

## Context menu functions for info charts

The following functions are available for the [Info Chart](#) analytic component on the category axis only when the user runs the application:

- for dimensions or dimension members
  - sort the dimension
    - sort by key or text ascending or descending ([Text/Key Ascending](#), [Text/Key Descending](#))
  - change the display of the dimension members
    - switch between different key and text combinations ([Key](#), [Text](#), [Key and Text](#), [Text and Key](#))
    - choice of text presentation type ([Short Text](#), [Medium Text](#), [Long Text](#), [Text](#))
  - change hierarchy of the dimension
    - select hierarchy (and activate it) ([Select Hierarchy](#))
    - expand and collapse the assigned hierarchy ([Expand All](#), [Collapse All](#))
    - expand to a specific hierarchy level ([Expand to Level](#))
  - filter dimension members
    - filter the member and move the dimension to the background filter ([Keep Member](#)) (only available if user has clicked on a dimension member)
    - filter the member and leave the dimension in the drilldown ([Keep Member on Axis](#)) (only available if user has clicked on a dimension member)
    - filter the member and swap the dimension with another dimension from the free axis (only available if user has clicked on a dimension member) ([Filter Member and Swap](#))
    - open the filter to filter the dimension ([Filter Members](#))
  - change drilldown
    - add another dimension from the free axis to the drilldown ([Drilldown By](#))
    - swap the current dimension with another dimension ([Swap With](#))
    - remove the current dimension from the drilldown ([Remove Drilldown](#))

---

## Context menu functions for pdf

The following functions are available for the technical component *PDF*:

- delete  
Use *Delete* to delete the functionality to export to PDF from the application.
- find references  
Use *Find References* if you want to know which elements of the application reference the PDF technical component. The results are displayed in the *Search Results* view.

## Context menu functions for text pool

The following functions are available for the technical component *Text Pool*:

- delete  
Use *Delete* to delete the text pool from the application.
- find references  
Use *Find References* if you want to know which applications reference the selected textpool. The results are displayed in the *Search Results* view.

## Related Information

[Working with the Initial View Dialog Box for Data Source Aliases \[page 94\]](#)

[Inserting a Copied Data source from SAP BusinessObjects Analysis, Edition for Microsoft Office \[page 93\]](#)

[Working with Global Scripts Objects and Global Script Functions \[page 297\]](#)

[Using the Context Menu \(Technical Component\) \[page 60\]](#)

[Using Planning Functions and Sequences \(Automated Planning\) \[page 145\]](#)

[Exporting to PDF \[page 125\]](#)

[Enabling Export to PDF in Analysis Applications \[page 126\]](#)

## 6.2 Using the Context Menu (Technical Component)

The technical component *Context Menu* enables the application user to navigate and analyse data at runtime. At design time, the context menu is automatically added to every new application, but can be removed if not required. Each application can have only one context menu. If you want to remove the context menu from the application, right-click *CONTEXT\_MENU* in the *Technical Components* folder of the *Outline* view and choose *Delete*.

Depending on the context, the following menu options are available:

- for dimensions or dimension members

- sort the dimension
  - sort by key or text ascending or descending (*Text/Key Ascending*, *Text/Key Descending*)
  - sort by hierarchy if a hierarchy is assigned to a dimension (*Sort by Hierarchy*)
- change the display of the dimension members
  - switch between different key and text combinations (*Key*, *Text*, *Key and Text*, *Text and Key*)
  - choice of text presentation type (*Short Text*, *Medium Text*, *Long Text*, *Text*)
- choose which display attributes are shown in the result set
- change hierarchy of the dimension
  - select hierarchy (and activate it) (*Select Hierarchy*)
  - expand and collapse the assigned hierarchy (*Expand All*, *Collapse All*)
  - expand to a specific hierarchy level (*Expand to Level*)
- switch the totals display of the dimension (*show totals*, *hide totals*, *Hide Totals if Only One Member*)
- filter dimension members
  - open the filter to filter the dimension (*Filter Members*)
  - clear the current filter (*Select All Members*)
  - define filters to get the TOP N or Bottom N values of a specified dimension based on the measure values (*Filter my Measure*)
  - filter the member and move the dimension to the background filter (*Keep Member*) (only available if user has clicked on a dimension member)
  - filter the member and leave the dimension in the drilldown (*Keep Member on Axis*) (only available if user has clicked on a dimension member)
  - filter the member and swap the dimension with another dimension from the free axis (only available if user has clicked on a dimension member) (*Filter Member and Swap*)
- change drilldown
  - add another dimension from the free axis to the drilldown (*Drilldown By*)
  - swap the current dimension with another dimension (*Swap With*)
  - remove the current dimension from the drilldown (*Remove Drilldown*)
  - swap the rows axis and the columns axis with each other (*Swap Axes*)
- for attributes
  - sort the dimension according to this attribute
  - change the member display of the attribute
- for result set data cells
  - If the query is input-enabled and the data cell is input-enabled, the cell can be locked (*Lock Value*)
- for structure members
  - sort the dimension members according to the measure values (*Measure Value Ascending* or *Measure Value Descending*)
  - change the drilldown (see the context menu entries for dimensions above)
  - filter the structure (see the context menu entries for dimensions above)
  - change the number format of the data cells belonging to this structure member
    - change the scaling factor (*Scaling Factor*)
    - change the decimal places (*Decimal Places*)
    - specify whether units and currencies are displayed in the crosstab and where they are displayed  
(*Units and Scaling Factors* > *Display Both in Header/Display Units in Data Cells/Do Not Display*)
  - add a calculation or change the totals calculation mode
    - add a new calculation based on one available measure ( *Add Dynamic Calculation* > *[Operator]* ).

- change the totals calculation mode (▮ *Calculate Totals As* ▸ *[Operator]* ▮)
- add a calculation by using two or more available measures as operands and an operator ▮ *Add Calculation* ▸ *[Operator]* ▮)
- navigation to jump targets (*Jump to*)  
 If Report-Report Interface targets (RRI targets) are specified for the query, the *Jump to* menu entry is displayed. If elements of the query have specific targets, the context menu for the jump targets contains an additional entry: *More...*, which retrieves these specific targets and displays them to the user, thus enabling the user to navigate to these targets.

## Related Information

[Creating a Generic Analysis Template for RRI Jump Targets \[page 304\]](#)

[Calculating New Measures at Runtime \[page 62\]](#)

### 6.2.1 Calculating New Measures at Runtime

Based on measures that are available in your crosstab, you can calculate new measures. There are two types of calculations.

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 an operand and you create a new measure based on this operand. For example, if you use sales volume per region as the 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. The newly created measures are added to the crosstab. You can edit the name and delete measures by using the context menu for these measures.

## Related Information

[Adding \(Simple\) Calculations \[page 63\]](#)

[Adding Dynamic Calculations \[page 64\]](#)

## 6.2.1.1 Adding (Simple) Calculations

### Context

By adding a (simple) calculation, you can create a new measure based on two or more available measures in your crosstab. The menu entry *Add Calculation* in the context menu of the crosstab is only available if the *Selection Type* property of the crosstab component has been set to *Multi* at design time.

### 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. In the context menu, choose ► *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 the value 80 and operand 2 has the 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 as a percentage. For example, operand 1 has the value 20 and operand 2 has the 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.

A measure value 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 query definition, the value is displayed as -200 in the crosstab. For calculations in Design Studio, the value 200 is used. If you select this checkbox, the value -200 is used for calculations.

## 6.2.1.2 Adding Dynamic Calculations

### Context

By adding a dynamic calculation, you can create a new measure based on one measure in your crosstab that works as operand.

### Procedure

1. Select the measure heading that you want to use.
2. Choose ► *Add Dynamic Calculation* ► *[Operator]* .

The following operators are available:

Table 5:

Operator	Description
<i>Moving Minimum Value</i>	<p>The new measure displays the lowest value available up to this point.</p> <p>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.</p>
<i>Moving Maximum Value</i>	<p>The new measure displays the highest value available up to this point.</p>
<i>Accumulative Sum</i>	<p>The new measure displays the sum of all values up to this point.</p> <p>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.</p>
<i>Accumulative Sum of Rounded Values</i>	<p>The new measure displays the sum of all rounded values up to this point.</p>
<i>Accumulative Count of All Values</i>	<p>The values per measure are counted and numbered starting with 1 for the first value. If there are 8 rows with values, including zeros, which contribute to the total, the new measure displays the numbers 1 to 8 for the 8 values.</p>



Operator	Description
<i>Accumulative Count of All Values that are Not Zero, Null or Error</i>	<p>The values per measure are counted and numbered starting with 1 for the first value, excluding values that are equal to zero.</p> <p>For values that are equal to zero, the last number is repeated.</p>
<i>Moving Average</i>	<p>The new measure calculates the average of all values up to this point.</p> <p>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.</p>
<i>Moving Average that is Not Zero, Null or Error</i>	<p>The new measure calculates the average of all values up to this point, excluding values that are equal to zero.</p>
<i>Rank Number</i>	<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>
<i>Olympic Rank Number</i>	<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 lowest 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 lowest value rank.</p>
<i>Percentage Contribution</i>	<p>The new measure calculates the percentage contribution of a value in the original measure to the overall result of the original measure.</p>

## Results

The new measure is added to the crosstab.

### 6.2.2 Creating Filters by Measure

By using the *Filter by Measure* entry in the context menu on a dimension in the crosstab, you can define filters to get the Top N or Bottom N values of a specified dimension, based on their measure values.

This means the filter is applied to the members of the selected dimension and does not affect totals or subtotals in your crosstab.

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 crosstab, some of these newly added members could match the Top 5 criteria and replace some of the previously displayed members.

## Prerequisites

This function is only available for SAP BW data sources or data sources based on SAP HANA HTTP connections via SAP HANA Info Access Service (InA).

Table 6:

SAP BW data sources	SAP HANA data sources
<p>The menu entry <i>Filter by Measure</i> is only available in the context menu if there is only a measure structure contained in the query.</p> <p>Therefore Filter by Measure is not available in the following scenarios:</p> <ul style="list-style-type: none"><li>queries with two structures</li><li>queries with no structure</li><li>queries with no structure and a measure restriction in the fixed filter (selection space)</li><li>queries with one non-measure structure (and a measure restriction in the fixed filter)</li></ul>	<p>SAP HANA data sources (via InA / HTTP connection) only offer a restricted feature set:</p> <ul style="list-style-type: none"><li>Only one measure based filter is supported per data source. Therefore the menu entry is automatically deactivated as soon as the data source has one measure-based filter and will only be activated again if this filter is removed</li><li>If one dimension in the drilldown (rows or columns) contains an active hierarchy, measure-based filters are not supported. Therefore this menu entry is deactivated.</li></ul>

## Creating a measure-based filter

- Right-click on a dimension header or dimension member in your crosstab and choose *Filter by Measure*. A new dialog box opens.
- In the dialog box, perform the following steps:
  - Choose the measure that you want to use as the basis for filtering.
  - Choose the operator (*Top N* or *Bottom N*).
  - Enter a value for the operator (filter criteria), for example, 5 for Top 5.
  - Choose *OK*.

A new measure-based filter is created for the dimension you have chosen. This means that the filter only filters the values for this dimension, for example, you filter the Top 5 customers based on their revenue by choosing the customer dimension.

If you have additional dimensions in the drilldown, they are not filtered.

---

## Editing a measure-based filter

1. Right-click on a dimension header or dimension member in your crosstab.
2. Chooses ► *Filter by Measure* ► *Edit* ► to edit an existing measure-based filter. A new dialog box opens.
3. Choose the measure, the operator and value for filter criteria according to your needs.
4. Choose OK. The measure-based filter is changed according to your input.

## Deleting a measure-based filter

1. Right-click on a dimension header or dimension member for which you have created a measure-based filter.
2. Choose ► *Filter by Measure* ► *Clear* ► to remove the measure-based filter that you have created for this dimension.

If you have multiple filters defined for multiple dimensions, you need to repeat this procedure for every dimension that contains a measure-based filter.

## (De)activating predefined conditions

Measure-based filters that were created in another tool (for example, BEx conditions created in BEx Query Designer or filters created in SAP BusinessObjects Analysis, edition for Microsoft Office) can only be (de)activated in Design Studio.

1. Right-click on a dimension header or dimension member in your crosstab.
2. Choose *Predefined Measure Filters* and click on any of the listed filters to activate or deactivate them.

# 7 Adding a Data Source

## Prerequisites

Before you can add data sources to the applications you have to create connections to BI backend, SAP HANA systems or universes containing the business data. SAP BusinessObjects Design Studio can access SAP HANA systems or SAP BW systems as BI backend systems.

## Context

You add a data source to be able to connect the various components with data. A data source can either be a query or query view of a BW system, an analytic or calculation view of an SAP HANA system (with the *Multidimensional Reporting* property set to *true*. For more information, see “Creating Analytic Views” in the SAP HANA Developer Guide on SAP Help Portal at [http://help.sap.com/hana\\_appliance](http://help.sap.com/hana_appliance) ), or a query based on a universe. The appearance of the dialog boxes can vary, depending on the chosen data source.

## Procedure

1. In the design tool, there are several methods and contexts for adding a data source to the application. Choose the method that suits the way you like to work in the design tool:
  - Click **Application** > **Add Data Source...** in the menu of the design tool.
  - Right-click **Data Sources** in the **Outline** tab of the design tool and click **New...**
  - If you have already added a component for displaying data (such as crosstab or chart) to your application, you can simply add a data source in the context of the chosen component. You do not need to assign the data source to the component in a further step:
    1. Click on the chart or crosstab component in the design area.
    2. Click on the **Data Source** property in the **Properties** view of the design tool and choose **Add...** in the menu.

The **Add Data Source** dialog box opens.

2. In the **Add Data Source** dialog box, perform the following steps:
  - a. Select a connection. For more information, see “Selecting a connection” in the *Related Topics* section.
  - b. Select a data source. For more information, see “Selecting a data source” in the *Related Topics* section.

### ➔ Tip

To view and select an entry from the history of your last ten choices, press **CTRL** + **Space** on your keyboard. You can filter the entries in the history view by typing the first letters of the required data source.

- c. *In the Data Source Alias* field, the system generates an alias for the data source by default. However, you can change the data source alias as required.

As you can use the same data source several times within one application, you work in the design tool with data source aliases as reference names.

3. Click *OK*.

## Results

You have added a data source with a data source alias as a reference name. You can now assign this data source to one or more components in your applications.

## Related Information

[Selecting a Connection \[page 721\]](#)

[Defining Connections to BI Backend Systems \[page 722\]](#)

[Selecting a Data Source \[page 71\]](#)

[Assigning a Data Source to a Component \[page 97\]](#)

## 7.1 Selecting a Connection

### Prerequisites

Before you can choose a connection, your administrator has to create OLAP data source connections to SAP HANA systems, SAP BW systems or universes containing business data.

If you want to use universe queries as data sources for analysis applications, your administrator needs to create the universes and the corresponding relational connections using the Information Design Tool (IDT) and publish them to the BI platform.

#### **i** Note

The BI platform server needs to be SAP BusinessObjects Business Intelligence 4.1. If you want to use universe queries as data sources in your analysis applications, you cannot use SAP BusinessObjects Business Intelligence 4.0.

In Design Studio you can use the following types of universes:

- relational
- single-source
- multi-source relational

For general information about universe access in SAP BusinessObjects Design Studio, see [Universe Access in Design Studio \[page 88\]](#)

If you want to create a planning application, you have to select a planning connection. For more information see [Selecting a Planning Connection \[page 143\]](#)

## Context

Connections represent BW or SAP HANA systems or universes. They are defined and configured by your administrator. Connections have to be active if you want to select a data source and use it immediately. However, you can also work with an inactive connection when designing and then validate the data sources for this connection later when the connection is active. The connection is automatically active when the backend system (SAP HANA, SAP BW, universe) is up and running.

## Procedure

1. In the [Connection](#) box, click [Browse....](#) The [Select Connection](#) dialog box opens where you can choose one of the created connections. The [Log on to <BW system>](#) or the [Log on to <SAP HANA system>](#) dialog box is displayed.

If you or your system administrator has created new data source connections on the platform (BI platform or SAP NetWeaver ) or locally, and you want these connections to be displayed in the [Select Connection](#) dialog box, press [Reload](#). The system adds your new connections to the existing connections list in the [Select Connection](#) dialog box.

2. This step depends on your choice of connection (SAP BW or SAP HANA system or universe). For a BW system, enter the client, your user credentials and the language. For an SAP HANA system, enter your user credentials and the language. If you chose a universe as your connection, you can skip step 2.

If the administrator has created a single sign-on connection for a BW system, you can connect to the BW systems without having to enter the BW username and password in the [Logon to <BW system>](#) dialog box (when inserting a BW data source in the design tool). You can change the client and the language of the BW system in this dialog box.

### Note

- Currently single sign-on for SAP HANA system connections is supported for the SAP BusinessObjects platform 4.1 but not for the 4.0 platform.
- Note that for OLAP connections of type SAP BW or SAP HANA the authentication type [Prompt](#) is not supported by Design Studio. So either choose [Pre-defined](#) or [Single-Sign-On](#) when creating an OLAP connection on the BI Platform.

## Results

You have selected a connection and can now select a data source based on this connection.

### Note

You can see all created connections in the design tool under **Tools > Preferences > Backend Connections**. To display recently defined connections in the corresponding table, click **Reload All Connections**.

## 7.2 Selecting a Data Source

### Context

You select a data source so that you can assign data to your component. Data sources can be SAP BW queries or query views, SAP HANA analytic or calculation views or universe queries. Depending on your connection choice, you have to perform different steps for BW/ and SAP HANA data sources on the one hand, and universe data sources on the other hand.

### Procedure

1. For SAP BW and SAP HANA data sources: In the **Data Source** box, use one of the following methods:
  - Type the name of a data source that you know.

#### Tip

To see the history of your last ten choices, press **CTRL** + **Space** on your keyboard.

- Type the name of a data source that you want to create later but which does not exist at the moment and click **Enter**. The system asks you to confirm that the data source does not show any data, as long as it is not active for the chosen connection, and that you can use the data source with the corresponding data source alias for further design steps. Choose the type for the data source (query, query view, InfoProvider) and click **Add**.
- Click **Browse**. The **Select Data Source** dialog box opens. Depending on the chosen connection (BW system or SAP HANA system), this dialog box will have a folder tab and/or search tab.

Option	Description
connection to BW system	<b>Folder</b> tab: You can specify if you want to look for queries or query views in the <b>InfoAreas</b> or <b>Roles</b> view. The <b>InfoAreas</b> view displays all InfoObjects (InfoAreas, InfoCubes, queries, and query views) in a tree structure. The <b>Roles</b> view displays your role-based objects in a tree structure. Select the view that suits your needs.  <b>Search</b> tab: You can search for the description or technical name of a data source. To retrieve data sources that begin with a specific string, you can type * after a partial string. You can also type a partial string without using *. The system will display any result that includes the partial string.
connection to SAP	<b>Folder</b> tab: All available data sources are displayed in a hierarchical structure. Select the one that suits your needs.

Option	Description
<b>HANA system</b>	<a href="#">Search</a> tab: You can search for the description or technical name of a data source. To retrieve data sources that begin with a specific string, you can type * after a partial string. You can also type a partial string without using *. The system will display any result that includes the partial string.

### ➔ Remember

The option of typing a data source name or browsing for a data source depends on how a connection has been configured. If the administrator has set one single data source for a connection, the system automatically displays this data source in the [Data Source](#) box. In this case, you cannot overwrite the entry or browse for another data source.

2. For universe data sources: Click [Edit Query Specification](#) to open the Query Panel. You can use the Query Panel to build, test, and preview the results of queries. You edit a query specification by adding dimensions and measures, setting filters, defining prompts, and so on. For further information about using the Query Panel, see [About the Member Selector \[page 74\]](#).
3. Click [OK](#).
4. For universe data sources: this step is optional but usually recommended.
  - a. Select and right-click the new universe data source in the [Outline](#) view.
  - b. Choose [Edit Initial View....](#) in the context menu of the new data source and define the initial geometry of your query, set additional filters, and so on.
  - a. Click [OK + Create Crosstab](#) or [OK](#) to leave the [Edit Initial View](#) dialog box. For further information about the Initial View dialog box, see [Working with the Initial View Dialog for Data Source Aliases \[page 94\]](#).

## Results

You have selected a data source that you can now assign to your component. The properties of the chosen data source (name, description and type) are displayed in the [Add Data Source](#) dialog and the properties view of the data source.

### ➔ Tip

You can easily exchange the data source: Click the symbol right to the data source name in the data source properties view. The [Exchange Data Source](#) dialog box opens where you can choose another data source.

## Related Information

[Assigning a Data Source to a Component \[page 97\]](#)



## 7.2.1 Creating a Query

### Prerequisites

This procedure assumes that you have opened the Query Panel (also referred to in the Design Studio as *Edit Query Specification* dialog box) on a published universe.

### Context

You can use this procedure to create queries and preview the results of universe queries.

For links to more detailed information about each step, see the Related Topics.

### Procedure

1. To select the objects you want to include in the query, drag objects from the universe on the left into the *Result Objects* pane.
2. For hierarchy result objects, select members to include or exclude in the results. To open the Member Selector, click the arrow to the right of the hierarchy object name.
3. To filter the results of the query, drag objects from the universe into the *Filter Objects* pane.

If a mandatory filter is defined on an object, the filter is triggered when you add the object to the *Result Objects* pane. The mandatory filter is visible in the query script, but not in the *Filter Objects* pane.

Non-mandatory, pre-defined filters are listed in the universe. You can drag these pre-defined filters into the *Filter Objects* pane to limit the results. The filter is visible in the query script.

You can also build business filters, including filters that use prompts. For detailed information, see the related topics.

4. For relational universes, you can build combined queries. To open the *Combined Queries* pane, click the *(Show/Hide) Combined Query Panel* icon in the upper left corner of the dialog box.
5. To preview the query results, click the refresh button in the *Data Preview* pane.

To change the layout of hierarchical data, click the *Result set display options* icon in the *Data Preview* pane and select an option from the list:

Option	Description
Flat layout	Displays repeated values for a level in every row.
Hierarchical layout	Displays repeated values once for a level.

6. Click *OK*. Now you can use the query as a data source for your application.

To define the initial geometry of your query and set additional filters, we recommend that you open and refine the query in the *Edit Initial View...* dialog box.

---

## Related Information

[Working with the Initial View Dialog Box for Data Source Aliases \[page 94\]](#)

[About the Member Selector \[page 74\]](#)

[Creating a Business Filter \[page 83\]](#)

[Filtering Data Using Prompts \[page 85\]](#)

## 7.2.2 About the Member Selector

The Member Selector lets you visualize and select members in a hierarchy. Use the Member Selector to:

- Select the members you want to appear in the query result set.
- Define members that will be excluded from queries.
- Define prompts to allow the selection of members to appear in the query each time you run the query.
- Select the members for a named set.
- Select the members when defining a Business Security Profile filter.

You open the Member Selector from hierarchy objects that you include in queries in the Query Panel. The Member Selector opens automatically when you edit named sets or filters for a Business Security Profile on a hierarchical business layer.

## Related Information

[Selecting hierarchy members \[page 76\]](#)

[About selecting hierarchy members \[page 74\]](#)

### 7.2.2.1 About selecting hierarchy members

In the Member Selector, you can select members in several ways:

- Select members explicitly in the hierarchy. For example, explicitly select the [California] and [Los Angeles] members of the [Geography] hierarchy.
- Select members implicitly using hierarchy relationships. For example, to select US states, you can select the child members of the [US] member.
- Select members included in a named set, for example Top Cities by Revenue, to include the cities that generate the most revenue.
- Select all members in a hierarchy level.
- Select all members up to a certain level in the hierarchy.
- Select calculated members.

The Member Selector contains three tabs:

Tab	Description
<a href="#">Members</a>	Displays the members arranged hierarchically. Use this tab to select members explicitly, by hierarchical relationships, and by specifying all members up to a given level.
<a href="#">Metadata</a>	Shows the hierarchy levels (if the hierarchy supports named levels), named sets, and calculated members.
<a href="#">Prompts</a>	Lets you define and modify prompts.


For information on how to select, display, search for, and sort hierarchy members, see the related topics.

## Related Information

[Selecting hierarchy members \[page 76\]](#)  
[Selecting members by hierarchy relationship \[page 76\]](#)  
[Selecting hierarchy members by level \[page 77\]](#)  
[Selecting named sets \[page 78\]](#)  
[Selecting calculated members \[page 79\]](#)  
[Searching for hierarchy members \[page 79\]](#)  
[Excluding hierarchy members \[page 80\]](#)  
[Defining a prompt to select members \[page 81\]](#)  
[Showing selected members in the Member Selector \[page 81\]](#)  
[Sorting hierarchy members \[page 82\]](#)  
[Setting display options \[page 82\]](#)  
[Showing estimated child count \[page 83\]](#)

## 7.2.2.2 Opening the Member Selector in the Query Panel

### Procedure



1. In the Query Panel, add the hierarchy object to the [Result Objects](#) pane.
2. To open the Member Selector, click the arrow to the right of the hierarchy object name: .
3. You can now select members in the hierarchy for inclusion or exclusion in a query. For descriptions of different ways to select members, see the related topic.

## Related Information

[About selecting hierarchy members \[page 74\]](#)

## 7.2.2.3 Selecting hierarchy members

### Procedure

1. In the Member Selector, click the [Members](#) tab to display the hierarchy members.
2. Select members in the hierarchy display.
3. To select all members in the hierarchy, click the [Select](#) icon , and select [Select All](#).
4. To select all members up to a specified level in the hierarchy, click the [Select](#) icon . You can identify the level in two ways:

Option	Description
<b>Select a named level</b>	This option is only available if the hierarchy has named levels. Select <a href="#">Select All Members until Named Level</a> and select the level from the submenu.
<b>Select a number of levels below the root</b>	Select <a href="#">Select All Members until</a> and select the number of levels from the submenu.

5. When you complete your selection, click [OK](#).

### Results

The selected members appear below the hierarchy object in the [Result Objects](#) pane of the Query Panel. When you run the query, only those members are included in the query result.

### Related Information

[Opening the Member Selector in the Query Panel \[page 75\]](#)

## 7.2.2.4 Selecting members by hierarchy relationship

### Procedure

1. In the Member Selector, click the [Members](#) tab to display the hierarchy members.
2. In the hierarchy, right-click the member for which you want to define the hierarchy relationship.
3. Select the relationship function from the menu:

### Note

*Children/Descendants* and *Parents/Ancestors* are mutually exclusive pairs. You cannot select both the children and the descendants of a member, and you cannot select both the parents and the ascendants of a member.

Relationship Function	Description
<i>Self</i>	Includes only the selected member. This is the default setting.
<i>Children</i>	Includes members one level below the selected member that have the selected member as their parent. The selected member is not included.
<i>Descendants</i>	Includes all members at all levels below the selected member. The selected member is not included.
<i>Descendants until Named Level...</i>	Includes the members at levels below the selected member until the named level you select. This option is only available if the hierarchy has named levels.
<i>Descendants until...</i>	Includes the members at levels below the selected member until the number of levels you select.
<i>Parent</i>	Includes the member that is one level above the selected member. The selected member is not included.
<i>Ancestors</i>	Includes all members at all levels above the selected member. The selected member is not included.
<i>Siblings</i>	Includes members at the same level that have the same parent as the selected member. The selected member is not included.
<i>Exclude</i>	Excludes members according to the relationship function (Self/Children/Descendants/Parent/Ancestors/Siblings).

## Related Information

[Opening the Member Selector in the Query Panel \[page 75\]](#)

### 7.2.2.5 Selecting hierarchy members by level

#### Prerequisites

To select members by level, the hierarchy must have named levels.

## Procedure

1. In the Member Selector, click the [Metadata](#) tab to display the hierarchy levels.

### Note

If the [Levels](#) folder does not display in the [Metadata](#) tab, the hierarchy is not level-based and you cannot select members by level.

2. Select levels in the [Levels](#) folder.
3. Click [OK](#).

## Related Information

[Opening the Member Selector in the Query Panel \[page 75\]](#)

### 7.2.2.6 Selecting named sets

## Prerequisites

To select members by named set, the hierarchy must have at least one named set defined. Named sets are defined in the business layer of the universe.

## Procedure

1. In the Member Selector, click the [Metadata](#) tab to display the named sets.

### Note

If the [Named Sets](#) folder does not display in the [Metadata](#) tab, the hierarchy has no named sets defined.

2. Select named sets in the [Named Sets](#) folder.
3. Click [OK](#).

## Related Information

[Opening the Member Selector in the Query Panel \[page 75\]](#)

## 7.2.2.7 Selecting calculated members

### Prerequisites

To select calculated members, the hierarchy must have at least one calculated member defined. Calculated members are defined in the business layer of the universe.

### Procedure

1. In the Member Selector, click the *Metadata* tab to display the calculated members.

#### Note

If the *Calculated Members* folder does not display in the *Metadata* tab, the hierarchy has no calculated members defined.

2. Select calculated members in the *Calculated Members* folder.
3. Click *OK*.

### Related Information


[Opening the Member Selector in the Query Panel \[page 75\]](#)

## 7.2.2.8 Searching for hierarchy members

### Context

Use the Search function in the Member Selector to select hierarchy members from a list of search results.

### Procedure

1. To open the *Member Search* dialog box, in the Member Selector *Members* tab, click the *Search* icon .
2. Enter text to search for in the *Search pattern* box.

You can use wildcards in the search:

Wildcard	Description
*	Matches any string of characters
?	Matches any one character

3. To search for text in the keys, select the [Search Keys](#) radio button.
4. Click [Search](#).
5. To select members from the search results, select the members in the [Search results](#) table.
6. Click [OK](#).

## 7.2.2.9 Excluding hierarchy members

### Procedure

1. In the Member Selector, select the members that you want to exclude.  
You can select members explicitly, by hierarchy relationship, by level, by named set, and calculated members.  
The selected members are listed in the [Summary](#) pane of the Member Selector.
2. In the [Summary](#) pane, select the [Exclude](#) option next to the members or member sets you want to exclude.
3. Click [OK](#).

### Results

Below the hierarchy object in the [Result Objects](#) pane of the Query Panel, the excluded members appear with a line drawn through the names to indicate that they are excluded from the query.

### Related Information

[Opening the Member Selector in the Query Panel \[page 75\]](#)

[Selecting hierarchy members \[page 76\]](#)

[Selecting members by hierarchy relationship \[page 76\]](#)

[Selecting hierarchy members by level \[page 77\]](#)

[Selecting named sets \[page 78\]](#)

[Selecting calculated members \[page 79\]](#)

[Searching for hierarchy members \[page 79\]](#)



## 7.2.2.10 Defining a prompt to select members

### Context

You can define a prompt to defer member selection to the time the query is run.

#### Note

When selecting members in response to a prompt, you can only select members explicitly. You cannot select members by hierarchy relationship.

### Procedure


1. In the Member Selector, click the *Prompt* tab.
2. Select *Enable Parameter* to defer member selection to when the query is run.  
You cannot access the other tabs in the Member Selector when the *Enable Parameter* option is selected.
3. Enter text for the prompt in the *Prompt Text* box.
4. If you want the prompt to select the previously-selected values by default when it displays, select *Keep last values selected*.
5. To define default values for the prompt, select *Set default values* and click *Edit*. In the *Select Parameter Values* dialog box, select default values for the prompt and click *OK*.
6. Click *OK*.

### Related Information

[Opening the Member Selector in the Query Panel \[page 75\]](#)

## 7.2.2.11 Showing selected members in the Member Selector

### Context

In the Member Selector *Members* tab, you can click the *Expand tree to show selections* icon  to show the selected members in the hierarchy display.

The display automatically expands to show the following members:

- Explicitly selected members.
- Members used to select related members. The related members implicitly selected are not necessarily shown. For example, if the member called France was used to select its children, the tree view expands to

show France. If the node France contains no explicitly selected members, the node is not expanded the show the implicitly selected children.

#### ➔ Tip

The *Expand tree to show selections* command does not collapse nodes that are already expanded. To reduce the length of the display, close all open nodes in the hierarchy display before clicking the icon.

## Related Information

[Opening the Member Selector in the Query Panel \[page 75\]](#)

### 7.2.2.12 Sorting hierarchy members

#### Context

By default in the Member Selector, hierarchy members are displayed sorted in the order they are stored in the database. To help find members in the hierarchy, you can sort the display in ascending or descending alphabetical order.

In the Member Selector *Members* tab, click the *Sort order* icon  and select the desired sort order.

The members are sorted locally in the Member Selector. Display of members in the query is not affected.

## Related Information

[Opening the Member Selector in the Query Panel \[page 75\]](#)

### 7.2.2.13 Setting display options

#### Context


By default, the Member Selector displays hierarchy member captions. You can set the display options to display unique names, or both captions and unique names.

In the Member Selector *Members* tab, click the *Member display options* icon  and select the desired display option.

## 7.2.2.14 Showing estimated child count

### Context

The Member Selector estimates the number of children for each member. By default, the estimates are hidden. You can show the estimated child count in the hierarchy display.

In the Member Selector *Members* tab, click the *Show/Hide estimated child count* icon  to toggle the display of child counts.

## 7.2.3 Filtering Data in the Query Panel

### 7.2.3.1 Creating a Business Filter

### Context

Business filters are filters based on objects in the business layer. They limit the data returned in the query.

Query filters have the following structure: filtered object, operator, operand. For example, in the following filter:

```
[Country] InList (US;France)
```

The **[Country]** dimension is the filtered object, **InList** is the operator, and the list of values (**US;France**) is the operand. The filter removes all values of **[Country]** other than **US** and **France** from the query result.

The following table describes the components of a filter:

Table 7:

Filter Component	Description
Filtered object	The filtered object is the business layer object whose values are filtered. Dimensions, attributes, measures, hierarchies, and hierarchy levels can be used as filtered objects.
Operator	The operator is used to compare the filtered object with the operand. For example, the <i>Equal To</i> operator retains only those values of the filtered object that correspond exactly to the value of the operand.
Operand	The operand supplies the value or values used to filter the filtered object.

### Procedure

1. In the *Edit Query Specification* dialog box, drag an object from the business layer to the *Filter Objects* pane. This is the filtered object.

2. In the *Filter Objects* pane, select an operator from the list.
3. In the *Filter Objects* pane, select an operand from the list.

Depending on the type of business layer and purpose of the filter, the following types of operands are available:

Table 8:

Operand type	Description
<i>Constant</i>	<p>Use the <i>Constant</i> operand to enter values directly into the filter. For example, you can use a constant to enter <b>France</b> into the filter:</p> <div style="background-color: #f0f0f0; padding: 10px; margin: 10px 0;"> <p><b>[Country]</b> Equal To <b>France</b></p> </div> <p>You can also enter the @Variable function to retrieve the value of a system variable or User Attribute. For example, to filter on the current user login name, enter the constant operand as @Variable('BOUSER'). For more information about @Variable, see the related topic.</p>
<i>List of Values</i>	<p>Use the <i>List of Values</i> operand to select values from the list associated with the filtered object. For example, if the filtered object is <b>[City]</b>, you can use the list of values to select one or more of the cities associated with the object.</p>
<i>Object</i>	<p>Use the <i>Object</i> operand to specify an object in the business layer. Drag and drop a business layer object to the operand position when defining the filter.</p> <div style="background-color: #fff9c4; padding: 10px; margin: 10px 0;"> <p><b>i Note</b></p> <p>You cannot select an object as an operand on some OLAP data sources or if the filtered object is a hierarchy.</p> </div>
<i>Prompt</i>	<p>Use the <i>Prompt</i> operand when you want to be prompted for a value when the query is refreshed. See the related topic about filtering using prompts.</p> <div style="background-color: #fff9c4; padding: 10px; margin: 10px 0;"> <p><b>i Note</b></p> <p>Prompt operands are not available if the business filter is defined for a Business Security Profile.</p> </div>

4. If you want to filter data on more than one criteria, add an additional filter by dragging another object to the *Filter Objects* pane.

By default, the filters are combined with the AND operator. To use the OR operator, double-click the box with the *And* operator.

#### **i Note**

The OR operator is not supported for OLAP data sources.

5. If you want to nest query filters, drag another business object and drop it onto an existing query filter in the *Filter Objects* pane.

Nesting query filters allows you to create more complex filter conditions than is possible when you combine filters at the same level. When you nest filters, you set the order in which they are evaluated. Nesting filters only makes sense once you have defined two filters at the same level.

## 7.2.3.2 Filtering Data Using Prompts

A prompt is a special type of query filter. It is a dynamic filter that displays a question every time you refresh the data in a query. You answer prompts by either typing or selecting the values you want to view before you refresh the data. The query then returns only the values you specified.

Prompts allow multiple users viewing a single document to specify a different sub-set of the database information and display it in the same report tables and charts. Prompts also reduce the time it takes for the data to be retrieved from the database.

When you define a prompt query filter, you can either build a new prompt, or use an existing prompt defined as a parameter in the business layer.

If you define more than one prompt in a query, you can change the order in which prompts are presented. Change prompt order in the query properties.

### Merged Prompts

When querying a business layer or universe, similar prompts are merged. For prompts to be merged, the following rules must be true:

- The prompts have the same prompt text.
- The prompts expect answers having the same data type.
- The prompts expect the same number of answers. (The number of answers to be given depends on the operator used to reference the prompt. For example, *Equal To* expects one answer. *Between* expects multiple answers.)

A single prompt message appears for merged prompts. The list of values displayed by the merged prompt is the list associated with the prompt that has the most display property constraints.

#### Note

All prompts in the query are candidates for merging: parameters defined in the business layer or data foundation, prompts defined as query filters, and prompts defined in the query expression of a business layer object with the `@Prompt` function.

### Related Information

[Creating a New Prompt to Filter Data \[page 86\]](#)

[Using an Existing Prompt to Filter Data \[page 87\]](#)

## 7.2.3.2.1 Creating a New Prompt to Filter Data

### Prerequisites

This procedure assumes you are creating a business filter in the *Edit Query Specification...* dialog box (also called Query Panel).

### Procedure

1. Drag the object you want to filter with a prompt and drop it onto the *Query Filters* pane.  
The query filter appears in outline in the *Query Filters* pane. The outline shows the filtered object, the operator and the type of filter applied to the object. (By default the filter is a constant.)
2. Select the filter operator from the list.

#### Note

The list of available operators depends on the type of filtered object.

3. Click the arrow at the right of the outline query filter and select *Prompt* from the menu to filter the object using a prompt.  
The *Edit Prompt* dialog box appears and the *New Parameter* option is selected by default.
4. Edit the prompt question in the *Prompt Text* box.
5. Select *Prompt with List of Values* to allow the user to select from a list of values when answering the prompt.  
The option is only available if the filtered object has an associated list of values in the universe.
6. Select *Select only from list* to restrict the user choice to values from the list of values.  
You can select this option only if the *Prompt with List of Values* option is selected.
7. Select *Keep last values* if you want the prompt to propose the last value that the user selected on the previous refresh. The first time the query is run, the default value (if set) is proposed.
8. Select *Optional prompt* to make the prompt optional. If the user does not supply a value for an optional prompt, the prompt is ignored.
9. Select *Set default values* if you want the prompt to propose values by default when it displays.
  - a. To enter or select the default values, click *Edit*.
  - b. If the filter object has an associated list of values, select the default values from the list.
  - c. If the filter object has no associated list of values, enter default values.
  - d. Click *OK* to save the default values.
10. Click *OK* to save the new prompt definition.

---

## Related Information

[Creating a Business Filter \[page 83\]](#)

### 7.2.3.2.2 Using an Existing Prompt to Filter Data

## Prerequisites

This procedure assumes you are creating a business filter in the [Edit Query Specification...](#) dialog box (Query Panel).

## Procedure

1. Drag the object on which you want to apply a prompt and drop it onto the [Query Filters](#) pane. The query filter appears in outline in the [Query Filters](#) pane.
2. Select the filter operator from the list.

#### Note

The list of available operators depends on the type of filtered object.

3. Click the arrow at the right of the Query Filter and select [Prompt](#) from the menu.
4. In the [Edit Prompt](#) dialog box, select the [Use Universe Parameter](#) option.
5. Select an existing parameter.

The list displays only those universe prompts that are compatible with the object you are filtering. For example, the filtered object and the universe prompt must have the same data type.
6. Click [OK](#) to save the prompt definition.

## Related Information

[Creating a Business Filter \[page 83\]](#)

---

## 7.3 Universe Access in Design Studio

### Supported Platform Release

SAP BusinessObjects Design Studio supports universe access for BI Platform 4.1 and 4.2.

### Supported Universe Types

With SAP BusinessObjects Design Studio, you can use the following types of universes:

- relational
- single-source
- multi-source relational

### General Workflow

Select a universe as a connection in the [Add Data Source](#) dialog box, define the query using the query panel in the [Edit Query Specification](#) dialog box and edit the initial layout of the query by using the [Edit Initial View...](#) dialog box.

The query definition in the query panel determines the data to be fetched using the universe. After the initial fetch, the user works with a local copy of the data held in memory for this session and the data is mapped to an OLAP-like result set. The initial fetch also determines the basis for needed projection functions (see the measures and the maximum result set sections below). The initial layout displayed is defined so that a measure dimension with all measures is created and moved to the column axis. All dimensions and attributes are moved to the free axis. You can change the initial layout with the initial view editor, but the standard restrictions of Design Studio with regards to layouts apply.

### Business Layer

- General business objects settings  
You can define the access level for every object in the universe. The access level specifies which objects the user can work with. Every object can have an associated list of values. These values are used for prompts (see the prompts section below). Restrictions on whether the object can be used in results/conditions/sort are only relevant for the query definition itself. They have no effect on how the objects can be used in the initial view editor or when executing an analysis application.
- Attributes  
Attributes are interpreted in the same way as dimension objects. This allows you to use attributes regardless of their dimensions in the resulting application.
- Measures



The user can define projection functions for measures. These projection functions are also used by default when navigating/aggregating in the OLAP-like result set after the initial fetch. The following projection functions are supported:

Table 9:

Average	Data is aggregated with average
Count	Number of occurrences is calculated
Max	Data is aggregated with maximum
Min	Data is aggregated with minimum
None	No aggregation available; if data needs to be aggregated, no value is displayed
Sum	Data is aggregated with summation

- Value help (list of values)  
Value helps defined in the universe are only used in the prompt context in Design Studio. They have no effect on other areas where member listing is performed.  
Only keys are supported in the value help. For example, if you have defined an LOV based on a query or static list with multiple columns in the business layer, only the key column is used for displaying the list of values.
- Supported data types  
Fields with the following data types are supported for usage in Design Studio:
  - Date
  - DateTime
  - LongText
  - Numeric
  - String
- Format handling  
Formats defined in the business layer for a business object are not supported. The formatting of objects depends on the viewing locale and the data type exposed by the universe for example, date formats, numeric formats. Timestamp data is displayed in the context of the user's timezone. For example, if you have defined a year using numeric as the data type, the year might be displayed as 2.013 ("," is the thousands separator). To avoid this, you can change the data type to String in your business layer.
- Navigation path  
Navigation path definitions are not interpreted.

## Query specification

You cannot change query properties. This includes settings like [Retrieve duplicate rows](#), predefining contexts and the order of prompts.

The system retains the sorting definitions from the query when possible. However, in some cases, the definitions can be edited using the initial view editor, in order to guarantee the required sorting.

## Execution

- Prompts

Prompts in Design Studio can appear in different situations:

- A new parameter is created in the query specification (query filter)
- A universe parameter is used in the query specification (query filter)
- In the query specification, you use a filter object, which contains a prompt
- A business layer object contains a prompt
- The data foundation contains a prompt

In addition, prompts can occur in the case of contexts and a value help needs a prompt (see value help chapter above)

A prompt can have an associated list of values. You select a value from a value help. If you set the property *Select only from list*, values that do not exist in the list of values are not allowed. The property *Keep last value* has no effect; the last value is always kept - as in the Design Studio session. You can set a prompt to optional. If the functionality of the @prompt business layer is used, you can define more settings in the prompt. Here only the selection modes Mono, Multi, and Leaf are allowed.

- Maximum Result Set

The result set has a fixed limit of 20000 rows or 200000 data cells. If either limit is exceeded, no data is displayed.

- Filter display

In Design Studio you can display the static filter values with script functions. However, only filter selections that produce a Cartesian product are displayed.

## 7.4 Selecting Data from a Data Source for Charts and SDK Extensions

### Prerequisites

You have added chart or SDK extensions to your application and a data source and assigned the data source to the charts or SDK extensions.

#### Note

The data selection for one data source is only available for charts and SDK extensions.

### Context

You can assign multiple charts or SDK extensions to a single data source. Each chart (or SDK extension) can display a different subset of the data contained in a data source, which holds the superset of the displayed data. For example, a data source could contain data for multiple regions (Europe, America, etc). You can then use different charts; each chart displays the data from a different region. To do this, you use the *Data Selection*

property of the chart component. This also allows you to reduce the number of data sources in an application and thereby improve the application performance.


## Procedure

1. Choose *Data Selection* in the components properties sheet. The *Select Data from ...* dialog box appears where the result set of the data source is displayed.
2. Specify your selection(s).
  - for charts: Select the rows and columns from the result set you want to be displayed in your component and choose *Show*. Your selection is now active.
  - for SDK extensions: The options of data selections offered in the dialog box depend on the used type of data-bound properties. Make your selection (s) to activate it (them).

### Note

Several types of data-bound properties allow you to restrict the selection of data values from a result set. Data-bound property types also help the SDK framework to check the feasibility of your selection and restrict the available selections in the *Select Data from...* dialog box. The following data-bound property types are available:

Table 10:

Property Type	Data Values
ResultCell	A single data value
ResultCellList	A single row or column of data values
ResultCellSet	A complex selection of data values from rows and columns (a Cartesian selection)  <div> <b>Note</b> For the ResultCellSet property type, there are limitations of selection options in the <i>Select Data from...</i> dialog box.</div>
ResultSet	All data values of the result set

For further information, see “SDK Extensions and Data Binding” in the *Developer Guide: Design Studio SDK* at <http://help.sap.com/boad>.

3. optional: If you want to delete your selection(s), choose the selection in the *Active Selections* area and choose *Delete*.
4. Click *OK*, to close the *Select Data from...* dialog box. The selected data is now displayed in the component.

---

## Related Information

[Chart \[page 338\]](#)

## 7.5 Deleting a Data Source

### Procedure

1. Choose the data source you want to delete in the [Data Sources](#) folder of the [Outline](#) view.
2. Right-click the chosen data source, and click [Delete](#).  
The system removes the corresponding data source in the [Data Source](#) folder.

### Results

You have deleted a data source. The system automatically removes this data source in all components that use the data source for data binding in the corresponding application. Other applications that use the same data source are not affected. If you want to use this data source again, click the [Undo](#) symbol in the toolbar.

---

## 8 Inserting a Copied Data source from SAP BusinessObjects Analysis, Edition for Microsoft Office

### Prerequisites

To use a data source created in SAP Business Objects Analysis, edition for Microsoft Office, the following prerequisites apply:

- Make sure that you have installed SAP BusinessObjects Analysis, edition for Microsoft Office and SAP BusinessObjects Design Studio on your machine.
- Make sure that you use both tools in either the local mode or with a BI platform. Otherwise you cannot insert an exported data source.

### Context

The edition for Microsoft Office enables you to create data sources using methods that are not available in the Design Studio you can use calculations, conditional formatting and exceptions to change a data source for example. You may then want to use this changed data source in the design studio.

### Procedure

1. In the edition for Microsoft Office: Once you have changed and saved the data source, right-click *Smart Copy* in the table. The data source is copied to the clipboard.
2. In the Design Studio, there are two ways to insert this data source:
  - Right-click the *Data Sources* folder (or any data source listed in this folder) in the *Outline* view and click *Smart Paste*. The data source in your clipboard is inserted into the *Data Source* folder. You can now assign this data source to components in the editor.
  - Right-click the *Layout* folder (or any element listed in this folder) in the *Outline* view and click *Smart Paste*. The data source in your clipboard is automatically inserted into a crosstab component and displayed in the editor.

## 9 Working with the Initial View Dialog Box for Data Source Aliases

The *Initial View* dialog box allows you to modify the initial state of any data source (BW, SAP HANA,...). If you have added a data source to your application, choose *Edit Initial View* or *Reset Initial View* (if you have already changed the initial view of the data source) in the context menu of the data source alias in the Outline view. In this dialog box, you can change the following data in the initial state of a data source:

- dimensions in the rows/columns and the order of dimensions
- measures displayed in the result set
- dimensions in the background filter
- display of total rows (above the members) in the rows axis and display of total columns (left of the members) in the column axis

For each measure you can change:

- the number of decimal places displayed
- the scaling factor used
- how totals are calculated
- the sorting order

### Note

This note is relevant for charts only. The number of decimal places displayed is generated based on the value in the first cell of each measure. If the first cell of a measure is empty, no decimal format can be generated for that measure in the *Edit Initial View* dialog box. To apply formatting to your measures in this case, you should use the format options available in the chart area *Additional Properties* tab. You can select format options from the *Format* dropdown list within the *Data Label*, *Tooltip*, *X-Axis* and *Y-Axis* additional properties.

For each dimension you can change:

- the additional result set attributes displayed
- the active hierarchy (if available) and its initial expansion level
- the member to be filtered (filter members)
- the filter by input string
- the member presentation (text, key, text/key, key/text)
- the totals display mode (show, hide, hide if only one member)
- the sorting type (by member, by attribute, by hierarchy)

### Note

The options only appear if you have added the attributes or activated existing hierarchies for the dimension, for example.

For each data source you can change the following global settings by expanding the *Global Data Source Settings* tile:

- display of negative values
  - -X
  - X-
  - (X)
- display of zero values
  - default (with currency and unit)
  - without currency/unit
  - custom

If you choose this option, you can change the display of zero values according to your needs.

### Caution

After you have modified the initial state of a data source using the [Initial View](#) dialog box, changes on the data source in the backend system will either result in runtime crashes, or the changes to the query might not be visible in your analysis application. It is also generally not possible to detect that changes in the backend have been made. The only solution to this is to reset the initial view (in the context menu of the data source in the [Outline](#) view) and recreate it from scratch.

## Live preview

You can see the effects of your changes in a live preview in a crosstab. The view is refreshed after each change.

## Pause refresh

If several small changes need to be made at the same time, you can pause all browser refreshes using the checkbox [Pause Refresh](#) in the upper right corner of the dialog box. Reactivate rendering once all changes have been made.

## Interaction within the dialog box

You can drag and drop the elements onto the trees on the left (structure, columns, rows, background filter), but not onto the live preview. You can also use the context menu for each element.

## Leaving the dialog box

If you want to leave the dialog box, you can:

- Click [Cancel](#) to discard your changes.
- Click [OK](#) to apply all modifications to the data source in your application.

- 
- Click **OK + Create Crosstab** to create a new crosstab component with the modified data source assigned

## Reset the initial view

If you have already modified the initial state of a data source using the [Edit Initial View...](#) dialog box, the context menu command [Reset Initial View](#) is enabled. When you select this command, all modifications made to the original state of the data source are reset. This means BW queries are displayed as specified in the BW system/Query Designer. With an SAP HANA data source, the initial view will contain highly aggregated measures, with no dimensions in the rows or columns.

## Related Information

[Chart Area Additional Properties \[page 343\]](#)



# 10 Assigning a Data Source to a Component

## Prerequisites

Before assigning a data source to a component, make sure you have completed the following steps:

- You have added at least one data source to your application. The data source is displayed in the [Data Sources](#) folder of the [Outline](#) view.
- You have added at least one component for displaying data (such as crosstab or chart) to your application.

## Context

By assigning a data source to a component you supply the component with data.

## Procedure

There are several ways to assign a data source to a component:

- Work in the [Outline](#) view: Drag the data source from the [Data Sources](#) folder and drop it onto the corresponding component in the [Layout](#) folder.
- Drag the data source from the [Data Sources](#) folder and drop it onto the corresponding component in the design area of the tool.

### Note

If you have added a data source by clicking the [Data Source](#) property of the component, you do not need to assign the data source to the component in a further step. In this case the system assigns the data source automatically to the component.

## Results

You have assigned a data source to a component. The component displays the data of the data source in the layout editor.

### Note

Switch to the [Data Binding](#) view in the Outline view to see which components have been assigned to a specific data source.

# 11 Binding the Properties of Standard Components to Data Sources

Data-bound components (also referred to as analytic components) - like crosstabs, charts or filter components and many SDK components - have a [Data Source](#) property that points to a data source. Some of these components also have one or more [Data Selection](#) properties that describe the subset of data to be used.

Unlike analytic components, basic components do not have a data source property. If you want basic components to display data from a data source, this must be set by scripting. While data-bound components are updated automatically when data is changed, it might be significantly more difficult to synchronize basic components in the same way. In this case, you have to know all the relevant events where a script has to update the basic component's properties.

An easier way to achieve this is to bind a property of a basic component in your application to a data source, by using the property-binding function in the [Properties](#) view of this component.

The [Binding](#) column in the [Properties](#) view displays a plus symbol for every property which can be bound (but currently is not bound). Properties without a plus symbol in the binding column cannot be bound at all. Pressing the plus symbol creates a new [Source](#) binding (and expands it automatically in the property sheet). In this way, you can create analysis applications without using scripts.

## i Note

In the special case of the [Items](#) property of a selection component (for example, List Box, Dropdown Box, Radio Button Group, Checkbox Group), you can also create a [Target](#) binding that in turn has an effect in the other direction: It filters the data source.

The Target binding for selection components can also be performed by dragging and dropping data sources (excluding SDK data sources) from the [Outline](#) view onto a selection component (in both the [Outline](#) view and the editor area). The dialog box [Create Filter Binding](#) is displayed where you have to select the dimension to bind. The source and target binding are configured automatically. In this way, the selection component can now act as a filter component for that data source and dimension).

Usually, when using drag and drop of data sources on selection components, only small adaptations are necessary in the property sheet (for example, setting the [All Members Text](#) property). Alternatively, you can set every property for the target binding manually in the [Properties](#) sheet.

After creating a binding, the plus symbol turns into a trashcan symbol that allows you to remove the binding again.

In the following chapters, you will find workflows for using the property-binding function:

- Displaying Cell Values
- Configuring a List Box for Filtering Data
- Configuring an SDK Component to Use Multiple Data Sources

## Related Information

---

[Displaying Cell Values \[page 99\]](#)

[Configuring a List Box for Filtering Data \[page 100\]](#)

[Configuring an SDK Component to Use Multiple Data Sources \[page 101\]](#)

## 11.1 Displaying Cell Values

### Context

This workflow describes how to bind a data cell value to a property, in order to display a result set cell value in a text box (component [Text](#)), for example. The effect is similar to calling `datasource.getData()` ; in a script.

### Procedure

1. Add a basic component, for example the [Text](#) component, to your application.
2. Choose a simple property, for example [Text](#) in the [Property](#) view, for this component.
3. Click on the plus symbol behind the [Text](#) property (in the [Binding](#) column of the [Properties](#) view). Clicking the plus symbol creates a new [Source](#) binding and expands it automatically in the property sheet).
4. Set the following properties for the [Source](#) binding:
  - [Type](#) (value [Data Cell Binding](#))  
This property is set automatically.
  - [Data Source](#)  
(optional) If there is more than one data source in your application, choose the data source that you want to bind from the [Data Source](#) dropdown box.
  - [Selected Data Cell](#)  
In the [Select Data](#) dialog box, select a single data cell that you want to bind from the table displayed or enter a selection expression into the input field. You can either select a data cell or toggle members to change the effective selection. And click [OK](#).  
The binding property (for example the [Text](#) property of the [Text](#) component) of the selected component changes to [<bound>](#). If the value does not fit your needs, proceed using a formatter function.
  - [Formatter Function](#)  
Click on the dropdown box of the [Formatter Function](#) and choose [Add](#). The [Create Script Function](#) dialog box is displayed.
5. Choose the Global Scripts Object (if you have more than one in the application) and type in a name for the script function. Choose [OK](#).
6. A dialog box is displayed. Write a function body using the given arguments `value` and `formattedValue`, for example, `return value / 1000 + " $";`.

The effects of the formatter function are not displayed in the editor of the design tool and are only displayed at runtime.

### Note

You can only use those global script functions as formatter functions which were created in the property view as described above. Functions created in this way are marked as having a read-only signature. This means the user cannot add, remove or change the input parameters, which the property-binding function relies on. Global script functions created via the Outline view cannot be used as formatter functions.

## 11.2 Configuring a List Box for Filtering Data

### Context

This workflow describes how to bind a List Box, Dropdown Box, Checkbox Group or Radio Button Group as a filter for a dimension. This is similar to calling `dataSource.getMembers()` ; `getMemberList()` ; and `dataSource.setFilter()` ; in some scripts.

### Procedure

1. Add a list component (for example, a List Box component) to your application.
2. In the *Property* view of this component, choose the *Items* property.
3. Click on the plus symbol in the *Binding* column of the *Items* property. The property sheet is expanded and the properties for *Source* and *Target* binding are displayed.
4. In the *Source* binding section set the *Data Source* and *Dimension* properties as required as well as all other relevant properties. The chosen dimension members are displayed in the list box component in the editor. To configure filtering, proceed with the following steps.
5. Navigate to the *Target* binding section and click on the plus symbol.
6. Choose at least one target data source (*Target Data Source*). It can be the same as the data source you used to fill the list box.
7. Choose a *Dimension* to filter. Usually this will be the same dimension as you used to fill the list box.
8. Execute the application. Choosing one or more entries in the list box will automatically filter the visualization component (for example, the crosstab).

### Note

If you want several dimension members to be selectable in the list box, set the *Selection Mode* property of the list box to *Multiple Selection*.

---

## 11.3 Configuring an SDK Component to Use Multiple Data Sources

### Context

This workflow describes how an SDK component receives data from multiple data sources. Note that this needs to be explicitly supported by an SDK component.

### Procedure

1. Add an SDK component or SDK data source that has multiple data-bound properties to your application.
2. If the component has a *Data Source* property, it can be set as default, but it can also remain set to *<None>*.
3. Choose a data-bound property and click on the plus symbol in the *Binding* column of the property. The data selection dialog box is displayed.
4. Choose the data source and the data range.
5. Repeat these steps for other data-bound properties.

---

# 12 Saving an Application

## Prerequisites

You have created a new application or changed an existing one.

## Context

You want to permanently save the changes you have made in a new/existing application.

## Procedure

Click ► [Application](#) ► [Save](#) ►. The system saves your application using the name you specified when you created it.

If you are working on several applications in different editors and you want to save all the changes you have made in one go, click ► [Application](#) ► [Save all.](#) ►

## Next Steps

If you want to save the application using a different name, use the save as function. For more information, see “Saving an application using a different name” in the *Related Topics* section.

### ➔ Tip

While you are working on your application, you can always execute it without needing to save it first. If you want to close your new/changed application, you have to save it first.

## Related Information

[Executing an Application \[page 724\]](#)

[Saving an Application Using a Different Name \[page 723\]](#)

## 13 Working with Templates

SAP BusinessObjects Design Studio offers you a set of standard (blank and predefined) and ready-to-run templates that serve various design and business needs. When you choose a template, the system automatically creates a copy of it. You can change the copy according to your needs.

You can also create user-defined templates and make them available as templates for other application designers. For more information, see [Exporting Applications As Templates \[page 309\]](#).

The rendering mode you have selected in the *New Application* dialog box determines which of the following templates are available:

- SAPUI5 mode (SAPUI5 Namespace sap.ui commons)
  - Standard Templates
    - Blank
    - Basic Analysis Layout
    - Basic Layout
    - Planning Layout
  - Ready-To-Run Templates
    - Data Discovery and Visualization
    - Generic Analysis
    - Online Composition
- SAPUI5 m (SAPUI5 Namespace sap.m)
  - Blank
  - Basic Layout

The deployment of your Design Studio, determines which of the following templates are available:

Table 11: Ready to Run Templates SAPUI5

	Local	BI Platform	SAP HANA	SAP NetWeaver
Data Discovery and Visualization	yes	yes	no	no
Generic Analysis	yes	yes	no	no
Online Composition	yes	yes	no	no

Table 12: Standard Templates SAPUI5

	Local	BI Platform	SAP HANA	SAP NetWeaver
Blank	yes	yes	yes	yes
Basic Layout	yes	yes	no	yes
Basic Analysis Layout	yes	yes	no	yes

	Local	BI Platform	SAP HANA	SAP NetWeaver
Planning Layout	yes	yes	no	yes

Table 13: Standard Templates SAPUI5 m

	Local	BI Platform	SAP HANA	SAP NetWeaver
Blank	yes	yes	yes	yes
Basic Layout	yes	yes	no	yes

## Prerequisites

Note the following points for your specific deployment of Design Studio :

- All standard templates and all ready-to-run templates are available if your Design Studio is deployed on the BI platform.
- All standard templates are available if your Design Studio is deployed on SAP NetWeaver.
- None of the standard and ready-to-run templates are available if your Design Studio is deployed on SAP HANA as a platform.

## Deployment on SAP NetWeaver

Before application designers can use the predefined templates in the design tool and make their generated analysis applications available on SAP NetWeaver, specific template files need to be uploaded to the MIME repository of SAP BW. For more information, see the chapter “Uploading Template Files to the MIME Repository” in the *Administrator Guide: SAP BusinessObjects Design Studio* on SAP Help Portal at <http://help.sap.com/boad>.

## 13.1 Using the Basic Analysis Layout Template

The Basic Analysis Layout template is a predefined Standard template used for ad-hoc slicing and dicing data sources. The template contains one crosstab, one chart and one spreadsheet by default. Drag and drop between the navigation panel and the crosstab is enabled.

There is one filter line, one navigation panel and a context menu which allow you to easily analyze your data. The feeding configurator and chart type picker enable you to design the chart.

You can easily adapt and extend the template by using a data source of your choice. To do so, open the Basic Analysis Layout template in the design tool and add a data source to the template. You can now run the template or adapt and change it.



---

When you run the template as it is, you can use the following functions:

## Switching the data visualization

To switch the visualization of your data, choose [Table](#)(crosstab), [Chart](#) or [Chart/Table](#) and [Spreadsheet](#) in the right corner of the application.

## Using the Show Designer icon

By choosing the [Show Designer](#) arrow icon, you can change the display of your data and the settings for the chart visualization. This means you can change the display of the measures and/or dimension in rows or columns by using drag and drop within the navigation panel. After you have rearranged the display of your data within the panel, press the [Pause Refresh](#) button to display the changes in the crosstab or chart. You can also see dimensions in the navigation panel that are not in the drilldown ([FREE](#)).

### Note

If you want to reverse a navigation step you have made in the application, you have the following options:

- Choose [Undo](#) (in the upper left corner of the application) to go back one navigation step; to the navigation state before you changed the application.
- Choose the entry [Back to Start](#) in the context menu, to go back to the very initial state of the application.

If you display the data in the table, you will find further drill-down and display options in the context menu of the table. The same is true for the spreadsheet.

## Using the filter line

If you want to set a filter, click on the [Add Filters](#) icon in the left of the filter line and choose one of the displayed dimensions. The filter dialog box opens where you can choose the appropriate dimension members.

To revert to the navigation state and drilldown state before the filter was applied, choose [Back to Start](#) in the context menu. It is also possible to remove each filter separately.

## Using prompts and displaying information

If your data source is designed for setting prompts, click on [Show Variable Prompts](#) in the left corner of the application to set the prompts.

Click on the [Show Application Information](#) icon in the right corner of the application to display information about your application. Here you will see for example, the technical name of your query, the prompts and filters you have set.

## Using the Actions icon

In the lower right corner of the application, you will find the [Export](#) icon, where you can use the [Export to Microsoft Excel](#), the [Export to CSV](#), or the [Export to PDF](#). Printing PDF files is not supported for the spreadsheet.

## 13.2 Using the Generic Analysis Ready-to-Run Template

The [Generic Analysis](#) template is a predefined ready-to-run template used for slicing and dicing data sources. It contains one crosstab, one chart and one spreadsheet by default. Drag and drop can be used between the navigation panel and the crosstab. A context menu is also available on the crosstab, which allows you to easily navigate and analyze your data.

The Generic Analysis Template has no data source assigned by default. You can keep the template as it is, execute it and then assign a data source at runtime by choosing a data source in the [Select Data Source](#) dialog box or by enhancing the application URL in the Web browser. This means the template remains generic and independent of any data sources. Enhance the application URL by entering the following data source URL parameter in the relevant form for the platform deployment of your Design Studio:

- for SAP NetWeaver: `&XQUERY=<TechnicalNameOfYourQuery>&XSYSTEM=<NameOfYourSystem>`

Example: `http://<applicationURL>&XQUERY=<queryID>&XSYSTEM=<systemID>`

For SAP NetWeaver deployment, we recommend that you use the Generic Analysis Template as the template for RRI jump targets. For more information, see the chapter *Creating a Generic Analysis Template for RRI Jump Targets* under the section *Advanced Design Tasks* in this guide.

- for SAP BusinessObjects Platform:

`&XSYSTEM=<cuid: YourUniqueCUID>&XQUERY=<TechnicalNameOfYourQuery>`

`http://<applicationURL>&XSYSTEM=cuid:<sysID>&XQUERY=<queryID>`

For further information on the CUID, see the *Administrator Guide: SAP BusinessObjects Design Studio based on SAP BusinessObjects BI Platform* on the SAP Help Portal at <http://help.sap.com/boad>.

### Note

For both deployments, there is also the URL parameter XTYPE. This parameter is initialized with the value "BW". However, it is also possible to use the value "HANA" and in this case the data source is a SAP HANA view.

You can also assign a data source to the template at design time when you open the template in your Design Studio. In this case, the [Load in Script](#) data binding property of the data source has to be set to [false](#).

## Switching the data visualization

To switch the visualization of your data, choose [Table](#)(crosstab), [Chart](#), [Chart/Table](#) or [Spreadsheet](#) in the right corner of the application.

## Using the Show Designer icon

By choosing the [Show Designer](#) arrow icon, you can change the display of your data and the settings for the chart visualization. This means you can change the display of the measures and/or dimension in rows or columns by using drag and drop within the navigation panel. After you have rearranged the display of your data within the panel, press the [Pause Refresh](#) button to display the changes in the crosstab or chart. You can also see dimensions in the navigation panel that are not in the drilldown ([FREE](#)).

### Note

If you want to reverse a navigation step you have made in the application, you have the following options:

- Choose [Undo](#) (in the upper left corner of the application) to go back one navigation step; to the navigation state before you changed the application.
- Choose the entry [Back to Start](#) in the context menu, to go back to the very initial state of the application.

If you display the data in the table, you will find further drill-down and display options in the context menu of the table. The same is true for the spreadsheet.

## Using the filter line

If you want to set a filter, click on the [Add Filters](#) icon in the left of the filter line and choose one of the displayed dimensions. The filter dialog box opens where you can choose the appropriate dimension members.

To revert to the navigation state and drilldown state before the filter was applied, choose [Back to Start](#) in the context menu. It is also possible to remove each filter separately.

## Using prompts and displaying information

If your data source is designed for setting prompts, click on [Show Variable Prompts](#) in the left corner of the application to set the prompts.

Click on the [Show Application Information](#) icon in the right corner of the application to display information about your application. Here you will see for example, the technical name of your query, the prompts and filters you have set.

## Using the Open Data Source Browser

By clicking the [Open Data Source Browser](#) icon, the [Select Data Source](#) dialog box is displayed where you can change your data source at runtime. You can either select a data source from the recently used data source list or from the system list. The system list varies in its appearance depending on the deployment of your Design Studio:

- If you are logged on locally to your design studio, all systems available in the SAP Log On dialog box are also displayed in the system list of the [Select Data Source](#) dialog box, together with SAP HANA connections.

- If your Design Studio is deployed on SAP NetWeaver, the BW system you are connected to is displayed in the system list.
- If your Design Studio is deployed on the BI platform, all available connections and folders, which you are authorized for, are displayed.

## Using bookmarks

If you have analyzed your data for a particular navigation status, you can set a bookmark, in order to reopen this navigation status of the query (with filter settings and prompts) at a later date, or to share it with others. To do so, select [Manage Bookmarks](#) in the right area of the application. The [Manage Bookmarks](#) dialog box opens where you can save, share or delete bookmarks. When you save bookmarks, the system automatically saves all relevant query information (such as system, query name and query type) to guarantee the uniqueness of the bookmark.

### Note

As bookmarks always relate to the chosen data source and the underlying generic analysis template prior to DS 1.6, they cannot be reused in the generic analysis template 1.6. Instead they have to be recreated.

## Using the Pin to Online Composition function (Create New Smart Object function) / portable fragment bookmarks)

The Pin to Online Composition function enables you to save portable fragment bookmarks (smart objects) with the generic analysis template. To do so, click on the [Pin to Online Composition](#) icon in the lower right corner of the application. The [Create New Smart Object](#) dialog box opens. By clicking on the [Save](#) button in the dialog box, the data visualization is saved in the table, chart, the chart/table or the spreadsheet display mode, but without the Designer area and the icon toolbar. This feature is not supported for NetWeaver deployment. Portable fragment bookmarks can be used in the Online Composition Template.

## Using the Actions icon

In the lower right corner of the application, you will find the [Export](#) icon, where you can choose the functions [Export to Microsoft Excel](#), [Export to CSV](#) or [Export to PDF](#). Printing PDF files is not supported for the spreadsheet.

## Related Information

[Creating a Generic Analysis Template for RRI Jump Targets \[page 304\]](#)

[Using the Online Composition Template \[page 109\]](#)

---

## 13.3 Using the Online Composition Template

The *Online Composition Template* is a predefined template used to allow end users to create dashboards using portable fragment bookmarks, also known as smart objects. The template comprises two pages. The first page contains a grid component that represents the saved state of the second page in the form of tiles and a menu. The second page contains the following:

- split cell
- fragment gallery
- tree component which reflects bookmark folder structure
- editable text box
- global filter
- menu

The *Online Composition Template* has no data source assigned by default. It uses the data sources stored with the bookmarks and can read them dynamically when the user runs the application.

### Creating a Composition

To create your virtual dashboard select the *Create* menu item at the bottom of the first page. This brings you to the second page, where you can drag and drop the required portable fragment bookmarks or smart objects from the *Fragment Gallery* onto the *Split Cell*. When you select *Save*, you will see your dashboard represented by a tile on the first page.

### Applying a Global Filter

You can allow the user to apply a global dimension filter across all portable fragment bookmarks or smart objects that they drop into their *Split Cell* container. Once a global filter is applied, the portable fragment bookmarks or smart objects that have been dropped into the *Split Cell* container and have these dimensions in common, will update to reflect this global filter - even if they are from different queries. All other portable fragment bookmarks dropped into the *Split Cell* container that do not have these dimensions in common will not be updated.

### Sharing your Dashboard

The template includes a way for users to share their newly created dashboards with colleagues. To share a dashboard simply select a tile and select the *Share* menu option on the first page. This opens a new window that includes a generated URL for you to share and a menu item allowing you to generate an email containing the URL.

---

## Opening a Dashboard

To open a dashboard in read mode, select a dashboard tile that you have already created and select [Open](#). This will open the dashboard in read mode.

## Deleting a Dashboard

To delete a dashboard select one from your available tiles on the first page and then select the menu option [Delete](#).

## Menu Items

- [Save As](#)  
Allows you to save changes to an existing dashboard.
- [Save](#)  
Allows you to save changes to an existing dashboard.
- [Refresh Icon](#)  
Refreshes the [Fragment Gallery](#) to make sure that you are looking at the most up to date list of portable fragment bookmarks or smart objects.
- [Cancel](#)  
Reverts you back to read mode.
- [Edit](#)  
Enables you to edit your online composition.
- [Back Icon](#)  
Reverts you back to the first page.

## Filtering the Fragment Gallery

You can populate the [Fragment Gallery](#) with different bookmarks or smart objects by selecting from the hierarchical tree of bookmark folders above the gallery. Selecting a folder from this list updates the [Fragment Gallery](#) with the bookmarks or smart objects available to you from that particular folder. This gives you a different selection of bookmarks or smart objects from which to choose, to create your online composition.

## Exporting to PDF

Select the [Export to PDF](#) button to export to PDF, a WYSIWYG version of the currently displayed page of your dashboard. You can customize the following details before exporting:

- [Page size](#)

- [Orientation](#)
- [Header Title](#)
- [Footer Text](#)

## Related Information

[Working with Online Composition \[page 150\]](#)

[Working with Fragment Gallery Components \[page 153\]](#)

[Working with Split Cell Containers \[page 155\]](#)

[Using the Generic Analysis Ready-to-Run Template \[page 106\]](#)

[Applying Global Filters \[page 156\]](#)

## 13.4 Using the Planning Layout Template



### Context

The *Planning Layout* template is a predefined standard template used for planning business data. Besides the functions of the Generic Analysis template, the Planning Layout template also has the following functions designed for planning applications:

- Switch to Display/Edit Mode
- Back to Saved State
- Planning Functions
- Refresh
- Save

The Planning Template cannot be used as a standard analysis template but has to be modified. As an application designer you need to perform the following steps after you have chosen the Planning Template in the [New Application](#) dialog box.

### Procedure

1. Assign a data source to the application using a planning query.
2. Define a planning connection in the [Properties](#) view under [Planning](#). The planning connection specifies the backend system, which the planning application refers to.
3. Depending to your planning scenario, add one or more planning functions to the application as required. For a Sales Revenue Planning application, you can add functions such as [Copy Actuals to Plan](#), [Delete Plan Data](#) and [Reevaluate Plan Data by XY Percent](#). To do this, choose  [Planning Objects](#)  [Add Planning Function...](#) in the [Outline](#) view of the Design Studio and type in or search for the planning function you need. For each planning function, add a [Script Alias](#) in the [Add Planning Function](#) dialog box. For example, PF\_COPY.

4. Now you can add the planning functions to the corresponding UI elements (*Planning Functions*). In the planning template, the planning functions are bound to buttons on the UI. This means that the planning functions you add are executed when the application user clicks on the buttons. In the design tool, you will find the corresponding buttons under the *ACTIONSHEET\_PLANNING\_FUNCTION* section in the *Outline* view:
  - a. Change the names of the planning function buttons by editing the *Text* property of each button, for example **Copy Initial Data** for the Copy planning function you have chosen in step 3.
  - b. Change the *On Click* event of each button as required by using the relevant method for the planning function. For example, for the Copy Initial Data button and the Delete button, you can add scripts like these:

```
PF_COPY. copyFilters (DS_1)
PF_COPY.execute();
```

or

```
PF_DELETE.execute();
```

If you do not know the planning function script alias, press **CTRL + Space** to see the list of all components and elements that can be used for the script.

5. If you do not use all the planning function buttons, delete the functions that you do not use in the *Outline* view.

You have made the necessary changes in the template. You can now run the application.
6. Save your changes and execute the planning template. The planning application is displayed in the planning mode by default. Click on the icon *Lock* if you want to view the data in read mode. If you want to plan data, switch back to the *Plan* mode by clicking the icon *Unlock*.
7. Optional: In the planning mode, you can perform manual planning actions such as bottom up and top down planning by entering the relevant data in the input-ready cells. To see the changes, click on *Refresh*(recalculate). Click on *Save* if you want to save the changed data in the backend.
8. To see the planning functions, click on *Planning Functions*. Use the functions as required.

## Results

You have changed the planning template according to your business needs and executed it in the Web browser.

### Note

Keep the following points in mind:

- in the display mode, the planning template behaves like the generic analysis template. This enables the application user to use the filter panel and analyze data and perform drilldowns.
- planning functions (or manual planning) can only be executed with a defined drilldown of dimensions.

## 13.5 Using the Data Discovery and Visualization Template


The *Data Discovery and Visualization Template* is used for online self-service data discovery and visualization based on SAP BW and SAP HANA data sources. Data enters the application as a dataset selected at runtime and



different chart types are selected to create a collection of visualizations based on the chosen dataset. You can create a dashboard that consists of visualizations associated with different datasets, offering a very rich graphical representation of your data. A WYSIWYG (What You See Is What You Get) version of the currently displayed page of the dashboard can be exported to PDF and the entire application can be shared via email within an organization. The template consists of two pages called *Visualize* and *Compose*.

#### Note

Before working with the *Data Discovery and Visualization Template*, you should refer to the following SAP Note:

SAP Note Number	Description
2169104 	Data Discovery and Visualization Template platform support in SAP BusinessObjects Design Studio

## Visualize Page

The *Visualize* page contains the following components:

- *Chart Type Picker*  
The *Chart Type Picker* is a palette of chart types, from which you can select the visualizations you want to use in your dashboards. You can change your chart type at runtime and immediately view your data in a different way. For more information on using the *Chart Type Picker*, you can refer to the chapter in this guide called *Chart Type Picker* within the *Basic Components* section.
- *Navigation Panel*  
Using the navigation panel, you can easily change how you drilldown into the data and see at a glance the navigational state of the data source at runtime. You can drag and drop dimensions from the *Navigation Panel* to the columns or rows area. By default all the measures are displayed in the visualization. To display only some of the measures, you must use the context menu to filter out the measures you do not need. For more information on using the *Navigation Panel*, you can refer to the chapter in this guide called *Navigation Panel* within the *Analytic Components* section.
- *Context Menu*  
The context menu offers you a variety of functions that help you create your visualizations and work with them efficiently at runtime, for example, enabling and expanding hierarchies. For more information on using the *Context Menu*, you can refer to the chapter in this guide called *Using the Context Menu (Technical Component)* within the *Working with the Outline View* section.
- *Filter Line*  
The *Filter Line* offers you a way to add, remove, view and edit filters and measures defined for your assigned data source. For more information on using the *Filter Line*, you can refer to the chapter in this guide called *Filter Line* within the *Analytic Components* section.
- *Layout Editor*  
The layout editor provides a What-You-See-Is-What-You-Get display, which enables you to view the working application during the creation process. As you change chart types or apply filters, your changes are immediately reflected in the *Layout Editor*.
- *Fragment Gallery*  
In the *Visualize* page, the *Fragment Gallery* is configured horizontally. It stores and displays portable fragment bookmarks or visualizations created by the user based on their selected dataset.

## Compose Page

The [Compose](#) page contains the following components:

- [Fragment Gallery](#)  
In the [Compose](#) page, the [Fragment Gallery](#) is configured vertically. It stores the collection of portable fragment bookmarks or visualizations, created in the [Visualize](#) page. You can drag and drop these visualizations from the [Fragment Gallery](#) into the [Split Cell](#) to create your own customized dashboards.
- [Split Cell](#)  
The [Split Cell](#) enables you to place and arrange portable fragment bookmarks, dragged from the [Fragment Gallery](#), in a tabular format.

## Selecting an Initial Data Source

The [Data Discovery and Visualization Template](#) has no data source assigned by default. You can keep the template as it is, execute it and then assign a data source at runtime. To assign a data source, run the template application and select [Recently Used Queries](#) or [Select a System](#). [Recently Used Queries](#) will display a list of all the queries you have already used to date. If there are no recently used queries listed, you can select the data source you want to assign to your template by using the [Select a System](#) panel. Once you have selected your data source from the list, select [Next](#). You can then select a query by searching or by selecting from a folder structure. Select [OK](#). The [Visualize](#) page opens.

## Changing a Data Source within an Application

You can create dashboards online using data from multiple data sources. This offers you greater flexibility and insight into your data. The [Fragment Gallery](#) displays a collection of visualizations associated with the selected dataset. However, you can display a different collection of visualizations in the [Fragment Gallery](#) in either the [Visualize](#) or [Compose](#) pages.

If you select a different dataset from the dropdown list at the top of the [Visualize](#) or [Compose](#) pages, the collection of visualizations in the [Fragment Gallery](#) changes to display the visualizations associated with the newly selected dataset. You can then manipulate those visualizations as you choose, for example, changing chart types or adding filters. Every time you change your selection in the dropdown list you are then working on the visualizations based on a different dataset.

## Deleting a Loaded Query and Associated Visualizations

You can delete a loaded query and all its associated visualizations from the [Visualize](#) page of the template. The visualizations or portable fragment bookmarks you delete from the template, will no longer appear in the available visualizations presented for selection in the [Fragment Gallery](#). Visualizations already saved in an existing dashboard will persist, even if the query associated with them is deleted from the dashboard. If you choose to delete the only query loaded in the [Visualize](#) room, you will be prompted to select a new query through the [Select Data Source](#) dialog box.

## Creating Visualizations

When you drag and drop dimensions to the rows or columns area, the default visualization in the [Layout Editor](#) changes automatically to reflect your selection. By default all the measures are displayed in the visualization. To display only some of the measures, you must use the context menu to filter out the measures you do not need. You can then select a different chart type from the [Chart Type Picker](#) and the visualization changes accordingly. To create another visualization, select the + icon in the [Fragment Gallery](#), manipulate the dimensions and measures, and select a chart type. Your visualizations are saved automatically as portable fragment bookmarks and are stored in the [Fragment Gallery](#). Images for the portable fragment bookmarks are automatically generated depending on the chart family type selected. Continue to add to your [Fragment Gallery](#) until you have created the number of visualizations required.

## Types of Visualizations

Type of Analysis	Description	Charts available
<a href="#">comparison</a>	Compares differences between values or shows a simple comparison of categorical divisions of measures. For example, use a bar chart to compare the differences in sales revenue between countries.	<ul style="list-style-type: none"><li>• bar chart</li><li>• bar chart with 2 x-axes</li><li>• column chart</li><li>• column chart with 2 y-axes</li><li>• stacked bar chart</li><li>• 100% stacked bar chart</li><li>• stacked column chart</li><li>• 100% stacked column chart</li><li>• combined bar line chart</li><li>• combined bar line chart 2 x-axes</li></ul>
<a href="#">trend</a>	Shows a trend in the data values (especially for dimensions that are time-based, such as Year) or the progression of your data and possible patterns. For example, you can use a line chart to view sales revenue trends of a product throughout a range of years.	<ul style="list-style-type: none"><li>• line chart</li><li>• horizontal line chart</li><li>• area chart</li><li>• horizontal area chart</li><li>• combined column line chart</li><li>• line chart with 2 y-axes</li><li>• combined column line chart with 2 y-axes</li></ul>
<a href="#">percentage</a>	Shows the percentage of parts in a whole or values as ratios to a whole. The legend shows the percentage and the total values. For example, use a pie chart to see who had the highest sales as part of a total sales value directly: Total sales = \$200, Paul had 10%	<ul style="list-style-type: none"><li>• pie chart</li><li>• multiple pie chart</li></ul>

Type of Analysis	Description	Charts available
	(\$20), David had 65% (\$130), and Susan had 25% (\$50)	
<i>correlation</i>	Shows the relationship between values or compares multiple measure values. For example, you can view the correlation of two measures and understand the impact of the first measure on the second measure.	<ul style="list-style-type: none"> <li>• scatter plot</li> <li>• bubble chart</li> </ul>
<i>other charts</i>	Shows a selection of other chart types available	<ul style="list-style-type: none"> <li>• radar chart</li> <li>• multiple radar chart</li> <li>• waterfall chart</li> <li>• horizontal waterfall chart</li> <li>• stacked waterfall chart</li> </ul>
<i>additional types</i>	Other types of visualizations apart from charts.	<ul style="list-style-type: none"> <li>• crosstab</li> </ul>

## Renaming a Visualization

Visualization titles are based on the query used to create them. To change the title of your visualization, select the cogwheel to the top right of the *Layout Editor* and select *Rename*. Select *Restore Default Title* to change the title back to the generated title. The generated title is based on the data source selection.

## Show or Hide Title

To hide or show the title of your visualization, you must select the menu option *Show/Hide Title* from the cogwheel to the upper right of the *Layout Editor*.

## Adding and Removing Filters

To add, remove, view and edit filters and measures defined for your assigned data source, select the **+** icon to the left of *Add Filters*. Select a dimension from the list in the dialog box. If a dimension contains a dimension hierarchy, select to expand the list to show all dimensions included in the hierarchy. The filters you select to apply to your data source will appear across the top of the *Layout Editor*. You can return to the dialog box to further modify your dimension filter, by selecting the applied filter from the *Add Filters* panel. You can remove a filter with the "X" icon beside the applied filter.

---

## Open Prompt Dialog

You can select [Open Prompt Dialog](#) from the cogwheel in the top right-hand corner of the [Layout Editor](#). It displays a list of required and optional variables. The values you select for the variables set filters that determine which data is included in a visualization. Multiple values can be selected when a variable allows for multiple values. The prompt values are saved automatically with the visualization.

## Creating Custom Applications

When you have created the collection of portable fragment bookmarks or visualizations you require, you can select the [Compose](#) page at the top of the template. The [Compose](#) page allows you to create custom dashboards using portable fragment bookmarks that are stored in the [Fragment Gallery](#). Select a title for your application and select [Create](#). You can then drag and drop portable fragment bookmarks from your vertical [Fragment Gallery](#) onto the [Split Cell](#) to create your own custom dashboard. The application is saved automatically. To create a new dashboard, simply select [New Story](#) button and repeat the process to create as many applications as you wish.

## Changing a Data Source within the Compose Page

If you select a different dataset from the dropdown list at the top of the [Compose](#) page, the collection of visualizations in the [Fragment Gallery](#) also changes to display the visualizations associated with the newly selected dataset. This allows you to then drag and drop into the [Split Cell](#) a combination of visualizations from different datasets.

## Switching Between Dashboards

To switch between applications, you can select from the dropdown list that appears at the top of the [Compose](#) page.

## Sharing a Dashboard

You can share your dashboard as a bookmark with other users by selecting the [Share](#) button. This allows you to do one of the following:

- Open the bookmark link directly in a separate browser by selecting the link icon.
- Select and copy the bookmark Url to their preferred browser. They can then use native browser functionality to add the Url to their favorites or share the bookmark by copying and pasting the Url into an email.
- Select the [Send by email](#) button to open an email directly from their current application browser. The subject field of the generated email is pre-populated with the title of both the application and the bookmark the user wants to share. The body of the email contains a copy of the actual bookmark link. The user can then edit the

---

email as required and share a bookmark of their application by sending the email to the appropriate recipients.

## Export to PDF

Select the [Export to PDF](#) button to export to PDF, a WYSIWYG version of the currently displayed page of your dashboard. You can customize the following details before exporting:

- [Page size](#)
- [Orientation](#)
- [Header Title](#)
- [Footer Text](#)

# 14 Executing an Application

## Context

In the design tool, there are several ways to execute an application:

- Choose ► [Application](#) ► [Execute Locally](#) ►.  
The application is displayed in a separate Web browser window, using a local Web server embedded within the design tool. This allows you to check the changes you made to the application before saving.
- Choose ► [Application](#) ► [Execute on BI Platform](#) ►.  
The application is displayed in a separate Web browser window, using the document link of the application and the current session of the BI platform. You do not have to log the session to the BI platform when executing the application on the BI platform. The analysis application is executed with the credentials you provided when logging on to the design tool.

### Note

If you have made unsaved changes to your application and you execute the application on the BI platform, the system informs you that the currently persisted (and probably outdated) version of the application will be displayed. You can still save your changes before executing the application.

- Choose ► [Application](#) ► [Execute on SAP Enterprise Portal](#) ►  
The application is displayed in a separate Web browser window. You are prompted to log on to SAP Enterprise Portal.
- Choose ► [Application](#) ► [Execute on SAP HANA](#) ►  
The application is displayed in a separate Web Browser window. You are prompted to log on to the SAP HANA system again.

---

# 15 Executing an Application on a Mobile Device

## Procedure

1. Click [Send to Mobile Device \(using QR code \\*\)](#). The dialog box [QR Code \\*](#) is displayed with the URL of the current application encoded.
2. If there are several network adapters active for your computer, the dropdown box [IP address to use](#) is displayed. Select the IP address you need.
3. Scan the QR code with a mobile device, for example an iPad (iPad 2 or higher), using one of the various QR code scanner apps.
  - When you click on the QR code, a new Web browser window with the document link opens, but without the current session of the platform. You have to log on to the platform. This is useful when you want to execute the application with different credentials, for example, to check access rights or personalized result sets.
  - To copy the document link to the clipboard, click the [Copy URL to clipboard](#) button. You can paste the URL into an e-mail, send the e-mail and the URL can be opened on a mobile device. If you want to add this URL to your bookmarks, you need to do this manually (and not by choosing the corresponding button on the mobile device).



## 16 Recording Applications for Offline Use

Application users can record any application workflow for offline use later on by pressing a key combination in the executed application. In general, the displayed state after each interaction step is saved during recording. After recording the application user can download the recorded steps as a .ZIP file and replay the interaction steps. During playback, it is possible to move forward or backward through the set of saved steps. Apart from moving through the steps, no other type of interaction is possible during playback.

### Prerequisites

Recording applications is possible with a Design Studio deployment on SAP NetWeaver, the BI platform or in the local mode. If you want to record applications, note the following points:

- The maximum number of recorded steps per application is 100. Once this limit is reached, the application user will be warned, and no further steps will be recorded. It will still be possible to download the currently recorded steps in this case.
- The Prompt dialog box as well as the context menu itself will not be shown during playback; any interaction steps containing the prompts dialog box are skipped. This means the playback skips from the step before the variable dialog box is shown to the step immediately after the dialog box is submitted or canceled.
- The following components do not support recording for offline use:
  - geo maps
  - all SDK components

### Enabling recording

The application user can press the key combination **Ctrl** + **Alt** + **R** to enable recording. Once the recording mode has been started, a control bar is available at the bottom of the application window. It opens when the user hovers with the mouse at the bottom of the window.



The control bar contains the following buttons (from left to right shown as in the picture above):

- Reset: Deletes all steps captured so far, but does not switch recording on or off.
- Toggle Button Recording Active: switches recording on and off. This icon changes appearance depending on whether recording is active. If recording is currently active, the icon is displayed as a pause button (like in a media player). If recording is currently inactive, the icon is displayed as a record button. This toggle button allows the user to skip steps, which are not required in the recording. When the user presses the pause button, the step immediately before the pause button was pressed is included in the recording, but the

---

pausing step itself is not. When the user presses the resume button, the resuming step is included in the recording.

- Stop: Turns off the recording feature altogether. This also removes the recording control bar and clears all recorded steps.
- Download: Downloads the recorded steps as a .ZIP file. This opens a browser download window. After the file has been downloaded, the set of captured steps is reset (the result is the same as clicking the Reset button).

## Opening the downloaded .ZIP playback file

The downloaded .ZIP file has the same name as the application. In order to start playback, the application user has to unzip the .ZIP file and open the file `index.html` contained in the ZIP file. All other files should not be directly accessed by the application user.

## Controlling playback

The playback control bar contains three buttons and a page/pages indicator:



From left to right as shown in the picture above:

- Go to first step: Displays the first recorded step.
- Previous step: Displays the step before the currently displayed step.
- Next Step: Displays the step after the currently displayed step.
- Current / Total page indicator

# 17 Printing an Analysis Application to a Browser


## Prerequisites

You have created an application and you have configured the settings for printing a copy of your application from your preferred browser. Calling `APPLICATION.print()`; and printing produces a WYSIWYG (What You See Is What You Get) version of the analysis application. This may cause truncation depending on the dimensions of the application. The following table describes the print settings (for different browsers) that need to be applied, in order to ensure that the whole application is printed.

Mode	IE(9)	Chrome	Firefox	Safari
local	<ol style="list-style-type: none"> <li>Choose <b>Settings</b> &gt; <b>Print</b> &gt; <b>Print preview</b>.</li> <li>See row of icons below <b>Print preview</b> dialog box title.</li> <li>Choose <b>Page setup</b>. You may need to adjust some or all of the following settings to ensure that the Design Studio application is printed correctly: <ul style="list-style-type: none"> <li>Select <b>portrait</b> or <b>landscape</b> as required.</li> <li>Ensure that <b>Print Background Colors</b> and <b>Images</b> are selected.</li> <li>Set all margins to 0 mm</li> </ul> </li> </ol>	<p>Open the <b>Print</b> dialog box by calling <code>APPLICATION.print()</code>:</p> <ol style="list-style-type: none"> <li>In the <b>Layout Section</b>, select <b>portrait</b> or <b>landscape</b> as required.</li> <li>In the <b>Margins</b> section, choose <b>Custom</b>.</li> <li>Hover over the <b>Page preview</b> and enter the required values for the margins in the four black boxes.</li> <li>In <b>Options</b>, ensure that <b>Background Colors</b> and <b>Images</b> are selected.</li> </ol>	<ol style="list-style-type: none"> <li>Choose <b>Menu</b> &gt; <b>Print</b> &gt; <b>Print preview</b>.</li> <li>Choose <b>Page setup</b> &gt; <b>Format &amp; Options</b> tab.</li> <li>Select the required page orientation and adjust the scale.</li> <li>Ensure that <b>Print Background Colors</b> and <b>Images</b> are selected.</li> <li>Choose <b>Margins &amp; Header/Footer</b>.</li> <li>Set all margins to 0 mm.</li> </ol>	<p>Open the <b>Print</b> dialog box by calling <code>APPLICATION.print()</code>:</p> <ol style="list-style-type: none"> <li>At the bottom of the dialog box, choose <b>Show Details</b>.</li> <li>Select the required page orientation and adjust the scale.</li> <li>Ensure that <b>Print Backgrounds</b> is selected.</li> </ol>
BI platform	Same as above.	<p><b>Note</b></p> <p>Printing in Chrome has been disabled.</p>	Same as above.	Same as above.

## Note

Printing is not supported when using the SAP BusinessObjects Mobile solution. You must read the following SAP Note before printing:

SAP Note Number	Description
<a href="#">2037087</a> 	Considerations when printing a SAP BusinessObjects Design Studio Analysis Application.

## Context

To call the browser print dialog box from an application, follow the steps below:

## Procedure

1. Create a component with an *onClick* property.
2. Include `APPLICATION.print()`; call in the *onClick* control of the component.
3. Execute the application.
4. In the application, select the component to call `print()`.

## 18 Exporting to PDF

You can use scripting methods, to allow the application user to export to PDF from their analysis applications to produce one the following export versions:

- A WYSIWYG (What You See Is What You Get) export version of the entire application screen.
- A WYSIWYG export version of a selected panel of the application.
- A WYSIWYG export version of multiple panel components of the application.
- A report style export version of all crosstabs and charts native to Design Studio, and custom SDK components contained in the application.

Exporting to PDF is supported on the following platforms - BI platform, SAP NetWeaver and local mode.

Exporting to PDF is activated through the optional *Technical Component* called *PDF*. It must be added in the *Outline* view of the designer application before this export to PDF functionality is available.

The export to PDF properties can be set in two ways:

- You can decide to set the export to PDF properties within the *PDF* technical component designer properties only. This means that when the application user selects the component scripted to call the export to PDF functionality, the application will be exported to PDF directly, without any property changes from the user.
- You can decide to allow the application user change their export to PDF property settings through a dialog box before exporting. The designer property values appear in the dialog by default. Any selections made by the user will overwrite any default values or values selected by the application designer in the *PDF* technical component designer properties.





For all export to PDF versions, you can specify orientation, paper size, header title, footer text and page number. For the report style export to PDF of all components in an application, you can also select to display optional metadata and also wrap column header text, row header text or both, for native crosstabs. You can also select not to wrap the text. An optional header image can be specified in the *PDF* technical component designer properties. This header image then automatically appears at the top of the PDF page when the application is exported to PDF.

### Note

- Legacy applications can also apply this export to PDF functionality.

### Note

Before working with the export to PDF feature, you should refer to the following SAP Notes:

SAP Note Number	Description
<a href="#">2150460</a> 	Using CSS with Export to PDF
<a href="#">2227751</a> 	Considerations when exporting to PDF in Design Studio 1.6
<a href="#">1177020</a> 	SAP BusinessObjects Design Studio - Sizing Information
<a href="#">2232036</a> 	Export to PDF - browser issue when exporting very large crosstabs

---

## Related Information

[Enabling Export to PDF in Analysis Applications \[page 126\]](#)

[WYSIWYG Export to PDF \[page 129\]](#)

[Scripting WYSIWYG Export to PDF \[page 131\]](#)

[Application Export to PDF \[page 132\]](#)

[Export to PDF Properties \[page 127\]](#)

## 18.1 Enabling Export to PDF in Analysis Applications

### Context

Exporting to PDF is activated through the optional *Technical Component* called *PDF*. It must be added in the *Outline* view of the designer application before this export to PDF functionality is available.

To activate the *PDF* technical component, follow these steps:

### Procedure

1. In the design tool, go to the *Outline* view.
2. Right-click *Technical Components* and choose *Create Child*.
3. Choose *PDF*.

### Results

The *PDF* technical component will now appear in the script editor as a global object, allowing you to script the export to PDF functionality into your application.

## Related Information

[Exporting to PDF \[page 125\]](#)

## 18.2 Export to PDF Properties

The type of export to PDF properties that can be set depends on the type of export to PDF output you choose. You can decide to allow these properties only to be set in the designer or by the application user. When the [Show Export Settings Dialog](#) is set to true, the default values or modified values selected in the designer will be presented to the application user when they run the application. The application user can then over-write these values by entering their own preferred property values, before exporting to PDF. When the [Show Export Settings Dialog](#) is set to false, the dialog is not presented to the application user when they run the application. When they call the component to export to PDF, their output is exported directly without the option of setting export to PDF properties.

### Export to PDF Properties

Table 14: Export to PDF Properties

Property	Property Value	Property Description
page settings	orientation <ul style="list-style-type: none"><li>landscape (default)</li><li>portrait</li></ul>	Sets how the export to PDF output is displayed.
	paper size <ul style="list-style-type: none"><li>a2</li><li>a3</li><li>a4 (default)</li><li>a5</li><li>letter</li><li>legal</li><li>ledger/tabloid</li></ul>	Sets the paper size of the export to PDF output. <div><b>Note</b> When using the option <a href="#">Ledger/Tabloid</a>, you must also select the orientation property value.</div>
header	image	Ellipsis button to allow you import a header image that is displayed by default in the header of the export to PDF output. <div><b>Note</b> This is always specified in the <a href="#">PDF</a> technical component designer properties only.</div>
	title	Sets the title that is displayed by default in the header of the export to PDF output.

Property	Property Value	Property Description
	show metadata <ul style="list-style-type: none"> <li>• true</li> <li>• false (default)</li> </ul>	<p>Optional information displayed in the PDF header for native charts and cross-tabs. The length of metadata text for variables, static and dynamic filters, is limited to 250 characters, or a maximum of two lines. Types of optional metadata include:</p> <ul style="list-style-type: none"> <li>• date</li> <li>• query name</li> <li>• variables</li> <li>• static filters</li> <li>• dynamic filters</li> </ul> <div> <p><b>i Note</b></p> <p>This property is available only when exporting to PDF a report style version of their application using the <code>PDF.exportApplication()</code> ; scripting method.</p> </div>
footer	text	Sets the text that is displayed by default in the footer of the export to PDF output.
	show page number <ul style="list-style-type: none"> <li>• true</li> <li>• false (default)</li> </ul>	Displays a page number in the footer of the export to PDF output.
crosstab export settings	wrap header text <ul style="list-style-type: none"> <li>• none (default)</li> <li>• column headers only</li> <li>• row headers only</li> <li>• both</li> </ul>	<p>Wraps the column or row header text in a native crosstab when the header text is too long to fit by default. The header text is wrapped over a maximum of two lines.</p> <div> <p><b>i Note</b></p> <p>This property is available only when exporting to PDF a report style version of their application using the <code>PDF.exportApplication()</code> ; scripting method.</p> </div>



Property	Property Value	Property Description
display	show export settings dialog <ul style="list-style-type: none"> <li>• true (default)</li> <li>• false</li> </ul>	<p>When set to true, <a href="#">Export Application Screen to PDF</a> dialog box is displayed to the application user to allow them to change the properties of their export to PDF output before exporting.</p> <div> <p><b>Note</b></p> <p>Property selections made by the user in this dialog box will overwrite the values selected in the designer.</p> </div> <p>When set to false, the export to PDF properties can be set in the designer only. When the user selects the component to call the export to PDF functionality, the export will occur directly, without input from the user.</p> <div> <p><b>Note</b></p> <p>This property is available in the designer only.</p> </div>

## 18.3 WYSIWYG Export to PDF

### WYSIWYG Application Export to PDF

You can export to PDF a WYSIWYG (What You See Is What You Get) version of your application. This will export to PDF a WYSIWYG version of the entire application screen, exactly as it appears on your screen. To call this export to PDF functionality, you can script a basic component using the scripting method `PDF.exportApplicationScreen()`;

You can then print your exported PDF using the print functionality of your PDF viewer of choice.

#### **Note**

If you call the `PDF.exportApplicationScreen()` ; function from within a [Popup](#) component, only the content of that [Popup](#) will be exported. Anything outside of the [Popup](#) will not be included in the WYSIWYG export to PDF.

## WYSIWYG Panel Export to PDF

Rather than export all the elements of your application, you can also select to export to PDF only the contents of a specific panel component in your application. To call this export to PDF functionality, you can script a component using the scripting method `PDF.exportPanelScreen(panel)`;

You can then print your exported PDF using the print functionality of your PDF viewer of choice.

## WYSIWYG Multiple Panel Export to PDF

Rather than export to PDF only the contents of a specific panel component in your application, you can also select to export multiple panel components. To call this export to PDF functionality, you can script a component using the scripting method `PDF.exportPanelScreen(panel)`; and input an array of panel components. This will result in the export to PDF of all the visible panel components included in the array, one panel component per page.

### Sample Code

#### Example 1

```
PDF.exportPanelScreen([PANEL_1, PANEL_2]);
```

### Sample Code

#### Example 2

```
var panels = [PANEL_1, PANEL_2];  
PDF.exportPanelScreen(panels);
```

You can then print your exported PDF using the print functionality of your PDF viewer of choice.

### Note

- Hidden components or non-active pages and tabs of *Pagebook* and *Tabstrip* components will not be exported to PDF.
- If a component to be exported to PDF shows an error, it will still be exported.

## Related Information

[Exporting to PDF \[page 125\]](#)

[Scripting WYSIWYG Export to PDF \[page 131\]](#)

## 18.3.1 Scripting WYSIWYG Export to PDF

### Context

You have created an analysis application and wish to export to PDF a screenshot of your application, a specific panel component or multiple panel components within your application, exactly as they appears on the screen. You have added the [PDF](#) technical component in the [Outline](#) view of your application.

### Procedure

1. In the scripting editor of the [On Click](#) property of a button component, add one of the following scripting methods:

Scripting method	Output
<code>PDF.exportApplicationScreen();</code>	WYSIWYG export to PDF of your application.  <b>i Note</b> If you call the <code>PDF.exportApplicationScreen();</code> function from within a <a href="#">Popup</a> component, only the content of that <a href="#">Popup</a> will be exported. Anything outside of the <a href="#">Popup</a> will not be included in the WYSIWYG export to PDF.
<code>PDF.exportPanelScreen(panel);</code>	WYSIWYG export to PDF of one specific panel components of your application.
<code>PDF.exportPanelScreen([panel_1, panel_2]);</code>	WYSIWYG export to PDF of multiple panel components of your application.

2. Set to true or false the property in the [PDF](#) technical component. Your choice will depend on whether you want to offer the application user the option of changing the export to PDF properties in the dialog box before exporting their screenshot, or if you want to allow these changes in the designer properties only.

### Results

You have scripted a basic component to enable the user to call the WYSIWYG export to PDF functionality. You have decided to whether to allow the user set the export to PDF properties in their application themselves, or to restrict this to within the designer properties only.

## Related Information

[WYSIWYG Export to PDF \[page 129\]](#)

[Enabling Export to PDF in Analysis Applications \[page 126\]](#)

## 18.4 Application Export to PDF





You can enable application users to export to PDF a report style version of their application. Native charts and crosstabs, and custom SDK components are exported to PDF with an optional header image, metadata, page number, header title and footer text. When exporting to PDF, an additional property called [Wrap Header Text](#) offers the option of wrapping the text that appears in the crosstab columns, rows or both, before exporting to PDF. You can also select not to wrap the text.

You can decide to enable the application user to change their export to PDF settings through a dialog box before exporting. The designer property values appear in the dialog by default. Any selections made by the user will overwrite any default values or values selected by the application designer in the [PDF](#) technical component designer properties. Or you can restrict the export to PDF property settings to within the PDF technical component designer properties only. The header image is always specified in the PDF technical component designer properties only. The orientation, paper size, header title, metadata, footer text, page number and header wrapping can be specified by the user in the [Export Application to PDF](#) dialog box or within the designer properties. To call this export to PDF functionality, you can script a basic component using the scripting method `PDF.exportApplication()` ;

Native charts are exported as images. Custom SDK components are exported as images. Native charts and custom SDK components are exported as one application component per page. Native crosstabs are exported with all of the data (no scrollbars), which will be represented as a formatted table in the PDF. Conditional formatting will be included in the exported PDF. For example, if a cell's background color is highlighted in red, then this will also be exported. The crosstab will be split by columns and rows where it does not fully fit onto the page. Row and column headers will be repeated for ease of use.

### Note

Before working with the export to PDF feature, you should refer to the following SAP Notes:

SAP Note Number	Description
<a href="#">2150460</a> 	Using CSS with Export to PDF
<a href="#">2227751</a> 	Considerations when exporting to PDF in Design Studio 1.6
<a href="#">1177020</a> 	SAP BusinessObjects Design Studio - Sizing Information
<a href="#">2232036</a> 	Export to PDF - browser issue when exporting very large crosstabs

---

## 18.4.1 Scripting Application Export to PDF Settings

### Context

You have created an analysis application and wish to allow the user to export to PDF their entire application, resulting in the export of one component per page. You have added the [PDF](#) technical component in the [Outline](#) view of your application.

### Procedure

1. Add `PDF.exportApplication()` ; in the scripting editor. For example, on the [On Click](#) property of a button component.
2. Header Images are optional. In the [PDF](#) technical component designer property called [Header Image](#), specify the location of the header image that will appear at the top of each page when the application is exported to PDF.

When the user calls the export functionality in their application, the [Export Application to PDF](#) dialog box is called. The user can then specify the properties of their export to PDF output before selecting the [Export to PDF](#) button in the dialog box.

### Results

You have scripted a basic component to enable the user to call the report style export to PDF functionality.

### Related Information

[Application Export to PDF \[page 132\]](#)

[Exporting to PDF \[page 125\]](#)

## 19 Exporting to SAP Lumira Desktop

You can allow the user to export their data source to a `.lums` Lumira file. This file can then be opened in SAP Lumira Desktop and used in the same way as if it had been imported from a CSV file.

The exported file is named `EXPORT_LUMSddmmyyhhmmssss`, which refers to the day, month, year, hour, minute, second and millisecond when the file was created. This level of detail is applied in order to guarantee a unique file name.

To implement this feature you can add a data source and from another component, for example, a button, call the `DS_1.export (exportType)` scripting method. When you execute in the browser, you can select the [Save As](#) option and this will save the generated `.lums` file to a local directory. In Lumira Desktop, you must select [Import to Folder](#). Select the `.lums` file and select [Yes](#) to open the dataset in Lumira Desktop.

### Note

You must read the following SAP Note before using the exporting to SAP Lumira Desktop:

SAP Note Number	Description
<a href="#">2169080</a> 	Considerations when exporting to SAP Lumira Desktop.

### Note

The functionality is supported on the BI platform, SAP NetWeaver and local mode.

## How the data is exported

The data is exported with the following changes:

- The data is flattened. All measures are moved to the columns and all dimensions are moved to the rows.
- All total values are not included.
- Values are exported without scaling factors and with up to 9 decimal places.
- Hierarchies will be output in the current selected state, but flattened.
- If the data source has been modified by filtering or removing measures via scripting, the export will reflect this.

---

## 20 Changing an Analysis Application

### Procedure

1. To open an existing application, click ► **Application** ► **Open** ⌵, select the relevant application and click **Open**.
2. Change the application according to your needs, for example, by adding, changing or removing components and data sources.
3. Save the changed application or click ► **Application** ► **Save As** ⌵ to save the changed application using a different name.

You can undo/redo any change you have made to an application by clicking the **Edit** menu actions, even after saving the application. The system updates the menu entries in the **Edit** menu dynamically according to the type of modification.

### Related Information

[Saving an Application Using a Different Name \[page 723\]](#)

## 20.1 Saving an Application Using a Different Name

### Prerequisites

You have made changes to an existing application.

### Context

You want to save the changed application using a different name. Perform one of the following steps:

### Procedure

Click ► **Application** ► **Save as...** ⌵. The **Save as** dialog box opens. The system suggests the current name for the application.

- For the BI platform mode, proceed as follows:

- Select the folder on the BI platform where you want to save your application to. Make sure that you have the relevant access rights for the chosen folder. If you encounter problems, contact your system administrator.
- In the [Name](#) box, type in a new name for the application and click [Save](#). Or select the application you want to rename in the list of existing applications and click the symbol ([Rename the application](#)). Now you can edit the relevant name directly in the list. Click [Save](#). The new application name is displayed in the outline view of the editor.
- In the list of existing applications, select the one that you want to use for your application and click [Save](#). The system asks you to confirm that you want to overwrite an existing application. Click [Overwrite](#). The chosen application name is displayed in the relevant tab in the editor.
- For the SAP NetWeaver or the local mode, proceed as follows:
  - In the [Name](#) box, type in a new name for the application and click [Save](#). The new application name is displayed in the outline view of the editor.
  - In the list of existing applications, select the one that you want to use for your application and click [Save](#). The system asks you to confirm that you want to overwrite an existing application. Click [Overwrite](#). The chosen application name is displayed in the relevant tab in the editor.

### Caution

Once you have confirmed that you want to overwrite an existing application, you cannot undo the changes.

You cannot select the name of an application that is open in another editor. Select another application name.

## Results

You have now saved the application using a new or different name.

## 20.2 Maximum Number of Steps Back or Resetting within an Application

### Prerequisites

You wish to allow users undo some steps or to revert all changes they made within their analysis application.

### Context

The scripting API can be used to allow application users to undo a configurable number of steps or revert all changes in their current analysis application. For example, if a user makes any changes to the data source, such as filtering on `KeyFigure` or `Type`, these changes can be undone or reset within the application.



## Procedure

1. Set ► *Application Properties* ► *Behavior* ► *Maximum Number of Steps Back* ► to an integer value more than zero. This property is set to zero or disabled by default. So, if you set the integer value to 10, the user will be able to undo 10 steps within their current application. The recommended maximum value is 20. Once the property is set to a value more than zero, the reset feature is then also enabled.
  2. Create an analysis application.
  3. Script a basic component, such as a *Button* using the `State.backOneStep()` ; method. This allows the user to revert back one step at a time with each selection of the basic component. The number of steps allowed, depends on the configuration of the *Maximum Number of Steps Back* application property.
  4. Script another basic component, such as a *Button* using the `State.backToStart()` ; method. This allows the user to revert all changes in the current application, back to the original state of the application.
  5. To disable the basic component for reverting back one step at a time, apply the method `State.isBackOneStepAvailable()` ;
  6. To disable the basic component for resetting the application apply the method `State.isBackToStartAvailable()` ;.
- Once the user has undone the number of steps configured by the *Maximum Number of Steps Back* application property, or already reset the application to the original state, the basic component will appear greyed out.

## Results

You have used API methods to allow users to revert changes made to their analysis application.

### Note

BW Integrated Planning write-back is not compatible with the maximum number of steps back or reset functionality described here.

### Note

Only changes made to the application in memory, such as changing a *Chart* type, can be undone or reset. Anything that persists to disk or platform, such as saving a bookmark, will not be undone or reset with this feature.

### Note

If the *Maximum Number of Steps Back* application property is set to a value greater than 20, an information message occurs and the property value is set back to the last acceptable number.

### Note

The maximum number of steps back and reset features are supported on the following platforms - BI platform, SAP NetWeaver and local. They are not supported on SAP HANA.

---

## Related Information

[Back One Step \(backOneStep\) \[page 688\]](#)

[Back To Start \(backToStart\) \[page 689\]](#)

[Is Back To Start Available \(isBackToStartAvailable\) \[page 689\]](#)

[Is Back One Step Available \(isBackOneStepAvailable\) \[page 690\]](#)

---

## 21 Deleting an Analysis Application

### Prerequisites

You have opened the application you want to delete.

### Procedure

1. Click ► *Application* ► *Delete* ►. The system asks you to confirm that you want to delete the application permanently.
2. Click *Delete*.

---

## 22 Searching in Applications

### Context

You can search within an analysis application and all its content (components, data sources, properties values and scripts) for a certain string or number.

### Procedure

1. Open the application that you want search.
2. Select in the menu of the Design Studio ► **Search** ► **Search Application...** . The **Search Application** dialog box is displayed.
3. Enter the your search string. The system displays the search results while you are typing the search string and highlights them.
4. If you want the system to perform a case-sensitive search and/or search any hidden components, select **Options** and select the respective checkbox.
5. If you want to keep the search result, press **Keep Results** in the dialog box. The results are displayed in the **Search Results** view.

---

## 23 Assigning Analysis Applications to the Mobile Category

### Context

Before users can access analysis applications on a mobile device using the SAP BusinessObjects Mobile solution, you need to assign the analysis applications to the mobile category on the BI platform.

### Procedure

1. In the BI launch pad, go to the folder that contains the analysis application that you want to assign to the mobile category.
2. Select the analysis application and click ► *More Actions* ► *Categories* ►.  
The *Categories* dialog box appears.
3. Choose the mobile category.
4. Click *OK*.

## 24 Creating Planning Applications

With SAP BusinessObjects Design Studio you can create desktop browser applications for planning business data. These planning applications support both manual and automated data entry and changes to data. The application user can enter the planning data manually in the crosstab (in cells or rows) or use planning functions and planning sequences (also known as planning objects) in SAP BW Integrated Planning to enter data automatically. For planning data, you have to use a BW backend system as the planning system.

After entering planning data, the application user can recalculate, reset or save the data. As an application designer, you integrate these functions into the planning application by using the following script methods for planning:

- clientReset
- hasClientChanges
- hasUnsavedChanges
- recalculate
- reset
- save

### Prerequisites

For creating planning applications, the following conditions must be met:

- You are using the Design Studio with BI Platform or SAP NetWeaver as the platform
- You are using a BW back end system as the planning system
- You are using a BW query or query view as a data source, which is defined as a planning query in the BEx Query Designer
- You are using a desktop browser application as planning application

For further information about administration tasks for planning applications, see “Administration tasks for Planning Applications” in the *Administrator Guide: SAP BusinessObjects Design Studio (based on SAP NetWeaver)* on the SAP help portal at <http://help.sap.com/boad>.

### Related Information

[Entering Data in the Crosstab \(Manual Planning\) \[page 144\]](#)

[Using Planning Functions and Sequences \(Automated Planning\) \[page 145\]](#)

[Selecting a Planning Connection \[page 143\]](#)

## 24.1 Selecting a Planning Connection

### Context

A planning connection (planning system) is required for all planning actions. If you have not selected a planning connection, you cannot change data, edit data or execute planning functions.

### Procedure

1. In the **Outline** view of your planning application, select the application in the structure.
2. Select [Planning Connection](#) in the **Properties** view of the application and choose a BW system from the dropdown box.
3. Select a suitable data source for the planning data by selecting [Add Data Source](#) in the **Outline** view or in the menu.

If you have not yet integrated data sources into your application, you can choose from all BW systems that provide data sources. If you have already added data sources, you can only choose between the systems already used in a data source. Although you can use data sources from several different backend systems in a planning application, there can only be ONE planning system. This means you can only plan for one system per application.

4. (optional) If you want to use a planning model that is available for the chosen planning connection, select the required model from the list of environments under the property [Planning Model](#). Under this property, you can select a planning model (Business Planning and Consolidation (BPC) environment and model) of SAP Business Planning and Consolidation, version for SAP NetWeaver, Unified. This makes it possible to create integrated planning solutions in a BW system, thereby providing high flexibility and usability for specialist users. Using this property causes the Design Studio's planning functionality on the server to behave differently. For more information, see [SAP Business Planning and Consolidation, version for SAP NetWeaver](#) on the SAP Help Portal.

### Related Information

[Selecting a Data Source \[page 71\]](#)

## 24.2 Entering Data in the Crosstab (Manual Planning)

In order to have input ready cells or rows in a crosstab, you have to bind the crosstab component to an input-enabled data source. Whether input ready cells are present also depends on the model in the BW back end and the initial view of the data source.

To enable the input readiness of the data source, you also have to use the data source alias method `configureInputReadiness`.

Input ready cells display an edit field which, when clicked on, allows the application user to enter text. When the user presses `Enter` or leaves the cell by navigating away from it, the system validates the input. If the input is invalid, the relevant input area of the cell is highlighted. If the input is valid, the entered value is formatted for the relevant data type (for example, a unit is added, the decimal display format is changed, ...). Besides selecting a cell by clicking it, the user can also use the tab key to move forward from cell to cell in a given row. If any value in an input ready cell has been modified, the first press of `Tab` validates the input, and the second press of `Tab` moves the focus to the next cell.

### Note

It is not possible to delete data in input-ready cells. When you enter a blank (empty) input, the cell is reset to its initial value. If you enter a 0 (zero) value (if permitted by the data type), the 0 is set as the new cell value.

In order to get new rows and configure their position in the crosstab, the following prerequisites must be met:

- Use a BW query or query view as a data source, which has a suitable initial view for getting new rows. For more information see “Planning Business Data Manually” on SAP Help Portal at <http://help.sap.com>
- Use the *Number of New Rows* and *Position of New Rows* properties of the crosstab. New rows can either be displayed at the *Bottom* or the *Top* of the crosstab.

In input-ready rows, it may also be possible to enter values in row header cells. These have an input help, which when clicked on, opens a value help for dimension members. When a user enters a value without using the value help, the external key of the dimension member must be entered.

## Related Information

[Configure Input Readiness \(configureInputReadiness\) \[page 574\]](#)

## 24.3 Editing Short Texts in Queries

With SAP BW 7.40 SPS 08, you can edit short texts in input-ready queries in the crosstab. This enables you to use characteristics (dimensions) of a suitable InfoProvider as key figures (measures) 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 the following example, free text entry is possible for cells in the *Comment (Amount)* column. Columns, in which you can edit short texts, do not support any settings. This means these columns cannot be sorted and they do not display a sorting icon. No formatting or sorting functions are available in the context menu.

		Amount	Comment (Amount)	Quantity	Comment (Quantity)
Cost Center	Cost Element	EUR		KG	
CCT_000001	1/CE00000001	<input type="text" value="250.00"/>	<input type="text" value="A comment"/>	<input type="text"/>	<input type="text"/>
	1/CE00000002	<input type="text" value="250.00"/>	<input type="text" value="Test"/>	<input type="text"/>	<input type="text"/>
	<b>Result</b>	<input type="text" value="500.00"/>	<b>NOT_EXIST</b>	<input type="text"/>	
CCT_000002	1/CE00000001	<input type="text" value="0.00"/>	<input type="text" value="Asd"/>	<input type="text" value="500.000"/>	<input type="text" value="good"/>
	1/CE00000002	<input type="text"/>	<input type="text"/>	<input type="text" value="500.000"/>	<input type="text" value="good"/>
	<b>Result</b>	<input type="text" value="0.00"/>	<b>Asd</b>	<input type="text" value="1,000.000"/>	<input type="text" value="good"/>
<b>Overall Result</b>		<input type="text" value="500.00"/>	<b>NOT_EXIST</b>	<input type="text" value="1,000.000"/>	<input type="text" value="good"/>
<input type="text"/>	<input type="text"/>		<input type="text"/>		<input type="text"/>
<input type="text"/>	<input type="text"/>		<input type="text"/>		<input type="text"/>

Figure 1: Example

#### Note

When editing short texts in queries, note the following points:

- Editing short texts in queries is not possible in a Design Studio deployed on SAP HANA. This is only possible for SAP BW-based data sources.
- There is no value help available for attribute fields that only allow you to enter specific values.
- You can only enter single-line texts. The application user cannot enter text with line breaks.
- Automatic text line wrapping in a cell is not supported.

For more information about editing short texts in queries and the definition in the query designer, see [Editing Short Texts in Queries](#) on SAP Help Portal.

## 24.4 Using Planning Functions and Sequences (Automated Planning)

Planning functions and sequences are planning objects defined in the Planning Modeler of SAP BW. These objects enable system-based processing and data generation. Functions can be executed immediately or in the background as a planning sequence. Various standard planning function types are delivered with the BW system. You can also define your own function types. A planning sequence is a sequence of planning functions. For more information about planning functions and sequences, see the following chapters on the SAP Help Portal:

- [Planning Functions](#)
- [Implementing a Planning Function Type](#)
- [Planning Sequence](#)

In the outline view, you can add an existing planning function or sequence for the selected planning system by choosing [Planning Models](#) [Add Planning Function...](#) or [Add Planning Sequence...](#) In the [Add Planning](#)

---

*Function* or *Add Planning Sequence* dialog box, enter the name of an existing planning function or sequence, or alternatively use the **Browse...** button. Now you can use this planning object in the events of other components in your application, for example in the OnClick event of a button component. The scripting methods relating to the planning function or planning object are listed under *PF\_\** or *PS\_\** in the content assistance of the script editor.

The following methods are available for planning functions:

- clearAllFilters
- clearFilter
- copyFilter
- execute
- getDimensionText
- getDimensions
- getFilterExt
- getFilterText
- getMemberList
- getMembers
- getVariableValueExt
- getVariableValueText
- getVariables
- setFilter
- setFilterExt
- setVariableValue
- setVariableValueExt

The following methods are available for planning sequences:

- execute
- getVariableValueExt
- getVariableValueText
- getVariables
- setVariableValue
- setVariableValueExt

## 24.5 Cell Locking

In analysis applications, you can lock input-ready cells in the crosstab to protect these cells from being changed. A locked cell cannot be changed manually when the crosstab is recalculated. Cell locking is a time-limited setting that only applies to the current user session. Locked cells are displayed as not editable with a lock icon



You can also undo cell locks.

There are two underlying implementations of locking cells in planning applications

- front-end cell locks (local cell locks in a query)  
The user sets the cell lock at runtime for the current result set. The cell locks remain in effect as long as no significant changes are made to the result set.

- back-end cell locks (global cell lock in a planning application)  
You or your administrator manage the cell locks only in the back-end system for one single query or all input-ready queries in the planning application. Locking cells in all queries of the planning application means a cell that is locked in one query can be displayed as locked in other queries and can be handled as locked.

### Caution

Once you have activated the global cell lock on the back end, the local cell lock will not work any more on the front end (Design Studio).

Remember that application users use the context menu entry *Lock Value* if they want to protect cells from being changed. They do not necessarily know which kind of cell lock implementation is used for the query.

For more information about cell locking in SAP BW, see “Cell Locks” in the SAP BW documentation on SAP Help Portal at <http://help.sap.com/>.

## 24.5.1 Front-End Cell Locks

Front-end cell locks in a query remain in effect until the application 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, the cells locks are removed:

- 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

For more information about front-end cell locking in SAP BW, see “Local Cell Locks in a Query (Front End)” in the SAP BW documentation on SAP Help Portal at <http://help.sap.com/>

## 24.5.2 Back-End Cell Locks

Back-end cell locks in a planning application allow you to use more navigation steps in a query while retaining the cells locks as front-end cell locks.

To activate the global cell lock on the back end, you or your administrator must set a parameter in the table RSADMIN. You can use the program SAP\_RSADMIN\_MAINTAIN to do this. Set the following parameter:

```
OBJECT = RSPLS_PQ_BACKEND_CELL_LOCKING
```

VALUE = X

### Caution

Once you have activated the global cell lock on the back end, the local cell lock will not work any more on the front end (Design Studio).

Cross query cell locks can be particularly suitable if a planning application contains multiple tabs or pages, 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 measures (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

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 removed:

- Changing the order of the dimensions included in a cell lock
- Changing 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
- Changing the query's dynamic filter (by restricting the variable values without restarting the query for example)
- Changing the filter using the *Prompts* dialog box
- Performing a planning function or planning sequence

For more information about back-end cell locking in SAP BW, see "Global Cell Locks in a Planning Application (Back End)" in the SAP BW documentation on SAP Help Portal at <http://help.sap.com/>.

## 24.5.3 Locking Cells

### Context

You can lock input-ready crosstab and total cells in planning applications. A locked cell cannot be changed manually when the crosstab is recalculated. The cells are locked for the current user session only. You can also unlock the cells during the session.

---

## Procedure

1. Select the crosstab cell you want to lock.

The selected cell must be input-enabled and must not have been changed since the last recalculation.

2. Open the context menu of the selected cell and choose [Lock cell](#).

If you want to unlock the cell, select it and open the context menu. You can now deselect the menu entry [Lock cell](#).

## Results

The selected cell is locked for changes and is displayed with a lock icon .

## 25 Working with Online Composition

The online composition feature allows application users to create and edit their own applications online, based on a selection of portable fragment bookmarks created by any user creating portable fragment bookmarks for the same group identifier. To create their own online applications, the user applies the drag and drop paradigm to drag portable fragment bookmarks from a [Fragment Gallery](#) component and drop them into a [Split Cell](#).

### Note

Online composition is not supported when working in [Main](#) (SAPUI5 Main) mode. For more information on SAPUI5 Main mode, you can refer to the chapter in this guide called *Using the SAPUI5 m Library*.

### Selecting Portable Fragment Bookmark Folders

The content of the [Fragment Gallery](#) can be filtered using a [Tree](#) component populated with bookmark folders. The folder structure returned in the [Tree](#) component, is based on the folder structure within the [Manage Folders](#) area of the Content Management Console on the BI platform. When the user selects one of these folders, the content of the [Fragment Gallery](#) changes to reflect the list of portable fragment bookmarks in the folder selected. If a user does not have access rights to a particular folder, they will not see it appear as an option for them to select. If a user has view rights only, they will be able to see the folder, but not overwrite the folder content in any way.

### Drag and Drop

Online composition relies on the portable fragment bookmark API methods. Portable fragment bookmarks only store the state of the container which has been expressly chosen by the application designer to be saved. As part of online composition, the [Split Cell](#) area adjusts itself to accommodate the portable fragment bookmarks as they are dropped in.

It is important to note that there are prescribed user interactions with the [Split Cell](#). An sample of those include the following:

- You can create content by dragging a portable fragment bookmark from the [Fragment Gallery](#) into a cell in the [Split Cell](#).
- You can replace the content in an existing cell by dropping a portable fragment bookmark from the [Fragment Gallery](#) into the same cell.
- When adding content, you should drop the portable fragment bookmark to the right or below an existing cell.
- If you add content to the right, your content is added as a new column in the [Split Cell](#).
- If you add content below, your content is added as a new row in the [Split Cell](#).

### Note

For more detailed information on populating the [Split Cell](#), you can refer to the chapter in this guide called *Working with Split Cell Containers*.

---

## Applying a Global Filter

You can allow the user to apply a global dimension filter across all portable fragment bookmarks that they drop into their [Split Cell](#) container. Once a global filter is applied, the portable fragment bookmarks that have been dropped into the [Split Cell](#) container and have these dimensions in common, will update to reflect this global filter - even if they are from different queries. For more information on global filters, you can refer to the chapter called [Applying Global Filters](#).

## Online composition use case:

The online composition feature hinges on the roles played by different personas within your organization. The scenario below describes a possible online composition workflow using three different personas - designer, power user and end user or consumer.

### Designer

As application designer, you can create a producer application and a consumer application and assign these applications to the same Group Identifier. You can also upload these applications to folders, which can be accessed by users with this same Group Identifier. Within the producer application, you can define the container that will be saved as a portable fragment bookmark. You can script a basic component to save portable fragment bookmarks with the `Bookmark.PortableFragmentBookmark.saveBookmark(groupIdIdentifier, ContainerComponent)`; scripting method. The portable fragment bookmarks can be added to the [Fragment Gallery](#) in the consumer application by using `getAllFragmentBookmarks(groupIdIdentifier)`; and then `FRAGMENTGALLERY_1.addItem(array);`.

### Power User

The power user creates a producer application to establish the content to be saved as portable fragments by defining the data source, filters and variable to be applied to the data. The power user can also define how this data will be visualized. These fragments created in the production application are then assigned to the appropriate bookmark folder and thereby available to any end user or consumer that shares the same Group Identifier.

### End User or Consumer

The end user opens the consumer application online and will select from the [Fragment Gallery](#) which portable fragment bookmarks they will drop into their own custom-made online application. If they wish to change the portable fragments available to them, they can select a different folder from within a [Tree](#) component. This will then populate the [Fragment Gallery](#) with a new list of portable fragment bookmarks from which to choose. The end user drags an item from the [Fragment Gallery](#) into the [Split Cell](#) and rearranges accordingly.

The end user can then save this new composition as a bookmark and call it up again by loading the bookmark.

## Using Templates

Examples of a consumer and producer applications can be found in the designer under the [Generic Analysis Ready-to-Run Template](#) and the [Online Composition Template](#).



### Note

For more information on using these templates, you can refer to the chapters called *Using the Generic Analysis Ready-to-Run Template* and *Using the Online Composition Template*.

## Online Composition Notes

### Note

Before working with bookmarks, you should refer to the following SAP Notes:

SAP Note Number	Description
<a href="#">2168984</a> 	Considerations when working with bookmarks in SAP BusinessObjects Design Studio.
<a href="#">2157239</a> 	Performing more than one free text search from your bookmark filter list will return zero items.

You must also be aware of the following when using the online composition feature:

### Note

If you save a chart component and another component in a container as a portable fragment bookmark, a multiple component icon will be generated to represent your portable fragment bookmark in the [Fragment Gallery](#). If your additional component is simply a [Textbox](#) and you expect your icon to reflect a chart, it is recommended that you do not include the [Textbox](#) within the same panel as your chart. You should put the [Textbox](#) component outside the panel. Then a chart component icon will be generated to represent your portable fragment bookmark in the [Fragment Gallery](#).

## Related Information

[Working with Fragment Gallery Components \[page 153\]](#)

[Working with Split Cell Containers \[page 155\]](#)

[Portable Fragment Bookmarks \[page 163\]](#)

[Assigning Bookmarks to Folders \[page 173\]](#)



## 25.1 Creating a Bookmark Based on a Query



You can use the script editor to assign a query and a system to a bookmarked application. This allows users running the application to distinguish between bookmarks that are associated with an application and bookmarks that are associated with the current query in the application.

When creating a bookmark, you can add an optional application identifier parameter called `appIdentifier`. This acts as a filter when calling for a list of bookmarks. Instead of returning a list of all bookmarks for an application, this added parameter allows you to give more context and to filter on bookmarks that refer to a certain query only.

When you save a bookmark, you can set a string as an attribute of the bookmark. When you then use the `getAllBookmarks` or `deleteAllBookmarks` scripting methods, you can supply that string and it is used to decide which bookmarks are listed or deleted. For example, `saveBookmark("Week5_net_sales, XSYSTEM + "_" + XQUERY);`

### Note

Before working with bookmarks, you should refer to the following SAP Notes:

SAP Note Number	Description
<a href="#">2168984</a> 	Considerations when working with bookmarks in SAP BusinessObjects Design Studio.
<a href="#">2157239</a> 	Performing more than one free text search from your bookmark filter list will return zero items.

### Note

- When adding an `appIdentifier` parameter, you must apply certain criteria, such as the following:
  - alpha-numeric characters only
  - must be at least 8 characters in length
- This parameter can be applied to any application, including the [Generic Analysis Ready-to-Run Template](#).

## Related Information

[Working with Online Composition \[page 150\]](#)

[Using the Generic Analysis Ready-to-Run Template \[page 106\]](#)

## 25.2 Working with Fragment Gallery Components

The *Fragment Gallery* is a basic component that is used to store portable fragment bookmarks for the user while they are using the online composition feature. The user can drag portable fragment bookmarks from the *Fragment Gallery* into the *Split Cell* to create their own virtual dashboard. If they wish to change the portable

fragments available to them, they can select a different folder from within a [Tree](#) component in their online application. This will then populate the [Fragment Gallery](#) with a new list of portable fragment bookmarks from which to choose. The end user can again drag an item from the [Fragment Gallery](#) into the [Split Cell](#) and rearrange accordingly.

## Configuration

The [Fragment Gallery](#) can be configured to display in the following ways:

- Single or Dual columns
- Vertical or Horizontal
- Display icons as:
  - Images
  - Images and text
  - Text only

### Note

For more information on the properties of the [Fragment Gallery](#), you can refer to the *Basic Components* section in this guide.

## Adding Portable Fragment Bookmarks to the Fragment Gallery

### Example

```
var bookmarks =  
Bookmark.PortableFragmentBookmark.getAllBookmarkInfosForApplication("groupIdentifier");  
  
bookmarks.forEach(function(element, index) {  
  
    FRAGMENTGALLERY_1.addItem(element);  
  
});
```

### Example

```
var bookmarks =  
Bookmark.PortableFragmentBookmark.getAllBookmarkInfosForApplication("groupIdentifier");  
  
FRAGMENTGALLERY_1.addItems(bookmarks);
```

#### **i** Note

1. If components are displaying off-centre after being dropped into the *Split Cell*, you must make sure that the component positioning within the bookmarked container is absolute.
2. Images for the portable fragment bookmarks are automatically generated depending on the component types in the bookmarked container. You can also apply scripting to allow a user specify an image from their own repository.

## Related Information

[Working with Split Cell Containers \[page 155\]](#)

[Working with Online Composition \[page 150\]](#)

[Portable Fragment Bookmarks \[page 163\]](#)

[Fragment Gallery \[page 406\]](#)

## 25.3 Working with Split Cell Containers

The *Split Cell* container component enables users to place and arrange portable fragment bookmarks, dragged from the fragment gallery, in a tabular format. Therefore the *Split Cell* container can only be used with the *Fragment Gallery* component and the portable fragment bookmarks, which can be created using the corresponding API methods.

#### **i** Note

The split cell container is not available for the SAP HANA mode.

#### **i** Note

When portable fragment bookmarks are inserted into a split cell container, the components contained in them may be renamed. In general, script references will be adapted to the new names. However, in case the name is given as a string, this renaming will not take place and could result in unexpected behaviour.

- working example: `PAGEBOOK_1.setSelectedPageIndex(0)`
- non-working example: `PAGEBOOK_1.setSelectedPageByName("PAGE_1")`

## Adding content to a split cell container

The split cell container component is initially always empty. If you drag an item from the fragment gallery onto the split cell container, the content of the portable fragment bookmark that this item represents is placed in a cell. This cell is added to the container. To fill the split cell container with an initial cell, you can drop an item anywhere. Depending on where you drop any other item, the layout of the container will change. If you drop an item onto the

---

top border of an existing cell, a new cell will be created above it. The same is true for left, right and bottom drops. You can also drop items onto the resize bar (see below) to create a cell between two cells. If you drop an item in the center of a filled cell, the content of this cell will be replaced.

#### ➔ Tip

Components that are stored in a fragment bookmark should be docked at all layout properties; set a numeric value for the left, top, right and bottom margin, and the value <auto> for the width and height. Otherwise the content of a split cell container cell can overlap with another cell.

## Resizing cells

At runtime the user can freely resize the cells by moving the mouse pointer between two cells. A resize bar is displayed. When the user drags the resize bar, the cells adjacent to the bar are changed in size accordingly.

## Deleting cells

If the user hovers with the mouse pointer over the top of a cell, a menu bar is displayed. To delete the cell, the user has to click the delete symbol in this bar.

## Moving cells

The user can move cells by hovering with the mouse pointer over the top of a cell until the menu bar is displayed. By dragging the bar, the user can move the cells and drop them within the container as described above.

## 25.4 Applying Global Filters

You can use a combination of scripting methods and the *Filter Line* component, to create a global filter that allows the application user to apply a filter across multiple data sources. This global filter can be dynamically added to their application.

Specifically in the area of online composition (self-service), you can allow the user to apply a global dimension filter across all portable fragment bookmarks that they drop into their *Split Cell* container. Once a global filter is applied, the portable fragment bookmarks that have been dropped into the *Split Cell* container and have these dimensions in common, will update to reflect this global filter - even if they are from different queries. All other portable fragment bookmarks dropped into the *Split Cell* container that do not have these dimensions in common will not be updated.

You can add scripting to the *On Drop* and *On Delete* events of the split-cell container, so that when the user drops a portable fragment bookmark onto the *Split Cell* container, the *Filter Line* is updated with a populated list of

dimensions associated with the data source assigned to that first portable fragment bookmark. Users can then choose from this list of dimensions the dimension they wish to use as a global filter across their online composition. The first dimension to be dropped into the split cell will be chosen as the source data source. All other data sources are target data sources. To allow the user to change the source data source used for the filter panel, you can also use scripting to populate a dropdown list of data sources added to the *Split Cell*. If the *Split Cell* container is empty, the *Filter Line* is empty.

Global filters are enabled in the *Filter Line* component in the *Data Discovery and Visualization Template* and the *Online Composition Template*.

## Related Information

[Filter Line \[page 386\]](#)

[Split Cell \[page 421\]](#)

[Using the Data Discovery and Visualization Template \[page 112\]](#)

[Using the Online Composition Template \[page 109\]](#)

## 25.4.1 Sample Scripting for Global Filters

You can script a list component display a list of all data sources assigned to the portable fragment bookmarks in the *Split Cell* container. The first data source in the list is the source data source i.e. the one used to populate the *Filter Line*. All others are target data sources.

You can also decide to display a list from which the user can select the target data source. The *Filter Line* property called *Target Data Sources* can also be used.

When a user selects a query from this list, the list of dimensions in the filter line reflects this change. All data sources from the *Split Cell* container are set as target data sources in order for the global filter selection to be applied to all. You can also script a component to allow the user to remove a target data source from the *Filter Line*.

## Getting a List of all Data Sources from the Split Cell and Populating the Filter Line

### Sample Code

```
var datasources = SPLITCELLCONTAINER_1.getDataSources();
var assignedDatasource = FILTERLINE_1.getDataSource();
if(assignedDatasource==undefined){
    if(datasources.length>0){
        var datasourcetoAssign = datasources[0];
        FILTERLINE_1.setDataSource(datasourcetoAssign);
    }
}
```

---

```
FILTERLINE_1.setTargetDataSources(datasources);  
BUTTON_5.onClick();  
var datasources = SPLITCELLCONTAINER_1.getDataSources();  
FILTERLINE_1.setTargetDataSources(datasources);  
BUTTON_5.onClick();
```

## 26 Working with Bookmarks

There are three types of bookmarks that can be applied when working in Design Studio analysis applications – standard bookmarks, fragment bookmarks and portable fragment bookmarks.

If an application user wishes to serialize the state of their entire application and persist that state in their BI platform or SAP NetWeaver backend, you can apply a standard bookmark to their analysis application. If they wish to serialize only a selected part of their analysis application, you can apply a fragment bookmark or portable fragment bookmark to their application. A user can also decide to create or edit one of their own applications online by working with the online composition feature. The online composition feature combines the functionality of the portable fragment bookmark, the [Tree](#), the [Filter Line](#), the [Fragment Gallery](#) and the [Split Cell](#) to empower the user to autonomously create their own online dashboards on demand.

All bookmark types created by users online can be assigned to a bookmark folder structure. This folder structure is created by the administrator on the BI platform in the Central Management Console and has user security applied on a folder level. Access is restricted according to the rights of individual users. Using the online composition feature, users can select a folder from a [Tree](#) component representing this folder structure to filter the portable fragment bookmarks they wish to use in their online application. If a user does not have access rights to a particular folder, they will not see it appear as an option for them to select. If a user has view rights only, they will be able to see the folder, but not overwrite the folder content in any way.



For more information on working with the online composition feature, you can refer to the chapter in this guide called *Using the Online Composition Feature*.

All bookmark types can be saved in local mode for testing and development purposes. You can use scripting methods to allow application users perform various functions online with their own standard and fragment bookmarks or with portable fragment bookmarks created by them or by other users for the same Group Identifier. All bookmark types have to be specifically called by the scripting API, unless they are being loaded from a Url.

For more information about the individual types of bookmarks, how to script for them and how to apply them, you can refer to the following chapters - *Standard Bookmarks*, *Fragment Bookmarks*, *Portable Fragment Bookmarks*, *Scripting for All Bookmark Types*.

### Note

Before working with bookmarks, you should refer to the following SAP Notes:

SAP Note Number	Description
<a href="#">2168984</a> 	Considerations when working with bookmarks in SAP BusinessObjects Design Studio.
<a href="#">2157239</a> 	Performing more than one free text search from your bookmark filter list will return zero items.

### Note

1. Bookmark Urls can only be loaded in local mode within a browser launched from the users own analysis application in the following circumstances:
  - While the designer is running

- In a different tab within the same browser
  - In a different browser as long as the designer is running and the designer session ID is manually added to the Url generated by the script method. Bookmark Urls generated in local mode do not include this transient session ID as the ID changes between designer sessions.
2. On Startup scripts and On Variable initialization do not run when loading a standard bookmark in an analysis application. This prevents startup scripts and variable initialization settings from overriding standard bookmark settings. This applies also to all bookmark types when they are being loaded via Url.


## Location of bookmarks persisted in the BW system

Bookmarks of analysis applications are persisted in the BW system in the following tables:

- RSWR\_DATA (Type = Z)
- RSWR\_DATA\_XREF
- RSAO\_T\_BOOKM\_EXT

### Note

The first two tables are also used by BEx bookmarks. For more information about the required SAP NetWeaver Support Packages, see the following SAP Note:

SAP Note Number	Description
<a href="#">2069853</a> 	Design Studio Netweaver: Bookmarking

## Related Information

[Standard Bookmarks \[page 160\]](#)  
[Portable Fragment Bookmarks \[page 163\]](#)  
[Fragment Bookmarks \[page 162\]](#)  
[Scripting for All Bookmark Types \[page 165\]](#)  
[Obsolete Bookmarks \[page 175\]](#)  
[Working with Online Composition \[page 150\]](#)  
[Tree \[page 415\]](#)  
[Applying Global Filters \[page 156\]](#)

## 26.1 Standard Bookmarks

Standard bookmarks are used to bookmark an entire analysis application. They are defined by the unique combination of analysis application, analysis application version and application user. You can apply standard bookmark scripting methods that allow the application user to serialize the state of their entire application and



then persist that state in the BI platform and SAP NetWeaver backend. A standard bookmark has to be specifically called by the scripting API. As the entire application is bookmarked, there is no requirement to select individual elements within the application. Standard bookmarks can also be saved in local mode (stored on disk) for testing and development purposes.


Standard bookmarks are always private. This means that they can only be shared by explicitly generating a bookmark Url and sending it to other users via email. Standard bookmarks are only visible via scripting to the user that created them.

They can be assigned to the bookmark folder structure created by the administrator.

Standard Bookmarks created and shared by other application users can be loaded via a Url in the browser bar in BI platform and SAP NetWeaver mode.

### Note

Before working with bookmarks, you should refer to the following SAP Note:

SAP Note Number	Description
2168984 	Considerations when working with bookmarks in SAP BusinessObjects Design Studio.

## Related Information

[Fragment Bookmarks \[page 162\]](#)

[Portable Fragment Bookmarks \[page 163\]](#)

[Scripting for All Bookmark Types \[page 165\]](#)

[Deleting Bookmarks \[page 166\]](#)

[Listing Bookmarks \[page 167\]](#)

[Saving a Bookmark \[page 170\]](#)

[Sharing a Bookmark \[page 172\]](#)

[Obsolete Bookmarks \[page 175\]](#)

[Assigning Bookmarks to Folders \[page 173\]](#)

## 26.1.1 Clearing Prompts on Standard Bookmark Load

You can keep, clear, hide and show both space and state variables when reloading a standard bookmark within your application. Each type of variable behaves differently, depending on the combination of variables in the application and how the *Prompts* application properties are set. When working with an application, you can set mandatory and optional prompts. With a mandatory prompt set, the prompts screen is called automatically. With optional prompts set, you can choose to show or hide prompts. The *Bookmark Loading* application property within the Prompts section, allows you to decide the behavior on loading a standard bookmark:

- Hide and Keep Prompts
- Hide and Clear Prompts

- Show and Keep Prompts
- Show and Clear Prompts

There are four different scenarios to consider:

- Space variables only:  
When your application contains only space variables, they can be cleared on loading a standard bookmark. If you select to show the prompt screen again, the application user can go ahead and make another selection on loading the standard bookmark.
- State variables only:  
State variables cannot be cleared on loading a standard bookmark. The [Bookmark Loading](#) application property values containing the clear prompts values will not apply to state variables. You can select to show prompts to allow application users to make another variable selection from state variables when they load the standard bookmark again.
- Space and state variables in the same application without merged prompts:  
When you have an application with a combination of space and state variables and the [Merge Prompts](#) application property is set to false, the clear prompts behavior will persist. When loading the standard bookmark, the space variable will clear and the state variable will not clear.
- Space and state variables in the same application with merged prompts  
When you have an application with a combination of space and state variables and the [Merge Prompts](#) application property is set to true, the following rules apply:
  - an application with a state data source and a state data source it behaves like a state variable.
  - an application with a space data source and a space data source it behaves like a space variable.
  - an application with a state data source and a space data source it behaves like a space variable.

## Related Information

[Properties of the Application \[page 325\]](#)

## 26.2 Fragment Bookmarks

Users may want to identify parts of their application that are to remain unchanged from a design perspective, while other parts of their application may change from time to time, such as [Text](#) fields. In order to achieve this, fragment bookmarks are used to serialize a fragment or part of an analysis application. If you wish to serialize only a selected part of an analysis application, you can choose any container component which contains the elements of the analysis application that are to be persisted. Any component or components inside the specified container, including the container itself, will be serialized.

Data sources connected to the data bound component in the selection will be included in the fragment bookmark. Any other data sources outside the serialized component will be excluded.

Fragment bookmarks can be assigned to a bookmark folder structure, which is created by the administrator.

It is important that the bookmarked container name remain unchanged and that the container persists as the host application versions change. For example, if you have saved your fragment bookmark within a [Panel](#) container, this [Panel](#) container must not be removed from the application into which you want to reload your fragment

bookmark. Fragment bookmarks are application and user specific. They are loaded only within the context of an application and so cannot be loaded standalone. Fragment bookmarks are available in local mode (stored on disk), BI platform mode (stored in CMS) and SAP NetWeaver mode (stored in SAP BW tables).

The container components that can be specified to accommodate a fragment selection include one of the following:


- grid layout
- pagebook
- panel
- tabstrip
- split cell

#### **i** Note

Fragment bookmarks created and shared by other application users can be loaded within the context of an application via a Url in the browser bar in BI platform and SAP NetWeaver mode.

#### **i** Note

Before working with bookmarks, you should refer to the following SAP Note:

SAP Note Number	Description
<a href="#">2168984</a> 	Considerations when working with bookmarks in SAP BusinessObjects Design Studio.

## Related Information

[Standard Bookmarks \[page 160\]](#)

[Portable Fragment Bookmarks \[page 163\]](#)

[Scripting for All Bookmark Types \[page 165\]](#)

[Assigning Bookmarks to Folders \[page 173\]](#)

## 26.3 Portable Fragment Bookmarks

A portable fragment bookmark is a type of bookmark that can be shared and reused between application users using a common reference called a Group Identifier.



You can decide which part of an analysis application can be saved as a portable fragment bookmark. This allows users to capture the state of a selected area of their application and make that available for use by other users in their applications. To allow other users to consume these portable fragment bookmarks in their own applications online, they must use the same Group Identifier.

Portable fragment bookmarks can be shared with other users like any other bookmark type, by using the share bookmark Url. For more information on sharing a bookmark, you can refer to the chapter in this guide called

*Sharing a Bookmark.* Portable fragment bookmarks are available in local mode (stored on disk), BI platform mode (stored in CMS) and SAP NetWeaver mode (stored in BW tables).

## Note

Before working with bookmarks, you should refer to the following SAP Notes:

SAP Note Number	Description
<a href="#">2168984</a> 	Considerations when working with bookmarks in SAP BusinessObjects Design Studio.
<a href="#">2157239</a> 	Performing more than one free text search from your bookmark filter list will return zero items.

## Note

- If there are multiple parts of the application to be saved, multiple portable fragment bookmarks must be created.
- Portable fragment bookmarks can only be loaded in the application where they were created. To load a portable fragment bookmark from a different application, refer to the chapter called *Using the Online Composition Feature*.

For further information on the online composition feature, you can also refer to the following chapters within this guide - *Working with Fragment Gallery Components* and *Working with Split Cell Containers*.

## Group Identifier

The `groupIdentifier` allows you to group or to apply a group to a set of portable fragment bookmarks. It is a common reference used to allow application users to make their portable fragment bookmarks available for consumption online by other application user groups. The Group Identifier is a way for an administrator to add another layer of authorization by ensuring that users from a particular business unit can only see portable fragment bookmarks relative to their business unit. As an application designer, you must associate all portable fragment bookmarks with a Group Identifier. User groups have access to the bookmark folders that store portable fragment bookmarks associated with the Group Identifier.

When scripting using the Group Identifier, a number of rules apply that must be followed. Whenever you enter the Group Identifier into the appropriate scripting method, a check will occur. Once the check passes, the script is executed. If the check fails, an information message appears.

The following are the rules that apply when scripting with the Group Identifier:

- a minimum of 8 characters must be used
- a minimum of 4 alphabetical characters must be used (upper or lower case acceptable)
- numbers are allowed
- underscores are allowed
- special characters are not allowed
- spaces are not allowed

## Related Information

[Working with Online Composition \[page 150\]](#)  
[Working with Fragment Gallery Components \[page 153\]](#)  
[Working with Split Cell Containers \[page 155\]](#)  
[Scripting for All Bookmark Types \[page 165\]](#)  
[Fragment Bookmarks \[page 162\]](#)  
[Sharing a Bookmark \[page 172\]](#)  
[Assigning Bookmarks to Folders \[page 173\]](#)  
[Save Bookmark \(saveBookmark\) \[page 466\]](#)



## 26.4 Scripting for All Bookmark Types

The scripting API can be used to allow application users to delete, list, load, save, share their bookmarks, and assign them to folders. For more detail, you can refer to the following chapters in this guide:

- [Deleting Bookmarks](#)
- [Listing Bookmarks](#)
- [Loading Bookmarks via Scripting](#)
- [Loading Bookmarks via Url](#)
- [Saving a Bookmark](#)
- [Sharing a Bookmark](#)
- [Assigning Bookmarks to Folders](#)

### Note

Before working with bookmarks, you should refer to the following SAP Notes:

SAP Note Number	Description
<a href="#">2168984</a> 	Considerations when working with bookmarks in SAP BusinessObjects Design Studio.
<a href="#">2157239</a> 	Performing more than one free text search from your bookmark filter list will return zero items.

## Related Information

[Deleting Bookmarks \[page 166\]](#)  
[Listing Bookmarks \[page 167\]](#)  
[Loading Bookmarks Using Scripting \[page 169\]](#)  
[Loading Shared Bookmarks Using a Url \[page 170\]](#)  
[Saving a Bookmark \[page 170\]](#)

---

[Sharing a Bookmark \[page 172\]](#)

[Assigning Bookmarks to Folders \[page 173\]](#)

## 26.4.1 Deleting Bookmarks

All bookmark types have a parent application. If the parent application is deleted, child bookmarks are deleted. You can use the scripting API to allow application users to delete their own bookmarks. The following scripting is used, depending on the type of bookmark involved:

### Deleting Standard Bookmarks

The following scripting is used to allow the application user to delete their own standard bookmarks:

- `Bookmark.deleteBookmark(id);`
- `Bookmark.deleteAllBookmarks();`

### Deleting Fragment Bookmarks

The following scripting is used to allow the application user to delete their own fragment bookmarks:

- `Bookmark.FragmentBookmark.deleteBookmark(id);`
- `Bookmark.FragmentBookmark.deleteAllBookmarks();`

### Deleting Portable Fragment Bookmarks

The following scripting is used to allow the application user to delete their own portable fragment bookmarks:

- `Bookmark.PortableFragmentBookmark.deleteBookmark(id)`
- `Bookmark.PortableFragmentBookmark.deleteAllBookmarks(groupIdIdentifier)`

#### **i** Note

1. Application users cannot delete bookmarks created by other application users.
2. Deleting a bookmark does not automatically remove it from a selection component. The selection component needs to be scripted separately to allow for the list of bookmarks to be refreshed after a bookmark is deleted.

---

## Related Information

[Delete All Bookmarks \(deleteAllBookmarks\) \[page 448\]](#)

[Delete All Bookmarks \(deleteAllBookmarks\) \[page 455\]](#)

[Delete Bookmark \(deleteBookmark\) \[page 456\]](#)

[Delete Bookmark \(deleteBookmark\) \[page 449\]](#)

[Delete All Bookmarks \(deleteAllBookmarks\) \[page 461\]](#)

[Delete Bookmark \(deleteBookmark\) \[page 462\]](#)

## 26.4.2 Listing Bookmarks

You can use the scripting API to allow application users to retrieve a list of their own bookmarks at runtime. The following script methods are used, depending on the type of bookmark involved:

### Listing Standard Bookmarks

The returned array contains a list of `BookmarkInfo` object types. A `BookmarkInfo` object contains `BookmarkId` `id`, `String` `name` and `String` `text`. Both `String` `name` and `BookmarkId` `id` refer to the bookmark id. `String` `text` refers to the bookmark title. Any selection component like a [Dropdown box](#) or a [List box](#) can be populated with the array returned from the `Bookmark.getAllBookmarks()` method.

```
• var array = Bookmark.getAllBookmarks();  
  
  array.forEach(function(element, index)  
  
    { DROPDOWN_1.addItem(element.name, element.text);  
  
  });
```

### Listing Fragment Bookmarks

Listing fragment bookmarks returns a list of all fragment bookmarks for the analysis application.

```
• var array = Bookmark.FragmentBookmark.getAllBookmarkInfos();  
  
  array.forEach(function(element, index)  
  
    { DROPDOWN_1.addItem(element.id, element.title);  
  
  });
```

## Listing Portable Fragment Bookmarks for all Applications

Applying the scripting method `Bookmark.PortableFragmentBookmark.getAllBookmarkInfos()`; returns a list of all portable fragment bookmarks specified by the Group Identifier parameter. You can populate the *Fragment Gallery* with all portable fragment bookmarks created by the user using the `FragmentGallery_1.addItem()`; scripting method.

- ```
var array =  
Bookmark.PortableFragmentBookmark.getAllBookmarkInfos("groupIdentifier");  
  
FRAGMENTGALLERY_1.addItem(array);
```

## Listing Portable Fragment Bookmarks for Current Application

Applying the scripting method

`Bookmark.PortableFragmentBookmark.getAllBookmarkInfosForApplication()`; returns a list of all portable fragment bookmarks for the current application only.

- ```
var array =  
Bookmark.PortableFragmentBookmark.getAllBookmarkInfosForApplication("groupIdentifier");  
  
FRAGMENTGALLERY_1.addItem(array);
```

For more information on the rules applied when using the Group Identifier, you can refer to the chapter in this guide called *Portable Fragment Bookmarks*.

For more information on working with the *Fragment Gallery* and using the online composition feature, you can refer to the chapter in this guide called *Using the Online Composition Feature*.

### Note

Deleting a bookmark does not automatically remove it from a selection component, so a deleted bookmark may still appear in a list. The selection component needs to be scripted separately to allow for the list of bookmarks to be refreshed after a bookmark is deleted.

## Related Information

[Portable Fragment Bookmarks \[page 163\]](#)

[Working with Online Composition \[page 150\]](#)

[Working with Fragment Gallery Components \[page 153\]](#)

[Get All Bookmarks \(getAllBookmarks\) \[page 450\]](#)

[Get All Bookmark Infos \(getAllBookmarkInfos\) \[page 456\]](#)

[Get All Bookmark Infos For Application \(getAllBookmarkInfosForApplication\) \[page 463\]](#)

[Get All Bookmark Infos \(getAllBookmarkInfos\) \[page 462\]](#)



## 26.4.3 Loading Bookmarks Using Scripting

You can use a scripting method to allow application users to load their own bookmarks within a running analysis application. The following script methods are used, depending on the type of bookmark involved:

### Loading Standard Bookmarks

To load a standard bookmark based on id, you can use the following method:

```
Bookmark.loadBookmark(id);
```

### Selecting a Standard Bookmark id from Dropdown List and Loading

```
var id = DROPDOWN_1.getSelectedValue();
```

```
Bookmark.loadBookmark(id);
```

### Loading Fragment Bookmarks

Applying the scripting method `Bookmark.FragmentBookmark.loadBookmark(id)` loads the state of an analysis application from a fragment bookmark.

### Loading Portable Fragment Bookmarks

Applying the scripting method `Bookmark.PortableFragmentBookmark.loadBookmark(id)` loads the state of an analysis application from a portable fragment bookmark.

#### **i** Note

It is important that the bookmarked container name remain unchanged and that the container persists. For example, if you have saved your bookmark within a *Panel* container, this container must not be removed from the application into which you want to load your bookmark.

#### **i** Note

Portable fragment bookmarks can only be loaded in the application where they were created. To load a portable fragment bookmark from a different application, refer to the chapter called *Using the Online Composition Feature*.

---

## Related Information

[Load Bookmark \(loadBookmark\) \[page 453\]](#)

[Load Bookmark \(loadBookmark\) \[page 459\]](#)

[Load Bookmark \(loadBookmark\) \[page 465\]](#)

[Working with Online Composition \[page 150\]](#)

## 26.4.4 Loading Shared Bookmarks Using a Url

Using a Url in the browser bar, application users can load bookmarks shared with them by other application users. When they open the Url from their preferred browser, they will see the shared analysis application. All bookmark types shared using a Url cannot be added directly to the users list of bookmarks. However, they can bookmark the shared analysis application and that bookmark is then listed when calling the script method

`Bookmark.getAllBookmarks()`; OR `Bookmark.FragmentBookmark.getAllBookmarkInfos()`; OR `Bookmark.PortableFragmentBookmark.getAllBookmarkInfos(groupIdIdentifier)`; . Alternatively, the Url can be saved to their browser favorites.

### Note

- It is important that the bookmarked container name remain unchanged and that the container persists. For example, if you have saved your bookmark within a *Panel* container, this container must not be removed from the application into which you want to load your bookmark.
- Obsolete bookmarks, can still be loaded if they have been saved as a Url. A message will appear when you run the application telling you that the bookmark is obsolete.

## Related Information

[Loading Bookmarks Using Scripting \[page 169\]](#)

[Sharing a Bookmark \[page 172\]](#)

[Obsolete Bookmarks \[page 175\]](#)

## 26.4.5 Saving a Bookmark

### Saving a Bookmark

You can use the scripting API to allow application users to save their own bookmarks. A new standard bookmark is created with an optional title for the current state and version of the analysis application. A new fragment or portable fragment will be created with an optional title for a part or parts of the analysis application. A default title will be automatically generated if not supplied. Via scripting, you can provide a way to allow users to create their own title.

## Saving Standard Bookmarks

The following scripting is used to allow the application user to save their own standard bookmarks with a unique, automatically generated title or with a title specified by them at runtime:

```
var id = Bookmark.saveBookmark();
```

```
var id = Bookmark.saveBookmark("title")
```

### Note

- When the application user saves their bookmark with a title that already exists, the corresponding bookmark with that title is overwritten.
- If a title is specified it must not contain an empty string.

## Saving Fragment Bookmarks

Scripting is used to allow the application user to save their own fragment bookmarks. Title is optional. If the title is not specified then a unique default title will be generated automatically. `Optional BookmarkInfo toOverWrite` is used if you want to overwrite an existing fragment bookmark.

```
var fragInfo = Bookmark.FragmentBookmark.saveBookmark(ContainerComponent);
```

```
Signature: FragmentBookmarkInfo saveBookmark(ContainerComponent component, optional  
String title ,optional String description, optional BookmarkId toOverwrite);
```

### Note

Title and description are not unique identifiers, so the user can apply any title or description.

## Saving Portable Fragment Bookmarks

Portable fragment bookmarks behave technically the same way as a fragment bookmark. The difference between fragment and portable fragment bookmarks, is that they are the only bookmark types that can be used within the [Fragment Gallery](#) as part of the online composition feature. For more information on online composition, see the following chapters - *Working with Online Composition*, *Working with Fragment Gallery Components* and *Working with Split Cell Containers*.

Title is optional. If the title is not specified then a unique default title will be generated automatically. The `groupIdentifier` allows you to group or to apply a group to a set of portable fragment bookmarks. This allows users to share portable fragment bookmarks across applications.

```
Bookmark.PortableFragmentBookmark.saveBookmark(groupIdentifier, ContainerComponent);
```

```
Signature: PortableFragmentBookmarkInfo saveBookmark(String groupIdIdentifier,
ContainerComponent component, optional String title, optional String description,
optional String image, optional PortableFragmentBookmarkInfo toOverwrite);
```

### **i** Note

- On the BI platform, if you wish to over-write the optional image applied when saving a portable fragment bookmark, you will need to specify in your scripting, a fully qualified path to the location of the required image. An example of such a path would be the following: "User Folders/Administrator/GEN (MIMES) /vizgal\_column.svg".
- When you save a bookmark it is saved to disk. Applying the scripting methods for the undo or reset features will not remove the bookmark. To remove the bookmark the appropriate method to delete bookmarks needs to be called.
- When scripting using the Group Identifier, a number of rules apply that must be followed. For more details on these rules, you can refer to the chapter called *Portable Fragment Bookmarks*.
- If a bookmark is saved while a popup is open, the popup will be visible when reloading the analysis application from that bookmark. Call the method `POPUP.hide()` to ensure the popup is closed on loading the bookmark.

## Related Information

[Fragment Bookmarks \[page 162\]](#)

[Working with Online Composition \[page 150\]](#)

[Working with Fragment Gallery Components \[page 153\]](#)

[Working with Split Cell Containers \[page 155\]](#)

[Maximum Number of Steps Back or Resetting within an Application \[page 136\]](#)

[Back One Step \(backOneStep\) \[page 688\]](#)

[Back To Start \(backToStart\) \[page 689\]](#)

[Fragment Gallery \[page 406\]](#)

[Save Bookmark \(saveBookmark\) \[page 453\]](#)

[Save Bookmark \(saveBookmark\) \[page 459\]](#)

[Save Bookmark \(saveBookmark\) \[page 466\]](#)

## 26.4.6 Sharing a Bookmark

To enable application users to share any bookmark type Url, you can implement one of the following scripting methods, depending on the type of bookmark being shared:

- To share a standard bookmark, you can applying the `Bookmark.shareBookmark(String URL)` scripting method.
- To share a fragment bookmark, you can apply the `Bookmark.FragmentBookmark.shareBookmark(String URL)` scripting method.
- To share a portable fragment bookmark, you can apply the `Bookmark.PortableFragmentBookmark.shareBookmark(String URL)` scripting method.

The Url parameter is the text to be displayed. At runtime when this script is called, application users will be presented with a [Share Bookmark](#) dialog box, containing the generated bookmark Url. They have three options when using the [Share Bookmark](#) dialog box:

- Open the bookmark link directly in a separate browser by selecting the link icon. This will bring the user to the log in screen of the bookmarked application.
- Select and copy the bookmark Url to their preferred browser. They can then use native browser functionality to add the Url to their favorites or share the bookmark by copying and pasting the Url into an email.
- Select the [Send by email](#) button to open an email directly from their current application browser. The subject field of the generated email is pre-populated with the title of both the application and the bookmark the user wants to share. The body of the email contains a copy of the actual bookmark link. The user can then edit the email as required and share a bookmark of their application by sending the email to the appropriate recipients.

To exit the [Share Bookmark](#) dialog box, application users should select the [Close](#) button.

#### **i** Note

- When sharing a fragment bookmark or portable fragment bookmark, it is important to note that users can only share one fragment bookmark Id . If users intend to share their fragment bookmark, they must ensure that everything they want to share is contained within one container component. Only one bookmark Url can be loaded at a time.
- Sharing a bookmark is most applicable in BI platform and SAP NetWeaver mode. However, it is technically possible to share a bookmark in local mode with other users on a different computer for testing and development purposes. Within the design tool under ► [Tools](#) ► [Preferences](#) ► [Application Design](#) ►, users should select the [Allow external access to embedded Web server](#) checkbox. To allow other users to view this bookmark Url, the designer session ID should also be manually added to the Url.

## Related Information

[Share Bookmark \(shareBookmark\) \[page 454\]](#)

[Share Bookmark \(shareBookmark\) \[page 460\]](#)

[Share Bookmark \(shareBookmark\) \[page 467\]](#)

## 26.4.7 Assigning Bookmarks to Folders

You can script a [Tree](#) component to display to the application user, a list of all bookmark folders available to them, based on their Group Identifier. You can also script a [Tree](#) component to allow the application user to save one of their own bookmarks to a selected folder. If a user does not have access rights to a particular folder, they will not see it appear as an option for them to select. If a user has view rights only, they will be able to see the folder, but not overwrite the folder content in any way. It is also possible to display the root folder within the [Tree](#) component.

## Assigning to a Folder using the Tree

This sample script could be used to assign bookmarks to a folder using a [Tree](#):

```
var folderId = TREE_1.getSelectedValue();
```

```
var bookmarkId = LISTBOX_1.getSelectedValue();
```

```
Bookmark.assignToFolder(folderId, bookmarkId);
```

## Get Child Folders of a Hierarchy

This sample script could be used to get the child folders of a hierarchy:

```
var id = TREE_1.getSelectedValue();
```

```
var newModel = Bookmark.getBookmarkFoldersTreeModel(false, id);
```

```
TREE_1.setModel(newModel);
```

## Setting a Tree with Bookmarks



This sample script could be used to set a [Tree](#) with bookmarks:

```
var model = Bookmark.getBookmarkFoldersTreeModel(true);
```

```
TREE_1.setModel(model);
```

### Note

Before working with bookmarks, you should refer to the following SAP Notes:

SAP Note Number	Description
<a href="#">2168984</a> 	Considerations when working with bookmarks in SAP BusinessObjects Design Studio
<a href="#">2157239</a> 	Performing more than one free text search from your bookmark filter list will return zero items.

---

## 26.5 Obsolete Bookmarks

### Obsolete Standard Bookmarks

Standard bookmarks can become obsolete when design time changes have been made to the analysis application and the application is then re-published. As a result of these changes, obsolete standard bookmarks are not returned to the user via the `Bookmark.getAllBookmarks()` method. However, the bookmark may still be loaded if the bookmark id is known. If an application user makes a change to their analysis application and tries to load one of their own standard bookmarks saved before this change is made, an information message appears. The information message informs the application user that the bookmark is obsolete and that a new bookmark should be created.

### Obsolete Personalization

If a user personalizes an application and the application subsequently changes, the personalization will be obsolete.

### Obsolete Fragment Bookmarks

Fragment bookmarks are never obsolete. This gives maximum flexibility.

#### Note

It is highly recommended that you only create a fragment bookmark using areas of an application that are unlikely to change in the future. If a change is made in the area that has been bookmarked, new user interface changes will not be reflected in the older version of this fragment bookmark.

### Obsolete Portable Fragment Bookmarks

In terms of becoming obsolete, portable fragment bookmarks behave in the same way as fragment bookmarks.

#### Note

- Obsolete bookmarks, can still be loaded if they have been saved as a Url. A message will appear when you run the application telling you that the bookmark is obsolete.

## Related Information

[Personalization \[page 176\]](#)

[Get All Bookmarks \(getAllBookmarks\) \[page 450\]](#)

## 26.6 Personalization

The personalization feature is available at runtime in local, BI platform and SAP NetWeaver mode. It allows application users to save the state of their analysis application and later restore that default state when reloading the analysis application. Personalization is defined by the unique combination of analysis application, analysis application version and application user. Everytime personalization is set, the current analysis application state is overwritten by the last personalization saved. There is no scripting method to load personalization. The following script methods apply to personalization:

- `State.setPersonalization()` ; . This method allows the application user to personalize their current analysis application.
- `State.deletePersonalization()` ; . This method allows the application user to delete the personalization of their analysis application.

### **i** Note

1. When scripting the personalization feature, you must provide application users with a way to call the `State.deletePersonalization()` ; method, to allow them to remove personalization when required. You must script the `State.setPersonalization()` ; and `State.deletePersonalization()` ; methods at the same time.
2. If a user personalizes an application and the application subsequently changes, the personalized bookmark will be obsolete. The application is then no longer personalized.
3. On Startup scripts and On Variable initialization do not run when personalizing an analysis application. This prevents startup scripts and variable initialization settings from overriding personalization settings.

## Related Information

[Set Personalization \(setPersonalization\) \[page 690\]](#)

[Delete Personalization \(deletePersonalization\) \[page 691\]](#)

[Obsolete Bookmarks \[page 175\]](#)



---

## 27 Working with Info Charts

### Context

You can add an [Info Chart](#) component to an analysis application to visually represent the data from a data source.

### Procedure

1. From the [Components](#) view, drag and drop the [Info Chart](#) component into the layout editor.
2. From the [Outline](#) view, drag and drop a data source onto the chart component.  
The info chart appears according to the properties of the default chart.
3. Select the info chart. In the [Properties](#) and [Additional Properties](#) views, you can modify and configure the chart properties.

### 27.1 Info Chart Settings

When getting started with your info chart, you can use the [Edit Initial View..](#) to specify the content (measures and dimensions) to be used in your info chart. The configuration and appearance of data in an info chart is determined by the chart type and the properties selected in the [Properties](#) and [Additional Properties](#) views.

### 27.2 Configuring Initial Settings for Info Charts

#### Context

Add to the initial view the measures and dimensions you want to see in your info chart. To work with columns and rows, use the [Edit Initial View...](#) dialog box to drag measures and dimensions to the [Columns](#) or [Rows](#) areas.

#### **i** Note

- In the [Edit Initial View...](#) dialog box, in the pane on the left, open the [Measures](#) node to see the full list of measures in your selected data source.
- Dual axis, scatter and bubble charts always require measures.

## Procedure

1. In the *Outline* view, right-click a data source, and in the context menu, select *Edit Initial View*.
2. In the *Edit Initial View...* dialog box, from the pane on the left side of the screen, drag measures and dimensions to the *Columns* or *Rows* areas.
3. Do one of the following:
  - Choose the *OK* button to return to the layout editor.
  - Choose the *OK + Create Crosstab* button to create a crosstab and return to the layout editor.

## 27.3 Using the Chart Configuration Dialog

### Context

The *Chart Configuration* dialog allows you to configure your chart at design time.

#### **i** Note

The *Chart Configuration* dialog is a combination of the *Chart Type Picker*, *Info Chart Feeding Panel* and a preview area of the selected *Info Chart*.

## Procedure

1. Assign a data source to your *Info Chart* and select the default info chart from within the *Layout Editor*
2. Within the *Properties* tab, select the *Chart Configuration* dialog to further configure your info chart settings.
3. Select the desired chart type from the *Chart Type Picker*. Each chart type collection offers a dropdown from which you can select the specific chart type required.

The info chart displayed in the preview area changes to reflect your info chart type selection.

4. Move measures and dimensions up and down within their respective areas in the *Info Chart Feeding Panel* component area to configure the binding of data to the info chart. Any changes made are directly reflected in the info chart preview area.
5. Select *OK*.

The chart type and the binding are applied to the info chart. The newly configured info chart is displayed in the *Layout Editor*.

6. To display the same conditional formatting in your chart as appears in your crosstab, set the *Conditional Formatting Visible* property to true. It is set to false by default.
7. To display totals in your charts, set the *Show Totals* property to true. It is set to false by default.

---

## Related Information

[Info Chart Feeding Panel \[page 400\]](#)

[Chart Type Picker \[page 399\]](#)

[Info Chart \[page 352\]](#)

## 27.4 Info Chart Types

You can use the [Chart Type Picker](#) to select the type of charts to be used in your application. The following chart types are available for selection:

### Bar and Column

- 100% Stacked Bar Chart
- 100% Stacked Column Chart
- Bar Chart
- Bar Chart with 2 X-Axes
- Column Chart
- Column Chart with 2 Y-Axes
- Stacked Bar Chart
- Stacked Column Chart
- 100% Stacked Bar Chart with 2 X-Axes
- 100% Stacked Column Chart with 2 Y-Axes
- Stacked Bar Chart with 2 X-Axes
- Stacked Column Chart with 2 Y-Axes

### Line

- Line Chart
- Area Chart
- Combined Bar Line Chart
- Combined Bar Line Chart with 2 X-Axes
- Combined Column Line Chart
- Combined Column Line Chart with 2 Y-Axes
- Horizontal Area Chart
- Horizontal Line Chart
- Line Chart with 2 Y-Axes
- Stacked Combined Bar Line Chart

- 
- Stacked Combined Column Line Chart
  - Stacked Combined Column Line Chart with 2 Y-Axes
  - Time Series Line Chart
  - Horizontal Line Chart with 2 X-Axes
  - Stacked Combined Bar Line Chart with 2 X-Axes

## Pie

- Pie Chart
- Doughnut Chart

## Bubble

- Bubble Chart
- Scatter Chart

## Map

- Heat Map
- Tree Map

## Trellis

- Trellis Area Chart
- Trellis Bar Chart
- Trellis Column Chart
- Trellis Horizontal Area Chart
- Trellis Horizontal Line Chart
- Trellis Line Chart

## Other

- Radar Chart
- Bullet Chart
- Vertical Bullet Chart

---

## Related Information

[Working with Info Charts \[page 177\]](#)

[Chart Type Picker \[page 399\]](#)

---

## 28 Working with Charts

### Context

You can add a chart component to an analysis application to display the data in a data source.

### Procedure

1. From the [Components](#) view, drag and drop the [Chart](#) component into the layout editor.
2. From the [Outline](#) view, drag and drop a data source onto the chart component.  
The chart appears according to the properties of the default chart.
3. Select the chart. In the [Properties](#) and [Additional Properties](#) views, you can modify and configure the chart properties.

### 28.1 Chart Settings

Several factors influence the structure and appearance of data in a chart. To design a chart well, you need to take into account both the measures and dimensions of a data source. Assigning these as columns or rows has a great impact on the appearance of your chart.

Other important factors that influence a chart's appearance are the chart type and its properties in the [Properties](#) and [Additional Properties](#) views.

### Related Information

[Chart](#) [page 338]

### 28.2 Configuring Initial Settings for Charts

#### Context

Use the initial chart settings tables to control the display of data in a chart. To work with columns and rows, use the [Edit Initial View...](#) dialog box to drag measures and dimensions to the [Columns](#) or [Rows](#) areas.

### **i** Note

1. When measures and dimensions are added together in either the *Columns* area or the *Rows* area, we assume in the following tables that measures are always the last item.
2. In the *Edit Initial View...* dialog box, in the pane on the left, open the *Measures* node to see the full list of measures in your selected data source.
3. In *Edit Initial View...*, if you have an empty measure or measures only in the *Columns* area, the crosstab will display metadata in the form of headers, but no rows or cells. When you run the analysis application, the chart will display dimensions in the axes, but no data.
4. If you have an empty dimension or dimensions only in the *Rows* area, the crosstab will display metadata in the form of one column header for each dimension and rows with metadata. When you run the analysis application, the chart will display dimensions in the legend, but no data.
5. If there are empty dimensions in the *Rows* and *Columns* areas, the crosstab will display a column header for each of the dimensions in the rows and columns. The crosstab will have no data. When you run the analysis application, the chart will display the legend and axes, but no data.
6. Dual axis, scatter and bubble charts always require measures.

## Procedure

1. In the *Outline* view, right-click a data source, and in the context menu, select *Edit Initial View*.
2. In the *Edit Initial View...* dialog box, from the pane on the left side of the screen, drag measures and dimensions to the *Columns* or *Rows* areas.
3. Do one of the following:
  - Choose the *OK* button to return to the layout editor.
  - Choose the *OK + Create Crosstab* button to create a crosstab and return to the layout editor.
4. Use the information in the chart tables below to configure your chart settings.

### 28.2.1 100% Stacked Bar

The 100% stacked bar charts are used to display the percentage that each component contributes to a total across categories. The table below contains the configuration chart display details for creating a 100% stacked bar chart. You can enter the details in the *Edit Initial View...* dialog box.

### **i** Note

Each stacked bar displays the stack segments as percentages of an entire bar, which always represents 100 in length.

Table 15: 100% Stacked Bar Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	Measures are displayed as stack segments in one bar, and in the legend.
None selected.	None selected.	One or more selected.	One or more selected.	One bar appears. Measures and dimensions appear as stack segments, and in the legend.
None selected.	One or more selected.	None selected.	None selected.	One bar appears for each measure. No stack segments appear.
None selected.	One or more selected.	One or more selected.	None selected.	Each measure appears as a bar. The dimensions appear as stack segments of each bar, and in the legend.
One or more selected.	None selected.	None selected.	One or more selected.	Each dimension or combination of dimensions appears as a bar. The measures appear as stack segments of each bar, and in the legend.
One or more selected.	None selected.	One or more selected.	One or more selected.	Each dimension or combination of dimensions in the <i>Columns</i> area appears as a bar. The measures and dimensions in the <i>Rows</i> area appear as stack segments of each bar, and in the legend.
One or more selected.	One or more selected.	None selected.	None selected.	Each measure appears as a bar. The bars are clustered for each dimension. No stack segments appear.
One or more selected.	One or more selected.	One or more selected.	None selected.	Each measure in the <i>Columns</i> area appears as a bar. Bars are clustered by dimensions in the <i>Columns</i> area. Each measure colors a group of bar clusters. Measures appear in the legend.



## 28.2.2 100% Stacked Column

The 100% stacked column charts are used to display the percentage that each component contributes to a total across categories. The table below contains the configuration chart display details for creating a 100% stacked column chart. You can enter the details in the [Edit Initial View...](#) dialog box.

### **i** Note

Each stacked column displays the stack segments as percentages of an entire column, which always represents 100 in length.

Table 16: 100% Stacked Column Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	Measures appear as stack segments in one column, and in the legend.
None selected.	None selected.	One or more selected.	One or more selected.	One column appears. Measures and dimensions appear as stack segments, and in the legend.
None selected.	One or more selected.	None selected.	None selected.	One column appears for each measure. No stack segments appear.
None selected.	One or more selected.	One or more selected.	None selected.	Each measure appears as a column. The dimensions appear as stack segments of each column, and in the legend.
One or more selected.	None selected.	None selected.	One or more selected.	Each dimension or combination of dimensions appears as a column. The measures appear as stack segments of each column, and in the legend.
One or more selected.	None selected.	One or more selected.	One or more selected.	Each dimension or combination of dimensions in the <a href="#">Columns</a> area appears as a column. The measures and dimensions in the <a href="#">Rows</a> area appear as stack segments of each column, and in the legend.

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
One or more selected.	One or more selected.	None selected.	None selected.	Each measure appears as a column. The columns are clustered for each dimension. No stack segments appear.
One or more selected.	One or more selected.	One or more selected.	None selected.	Each measure in the <a href="#">Columns</a> area appears as a column. Columns are clustered by dimensions in the <a href="#">Columns</a> area. Each measure colors a group of column clusters. Measures appear in the legend.

## 28.2.3 Area

Area charts are used to emphasize an amount of change over a certain period or to show total values for a trend. When an area chart includes sums of values, you can see the relationship of the individual components to a whole. The table below contains the configuration chart display details for creating an area chart. You can enter the details in the [Edit Initial View...](#) dialog box.

Table 17: Area Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	Measures appear as a vertical line, and in the legend.
None selected.	None selected.	One or more selected.	One or more selected.	Measures and dimensions appear as a vertical line, and in the legend.
None selected.	One or more selected.	None selected.	None selected.	Measures appear on the label axis. The area from 0 on the value axis to the plotted measure amounts is filled with color.
None selected.	One or more selected.	One or more selected.	None selected.	Measures appear on the label axis. Dimensions are plotted from 0 on the value axis and each area is filled with a different color. Dimensions appear in the legend.

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
One or more selected.	None selected.	None selected.	One or more selected.	Dimensions appear on the label axis. Measures are plotted from 0 on the value axis and each is filled with a different color. Measures appear in the legend.
One or more selected.	None selected.	One or more selected.	One or more selected.	Dimensions in the <i>Columns</i> area appear on the label axis. Measures are clustered by dimensions and plotted from 0 on the value axis. Each measure area is filled with a different color. Measures and dimensions in the <i>Rows</i> area appear in the legend.
One or more selected.	One or more selected.	None selected.	None selected.	Measures are clustered by dimensions on the label axis. Measures are plotted from 0 on the value axis and the area is filled with color.
One or more selected.	One or more selected.	One or more selected.	None selected.	Measures are clustered on the label axis by the dimensions in the <i>Columns</i> area. Measures are plotted according to the dimensions in the <i>Rows</i> area. The measure areas are filled with color according to these row dimensions, whose colors appear in the legend.

## 28.2.4 Bar

Bar charts are used to display the differences between items. Bar charts are useful for displaying durations. The table below contains the configuration chart display details for creating a bar chart. You can enter the details in the *Edit Initial View...* dialog box.

Table 18: Bar Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	Measures are displayed as bars, and in the legend.
None selected.	None selected.	One or more selected.	One or more selected.	Dimensions appear as bars for each measure. Each dimension has a color that is repeated for each measure. Dimensions and measures appear in the legend.
None selected.	One or more selected.	None selected.	None selected.	Each measure appears as a bar.
None selected.	One or more selected.	One or more selected.	None selected.	Bars are displayed in clusters. Each measure represents a cluster. Dimensions in the <i>Rows</i> area appear in the legend.
One or more selected.	None selected.	None selected.	One or more selected.	Bars are displayed in clusters. Each dimension represents a cluster. Measures appear in the legend.
One or more selected.	None selected.	One or more selected.	One or more selected.	Bars appear in clusters. Each dimension represents a cluster. The measures appear in the legend.
One or more selected.	One or more selected.	None selected.	None selected.	Bars appear as clusters of dimensions for each measure.
One or more selected.	One or more selected.	One or more selected.	None selected.	Bars appear as clusters of dimensions for each dimension in the <i>Columns</i> area. Each measure represents a cluster. Dimensions in the <i>Rows</i> area appear in the legend.

## 28.2.5 Bar Combination

A bar combination chart combines the features of a bar chart and a line chart. Use bar combination charts to display the values in different categories. The table below contains the configuration chart display details for creating a bar combination chart. You can enter the details in the *Edit Initial View...* dialog box.

## i Note

In the table below, in the *Chart Display* column, certain settings cause either dots or lines to be displayed. In these cases, you can display lines by adding more data in the *Edit Initial View...* dialog box, in the *Rows* or *Columns* area.

Table 19: Bar Combination Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	The first measure appears as a bar, and the subsequent measures as dots or lines.
None selected.	None selected.	One or more selected.	One or more selected.	The first dimension appears as bars. Subsequent dimensions and measures appear as horizontally placed dots or lines that are colored according to the dimensions. Measures and dimensions appear in the legend.
None selected.	One or more selected.	None selected.	None selected.	A bar appears for each measure. No line appears.
None selected.	One or more selected.	One or more selected.	None selected.	Measures appear on the value axis. Dimensions appear as bars that are clustered by dimensions and measures. No lines appear. Dimensions appear in the legend.
One or more selected.	None selected.	None selected.	One or more selected.	The first measure appears as bars. Subsequent measures appear as horizontal lines that are plotted by dimensions. Measures appear in the legend.
One or more selected.	None selected.	One or more selected.	One or more selected.	The first measure appears as bars that are clustered according to dimensions. Subsequent measures appear as dots or lines.
One or more selected.	One or more selected.	None selected.	None selected.	A bar appears for each measure. Measures are clustered by dimensions. No line appears.

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
One or more selected.	One or more selected.	One or more selected.	None selected.	Clusters of bars appear according to measures and dimensions. No line appears.

## 28.2.6 Bubble

Bubble charts can display three or four measures of data. Each bubble is plotted using the first two values as its label-axis / data-axis location and the third and fourth as bubble size. The tables below contain the configuration chart display details for creating a bubble chart. You can enter the details in the [Edit Initial View...](#) dialog box.

### Note

When you are working with the bubble chart initial settings, bear the following in mind:

- You must use a minimum of three measures. The first measure becomes the label-axis setting, the second measure becomes the value-axis setting, and the third and fourth measures become the bubble width and height.
- Location of the measures does not have any impact on the chart. However, you usually add measures in the [Rows](#) area. If you add measures in the [Columns](#) area, you must set the [Swap Axes](#) property to **true**.
- You can add dimensions to the same area to which you have added measures.
- The first dimension in the [Columns](#) area displays as different colors. If you add more dimensions, more bubbles appear in the chart.
- The first dimension in the [Rows](#) area displays as different shapes. If you add more dimensions, these appear as extra bubbles or other shapes.
- If you have less than three measures, a scatter chart is recommended.

Table 20: Bubble Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	Three measures selected.	The chart displays one bubble.
None selected.	Three measures selected.	None selected.	None selected.	The chart displays one bubble.
One or more selected.	None selected.	None selected.	Three measures selected.	The chart displays more than one bubble.
None selected.	Three measures selected.	One or more selected.	None selected.	The chart displays more than one bubble.

### Note

For this setting you must set the [Swap Axes](#) property to **true**.

## 28.2.7 Column

Column charts are used to display the differences between items or to display durations. The table below contains the configuration chart display details for creating a column chart. You can enter the details in the [Edit Initial View...](#) dialog box.

Table 21: Column Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	Measures are displayed as columns, and in the legend.
None selected.	None selected.	One or more selected.	One or more selected.	Dimensions appear as columns for each measure. Each dimension has a color that is repeated for each measure. Dimensions and measures appear in the legend.
None selected.	One or more selected.	None selected.	None selected.	Each measure appears as a column.
None selected.	One or more selected.	One or more selected.	None selected.	Columns are displayed in clusters. Each measure represents a cluster. Measures appear in the legend.
One or more selected.	None selected.	None selected.	One or more selected.	Columns are displayed in clusters. Each dimension represents a cluster. Measures appear in the legend.
One or more selected.	None selected.	One or more selected.	One or more selected.	Columns appear in clusters. Each dimension represents a cluster. The measures appear in the legend.
One or more selected.	One or more selected.	None selected.	None selected.	Columns appear as clusters of dimensions for each measure.
One or more selected.	One or more selected.	One or more selected.	None selected.	Columns appear as clusters of dimensions for each dimension in the <a href="#">Columns</a> area. Each measure represents a cluster. Dimensions in the <a href="#">Rows</a> area appear in the legend.

## 28.2.8 Column Combination

A column combination chart combines the features of a column chart and a line chart. Use column combination charts to display the values in different categories. The table below contains the configuration chart display details for creating a column combination chart. You can enter the details in the [Edit Initial View...](#) dialog box.

### **i** Note

In the table below, in the [Chart Display](#) column, certain settings cause either dots or lines to be displayed. In these cases, you can display lines by adding more data in the [Edit Initial View...](#) dialog box, in the [Rows](#) or [Columns](#) area.

Table 22: Column Combination Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	The first measure appears as a column, and the subsequent measures as dots or lines.
None selected.	None selected.	One or more selected.	One or more selected.	The first dimension appears as columns. Subsequent dimensions and measures appear as vertically placed dots or lines that are colored according to the dimensions. Measures and dimensions appear in the legend.
None selected.	One or more selected.	None selected.	None selected.	A column appears for each measure. No line appears.
None selected.	One or more selected.	One or more selected.	None selected.	Measures appear on the label axis. Dimensions appear as columns that are clustered by dimensions and measures. No lines appear. Dimensions appear in the legend.
One or more selected.	None selected.	None selected.	One or more selected.	The first measure appears as columns. Subsequent measures appear as horizontal lines that are plotted by dimensions. Measures appear in the legend.



Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
One or more selected.	None selected.	One or more selected.	One or more selected.	The first measure appears as columns that are clustered according to dimensions. Subsequent measures appear as dots or lines.
One or more selected.	One or more selected.	None selected.	None selected.	A column appears for each measure. Measures are clustered by dimensions. No line appears.
One or more selected.	One or more selected.	One or more selected.	None selected.	Clusters of columns appear according to measures and dimensions. No line appears.

## 28.2.9 Dual Axis

Use dual axis charts to assign a different measure to each of the two vertical axes. This provides a broader view of the data. The following charts are available in dual axis mode:

- [Bar](#)
- [Column](#)
- [Combination](#)
- [Horizontal Combination](#)
- [Line](#)
- [Horizontal Line](#)

The [Dual Axis](#) chart data mapping works in the same way as for the single axis equivalent of the chart type with one exception; in order for the axis colors to match the displayed measures colors, measures cannot be in the [Columns](#) (x-axis). Put the measures in the [Legend](#).

- You can set the color for each axis and you can change which measure shows on each axis.
- In combination dual axis charts, you can change the axis and the shape of a given measure.
- To create a dual axis chart, you enter the details in the [Edit Initial View...](#) dialog box.
- For more information, see the relevant chart type documentation for initial settings for each chart type.

### Note

There need to be at least 2 measures in the initial view to display a dual axis chart. When there is 1 or no measure in the [Columns](#) or [Rows](#) area, no chart appears in the layout editor.

## 28.2.10 Horizontal Area

Horizontal area charts are similar to area charts. The difference is that they display the data area horizontally across the chart's label axis. Use horizontal area charts to emphasize an amount of change over a certain period or to show total values for a trend. When a horizontal area chart includes sums of values, you can see the relationship of the individual components to a whole. The table below contains the configuration chart display details for creating a horizontal area chart. You can enter the details in the [Edit Initial View...](#) dialog box.

Table 23: Horizontal Area Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	Measures appear as a horizontal line, and in the legend.
None selected.	None selected.	One or more selected.	One or more selected.	Measures and dimensions appear as a horizontal line, and in the legend.
None selected.	One or more selected.	None selected.	None selected.	Measures appear on the label axis. The area from 0 on the value axis to the plotted measure amounts is filled with color.
None selected.	One or more selected.	One or more selected.	None selected.	Measures appear on the label axis. Dimensions are plotted from 0 on the value axis and each area is filled with a different color. Dimensions appear in the legend.
One or more selected.	None selected.	None selected.	One or more selected.	Dimensions appear on the label axis. Measures are plotted from 0 on the value axis and each is filled with a different color. Measures appear in the legend.
One or more selected.	None selected.	One or more selected.	One or more selected.	Dimensions in the <a href="#">Columns</a> area appear on the label axis. Measures are clustered by dimensions and plotted from 0 on the value axis. Each measure area is filled with a different color. Measures and dimensions in the <a href="#">Rows</a> area appear in the legend.

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
One or more selected.	One or more selected.	None selected.	None selected.	Measures are clustered by dimensions on the label axis. Measures are plotted from 0 on the value axis and the area is filled with color.
One or more selected.	One or more selected.	One or more selected.	None selected.	Measures are clustered on the label axis by the dimensions in the <a href="#">Columns</a> area. Measures are plotted according to the dimensions in the <a href="#">Rows</a> area. The measure areas are filled with color according to these row dimensions, whose colors appear in the legend.

## 28.2.11 Horizontal Line

Horizontal line charts are similar to line charts. The difference is that they display the data area horizontally across the chart's label axis. Use horizontal line charts to display trends in data at equal intervals. Also, horizontal line charts can be useful for displaying durations. In a horizontal line chart, the data label axis is vertical and the data value axis is horizontal. The table below contains the configuration chart display details for creating a horizontal line chart. You can enter the details in the [Edit Initial View...](#) dialog box.

### Note

If a measure or dimension contains no value (null), the line will end at the previous value and begin again with the following value. The line will appear to be broken at the null point.

Table 24: Horizontal Line Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	Measures appear as dots and in the legend.
None selected.	None selected.	One or more selected.	One or more selected.	Measures appear as horizontally placed dots that are colored according to the dimensions. Measures and dimensions appear in the legend.

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	One or more selected.	None selected.	None selected.	Measures appear on the label axis. Line is plotted according to values on the value axis.
None selected.	One or more selected.	One or more selected.	None selected.	Measures appear on the label axis. Lines representing dimensions appear in different colors. Dimensions appear in the legend.
One or more selected.	None selected.	None selected.	One or more selected.	Measures appear as vertical lines that are plotted by dimensions. Measures appear in the legend.
One or more selected.	None selected.	One or more selected.	One or more selected.	Dimensions in the <a href="#">Columns</a> area appear on the label axis. Measures and dimensions in the <a href="#">Rows</a> area appear as dots, and in the legend.
One or more selected.	One or more selected.	None selected.	None selected.	A line is plotted according to measures, which are clustered by dimensions.
One or more selected.	One or more selected.	One or more selected.	None selected.	Dimensions are clustered by measures and appear as dots or lines. Measures appear in the legend.

## 28.2.12 Horizontal Waterfall

Horizontal waterfall charts are similar to waterfall charts. The difference is that they display the data area horizontally across the chart's label axis. Use horizontal waterfall charts to display how an initial value is affected by a series of intermediate positive or negative values. The tables below contain the configuration chart display details for creating a horizontal waterfall chart. You can enter the details in the [Edit Initial View...](#) dialog box.

Table 25: Horizontal Waterfall Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	Measures in the <a href="#">Rows</a> area are displayed as bars.

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	One or more selected.	None selected.	None selected.	Measures in <i>Columns</i> area are displayed as bars.
None selected.	One or more selected.	No more than one is selected.	None selected.	Measures in <i>Columns</i> area are displayed as bars according to the dimension in the <i>Rows</i> area.  <b>i Note</b> When you add measures to the <i>Columns</i> area, you must add either <b>no dimensions</b> or <b>one dimension</b> in the <i>Rows</i> area; otherwise the chart cannot be displayed.
No more than one is selected.	None selected.	None selected.	One or more selected.	Measures in the <i>Rows</i> area are displayed as bars according to the dimension in the <i>Columns</i> area.  <b>i Note</b> When you add measures to the <i>Rows</i> area, you must add either <b>no dimensions</b> or <b>one dimension</b> in the <i>Columns</i> area; otherwise the chart cannot be displayed.
No more than one is selected.	None selected.	One or more selected.	One or more selected.	Only the first measure in the <i>Rows</i> area is displayed as one bar.
One or more selected.	One or more selected.	No more than one is selected.	None selected.	From the <i>Columns</i> area, a combination of the first measure and the first value of each dimension is displayed as a bar.

### Caution

When the following settings are applied, no chart appears in the layout editor.

Table 26: Horizontal Waterfall Chart Initial Settings: Not Supported

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	One or more selected.	None selected.	Chart is not displayed.
None selected.	None selected.	One or more selected.	One or more selected.	Chart is not displayed.
Two or more are selected.	None selected.	None selected.	One or more selected.	Chart is not displayed.
Two or more are selected.	None selected.	One is selected.	One or more selected.	Chart is not displayed.
One or more selected.	One or more selected.	None selected.	None selected.	Chart is not displayed.
One or more selected.	One or more selected.	Two or more are selected.	None selected.	Chart is not displayed.

## 28.2.13 Line

Line charts are used to display trends in data at equal intervals, or to display durations. The table below contains the configuration chart display details for creating a line chart. You can enter the details in the [Edit Initial View...](#) dialog box.

### Note

In the table below, in the *Chart Display* column, certain settings cause either dots or lines to be displayed. In these cases, you can display lines by adding more data in the [Edit Initial View...](#) dialog box, in the *Rows* or *Columns* area.

### Note

If a measure or dimension contains no value (null), the line will end at the previous value and begin again with the following value. The line will appear to be broken at the null point.

Table 27: Line Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	Measures appear as dots or lines and in the legend.
None selected.	None selected.	One or more selected.	One or more selected.	Measures appear as vertically placed dots or lines that are colored according to the dimensions. Measures and dimensions appear in the legend.

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	One or more selected.	None selected.	None selected.	Measures appear on the label axis. Line is plotted according to values on the value axis.
None selected.	One or more selected.	One or more selected.	None selected.	Measures appear on the label axis. Lines representing dimensions appear in different colors. Dimensions appear in the legend.
One or more selected.	None selected.	None selected.	One or more selected.	Measures appear as horizontal lines that are plotted by dimensions. Measures appear in the legend.
One or more selected.	None selected.	One or more selected.	One or more selected.	Dimensions in the <i>Columns</i> area appear on the label axis and are colored according to the dimensions in the <i>Rows</i> area. Measures and dimensions appear as dots or lines, and in the legend, according to the dimensions in the <i>Rows</i> area.
One or more selected.	One or more selected.	None selected.	None selected.	A line is plotted according to measures, which are clustered by dimensions.
One or more selected.	One or more selected.	One or more selected.	None selected.	Dimensions are clustered by measures and appear as dots or lines. Measures appear in the legend.

## 28.2.14 Multiple Pie

Multiple pie charts are used to display several sets of data at the same time. Note the following regarding multiple pie charts:

- Dimensions in the *Rows* area of the *Edit Initial View...* dialog box appear as pie slices in the chart.
- Dimensions in the *Columns* area of the *Edit Initial View...* dialog box appear as a vertical column of pie charts in the chart.
- Each measure in the *Columns* area in the *Edit Initial View...* dialog box appears as a horizontal row of pie charts in the chart.

The table below contains the configuration chart display details for creating a multiple pie chart. You can enter the details in the [Edit Initial View...](#) dialog box.

Table 28: Multiple Pie Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	One column of pies without pie slices is displayed.
None selected.	None selected.	One or more selected.	One or more selected.	A column of pie charts appears, one for each measure. Dimensions are displayed in the legend and as pie slices.
None selected.	One or more selected.	None selected.	None selected.	A column of pie charts appears, one for each measure. No pie slices appear.
None selected.	One or more selected.	One or more selected.	None selected.	A column of pie charts appears, with one pie chart for each measure. Dimensions appear as pie slices, and in the legend.
One or more selected.	None selected.	None selected.	One or more selected.	Rows and columns of pie charts appear according to the number of measures and dimensions. No pie slices are displayed.
One or more selected.	None selected.	One or more selected.	One or more selected.	Rows and columns of pie charts appear. The number of rows is determined by the number of measures, and the columns are determined by the dimensions in the <a href="#">Rows</a> area. The dimensions in the <a href="#">Rows</a> area appear in the legend.
One or more selected.	One or more selected.	None selected.	None selected.	Rows and columns of pies without pie slices appear according to the number of measures and dimensions.
One or more selected.	One or more selected.	One or more selected.	None selected.	Rows and columns of pie charts appear according to the measures and dimensions in the <a href="#">Columns</a> area. These dimensions appear in the legend, and the pie charts are colored accordingly with no slices if there is only one dimension in the <a href="#">Rows</a> area. If there is more than one dimension in the <a href="#">Rows</a> area, they appear as pie slices.

## 28.2.15 Multiple Radar

Multiple radar charts display more than one radar chart, where each individual chart represents one measure. The table below contains the configuration chart display details for creating a multiple radar chart. You can enter the details in the [Edit Initial View...](#) dialog box.

### i Note

When you are working with the multiple radar chart initial settings, note the following:

- You must use at least two measures. The measures are plotted on the dimension radii, creating polygons.
- The more dimensions you add, the more radii appear in the chart.
- You cannot add dimensions to the same area that you have added measures.



- Usually, you add measures to the [Rows](#) area. If you add measures in the [Columns](#) area, you must set the [Swap Axes](#) property to **true**.

Table 29: Multiple Radar Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
One or more selected.	None selected.	None selected.	Two or more measures selected.	<p>The chart displays multiple radar charts:</p> <ul style="list-style-type: none"> <li>Each measure represents one individual radar chart.</li> <li>The data for each measure is plotted around the radii of the chart.</li> <li>Each dimension represents radii in the individual charts.</li> </ul>

## 28.2.16 Pie

Use pie charts to display the size of items in a data series relative to the sum of the items. Each item is displayed as a slice of the pie.

The table below contains the configuration chart display details for creating a pie chart. You can enter the details in the [Edit Initial View...](#) dialog box.

### **i** Note

When there are no measures in the [Columns](#) nor the [Rows](#) area, only a dot appears in the layout editor.

Table 30: Pie Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	Measures are displayed as pie slices. Measures appear in the legend.
None selected.	None selected.	One or more selected.	One or more selected.	Measures and dimensions are displayed as pie slices.
None selected.	One or more selected.	None selected.	None selected.	Measures are displayed as pie slices.
None selected.	One or more selected.	One or more selected.	None selected.	Dimensions are displayed as pie slices.
One or more selected.	None selected.	None selected.	One or more selected.	Dimensions are displayed as pie slices, and in the legend.

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
One or more selected.	None selected.	One or more selected.	One or more selected.	Dimensions in the <a href="#">Columns</a> area are displayed as slices, and appear in the legend.
One or more selected.	One or more selected.	None selected.	None selected.	Dimensions and measures appear as pie slices. Measures appear in the legend according to dimensions.
One or more selected.	One or more selected.	One or more selected.	None selected.	Measures appear as pie slices only if there are at least two dimensions in the <a href="#">Rows</a> area. Otherwise, no pie slices appear.

## 28.2.17 Radar

Radar charts compare the cumulative values of more than one data series. A radar chart displays the data of measures on axes starting from the same point. The radar chart contains a sequence of spoke-like elements, called radii, with each radius representing one of the measures. A line connects the data values for each radius. The table below contains the configuration chart display details for creating a radar chart. You can enter the details in the [Edit Initial View...](#) dialog box.

### Note

When you are working with the radar chart initial settings, note the following:

- You must use at least two measures. The measures are plotted on the dimension radii, creating polygons.
- You cannot add dimensions to the same area that you have added measures.
- Usually, you add measures in the [Rows](#) area. If you add measures in the [Columns](#) area, you must set the [Swap Axes](#) property to **true**.
- The more dimensions you add, the more radii appear in the chart.

Table 31: Radar Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
One or more selected.	None selected.	None selected.	Two or more measures selected.	The chart displays the measures as rings, and the radii as dimensions.

## 28.2.18 Scatter

Scatter charts display a group of dots. Scatter charts can display two measures of data. Each dot is plotted using measure values as its label axis / value axis location. The table below contains the configuration chart display details for creating a scatter chart. You can enter the details in the [Edit Initial View...](#) dialog box.

### **i** Note

When you are working with the scatter chart initial settings, note the following:

- You must use two measures. The first measure becomes the label axis setting; the second measure becomes the value axis setting.
- Usually, you add measures in the [Rows](#) area. If you add measures in the [Columns](#) area, you must set the [Swap Axes](#) property to **true**.
- You cannot add dimensions to the same area that you have added measures.
- In rows, the first dimension sets the legend color.
- In columns, the first dimension sets the legend shape.
- Extra dimensions are used in the tooltip.
- The more dimensions you add, the more dots appear in the chart.
- For more complex data sources with measures and dimensions in columns and dimensions in rows, the result is that the dimensions map to the legend but are rendered as shapes. This chart is similar to the reverse scenario with measures and dimensions in rows and measures in columns, which are then rendered as shapes.

Table 32: Scatter Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	Three measures selected.	The chart displays one dot.
None selected.	Three measures selected.	None selected.	None selected.	The chart displays one dot.
One or more selected.	None selected.	None selected.	Three measures selected.	The chart displays more than one dot.
None selected.	Three measures selected.	One or more selected.	None selected.	The chart displays more than one dot.

### **i** Note

For this setting you must set the [Swap Axes](#) property to **true**.

## 28.2.19 Stacked Bar

Stacked bar charts are used to display the relationship of each item to the whole, comparing the amount of each component to a total across categories. The table below contains the configuration chart display details for creating a stacked bar chart. You can enter the details in the [Edit Initial View...](#) dialog box.

Table 33: Stacked Bar Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	Measures appear as stack segments in one bar, and in the legend.
None selected.	None selected.	One or more selected.	One or more selected.	One bar appears. Measures and dimensions appear as stack segments, and in the legend.
None selected.	One or more selected.	None selected.	None selected.	One bar appears for each measure. No stack segments appear.
None selected.	One or more selected.	One or more selected.	None selected.	Each measure appears as a bar. The dimensions appear as stack segments of each bar, and in the legend.
One or more selected.	None selected.	None selected.	One or more selected.	Each dimension or combination of dimensions appears as a bar. The measures appear as stack segments of each bar, and in the legend.
One or more selected.	None selected.	One or more selected.	One or more selected.	Each dimension or combination of dimensions in the <a href="#">Columns</a> area appears as a bar. The measures and dimensions in the <a href="#">Rows</a> area appear as stack segments of each bar, and in the legend.
One or more selected.	One or more selected.	None selected.	None selected.	Each measure appears as a bar. The bars are clustered for each dimension. No stack segments appear.

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
One or more selected.	One or more selected.	One or more selected.	None selected.	Each measure in the <a href="#">Columns</a> area appears as a bar. Bars are clustered by dimensions in the <a href="#">Columns</a> area. Each measure colors a group of bar clusters. Measures appear in the legend.

## 28.2.20 Stacked Column

The stacked column charts display the relationship of each item to the whole, comparing the amount of each component to a total across categories. The table below contains the configuration chart display details for creating a stacked column chart. You can enter the details in the [Edit Initial View...](#) dialog box.

Table 34: Stacked Column Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	Measures appear as stack segments in one column, and in the legend.
None selected.	None selected.	One or more selected.	One or more selected.	One column appears. Measures and dimensions appear as stack segments, and in the legend.
None selected.	One or more selected.	None selected.	None selected.	One column appears for each measure. No stack segments appear.
None selected.	One or more selected.	One or more selected.	None selected.	Each measure appears as a column. The dimensions appear as stack segments of each column, and in the legend.
One or more selected.	None selected.	None selected.	One or more selected.	Each dimension or combination of dimensions appears as a column. The measures appear as stack segments of each column, and in the legend.

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
One or more selected.	None selected.	One or more selected.	One or more selected.	Each dimension or combination of dimensions in the <a href="#">Columns</a> area appears as a column. The measures and dimensions in the <a href="#">Rows</a> area appear as stack segments of each column, and in the legend.
One or more selected.	One or more selected.	None selected.	None selected.	Each measure appears as a column. The columns are clustered for each dimension. No stack segments appear.
One or more selected.	One or more selected.	One or more selected.	None selected.	Each measure in the <a href="#">Columns</a> area appears as a column. Columns are clustered by dimensions in the <a href="#">Columns</a> area. Each measure colors a group of column clusters. Measures appear in the legend.

## 28.2.21 Stacked Waterfall

Stacked waterfall charts are used to display how an initial value is affected by a series of intermediate positive or negative values. Chart elements for each data series are displayed as stack components. Each column represents the cumulative value of the components it contains. The tables below contain the configuration chart display details for creating a stacked waterfall chart. You can enter the details in the [Edit Initial View...](#) dialog box.

Table 35: Stacked Waterfall Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	One or more selected.	Measures in the <a href="#">Rows</a> area are displayed as columns. No stack segments appear.
None selected.	One or more selected.	None selected.	None selected.	Measures in <a href="#">Columns</a> area are displayed as columns. No stack segments appear.

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	One or more selected.	No more than one is selected.	None selected.	<p>Only the first measure in the <i>Columns</i> area is displayed as columns according to the dimension in the <i>Rows</i> area. No stack segments appear.</p> <div> <p><b>i Note</b></p> <p>When you add measures to the <i>Columns</i> area, you must add either <b>no dimensions</b> or <b>only one dimension</b> in the <i>Rows</i> area; otherwise the chart cannot be displayed.</p> </div>
No more than one is selected.	None selected.	None selected.	One is selected.	<p>Measures in the <i>Rows</i> area are displayed as columns according to the dimension in the <i>Columns</i> area. No stack segments appear.</p> <div> <p><b>i Note</b></p> <p>When you add measures to the <i>Rows</i> area, you must add either <b>no dimensions</b> or <b>one dimension</b> in the <i>Columns</i> area; otherwise the chart cannot be displayed.</p> </div>
No more than one is selected.	None selected.	One or more selected.	One or more selected.	<p>Dimensions in the <i>Columns</i> area are displayed as columns. Measures in the <i>Rows</i> area appear as stack segments according to the dimensions in the <i>Rows</i> area, which also appear in the legend.</p>

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
One or more selected.	One or more selected.	No more than one is selected.	None selected.	Dimensions in the <a href="#">Rows</a> area appear as columns according to the values of the first measure. The columns are colored according to the dimensions in the <a href="#">Columns</a> area, which also appear in the legend.
One or more selected.	None selected.	Two or more selected.	One is selected.	Dimensions in the <a href="#">Columns</a> area are displayed as columns. In the <a href="#">Rows</a> area, both dimensions of the measure are displayed as stack segments and appear in the legend.
One is selected.	One is selected.	One is selected.	None selected.	Dimensions in the <a href="#">Columns</a> area appear as columns according to the values of the measure. The columns are colored according to the dimensions in the <a href="#">Rows</a> area, which also appear in the legend.
Two or more are selected.	One is selected.	One is selected.	None selected.	Dimensions in the <a href="#">Rows</a> area are displayed as columns. In the <a href="#">Columns</a> area, the dimensions of the measure are displayed as stack segments and in the legend.

### Caution

When the following settings are applied, no chart appears in the layout editor.

Table 36: Stacked Waterfall Chart Initial Settings: Not Supported

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	None selected.	None selected.	Chart is not displayed.
None selected.	None selected.	One is selected.	None selected.	Chart is not displayed.
None selected.	None selected.	One is selected.	One is selected.	Chart is not displayed.
None selected.	One is selected.	Two or more selected.	None selected.	Chart is not displayed.
One is selected.	None selected.	None selected.	None selected.	Chart is not displayed.
Two or more selected.	None selected.	None selected.	One is selected.	Chart is not displayed.
Two or more selected.	None selected.	One is selected.	One is selected.	Chart is not displayed.



Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
Two or more selected.	None selected.	Two or more selected.	One is selected.	Chart is not displayed.
One is selected.	One is selected.	None selected.	None selected.	Chart is not displayed.
One is selected.	One is selected.	Two or more selected.	None selected.	Chart is not displayed.
Two or more selected.	One is selected.	Two or more selected.	None selected.	Chart is not displayed.

## 28.2.22 Waterfall

Waterfall charts are used to display how an initial value is affected by a series of intermediate positive or negative values. Generally, waterfall charts are used for time or duration-related data, and horizontal waterfall charts are used for other types of data. The tables below contain the configuration chart display details for creating a waterfall chart. You can enter the details in the [Edit Initial View...](#) dialog box.

Table 37: Waterfall Chart Initial Settings

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected	None selected	None selected	One or more selected	Measures in the <a href="#">Rows</a> area are displayed as columns.
None selected	One or more selected	None selected	None selected	Measures in <a href="#">Columns</a> area are displayed as columns.
None selected	One or more selected	No more than one is selected	None selected	Only the first measure in <a href="#">Columns</a> area is displayed as columns according to the dimension in the <a href="#">Rows</a> area.
				<div> <i>i</i> <b>Note</b> </div> <div> When you add measures to the <a href="#">Columns</a> area, you must add either <b>no dimensions</b> or <b>only one dimension</b> in the <a href="#">Rows</a> area; otherwise the chart cannot be displayed. </div>

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
No more than one is selected	None selected	None selected	One or more selected	Measures in the <i>Rows</i> area are displayed as columns according to the dimension in the <i>Columns</i> area.  <b>Note</b> When you add measures to the <i>Rows</i> area, you must add either <b>no dimensions</b> or <b>only one dimension</b> in the <i>Columns</i> area; otherwise the chart cannot be displayed.
No more than one is selected.	None selected.	One or more selected.	One or more selected.	Only the first measure in the <i>Rows</i> area is displayed as columns, according to the dimension in the <i>Columns</i> area.
One or more selected.	One or more selected.	No more than one is selected.	None selected.	From the <i>Columns</i> area, a combination of the first measure and the first value of each dimension is displayed as a column.



### Caution

When the following settings are applied, no chart appears in the layout editor.

Table 38: Waterfall Chart Initial Settings: Not Supported

Column Dimensions	Column Measures	Row Dimensions	Row Measures	Chart Display
None selected.	None selected.	One or more selected.	None selected	Chart is not displayed
None selected.	None selected.	One or more selected.	One or more selected	Chart is not displayed
One or more selected.	None selected.	None selected.	None selected	Chart is not displayed
Two or more are selected.	None selected.	None selected.	One or more selected	Chart is not displayed
One or more selected.	None selected.	One or more selected.	None selected	Chart is not displayed
Two or more are selected.	None selected.	One is selected.	One or more selected	Chart is not displayed
One or more selected.	One or more selected.	None selected.	None selected	Chart is not displayed
One or more selected.	One or more selected.	Two or more are selected.	None selected	Chart is not displayed

## 28.3 Configuring Additional Chart Types

### Context

A selection of *Additional Chart Types..* can be selected from the *Chart Type* property dropdown list. Use the *Additional Charts* dialog box to populate these additional chart types. You have created an application and wish to select and populate an additional chart type.

#### **i** Note

- If you select and populate a chart type from the *Additional Chart Types..* dropdown list, it is recommended that you reconfigure your axes using the *Additional Charts* dialog box. You should not reconfigure your axes by changing the value of the *Swap Axes* property in the *Properties* tab. Changing the *Swap Axis* property here may result in your chart being rendered incorrectly.
- If you have created and installed new chart extensions using the SAP Lumira SDK, these extensions will appear in the *Additional Chart Types..* dropdown list.. For more information about creating SAP Lumira SDK extensions, see the *SAP Lumira Visualization Extensions Developer Guide* on the SAP Help Portal at <http://help.sap.com/lumira> . Extensions developed with the SAP Lumira SDK can only be deployed to the BI platform and SAP NetWeaver. They cannot be deployed to SAP HANA.
- The *Time Based Line* chart is only available after you install the Real-Time package. For more information, you can refer to the chapters called “Working with Real-Time Dashboards” and “Time Based Line Chart” in this guide.
- If you have dimensions in rows and/or columns in the *Initial View Definition* pane, the *Chart Population* pane must reflect the identical assignment of dimensions. For example, if *Column Dimensions* contains the dimension COUNTRY and *Row Dimensions* contains the dimension REGION, the dimensions COUNTRY and REGION must be assigned in the *Chart Population* pane. If, for example, the dimension COUNTRY is not assigned in the *Chart Population* pane, only the data from the first member of the dimension COUNTRY will display in your chart. To ensure that your *Initial View Definition* pane only displays the dimensions you need for your chart, you must ensure that the dimensions in your *Edit Initial View..* are assigned appropriately.

To configure *Additional Chart Types..* you must follow the steps below.

### Procedure

1. In the *Properties* view, under *Chart Type*, select the last element in the dropdown list called *Additional Chart Types..*
2. In the *Additional Charts* dialog box, select the required *Chart Type* from the dropdown list.
3. From the *Initial View Definition* pane on the left side, move measures, columns and rows to the *Chart Population* pane on the right side of the dialog box. To move measures, columns and rows into the correct area within the *Chart Population* pane, use the *Assign Measure here* or *Assign Column or Row Axis here* prompts.
4. If the *Initial View Definition* pane has dimensions in rows and columns, the *Chart Population* pane should also contain the same rows or columns in an axis.

- 
5. The assignment of measures, rows and columns is based on the additional chart type you have selected.
  6. Choose ► **Apply** and ► **OK**.
  7. The additional chart types currently available by default include:
    - **trellis:** A trellis chart is a series or grid of small similar graphics or charts, allowing them to be easily compared. The following are trellis chart types:
      - bar
      - column
      - line
      - area
      - horizontal line
      - horizontal area
    - **heat map:** A heat map is a chart that represents by colors, the individual values contained in a matrix.
    - **treemap:** Treemaps display hierarchical (tree-structured) data as a set of nested rectangles. Each branch of the tree is given a rectangle, which is then tiled with smaller rectangles representing sub-branches.
    - **time based line:** Time based line charts decouple the plot area from the X-Axis, allowing the spacings in the plot area be independent from the spacings on the axis. The data shown in the plot area is placed according to its time dimension rather than the index in the data.

## Results

You have configured an additional chart type.

## Related Information

[Installing CVOM Chart Extensions for SAP BusinessObjects Design Studio \[page 303\]](#)

[Deploying SDK Extensions \[page 301\]](#)

[Working with Real-Time Dashboards \[page 262\]](#)

## 28.4 Conditional Formatting for Charts

### Context

You can apply conditional formatting rules to measures or dimension members within a selected chart. These rules change the appearance of the chart when specific conditions are met. You can set conditional formatting rules within ► **Chart Properties** ► **Conditional Formatting**. Your analysis application contains a chart to which you wish to apply conditional formatting rules.

---

To apply conditional formatting to your chart, follow the steps below:

## Procedure

1. Within the *Layout Editor*, select the chart to which you wish to apply conditional formatting.
2. Within the *Properties* panel, choose the *Conditional Formatting* ellipsis button.  
The *Conditional Formatting* dialog box opens.
3. Choose *New Rule*.  
A *New Rule* dialog box opens.
4. In the *Name* textbox, enter a title to identify your new conditional formatting rule.
5. Within the *Rule Description* area of the *Conditional Formatting* dialog box, select either Measure or Dimension Member from the dropdown list. The dropdown list is populated from the data set you have applied to your chart.
6. There are three types of conditional formatting rules that you can apply to your chart. Select one of the following:
  - is equal to
  - is greater than
  - is less than
7. Input the appropriate value to be assigned to the new rule.
8. Within the *Preview* area of the dialog box, choose the *Format* button to assign a color to your rule. Choose a color from the pre-defined color picker or define a custom color by choosing the *Define Custom Colors* button. You can then save your custom color to the color picker. Select *OK*.  
The color you selected is previewed in the *Preview* area.
9. Select *OK* again.  
Your new rule is listed in the *Conditional Formatting* dialog box. The *Conditional Formatting* property value field is populated with the JSON text.
10. To edit existing conditional formatting rules, highlight the rule to be edited and choose the *Edit Rule* button.  
You can also activate the edit function by doubleclicking the rule within the *Conditional Formatting* dialog box.
11. To delete existing conditional formatting rules, highlight the rule to be deleted and choose the *Delete Rule* button.
12. Use the up-down arrows to change the order in which the rules appear in the list.  
The rules are applied to the chart in the order in which they appear in the *Conditional Formatting* dialog box list.

## Results

You have applied conditional formatting to your chart.

---

## Related Information

[Chart \[page 338\]](#)

## 29 Working with Geo Maps



### Context

You can add a [Geo Map](#) analytic component to an analysis application to allow users to display different layers of geographical information on a map. Depending on how you configure the [Geo Map](#) properties and apply scripting, each layer displays different information when the user interacts with it. Each layer is connected to a different data source.

Connected data sources must contain geo-specific dimensions or measures, for example, countries, longitude and latitude.

#### Note

You must read the following SAP Notes before using the [Geo Map](#) component:

SAP Note Number	Description
<a href="#">2238090</a> 	Working with SAP BusinessObjects Design Studio Geo Maps.
<a href="#">2239530</a> 	Design Studio RTL support for Geo Maps.

### Procedure

1. From the [Components](#) view, drag and drop the [Geo Map](#) component into the layout editor.
2. From the [Outline](#) view, add into the [Data Sources](#) folder, the data sources you wish to work with in your [Geo Map](#).

The added data sources are listed in the [Data Source](#) dropdown list in the [Additional Properties](#) tab.

3. Before the user can interact with the layers of [Geo Map](#), you must configure the [Edit Initial View...](#) for each data source to ensure that only one geo dimension appears in the [Rows](#). The measures should appear in the [Columns](#) area.
4. Select the [Geo Map](#) component in the layout panel. In the [Properties](#) and [Additional Properties](#) views, you can configure the [Geo Map](#) properties. For more information on each of these properties, you can refer to the chapter in this guide called [Geo Map](#).
5. You can allow users to interact in multiple ways with the different [Geo Map](#) layers by using the `onSelect()` ; scripting method. For more information on interacting with the [Geo Map](#), you can refer to the chapter in this guide called [Interacting with Geo Maps](#).

### Related Information

[Geo Map \[page 390\]](#)

[Geo Map Settings \[page 216\]](#)

[Configuring Initial Geo Map Settings \[page 216\]](#)

[Interacting with Geo Maps \[page 222\]](#)

[Configuring Geo Map Layers \[page 217\]](#)

## 29.1 Geo Map Settings

To create a well-designed *Geo Map*, you must use data sources containing geo-specific dimensions or measures and you need to have a very good idea of how you would like to display your data. You can choose to display data as shapes (polygons, multi-polygons, lines or multi-lines), points or charts (bubble or pie).

Other important factors that influence a *Geo Map*'s appearance are the selected properties in the *Properties* and *Additional Properties* views.

## 29.2 Configuring Initial Geo Map Settings

### Context

You must use the *Edit Initial View..* to configure the initial settings for the *Geo Map*.

#### **i** Note

- There must be only one geo-specific dimension located in the *Rows* area of the *Edit Initial View...* of your SAP BW or SAP HANA query. The measures should appear in the *Columns* area.
- For pie chart layers, you require two dimensions in the *Rows* area of the *Edit Initial View...*. The first dimension is the geo-specific dimension and the second dimension is used to determine the pie slices.
- If your geo dimension SAP BW attributes contain longitude and latitude, longitude and latitude are assigned to your map layer based on these BW attributes. Your SAP BW attributes should then only be used to get longitude and latitude values and should not be used elsewhere. If your geo dimension does not contain latitude and longitude BW attributes, then longitude and latitude values for your *Geo Map* should be contained in a measure.

It is not possible to display shapes based on SAP BW attributes, as there can be many duplicates. Shapes can only be based on dimension members, not SAP BW attributes.

### Procedure

1. In the *Outline* view, right-click a data source, and in the context menu, select *Edit Initial View...*
2. In the *Edit Initial View...* dialog box, from the pane on the left side of the screen, drag one geo dimension to the *Rows* area.



3. If your SAP BW attributes contain longitude and latitude, expand the geo dimension and its attributes. Right click on the underlying longitude and latitude and select [Add](#).
4. Drag the geo dimension to the [Rows](#) pane.
5. You should hide totals. To do this, right click on the dimension, select ► [Totals Display](#) ► [Hide Totals](#) ▾.
6. Select the [Geo Map](#) in the layout pane. In the [Properties](#) tab, enter the URL for your selected basemap in the [Basemap URL](#) property. It is also an option to leave this blank, if no basemap is required.
7. If required by the basemap provider, you should enter a reference to the relevant basemap copyright information.
8. The [Map Legend Visible](#) property is set to true by default. If you wish to hide the legend on all your [Geo Map](#) layers, you can set this property value to false.

## 29.3 Configuring Geo Map Layers

Once you have selected your data sources, configured your initial view and set the basemap url, basemap copyright (when required) and set your map legend visibility in the [Properties](#) tab, you can now begin to add layers to your [Geo Map](#). Adding or removing layers can be done by using the relevant buttons in the [Additional Properties](#) tab. This layered approach to adding data to your [Geo Map](#) allows the application user to drill down into and interact with the data visualization. Each layer has a unique ID. You can show, hide, rename, and remove the different layers in your [Geo Map](#). You can also change the order in which the layers are applied to the [Geo Map](#) by using drag and drop in the [Additional Properties](#) tab. When you then run the application, the [Geo Map](#) legend reflects the new order of the layers. For each layer, you can select, whether you want to display data in the form of shapes, points, bubble charts or pie charts. Data sources are assigned to individual [Geo Map](#) layers.

### **i** Note

- When configuring each layer, you must ensure that the [ID](#) property contains valid css selectors. A name must begin with an underscore, a hyphen, or a letter, followed by any number of hyphens, underscores, letters, or numbers. If the first character is a hyphen, the second character must be a letter or underscore. The name must be at least two characters long.
- When scripting the [Geo Map](#), the scripting methods `getSelectedLayer()` ; and `getSelectedMember()` ; should be used together for optimal results.

## Binding Data Sources

You can bind different data sources to different layers of your [Geo Map](#). One data source is assigned per layer. However, the same data source can also be used in different layers. Data source selection can be accessed from the [Geo Map Additional Properties](#). You can also use the [Data Selection](#) ellipsis button if you want to have a more fine-grain control over the selection of data in the data source. For example, if you are looking at a layer in the [Geo Map](#) that displays capital cities in the world, you might want to restrict the data selection. When you select your data source, you need to ensure that in the [Edit Initial View...](#), you have a geographic dimension in the [Rows](#) area. The measures should appear in the [Columns](#) area. For more information on using the [Data Selection](#) property, you can refer to the chapters called *Chart* or *Info Chart* within the *User Interface Reference* section.

## GeoJSON

GeoJSON is a standard format for encoding collections of simple geographical features. A GeoJSON file contains a list of polygons, each of which contains a list of properties or metadata that define the polygons and a list of the coordinates, that will be drawn on the map. You need to define how to match the dimension member to the polygon within the GeoJSON file. You have the option of using a custom GeoJSON file, which can be used in local mode, or uploaded onto your BI platform or SAP NetWeaver platform. This file is then accessed from the [Custom GeoJSON File](#) property within the [Additional Properties](#) tab. If you do not specify a custom GeoJSON file, a default GeoJSON file containing countries is used. It contains a standard set of properties such as ISO names and full names. For example, 'DE' is the ISO name for 'Germany'. In the [GeoJSON Mapping Property](#) within the [Additional Properties](#) tab, you will see by default, sample measure names aligned with the corresponding GeoJSON name. You can use this dropdown list to select the appropriate GeoJSON property name for use in your [Geo Map](#). You can use the [GeoJSON Mapping Type](#) to determine how the dimension members will be mapped by selecting either key or text.

## Related Information

[Working with Geo Maps \[page 215\]](#)

### 29.3.1 Configuring a Shapes Layer

#### Context

A shape layer or choropleth layer depicts a collection of polygons, multi-polygons, lines or multi-lines on a map, for example a country, sales region etc. In the [Additional Properties](#) of the [Geo Map](#), the shape layer is the default layer type. To display shapes in a [Geo Map](#), a shapes layer needs to reference a GeoJSON file containing the required shapes. You can use a custom GeoJSON file or you can use the default shapes provided with Design Studio. Shapes are based on dimension members in the [Edit Initial View...](#) For example, if you want to display the states of a country, the dimension 'State' should be in the [Rows](#) area.

#### Procedure

1. Select the [Geo Map](#) component in the [Layout](#) pane and select the [Additional Properties](#) tab.

The additional properties associated with the shapes layer opens by default.

2. Select the [Add Layer](#) button if you have already added a layer and wish to add another one.

A new layer of additional properties is opened above the previous layer.

3. Provide a unique ID for your shapes layer in order to make it easy to identify when all layers are collapsed.
4. The [Show Layer](#) checkbox is selected by default. If you want this layer to be invisible to the application user, you must deselect this checkbox.

5. Select the data source you wish to assign to this shapes layer. Once you have selected the data source, you can also use the [Data Selection](#) ellipsis button to further restrict or modify the number of measures that appear in the [Measure](#) dropdown list for selection.
6. Select from the [Measure](#) dropdown list, the measure you wish to assign to the shape layer. Measures determine the color of the shape to be represented on the [Geo Map](#).
7. Select the shape color gradient using the [Start Color](#) and [End Color](#) color pickers or by entering HEX values. The color gradient is displayed on the map based on the assigned measure value from the data source specified. The color properties are black and white by default. The highest value will use the [End Color](#).

The shapes in this layer will display on the [Geo Map](#) in the color selected.

8. Within the [Classification Type](#) property, select from the two options Quantile or Equal Interval.
9. If you are on the BI platform or the [SAP NetWeaver](#) platform, you can select the ellipsis button within the [Custom GeoJSON File](#) property to select the location of your custom GeoJSON file. Selecting the [Custom GeoJSON File](#) ellipsis button in local mode, opens a [Select Local GeoJSON](#) dialog box, from where you can select your locally stored custom GeoJSON file.
10. Use the [GeoJSON Mapping Property](#) dropdown list to select the property in the GeoJSON file that will be used to display the shape. To assist you with your selection, a text value is available beside the first property within the GeoJSON file. Select the drop down arrow to view it.
11. You can specify how the dimension members will be mapped by selecting either key or text from the [GeoJSON Mapping Type](#) property.

## Related Information

[Configuring Geo Map Layers \[page 217\]](#)

## 29.3.2 Configuring a Points Layer

### Context

You can display data as precise locations on a [Geo Map](#) points layer using values for longitude and latitude. It is necessary that the data source assigned to the layer contains latitude and longitude. To ensure that latitude and longitude coordinates are included in your [Geo Map](#), the SAP BW attributes must be displayed in the [Edit Initial View....](#) If the values for longitude and latitude are in the measures, the geographic dimension should be in the [Rows](#).

There are two options available to you when assigning data to a points layer in your [Geo Map](#):

- You can assign data to a points layer from the [Geo Map](#) additional properties [Latitude](#) and [Longitude](#). These properties display dimension members as measures, allowing you to select how to represent longitude and latitude as custom or default points on a map.
- You can also assign data to a points layer from the SAP BW attributes in the [Edit Initial View...](#)

## Procedure

1. Select the [Geo Map](#) component in the [Layout](#) pane and select the [Additional Properties](#) tab.  
A shapes layer is open by default.
2. If you have already added a layer, and wish to add another one, select the [Add Layer](#) button.  
A new layer is opened above the previous layer.
3. Select the [Points](#) layer type.  
The [Additional Properties](#) are changed to refer only to the points layer type.
4. Provide a unique ID for your points layer in order to make it easy to identify when all layers are collapsed.
5. The [Show Layer](#) checkbox is selected by default. Uncheck this checkbox if you want this layer to be invisible to the user online.

### 6. Note

The longitude and latitude will not appear in the [Data Selection](#) dialog box, if you select the ellipsis button beside the [Data Source](#) property.

Select the data source you wish to assign to this points layer. Once you have selected the data source, you can also use the [Data Selection](#) ellipsis button to further restrict or modify the number of measures that appear in the dropdown list for selection for the [Geo Map](#).

7. Select from the dropdown lists, the longitude value for the [Longitude](#) property and the latitude value for the [Latitude](#) property. The longitude and latitude both appear for the two dropdown lists for the longitude and latitude above of the data selection made.
8. Select a color to customize the color of the marker that indicates the point on the [Geo Map](#).
9. You can upload to the platform any pixel based image file to replace the standard [Geo Map](#) marker, however it is recommended to use PNG files. You can have a different marker for each points layer. Select the ellipsis button beside the [Custom Marker File](#) to select custom markers that are stored on the platform. Selecting the [Custom GeoJSON File](#) ellipsis button in local mode, opens a [Select Local GeoJSON](#) dialog box, from where you can select your locally stored custom GeoJSON file.

The points marker and corresponding legend changes to display the customized marker you have selected.

10. To select an anchor point for the custom marker, select from the options provided in the [Vertical Anchor](#) dropdown list.
11. Use the [GeoJSON Mapping Property](#) dropdown list to select the property in the GeoJSON file that will be used to display the point. To assist you with your selection, a text value is available beside the first property within the GeoJSON file. Select the drop down arrow to view it.
12. You can specify how the dimension members will be mapped by selecting either key or text from the [GeoJSON Mapping Type](#) property.

## Related Information

[Configuring Geo Map Layers \[page 217\]](#)

## 29.3.3 Configuring a Bubble or Pie Chart Layer

### Context

In the context of the [Geo Map](#) component, bubble or pie chart layers are similar to points layers, but with a measure assigned to them. The data source assigned to a bubble or pie chart layer must contain longitude and latitude or reference a GeoJSON file. If longitude and latitude are present, the bubble or pie chart will be displayed at these points or coordinates. If a GeoJSON file is referenced, the bubble or pie chart will be positioned in the middle of the shape. The bubble size and pie slices change, depending on the measures. Both chart layer types display a legend.

#### **i** Note

- There must be only one geo-specific dimension located in the [Rows](#) area of the [Edit Initial View...](#) of your SAP BW or SAP HANA query. The measures should appear in the [Columns](#) area.
- For pie chart layers, you require two dimensions in the [Rows](#) area of the [Edit Initial View....](#) The first dimension listed is the geo-specific dimension and the second dimension is used to determine how the pie is sliced.

### Procedure

1. Select the [Geo Map](#) component in the [Layout](#) pane and select the [Additional Properties](#) tab.  
A shapes layer is open by default.
2. If you have already added a layer, and wish to add another one, select the [Add Layer](#) button.  
A new layer is opened above the previous layer.
3. Change the layer type to a bubble or pie chart by selecting the corresponding icon.  
The [Additional Properties](#) are changed to refer only to the bubble or chart layer type.
4. Provide a unique ID for your charts layer in order to make it easy to identify when all layers are collapsed.
5. The [Show Layer](#) checkbox is selected by default. If you want this layer to be invisible to the user online, you must uncheck this checkbox.
6. Select the data source you wish to assign to this bubble or pie chart layer. Once you have selected the data source, you can also use the [Data Selection](#) ellipsis button to further restrict or modify the number of measures that appear in the dropdown list for selection for the [Geo Map](#).
7. If you want to display the bubble or pie chart as a location, in the [Longitude](#) and [Latitude](#) dropdown menus, select the relevant measure, containing latitude and longitude values.

#### 8. **i** Note

The [Color](#) property is only available for the bubble chart layer, not for the pie chart layer.

Select a color to customize the color of the bubble charts in the [Geo Map](#) layer.

9. Select a measure in the [Measure](#) dropdown menu. The size of the bubbles and how the pie is sliced are determined by this measure.

- 
10. If you want to display the bubble or pie chart at the center of a shape or polygon, and you are on the BI platform or the [SAP NetWeaver](#) platform, you can select the ellipsis button within the [Custom GeoJSON File](#) property to select the location of your custom GeoJSON file. Selecting the [Custom GeoJSON File](#) ellipsis button in local mode, opens a [Select Local GeoJSON](#) dialog box, from where you can select your locally stored custom GeoJSON file.
  11. Use the [GeoJSON Mapping Property](#) dropdown list to select the property in the GeoJSON file that will be used to display the bubble or pie chart. To assist you with your selection, a text value is available beside the first property within the GeoJSON file. Select the drop down arrow to view it.
  12. You can specify whether key or text of the geo dimension member is used when mapping to the GeoJSON file by selecting from the [GeoJSON Mapping Type](#) property.

## Related Information

[Configuring Geo Map Layers \[page 217\]](#)

[Configuring Initial Geo Map Settings \[page 216\]](#)

## 29.4 Interacting with Geo Maps

Through scripting and working with the [Geo Map](#) properties, you can allow the application user to interact with the map in a number of ways.

### Drilldown into different layers

You can script the [Geo Map](#) so that when the user clicks on the map, the map returns a different layer. This hides or shows a layer based on an [On Select](#) event and allows the user to drilldown into the map layers to reveal different data.

### Change the basemap

You can script a component in the [Geo Map](#) to allow the user to select a different basemap while running the application.

### Center the map

You can script the [Geo Map](#) to change the center the map around the data contained in a layer, based on where the user clicks on the map. This repositions the map, so that the area that contains the data becomes the center of the map.

---

## Pan and zoom

The user can use the mouse and/or the default zoom buttons to pan across the entire map and to zoom in and out on the selected area on the map.

## Tooltips and legends

As the user hovers over different areas on the map, the tooltip changes to display the geo-specific dimension and its corresponding measure value. If you have scripted the layers to show the legend, this information is also displayed in the legend.

## 30 Working with Drag and Drop in Applications and Crosstabs

### Prerequisites

When working with drag and drop functionality in applications and crosstabs, please note the following points:

- If your Design Studio is based on SAP HANA as platform, drag and drop operations for dimension members are not supported. Only dimensions can be dragged and dropped.
- Drag and Drop cannot be used when the crosstab is used in the planning mode, that means if the crosstab contains input ready cells.
- Drag and Drop is not supported for mobile scenarios (for example, applications running on the iPhone or iPad)

### Enabling/Disabling Drag and Drop

Drag and Drop can be enabled and disabled on application level and on component level:

- On Application Level  
Use the application property *Drag and Drop between Components* to specify if drag and drop operations between different components are allowed. This property is set to *false* by default, which means that drag and drop operations cannot be carried out between components. Drag and Drop operations within one single component are still possible, if the crosstab (component) level property is set to *true* (see next section). Set this property to *true* if you want to enable drag and drop operations between components (for example, between the Navigation Panel and the Crosstab).
- On Crosstab (component) Level  
Use the crosstab property *Drag and Drop enabled* if you want to enable drag and drop operations within the crosstab. The property is set to *false* by default, what means that the crosstab does not allow any drag and drop operations. If this property is set to *true*, the crosstab:
  - enables internal drag and drop operations. Thus the application user can drag and drop dimensions and members within the crosstab and remove dimensions and members by dragging and dropping them outside the area of the crosstab.
  - accepts external drops of dimensions from other components (for example, the navigation panel). This only works if the application property *Drag and Drop between Components* is set to *true* as well.

For further information about the respective application and crosstab properties, see [Properties of the Application \[page 325\]](#) and [Crosstab \[page 372\]](#).

In the following documentation chapters, you will find more specific information about different drag and drop aspects.



# 30.1 Dragging a Dimension or Dimension Member

## General aspects

Dimensions or members that can be dragged can be easily identified: the mouse cursor changes when you hover above them. Drag and drop operations in the crosstab can only be performed for one single dimension or one single dimension member. You cannot select multiple members and drag the members in the selection around the crosstab, even if the *Selection Type* property of the crosstab has been set to *multi*.

The mouse cursor changes to the "move" cursor type ( dimension "Product", for example):

		Prod
Calendar Year/Month	Product	Price per UM
JAN 2003	Automatic umbrella	9.18
	Camera Connector	45.90
	Flatscreen Vision I	1,064.00

When you start to drag a dimension or member, the drag ghost displays all cells that belong to the respective dimension. This means text fields and attributes are displayed. In this example, the dimension "Product" has the attribute "Price per UM", which is displayed in the drag operation as well:

			Product group
Calendar Year/Month	Product	Price per UM	
JAN 2003	Product		Phone
	Automatic umbrella	9.18	Fax
			Internet
			Others
	Camera Connector	45.90	Phone
			Fax
			Internet
			Others
	Flatscreen Vision I	1,064.00	Phone
			Fax
			Internet

For all drag (and drop) operations, dimension and member data with all related fields like texts or attributes are always treated as a single unit. In this example, this means it is not possible to drag and drop “Price per UM” separately, because it is an attribute for dimension “Product”. This behavior is also valid for the “Product” cell that represents the “Product” dimension's text. Therefore, you can grab and drag any cell (text, attribute) for a dimension, in order to move the whole cell block that belongs to that dimension.

Drag and drop operations within the crosstab as well as drag drop operations from the navigation panel to the crosstab can be canceled before dropping the element by pressing the **ESC** key on the keyboard.

## The dimension header split cell

Typically, when no scaling factor is specified on the rows or on the columns axis in the drilldown, the pivot cell of the dimension header hosts the respective dimension from the rows and from the columns axis. In order to drag a dimension out of this cell, you need to render this cell as a "split" cell with a diagonal separator. The two areas created by this separator determine which dimension will be dragged when dragging is started from a position within the cell. In the following example, the “Currency” dimension on the rows axis is dragged when starting the drag operation from the lower left cell area:

		0BC_COUNT		
		K4/2004		
	Fiscal year	Calendar year, 4 spec. periods 2004		
	Currency   Calendar Year/Month	JAN 2004	FEB 2004	MAR 2004
	Euro			
	Euro			
	Euro			
	Euro			
	Euro			
	Euro			
	Euro	1		
	Euro			
	Euro			
	Euro	1		
	Euro			

Starting from the same cell, but using the upper right cell area, the “Calendar Year/Month” dimension on the columns axis is dragged:

		0BC_COUNT		
		K4/2004		
		Calendar year, 4 spec. periods 2004		
	Calendar Year/Month			
	Fiscal year			
	Currency   Calendar Year/Month	JAN 2004	FEB 2004	MAR 2004
	Euro			
	Euro			
	Euro			
	Euro			
	Euro			
	Euro			
	Euro	1		
	Euro			
	Euro			
	Euro	1		
	Euro			

Structures usually do not have a text rendered in the crosstab. The drag ghost always displays a text, and hence a split cell, as shown below, allows you for example to grab and drag the measure structure out of the pivot cell:

Calendar Year/Month	JAN 2004			
Fiscal year	1/4/2004	Measures		
	Calendar year, 4 spec. periods 2004			
Currency	0BC_COUNT	0BC_DATE2	0BC_DATE3	
Euro				
Euro				
Euro				
Euro				
Euro				
Euro				
Euro	1	1/13/04	1/13/04	
Euro				
Euro				
Euro	1	1/29/04	1/29/04	
Euro				

## 30.2 Removing a Dimension or Dimension Member

You can remove a dimension or dimension member from the drilldown by dragging the element out of the visible crosstab area. When you do this, the drag ghost will show a trash can symbol to indicate that dropping the element at the current position will remove the element:

			Formula 3			CALC			FORMULA		
			Product group	DS30	DS20	Overall Result	DS30	DS20	Overall Result	DS30	DS20
				Hardware software	Accessories+space		Hardware software	Accessories+space		Hardware software	Accessories+space
Calendar Year/Month	Product	Price per UM	ST	ST	ST	\$	\$	\$	\$	\$	\$
JAN 2003	Automatic umbrella	9.18	Phone		1,704,488	1,704,488		1,955,470.00	1,955,470.00		7,821,880.00
			Fax		2,480,748	2,480,748		2,846,088.50	2,846,088.50		11,384,354.00
			Internet		1,255,412	1,255,412		1,440,321.00	1,440,321.00		5,761,284.00
			Others		599,876	599,876		688,267.50	688,267.50		2,753,070.00
	Camera Connector	45.90	Phone		5,294,964	5,294,964		3,032,347.50	3,032,347.50		12,129,390.00
			Fax		7,436,492	7,436,492		4,258,818.00	4,258,818.00		17,035,272.00
			Internet		4,142,948	4,142,948		2,372,611.50	2,372,611.50		9,490,446.00
			Others		1,874,248	1,874,248		1,073,347.50	1,073,347.50		4,293,390.00
	Flatscreen Vision I	1,064.00	Phone		2,136,620	2,136,620	2,817,174.50		2,817,174.50	11,268,698.00	
			Fax		3,145,856	3,145,856	4,147,696.50		4,147,696.50	16,590,786.00	
			Internet		4,545,556	4,545,556	3,133,448.00		3,133,448.00	8,525,648.00	
			Others		1,874,248	1,874,248					

### Note

Keep in mind that structures cannot be removed from the drilldown unless they only contain a single member. If you want to remove members of a structure (for example, the measure structure), at least one structure member must remain in the structure. Otherwise you cannot remove a member.

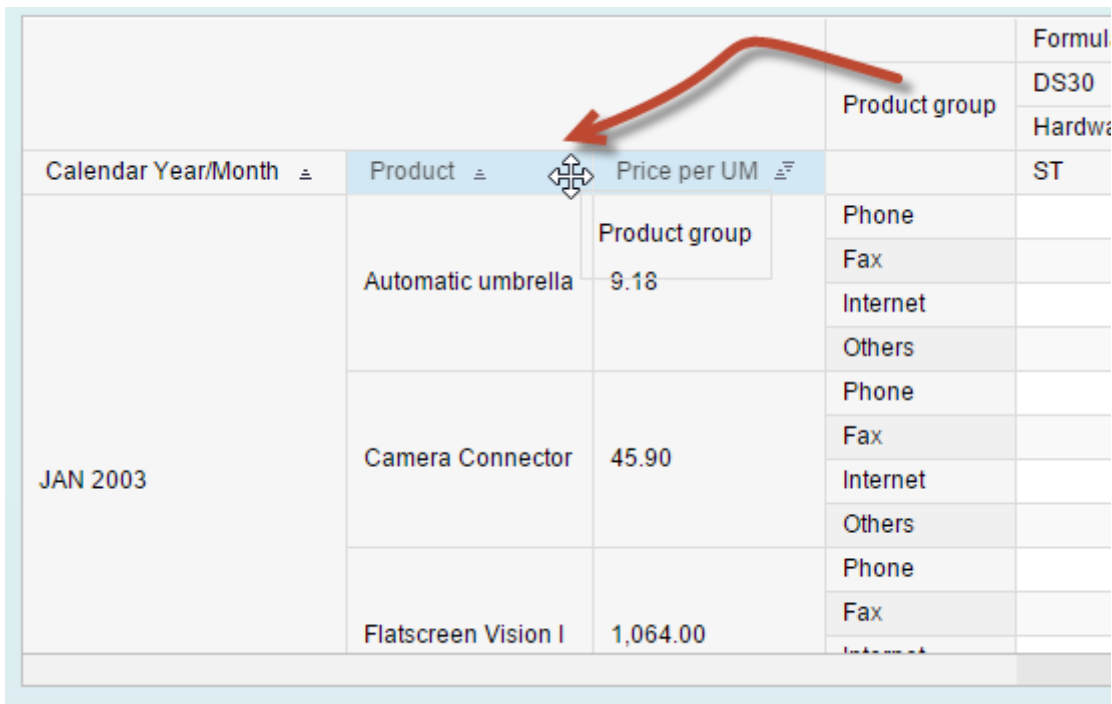
## 30.3 Dropping Dimensions or Dimension Members

Drop targets for dimensions and dimension members can be other dimensions (or, in the case of members, other members), or the respective drop areas between dimensions and members. Dropping a dimension on another dimension or a member on another member exchanges the two elements, whereas dropping a member/dimension on a respective drop area inserts the member/dimension at the respective drop area position.

A valid drop target is highlighted accordingly, for example, when you drag the “Product Group” dimension onto the “Product” dimension or the “Phone” member onto the “Internet” member, as in the following examples:

### Exchange operations

Dimension drop:



Calendar Year/Month	Product	Price per UM	Formula
JAN 2003	Automatic umbrella	Product group	DS30
			Hardware
			ST
	Camera Connector	Phone	
		Fax	
		Internet	
		Others	
	Flatscreen Vision I	Phone	
		Fax	
		Internet	
		Others	

#### **i** Note

The drop target highlighting also highlights all cells that belong to the dimension, including text/key and attribute field cells.

A "split cell" that is a pivot cell that hosts both a dimension on the rows axis and columns axis (for more information, see section *The dimension header split cell* in [Dragging a Dimension or Dimension Member \[page 225\]](#)), cannot be a drop target. In order to drop onto a dimension that is contained in a split cell, you first have to swap or move the dimension out of this cell.

Member drop:

			Formula 3	
			Product group	DS30 DS20
				Hardware software Accessori
Calendar Year/Month	Product	Price per UM	ST	ST
JAN 2003	Automatic umbrella	9.18	Phone	
			Fax	
			Internet	
			Others	
	Camera Connector	45.90	Phone	
			Fax	
			Internet	
			Others	
	Flatscreen Vision I	1,064.00	Phone	2,136,620
			Fax	3,145,856
			Internet	4,646,568
			Others	

If you release the left mouse button at the respective positions, the dimensions/members are exchanged with each other.

## Insert/move operations

You can perform an insert operations by dragging elements and dropping them on drop areas, which are displayed as horizontal or vertical lines when you hover over them:

The following example shows how to drop a dimension on the rows axis by dragging the “Product Group” dimension between “Calendar Year/Month” and “Product”.

### Note

Since “Product” and “Price per UM” belong to the same dimension (“Product”), you cannot position “Product Group” between these two, and hence no drop area appears when you hover over the respective position. The cell to the left of the drop area, is grayed out, except when the drop area at the very left (or very right for RTL) or the drop area at the very top is highlighted. In this case, the grayed out cell is the cell next to (left, right or below) the cell where the dragged dimension will be inserted.

Calendar Year/Month	Product	Price per UM
JAN 2003	Automatic umbrella	9.18
	Camera Connector	45.90
	Flatscreen Vision I	1,064.00

The following example shows how to drop a dimension on the columns axis by positioning the “Distribution Channel” structure between “Product Group” and the measure structure.

If you release the left mouse button, the dragged dimension is moved into the new position.

If you want to insert members, proceed like in the following examples:

- Moving dimension member “Others” ahead of “Fax”. This means insert the member between “Phone” and “Fax”.

			Formula 3	
			Product group	DS30
				Hardware software
				Access
Calendar Year/Month	Product	Price per UM		ST
JAN 2003	Automatic umbrella	9.18	Phone	
			Fax	
			Internet	
			Others	
	Camera Connector	45.90	Phone	
			Fax	
			Internet	
			Others	
	Flatscreen Vision I	1,064.00	Phone	2,136,620
			Fax	3,145,856
			Internet	1,646,500
			Others	

- This procedure also works for other elements on the rows axis and columns axis, like inserting “DS30” between “DS20” and the “Overall Result”.

	Product group	Formula 3		CALC	
		DS30	DS20	Overall Result	DS30
		Hardware software	Accessories+space	DS30	Hardware sof
Price per UM ₪		ST	ST	ST	\$
9.18	Phone		1,704,488	1,704,488	
	Fax		2,480,748	2,480,748	
	Internet		1,255,412	1,255,412	
	Others		599,876	599,876	
45.90	Phone		5,294,964	5,294,964	
	Fax		7,436,492	7,436,492	
	Internet		4,142,948	4,142,948	
	Others		1,874,248	1,874,248	
1,064.00	Phone	2,136,620		2,136,620	2,817,1
	Fax	3,145,856		3,145,856	4,147,6
	Internet	4,646,568		4,646,568	3,434,4

## 30.4 Drag and Drop Constraints

When using drag and drop, please note the following points:

- Drag and drop only works for single elements under the mouse cursor. This means only one dimension/ dimension member at a time can be dragged and dropped.
- Structures can only be removed if they contain no more than one member.
- Dimensions can only be dropped on dimensions and the respective drop areas.
- Dimension members can only be dropped on dimension members and the respective drop areas.



- Members can only be dragged and dropped within the previous peer dimension's member space. Example:

			Formula 3		
			Product group	DS30	DS20
				Hardware software	Access
Calendar Year/Month	Product	Price per UM		ST	ST
JAN 2003			Phone		
			Fax		
			Internet		
			Others		
	Automatic umbrella	9.18	Phone		
			Fax		
			Internet		
			Others		
	Camera Connector	45.90	Phone		
			Fax		
			Internet		
			Others		
	Flatscreen Vision I	1,064.00	Phone		2,136,620
			Fax		3,145,856
			Internet		1,646,500
			Others		

In this example, the structure member "Others" can only be moved within the space that is spanned by the previous peer dimension's member, which is "Camera Connector".

- Hierarchy members can only be dragged and dropped within the same drill level
- Hierarchy members cannot be removed.
- A "split cell" (pivot cell) is not a drop target.
- "Result" members cannot be dragged and cannot be drop targets for other members.

## 30.5 Dragging and Dropping Between Crosstab and Navigation Panel

If you want to drag and drop between the navigation panel and the crosstab, note the following points:

- You first have to enable drag and drop between different components by setting the application property *Drag and Drop between Components* to *true*. For more information, see section *Enabling/Disabling Drag&Drop in Working with Drag and Drop in Applications and Crosstabs* [page 224]
- Only dimensions can be dragged from the navigation panel and dropped on crosstab dimensions and their respective drop areas.
- You can cancel the drag operation by pressing the **ESC** key on the keyboard.
- The navigation panel can only act as a dimension drag source for the crosstab. Members cannot be dragged from the navigation panel, and the navigation panel is not a drop target for members or dimensions that are dragged from a crosstab.
- If dimensions, which are already contained on any axis of the crosstab drilldown, are dragged from the navigation panel to the crosstab, the crosstab will not accept the drop, neither for dimension cells nor for drop areas. Only dimensions that are not already in the drilldown are accepted. Thus, if you want to reorder dimensions, you need to do this either within the crosstab or within the navigation panel. You cannot reorder by dragging and dropping a dimension already contained on an axis from the navigation panel to the crosstab.

## 30.6 Drag and Drop in Crosstabs with Property "Display Repeated Texts"

If you set the crosstab property *Display Repeated Texts true*, this affects how drag and drop works with members. Dragging and dropping of dimensions will remain unaffected.

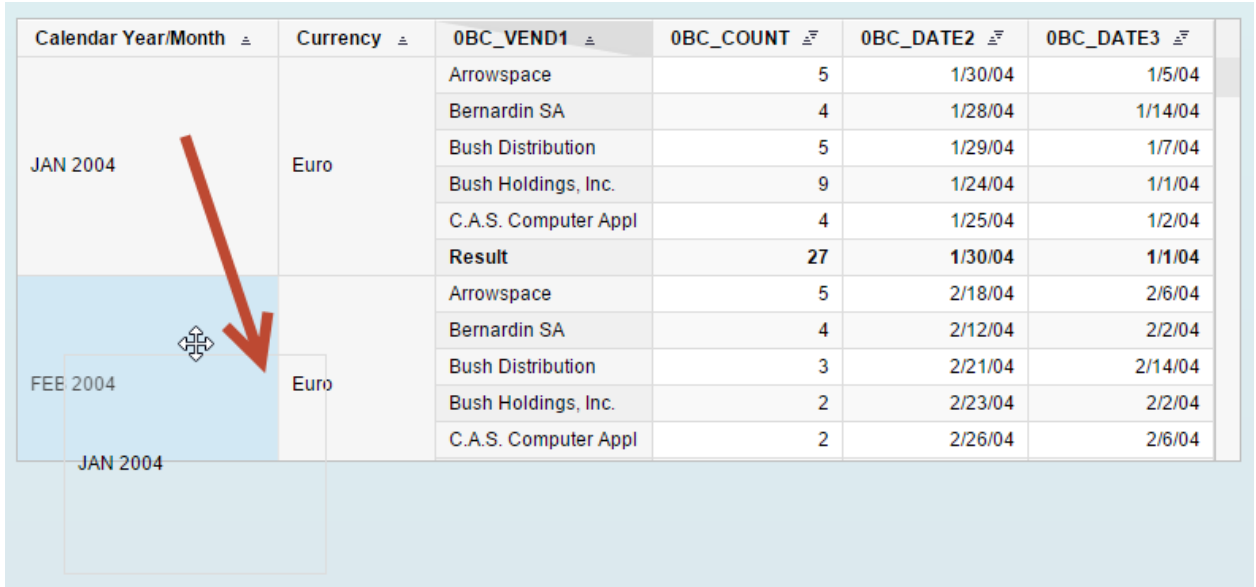
In a crosstab with repeated texts, all drag and drop operations are applied logically to the dimension members with the finest drilldown granularity, despite the fact that the visualization of the drag and drop operation is applied to a whole crosstab row or column, and you can start by grabbing any cell in the respective row or column. Hence, it is not possible to drag and drop the lines everywhere. Instead the system checks what is allowed and what is not allowed, based on the finest drilldown granularity dimension member. The following examples explain this behavior:

Calendar Year/Month	Currency	OBC_VEND1	OBC_COUNT	OBC_DATE2	OBC_DATE3
JAN 2004	Euro	Arrowspace	5	1/30/04	1/5/04
JAN 2004	Euro	Bernardin SA	4	1/28/04	1/14/04
JAN 2004	Euro	C.A.S. Computer Appl	5	1/29/04	1/7/04
JAN 2004	Euro	Bush Distribution	9	1/24/04	1/1/04
JAN 2004	Euro	Bush Holdings, Inc.	4	1/25/04	1/2/04
JAN 2004	Euro	C.A.S. Computer Appl	27	1/30/04	1/1/04
JAN 2004	Euro	Result			
FEB 2004	Euro	Arrowspace	5	2/18/04	2/6/04
FEB 2004	Euro	Bernardin SA	4	2/12/04	2/2/04
FEB 2004	Euro	Bush Distribution	3	2/21/04	2/14/04
FEB 2004	Euro	Bush Holdings, Inc.	2	2/23/04	2/2/04
FEB 2004	Euro	C.A.S. Computer Appl	2	2/26/04	2/6/04

To start drag and drop, the user can grab and start dragging any cell in the row, for example "JAN 2004" or "Euro" or "C.A.S Computer Appl". The drag visualization shows the whole line, but the drag and drop operation is effectively applied to the member of "OBC\_VEND1" because this is the finest granularity dimension on the rows in the given drilldown. Therefore the user is actually logically dragging "C.A.S Computer Appl".

As a result, the line cannot be dragged and dropped somewhere outside of the "JAN 2004" block of "Calendar Year/Month". Member reordering of OBC\_VEND1 members is only possible within this block. This behavior is very similar to the behavior you would observe if the property *Display Repeated Texts* would be set to *false*.

In fact, using the property *Display Repeated Texts* in the crosstab is even more restrictive when using drag and drop for members. The following example, show the same crosstab as in the example above but with the property *Display Repeated Texts* set to *false*:



Calendar Year/Month	Currency	OBC_VEND1	OBC_COUNT	OBC_DATE2	OBC_DATE3
JAN 2004	Euro	Arrowspace	5	1/30/04	1/5/04
		Bernardin SA	4	1/28/04	1/14/04
		Bush Distribution	5	1/29/04	1/7/04
		Bush Holdings, Inc.	9	1/24/04	1/1/04
		C.A.S. Computer Appl	4	1/25/04	1/2/04
		Result	27	1/30/04	1/1/04
FEB 2004	Euro	Arrowspace	5	2/18/04	2/6/04
		Bernardin SA	4	2/12/04	2/2/04
		Bush Distribution	3	2/21/04	2/14/04
		Bush Holdings, Inc.	2	2/23/04	2/2/04
		C.A.S. Computer Appl	2	2/26/04	2/6/04

In this example, the “JAN 2004” member can be exchanged with the “FEB 2004” member because the drag and drop operation operates on the whole block, which these members span in the dimensions to the right of the crosstab.

Since the drag and drop operation does not operate on a selection in the crosstab, it is not possible to achieve this behaviour as described above when the property *Display Repeated Texts* is set to *true*. This is because there would only be a single row drag and drop action that is applied to a OBC\_VEND1 member, and dragging and dropping a single “JAN 2004” row between multiple “FEB 2004” rows would result in an invalid state.

## 31 Working with Scorecards








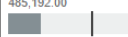



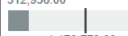







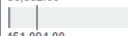
The Scorecard component is used for creating dashboards and reports with controlled visualization content and restricted navigation possibilities. Scorecards can help you to make complex result sets browsable by using user friendly visualizations (like charts) and highly customizable layouting properties.

It is not recommended to use the Scorecard component in applications with dynamic data sources and data sources with free multidimensional navigation (slice and dice). The predefined data selections in the Scorecard can turn invalid in such scenarios.



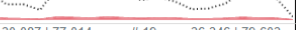









### Using the Scorecard

You can add a Scorecard component to an analysis application to allow users to view the data as a highly customizable table as well as micro charts in columns. Depending on your configuration of the Scorecard column properties, each column can display information in the form of a text, image or charts. Each column property can be connected to a selection of the assigned data source.

The following pictures show examples of applications using scorecards. This first one shows a scorecard with several comparison charts and the second trend charts :

Region	Product Group	Billed Quantity	Billed Quantity vs Sales Value	Production Costs Percentage	Difference Sales vs Costs	Billed Quantity vs Sales Value
West   80504	Bag & Outdoor   DS10	25,294.00	 1,178,447.00	 106,023.00 1,284,470.00	 Δ 1,178,447.00	 106,023.00 1,284,470.00
West   80504	Accessories   DS20	119,894.00	 783,555.00	 485,192.00 1,268,747.00	 Δ 783,555.00	 485,192.00 1,268,747.00
West   80504	Office   DS30	93,871.00	 857,843.00	 312,936.00 1,170,779.00	 Δ 857,843.00	 312,936.00 1,170,779.00
West   80504	Total	239,059.00	 2,819,845.00	 904,151.00 3,723,996.00	 Δ 2,819,845.00	 904,151.00 3,723,996.00
North East   80502	Bag & Outdoor   DS10	8,739.00	 414,262.00	 36,832.00 451,094.00	 Δ 414,262.00	 36,832.00 451,094.00


< 1 2 3 4 >

Region	Product Group	Billed Quantity	Billed Quantity by Customers	Product Distribution (Amount vs Revenue)	Revenue and Target by last Months
West   80504	Bag & Outdoor   DS10	25,294.00	⇐8,034.00 # 19 424.00⇒ 	⇐424.00   424.00 9,532.00   9,532.00 	⇐8,034   92,773 # 19 424   11,481⇒ 
West   80504	Accessories   DS20	119,894.00	⇐30,087.00 # 19 36,346.00⇒ 	⇐4,661.00   4,661.00 41,036.00   41,036.00 	⇐30,087   77,014 # 19 36,346   79,603⇒ 
West   80504	Office   DS30	93,871.00	⇐17,523.00 # 19 20,905.00⇒ 	⇐3,005.00   3,005.00 33,242.00   33,242.00 	⇐17,523   55,733 # 19 20,905   79,602⇒ 
West   80504	Total	239,059.00			
North East   80502	Bag & Outdoor   DS10	8,739.00	⇐7,363.00 # 7 6,990.00⇒ 	⇐2,278.00   2,278.00 7,596.00   7,596.00 	⇐7,363   91,038 # 7 6,990   86,163⇒ 

< 1 2 3 4 >

#### i Note

Please read the following SAP Note before using the [Scorecard](#) component:

SAP Note Number	Description
2226283 	Working with SAP BusinessObjects Design Studio 1.6 + Scorecard

1. Drag and drop the [Scorecard](#) component from the [Components](#) view into the layout editor.
2. Add a data source to your application (in the [Outline](#) view right-click [Data Sources](#) and choose [Add Data Source...](#)) and assign it to the Scorecard.
3. (Optional) You can configure the Data Source by using the [Edit Initial View...](#) function (right-click the data source in the [Data Source](#) folder and choose [Edit Initial View...](#)). For best fit, place the dimensions in rows and measures in columns.
4. Select the [Scorecard](#) component in the [Layout](#) folder of the [Outline](#) view and configure the Scorecard properties in the [Properties](#) and [Additional Properties](#) views.
5. (Optional) You can allow users to interact in multiple ways with the Scorecard component by using the `onSelect()` ; scripting method. For more information on interacting with the Scorecard, see [Interacting with Scorecards \[page 242\]](#).

## 31.1 Scorecard Configuration

To create a well-designed Scorecard, you must use a data source containing dimensions in rows. These dimensions are made available for the row scope definition. For best fit, place the measures in columns. You can also place the measures in rows. In this case, the role of measures changes and they are handled as normal members of a measure dimension.

As soon the data source drilldown is defined, you can start configuring the scorecard.

To configure a scorecard, follow the steps in sequential order.

1. Define the row scope
2. Generate the initial scorecard

### Caution

Once you have started the configuration of the scorecard, it is not recommended to make any changes in the data source definition (for example, removing or adding dimensions). For this reason, it is recommended to use a separate data source which is assigned to the scorecard component or to assure that there is no runtime functionality for this data source which can influence the drill down state. Filtering on the dimensions is allowed and will not invalidate the scorecard component.

## Related Information

[Defining the Row Scope \[page 238\]](#)

[Generating the Initial Scorecard \[page 238\]](#)

## 31.1.1 Defining the Row Scope

In the *Additional Properties* sheet of the scorecard, choose a dimension of your data source out from the dropdown box for the row scope definition. You can select the first, second or any other dimension from the rows. The chosen row in the scorecard has the scope of the selection. Every row will become a single member selection of the chosen dimension and all dimensions which are placed higher in the rows. The other dimensions which are not contained in the row scope definition are free for any use in the columns (for example, for charts).

### Example

The data source assigned to the scorecard has four dimensions in the rows (in the sequential order): "Customer", "Region", "Product Group" and "Time".

You have selected as row scope the dimension "Customer". The system assigns every row to a concrete customer as the member of the dimension Customer (for example, "SAP SE"). You can then create charts in other columns for "Region", "Product Group" and "Time Dimensions".

You have selected as row scope the dimension "Region". The system assigns every row to a concrete customer and region as combination of the members of the dimensions "Customer" and "Region" (for example, "SAP SE | EUROPE" or "SAP SE | ASIA"). You can then create charts in other columns for product group and time dimensions. As the dimension "Region" is already in the row scope, it cannot be used for any other charts. You can assign a text to the members of the region dimension as text content.

## 31.1.2 Generating the Initial Scorecard

The scorecard component is a highly customizable component. It does not reflect the data source result set directly like the crosstab component does. It allows you to specify every column of the scorecard separately and decide which result set selection should be used for any visualization in the concrete column.

The configuration of columns can be done manually by using the *Add New Column* function or you can generate an initial scorecard for a quicker startup by clicking on the respective button.

When you use the manual creation, you can define column by column of the scorecard.

When you use the automatic generation of an initial scorecard, the following columns are generated by the system based on the result set:

- for every dimension included in the rows scope – one column with dimension members
- for the first measure – one dimension displaying the measure sum of the row scope definition
- for the first dimension which is outside of the row scope – a column with trend chart displaying the measure members for that dimension as line or column chart.
- if the data source has more than one dimension – a column with comparison chart displaying the delta between the first and second measure

### Note

Keep in mind that the automatic generation has only a "show case" character to speed up the initial creation. It does not analyze in full extend any specialties of the data source – for example, it is assuming that all dimensions have result members activated. You have to check all selections and properties of the generated columns and assure that the selected data is fulfilling your requirements.

## 31.2 Basic Scorecard Concepts

The Scorecard consists of the following areas:

- **Group Headers**  
Group Headers allow you to group columns in custom groups, which can be used for special descriptions.
- **Columns**  
Columns allow you to visualize the result set, based on chosen selections and visualizations.

A column is defined by:

- the column header. The header allows you to place a description of the column.
- the column content. The column content allows you to visualize the data per row, based on the row scope definition and the chosen visualization.

While configuring the columns, you can

- add a new column after the last existing column
- insert a new column before the selected one
- delete existing columns
- copy existing column (copy is inserted to the right of (in case of RTL to the left of) the existing column)
- move a column up and down (the group assignment is adjusted automatically)

When configuring group headers, you can:

- group two or more group headers together
- ungroup a header, which was grouped before

## 31.3 Configuring Initial Scorecard Settings

Once you have generated an initial scorecard, you can start to configure the scorecard settings in detail. To configure the scorecard, you have to set the properties in the [Properties](#) and the [Additional Properties](#) sheet.

The properties in the properties sheet affect global settings like:

- heights for group headers, headers and rows
- interaction behavior
- navigation behavior

The properties in the additional properties sheet are used for the column definition and customizing.

### 31.3.1 Configuring Scorecard Columns

To configure a scorecard column, select a column in the [Columns and Group Headers](#) area in the Additional Properties sheet and click on the [Column Properties](#) tab.

A column in the scorecard component has a [Width](#) property in pixels and can be configured with [Automatic Column Size](#). All columns, which are marked as [Auto Size](#), are resized accordingly (the column width is increased

---

or decreased to fit the available free space and the column width ratio is retained). A column can also be hidden (property [Column Hidden](#)). To make a column visible, use the corresponding scripting method.

## 31.3.2 Configuring Scorecard Headers and Group Headers

As a header is bound to the column, you need to select the column first, in order to specify the header. Select a column in the [Columns and Group Headers](#) area in the [Additional Properties](#) sheet, click on the [Header Layout](#) tab, to see all available properties and set them according to your needs. The header layout properties consist of the [Column Header Layout](#) and the [Column Header Font](#) properties.

You can hide the header area if you want to show merged columns with a single group header description.

The group header is initially invisible and can be activated by selecting the [Show Group Header](#) function. Then you can select a group header that you want to define. Group headers are special because they can be merged between columns. This allows you to use a description across multiple columns. There are no other special settings for the group header. The [Group Header Content](#) is a text, which can be directly maintained or bound to the data selection.

## 31.3.3 Configuring the Scorecard Cell Content

Each cell (whether it is a header, group header or content cell) has a bunch of properties for layout and text design. Text properties are only available for cells which support the text inside.

In the layout area, you can find different properties for the borders, padding, and background settings. In the text area, you can define the font name, size, color and all other settings.

There are two special settings for the cell layout and cell font:

- Cell Visible  
This property hides the cell, other properties do not apply
- Text Wrapping  
This property allows the text to be wrapped. A wrapped text is not positioned in the vertical middle, but to the top of the cell. You can influence the position by changing the top padding of the cell.

## 31.3.4 Defining the Scorecard Column Content

The main content of the scorecard are the columns. A column consists of four parts:

- column content
- column cell layout
- column header layout
- column properties

The column content is dependent of the selected column template. The scorecard has three column template types:



- content display for values, texts and images – for single selection content
- comparison chart as Bar, Bullet, Delta Bar, Fraction Pie – for comparison of two single selections
- trend chart as Line, Column, Bubble – for list of multiple cells selection

The properties that are available for every column template type can be maintained manually (in this case, the properties are set with same value for all rows) or they can be bound to a selection and are evaluated for each row. The binding allows you to reflect the data conditions in the row visualizations.

After each property description you can find functions which helps you with the configuration:

- common properties (like cell background) can be applied to all columns
- header properties can be applied also to group headers
- group header properties can be applied to all group headers
- all properties can be reset to a default value (the configuration will be removed)

### 31.3.5 Defining the Textual Content

The template for textual content can be used to display members from dimensions, attributes and the data from selected measures. It allows you to set basic properties for the content, and activate icons or images. The main property is the *Text* property. You can select any content to display as text in the cell.

### 31.3.6 Configuring Trend Charts

Trend charts can help you to visualize data series from dimensions which are not included in the row scope definition.

The column chart can be used when you have a small number of data points and only one axis.

The line chart can be used when you have a bigger number of data points and only one axis. In addition, you can add target data points which are visualized by a dotted line.

The bubble chart can be used when you want to compare the distribution of data and you have more than one axis (two or three). The bubble chart displays the actual data in the Y axis, the target data in the X axis and can visualize the bubble size on an additional Z axis.

### 31.3.7 Configuring Comparison Charts

Comparison charts can help you to visualize data and compare it to other data.

The bar chart is used for single value display.

The bullet chart is used to display two values (actual and target) with a third value in the background (forecast). In addition you can define up to five threshold values.

The delta bar chart is used to compare the delta value between two values.

The fraction pie chart is used to express a fraction as a percentage of the pie.

## 31.4 Interacting with Scorecards

The scorecard component supports interaction in the form of selections.

Cell selection is activated by default. In the *onSelect* event you can access the current selection of the dimensions specified in the row scope and the ID of the selected column.

You can use the property *Selection Type* to activate the row selection. The property is set to *None* by default. You can activate *Single Selection*, *Multiple Selection* and *Multiple (Toggle) Selection*.

In the event you get the current selection of the dimensions specified in the row scope. If the user selects a cell, he/she can access the selected column ID. If the user selects a row, the column ID is empty.

## 31.5 Binding the Properties of Scorecards to Data Sources

Property binding allows filling in the value of the property by using the data source. Depending on the property type, there are different ways to fill the proper content for the scorecard property:

- direct binding: The bound value is used directly as property value
- indirect (conditional) binding: The bound value must be translated into the proper property value

Additionally, depending on the property type it is also possible to bind an aspect item of the data set to the property. The data set "aspect" can include the following items:

- value of a data cell
- content of master data information
- content of a dimension or measure
- other metadata content which is available on the data set

### Property Binding Basics

Almost all scorecard properties in the *Additional Properties* sheet can be bound to a data source. The binding reads out the value of the chosen aspect and places the value into the property value. In case of direct binding the value is used directly in the property. All values will be validated according to the property type. If the bound value is returning a value which cannot pass the validation, the default value will be applied. It is also possible to specify conditional binding. There are up to nine conditions possible.

### Conditional Binding

Conditional binding enables the creation of indirect binding of data source aspects into the property value. You can specify up to nine different conditions. The system will read out the bound value from the data source and evaluate the conditions taking the *Conditional Format Value* of the conditional binding key which equals to the bound value.

Example: You want to bind a background color to the conditional format. If the format is returning "3" you want to make the color "red". For this you have to select the binding type [Conditional Format Value](#) and select its properties. Then you have to activate conditional binding and insert a new entry (in addition to the default entry). The key of the entry must be **3** and the value **red**.

#### **i** Note

Conditional binding can be used for all binding types, there is nevertheless a restriction to nine values.

## 31.5.1 Properties with Single Values

The majority of the properties which can be bound to a data source are of the single value type. This means, in case you bind these properties to an aspect of the data source, these properties must get a single value as content. There are multiple binding types to select for single value properties:

- [Single data cell value](#)
- [Conditional Format Value](#)  
-> details
- [Attribute Member Content](#)
- [Dimension Member Content](#)
- [Multiple Cell Count](#)
- [Dimension Metadata Content](#)
- [Measure Metadata Content](#)
- [Attribute Metadata Content](#)

An example for a single value property is the [Cell Background](#).

Depending on the binding type you have selected for a single value property, you have to set further properties relating to the chosen binding type.

### Related Information

[Binding Type Settings \[page 244\]](#)

## 31.5.2 Properties with Multiple Values

Properties for values in charts enable the selection of multiple cells. The result of this binding is represented by multiple cells with values. For this property type, you can only select [Multiple Cell Values](#) as the binding type. An example for a property with multiple values is "Actual Values" in trend charts.

## Related Information

[Binding Type Settings \[page 244\]](#)

### 31.5.3 Binding Type Settings

#### Binding Type Settings for Single Value Properties

Depending on the binding type you have selected for a single value property, you have to set further properties relating to the chosen binding type:

- binding type: Single Data Cell Value

Table 39:

Property	Description
Selected Data Cell	Specifies the cell that should provide its value (content)
Content Type	<p>Specifies the type of content to be used in the cell:</p> <ul style="list-style-type: none"><li>○ Value as Number The mere value, unformatted as float value is displayed. The scaling factor defined in the initial view is applied. Float values use the "." character as decimal separator; thousands separators are not applied and all decimals with value 0 are ignored. The value as number is language independent. Example: "241.80" becomes "241.8"; 241.000 becomes "241".</li><li>○ Value as Formatted String The value is displayed as in the crosstab component, respecting the scaling factor and the decimal places. Those values are language dependent and are not recommended for use in conditional binding.</li></ul>
Selection is Relative in Row	This property is selected by default. In this case the chosen cell applies the row scope definition. When you uncheck this property, the selection is always bound to the selected cell – in every row.

- binding type: Conditional Format Value  
This binding type can be used to access the content of the conditional formatting (BEx Exceptions) for the selected data cell.

Table 40:

Property	Description
Selected Data Cell	Specifies the cell that should provide its value (content)
Conditional Format	Specifies the conditional format which should be selected from the cell.

- binding type: Attribute Member Content  
This binding type can be used to access the attributes of a dimension.

Table 41:

Property	Description
Dimension	Specifies the dimension which should provide its attributes.
Attribute	Specifies the attribute which should provide its content.
Attribute Content	Specifies the content of the attribute – text or key.

- binding type: Dimension Member Content  
This binding type can be used to access the members of a dimension.

Table 42:

Property	Description
Dimension	Specifies the dimension which should provide its members.
Member Display	Specifies the content of the member object: <ul style="list-style-type: none"> <li>◦ Value Presentation the member presentation selected in the query or the initial view</li> <li>◦ Value Key the member key</li> <li>◦ Member Type the type of the member; can be used for hierarchy members</li> <li>◦ Member Node State the state of the member, can be used for hierarchy members</li> <li>◦ Member Level the level of the member; can be used for hierarchy members</li> </ul>

Property	Description
Member Presentation Part	<p>Specifies the Member Presentation Part. Only if you have selected <i>Value Presentation</i> for the <i>Member Display</i>, the following choices are available:</p> <ul style="list-style-type: none"> <li>○ Default The default is the complete member presentation.</li> <li>○ First Part This is the first part of the value presentation.</li> <li>○ Last Part This is the last part of the value presentation.</li> </ul>

- binding type: Multiple Cell Count  
This binding type can be used to access the count of the selected cells.

Table 43:

Property	Description
Selected Data Cells	Specifies the selected data cells.
Selection Mode	Specifies the behavior of the selection – with or without results.

- binding type: Measure Metadata Content  
This binding type can be used to access the dimension metadata.

Table 44:

Property	Description
Measure	Specifies the dimension which should provide its metadata.
Measure Display	<p>Specifies the content of the metadata – text or key. In addition, the following measure attributes can be selected:</p> <ul style="list-style-type: none"> <li>○ Unit of Measure</li> <li>○ Scaling Factor</li> <li>○ Format String</li> </ul>

- binding type: Attribute Metadata Content  
This binding type can be used to access the attribute metadata of a dimension.

Table 45:

Property	Description
Dimension	Specifies the dimension which should provide its attributes.
Attribute	Specifies the attribute which should provide its content.
Attribute Display	Specifies the content of the attribute – text or key.

## Binding Type Settings for Multiple Values Properties

Depending on the binding type you have selected for a multiple values property, you have to set further properties relating to the chosen binding type:

This binding type can be used to access multiple cells. This property is used for selections in trend charts.

Table 46:

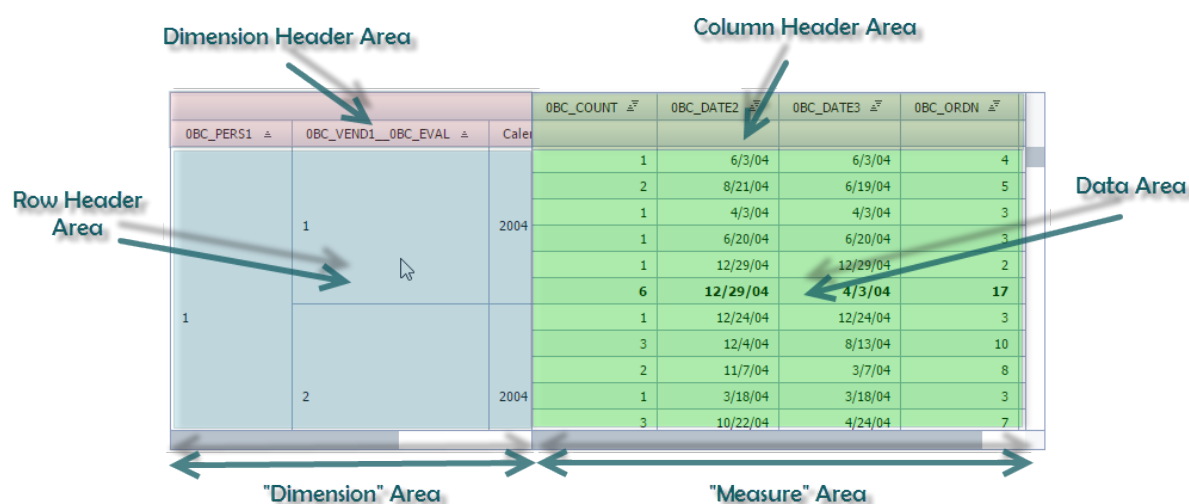
Property	Description
Selected Data Cells	Specifies the selected data cells.
Selection Mode	Specifies the behavior of the selection – with or without results.

## 32 Using Header Scrolling in Crosstabs

With SAP BusinessObjects Design Studio 1.6, the header scrolling function of the crosstab component has been changed. Incompatible changes and improvements with regard to earlier versions (Design Studio 1.4 and DS 1.5) have been made in this feature, so starting from release 1.6, this documentation is the only source of reference.

### Terminology and Basic Concepts

The following picture displays the main concepts and their terms when using the header scrolling function:



- "Dimension Area" represents the combined dimension/row header area of the crosstab
- "Measure Area" represents the combined column header/data area of the crosstab
- "Content Width" represents the overall width of the combined "Dimension Area" and "Measure Area" that results from setting the crosstab's *Width* property. If the width property is set to *auto*, the "" will be dependent on the crosstab's container width and/or the width of the browser window. Content Width

### Crosstab Properties

ContentThe header scrolling function of the crosstab is controlled by the following properties:

- Properties in section *User Interactivity*
  - *Horizontal Scrolling for Header Enabled*: true/false (default: false)
  - *Horizontal Header Resizing Enabled*: true/false (default: false)
- Properties in section *Display*
  - *Maximum Width of Header Area*: width in pixel / auto (default: auto)



When using these properties, note the following:

- If scrolling of the dimension area is needed, the property *Horizontal Scrolling for Header Enabled* needs to be set to *true*. Only if this property is set to "true", the remaining properties (*Horizontal Header Resizing Enabled* and *Maximum Width of Header Area*) are taken into account to enable finer control of the way the dimension area behaves.
- If *Horizontal Header Resizing Enabled* is set to *true*, the vertical border between dimension area and measure area can be moved horizontally by the user by dragging and dropping that border horizontally, thus altering the width ratio between dimension area and measure area. This happens so that the crosstab overall width remains unaltered if possible. However, if the user makes the dimension area width quite small and there are not enough columns in the measure area to fill the remaining horizontal space to fill all of the content width, the crosstab will horizontally -shrink. If the dimension area is enlarged horizontally again, the crosstab will horizontally expand up to the content width again. If *Maximum Width of Header Area* is set to a pixel value, the user cannot enlarge the dimension area beyond that value (see also the following "Maximum Width of Header Area" bullet point).
- "Maximum Width of Header Area":
  - When the property is set to "auto", the width of the dimension area can take any value within the horizontal boundaries of a fully rendered dimension area.
  - When the property has been set to a pixel value, the width of the dimension area can only be lower than or equal to that value. The automatic dimension area width determination and the user's possibility to alter the width of the dimension area if *Horizontal Header Resizing Enabled* is set to "true" take a non-auto pixel width setting into account. If that pixel value is set, the crosstab tries to render the dimension header area width in this size. However, if the property's pixel value is larger than the fully rendered width of the dimension area, the dimension area only takes Maximum Width up as much horizontal space as it actually needs. TheMaximum Width up as much horizontal space as it actually needs. The of Header Area dimension area width is not artificially enlarged (cells made wider) to stretch out to the given pixel width.

#### **i** Note

- There is no need for any type of horizontal scrolling in the crosstab (which means if dimension area and measure area fully fit into the content width, will not be applied).
- The property *of Header AreaMaximum Width of Header Area* cannot be set to 0 and the dimension area cannot be excluded from the crosstab by trying to do so.
- The actual rendered width of the dimension area in case it is limited by a pixel setting may differ by a few pixels from the exact property value, depending on the SAPUI5 theme used.

## 32.1 "Horizontal Scrolling Enabled" Set to "False"

If you set the crosstab property *Horizontal Scrolling Enabled* to *false*, the crosstab applies the default rendering, which means the dimension area is always rendered to its full horizontal extent, and whatever horizontal space remains of the content width is used for the measure area. If the width of the dimension area exceeds the content width, there is no scrolling of the dimension area available and columns extending beyond the crosstab boundaries are not displayed or reachable.

Also, if the dimension area takes up most of the content width so that only a very small portion of the measure area can be rendered, the dimension area does not shrink (and is hence scrollable) to make more room available for the measure area.

## 32.2 "Horizontal Scrolling Enabled" Set to "True"

If you set the crosstab property *Horizontal Scrolling Enabled* to *true*, consider the following two scenarios:

### 1. Width of the dimension area and the measure area combined is smaller than the content width

In this case there is no need for any scrolling at all. Both dimension area and measure area render to their full horizontal extent without any scrollbars. Also, the application user will not be able to manually move the border between dimension area and measure area even if the property *Horizontal Header Resizing Enabled* is set to "true". The *Maximum Width of Header Area* property is not applied either.

### 2. Width of the dimension area and the measure area combined exceeds the content width

Depending on whether the width of the dimension area is limited or not, take these two scenarios into account:

- *Maximum Width of Header Area* set to "auto", general strategy
- *Maximum Width of Header Area* set to a pixel value

## Related Information

[General Strategy if "Maximum Width of Header Area" Set to "Auto" \[page 250\]](#)

[Behavior after Manual Changing of Header With \[page 254\]](#)

[Crosstabs With a Dimension Area or a Measure Area Only \[page 259\]](#)

["Maximum Width of Header Area" Set To A Pixel Value \[page 260\]](#)

### 32.2.1 General Strategy if "Maximum Width of Header Area" Set to "Auto"

The general strategy is to distribute the widths of the dimension area and the measure area roughly in a 1:1 ratio across the available content width (it is not always exactly 1:1!). However, this is not always possible, for example, if the dimension area has many columns and the measure area has only few columns or just a single column. In that case, the dimension area width is determined so that the full width of the measure area can be rendered within the content width. This might result in the scrollable dimension area being much wider than the measure area.

Examples:

- Crosstab with a roughly 1:1 width distribution between dimension area and measure area

OBC_PERS1	OBC_VEND1__OBC_EVAL	OBC_COUNT	OBC_DATE2	OBC
1	1	1	6/3/04	
		2	8/21/04	
		1	4/3/04	
		1	6/20/04	
		1	12/29/04	
		6	12/29/04	
	2	1	12/24/04	
		3	12/4/04	
		2	11/7/04	
		1	3/18/04	
		3	10/22/04	
		1	6/26/04	

- Crosstab with scrollable dimension area and fully rendered measure area that only has two columns.  
Dimension area width > 50% of content width

OBC_PERS1	OBC_VEND1__OBC_EVAL	Calendar Year	OBC_	OBC_COUNT	OBC_DATE2
1	1	2004	bank c	1	6/3/04
			by bill	2	8/21/04
			check	1	4/3/04
			credit	1	6/20/04
			EC car	1	12/29/04
			<b>Resu</b>	<b>6</b>	<b>12/29/04</b>
	2	2004	bank c	1	12/24/04
			by bill	3	12/4/04
			cash	2	11/7/04
			check	1	3/18/04
			coupo	3	10/22/04
			credit	1	6/26/04

- Shrinking the above example with the two columns measure area. Now, the full measure area cannot be rendered any more without the dimension area width falling below 50% of the Content Width. Hence, a 1:1 width distribution is rendered again

0BC_PERS1	0BC_VE	0BC_COUNT	0BC_D
1	1	1	
		2	
		1	
		1	
		1	
		6	1
	2	1	
		3	
		2	
		1	
		3	
		1	

If the property *Horizontal Header Resizing Enabled* is set to "true", the user may move the border between dimension area and measure area. Depending on how many columns are available in the measure area, shrinking the dimension area may alter the content width so that the crosstab width gets smaller.

Examples:

- Grabbing the border between dimension area and measure area

0BC_PERS1	0BC_VEND1__0BC	0BC_ORDN	0BC_ORDW
			KG
1	1	4	1.460
		5	10.050
		3	2.220
		3	5.180
		2	4.480
		17	23.390
	2	3	7.710
		10	20.130
		8	13.510
		3	8.390
		7	12.380

- Manually enlarged dimension area

			0BC_ORDN
0BC_PERS1	0BC_VEND1__0BC_EVAL	Calendar Y	
1	1	2004	
	2	2004	

- Manually shrunk dimension area

		0BC_ORDN	0BC_ORDW	0BC_TURN
0BC_PERS1		KG	EUR	
1	1	4	1.460	54.04
		5	10.050	135.93
		3	2.220	73.27
		3	5.180	50.36
		2	4.480	18.40
		<b>17</b>	<b>23.390</b>	<b>332.00</b>
	2	3	7.710	15.82
		10	20.130	146.11
		8	13.510	88.65
		3	8.390	95.50
		7	12.380	173.37

- The default layout when the measure area is small enough

		OBC_ORDN	OBC_ORDW
OBC_PERS1	OBC_VEND1__OBC_EVAL		KG
1	1	4	1.460
		5	10.050
		3	2.220
		3	5.180
		2	4.480
		17	23.390
	2	3	7.710
		10	20.130
		8	13.510
		3	8.390
		7	12.380

- The dimension area of the above example has now been manually shrunk. Observe that the overall crosstab width is now smaller since there are not enough columns in the measure area to fill the width up to the available content width.

		OBC_ORDN	OBC_ORDW
OBC_PERS1			KG
1		4	1.460
		5	10.050
		3	2.220
		3	5.180
		2	4.480
		17	23.390
		3	7.710
		10	20.130
		8	13.510
		3	8.390
		7	12.380

## 32.2.2 Behavior after Manual Changing of Header With

Once the user has altered the dimension area width manually by using the border handle, all following width resize operations that may be applied to the crosstab as a whole will try to preserve that dimension area width, except in case the crosstab has been enlarged so that the dimension area can be rendered without a scrollbar. If the crosstab is then shrunk again, the automatic algorithm applies and any previously made dimension area width setting will be lost.

When the crosstab width is shrunk so that the content width becomes smaller than the dimension area width that was set by the user, the user setting will also be discarded and the auto-width distribution strategy applies.

When the dimension area width is manually altered, subsequent resize operations on the crosstab as a whole might lead to the crosstab not taking up the full available space.

Please see the following example sequence:

1. Two identical crosstabs, but in the lower crosstab the dimension area was manually expanded by the user.

		OBC_ORDN	OBC_O
OBC_PERS1	OBC_VEND		KG
1	1	4	
		5	
		3	
		3	
		2	

		OBC_ORD
OBC_PERS1	OBC_VEND1__OBC_EVAL	
1	1	

2. Increasing the crosstab width. Note how the upper crosstab uses the auto 1:1 width distribution while the lower crosstab preserves the dimension area width set by the user.

		0BC_ORDN	0BC_ORDW
0BC_PERS1	0BC_VEND1__0BC_EV		KG
1	1	4	1.460
		5	10.050
		3	2.220
		3	5.180
		2	4.480

		0BC_ORDN	0BC_ORDW
0BC_PERS1	0BC_VEND1__0BC_EVAL		KG
1	1	4	1
		5	10
		3	2
		3	5
		2	4

3. Further increasing the crosstab width. Note how in the upper crosstab the automatic algorithm fills the available content width, while the lower crosstab preserves the user's dimension area width setting, thus resulting in the crosstab not using up all the available space.

			0BC_ORDN	0BC_ORDW
0BC_PERS1	0BC_VEND1__0BC_EVAL	Calendar Y		KG
1	1	2004	4	1.460
			5	10.050
			3	2.220
			3	5.180
			2	4.480

		0BC_ORDN	0BC_ORDW
0BC_PERS1	0BC_VEND1__0BC_EVAL		KG
1	1	4	1.460
		5	10.050
		3	2.220
		3	5.180
		2	4.480

4. Increasing the crosstab width so everything can be rendered without any scrollbars. In this case, the user set header width is also discarded since header scrolling is no longer necessary.



				0BC_ORDN $\Sigma^{\overline{}}$	0BC_ORDW $\Sigma^{\overline{}}$
0BC_PERS1 $\Delta$	0BC_VEND1__0BC_EVAL $\Delta$	Calendar Year $\Delta$	0BC_PAYM $\Delta$		KG
1	1	2004	bank collection	4	1.460
			by bill	5	10.050
			check	3	2.220
			credit card	3	5.180
			EC card	2	4.480
			<b>Result</b>	<b>17</b>	<b>22.390</b>

				0BC_ORDN $\Sigma^{\overline{}}$	0BC_ORDW $\Sigma^{\overline{}}$
0BC_PERS1 $\Delta$	0BC_VEND1__0BC_EVAL $\Delta$	Calendar Year $\Delta$	0BC_PAYM $\Delta$		KG
1	1	2004	bank collection	4	1.460
			by bill	5	10.050
			check	3	2.220
			credit card	3	5.180
			EC card	2	4.480
			<b>Result</b>	<b>17</b>	<b>22.390</b>

5. Again horizontally shrinking the crosstab shows that the user-set dimension area width is not being preserved, thus resulting in both crosstabs behaving exactly the same.

		0BC_ORDN $\Sigma^{\overline{}}$	0BC_ORDW $\Sigma^{\overline{}}$
0BC_PERS1 $\Delta$	0BC_VEND1__0BC_EVAL		KG
1	1	4	1.460
		5	10.050
		3	2.220
		3	5.180
		2	4.480

		0BC_ORDN $\Sigma^{\overline{}}$	0BC_ORDW $\Sigma^{\overline{}}$
0BC_PERS1 $\Delta$	0BC_VEND1__0BC_EVAL		KG
1	1	4	1.460
		5	10.050
		3	2.220
		3	5.180
		2	4.480

6. Moving the lower crosstab's handle to shrink the dimension area. Note the size change in the crosstab due to insufficient columns in the measure area.

		OBC_ORDN	OBC_ORDW
OBC_PERS1	OBC_VEND1__OBC_EVAL		KG
1	1	4	1.460
		5	10.050
		3	2.220
		3	5.180
		2	4.480

		OBC_ORDN	OBC_ORDW
OBC_PERS1	OBC_VEND1		KG
1	1	4	1.460
		5	10.050
		3	2.220
		3	5.180
		2	4.480

7. Shrinking the crosstab so that the content width gets smaller than the user-set dimension area width. In this case, the user-set dimension area width is discarded and the auto-layout applies, resulting in upper and lower crosstab behaving identically again.

	OBC_ORDN
OBC_PERS1	
1	

	OBC_ORDN
OBC_PERS1	
1	

Also, be aware that through setting the dimension area width manually and then shrinking the crosstab, situations like the following may arise where the measure area gets very small while the auto-layout constantly provides a usable layout: Unusable measure area due to manual dimension area width setting in the lower crosstab. Upper crosstab with auto-layout provides usable width distribution.

	0BC_ORDN	
0BC_PERS1		
1		4
		5
		3
		3
		2

0BC_PERS1	0BC_VEND	
1	1	

### 32.2.3 Crosstabs With a Dimension Area or a Measure Area Only

In some cases, there may only be a dimension area or a measure area without the respective other area present, for example, if there is only the measure structure on the columns axis of the drilldown as shown in the following picture:

0BC_DATE3	0BC_DATE2	0BC_COUNT	0BC_TURN	0BC_ORDN	0BC_ORDW
			EUR		KG
1/1/04	12/31/04	2,000	109,393.65	5,935	10,684.600

When the crosstab is shrunk horizontally, scrolling is also displayed for this case, but even if the property *Horizontal Header Resizing Enabled* is set to *true* and/or there is a pixel value set for the property *Maximum Width of Header Area*, those properties are not applied. Hence, the user is not able to move any area border interactively (because there is none) and no maximum header width pixel value is taken into account. The result may appear as follows:

C_COUNT	0BC_TURN	0BC_ORDN	0BC_ORDW
	EUR		KG
2,000	109,393.65	5,935	10,684.600

The above crosstab has shrunk so that it becomes horizontally scrollable. No maximum header width is applied and no interactive resizing of an area border is possible since there is only one area.

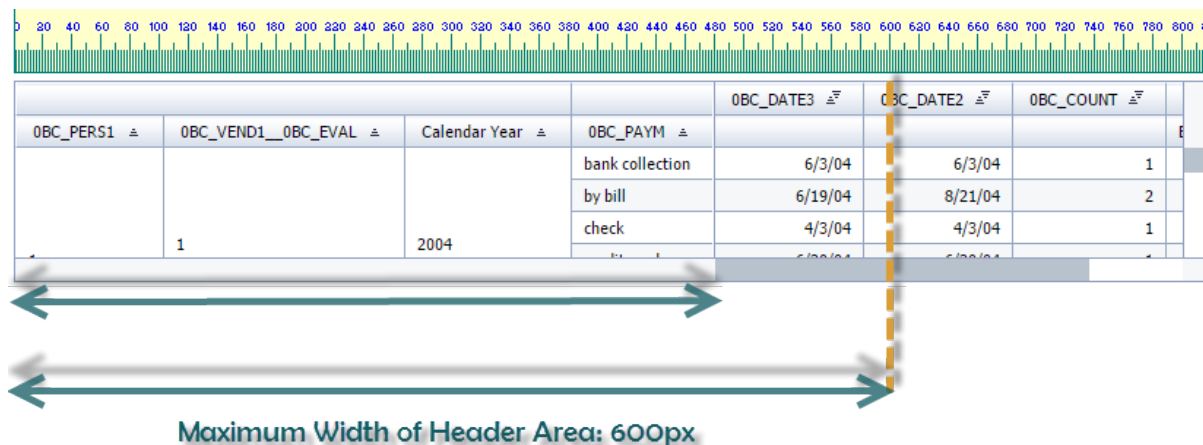
## 32.2.4 "Maximum Width of Header Area" Set To A Pixel Value

All given facts and aspects described in the sections if the property *Maximum Width of Header Area* is set to *auto* are also valid if the same property is set to a pixel value. The only difference is that the width of the dimension area never exceeds the given pixel value. Also, the user cannot drag the border between dimension area and measure area beyond what would be allowed by the maximum width setting. The crosstab tries to make the dimension area width match the width setting of *Maximum Width of Header Area* if possible. This means, if the dimension area provides enough data or wide enough cells to actually take up the space required to fill the width given through the *Maximum Width of Header Area* property.

Besides, the behavior of *Maximum Width of Header Area* is similar to if the Dimension Area width was set interactively by the user as described above.

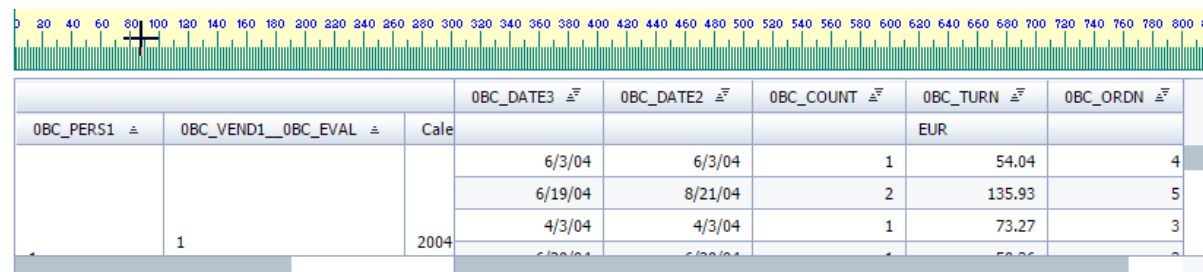
### Examples:

- *Maximum Width of Header Area* set to 600px. However, the dimension area is not wide enough (only around 480px) to extend this far. Hence, the dimension area width grows no larger than its actual size.



OBC_PERS1	OBC_VEND1__OBC_EVAL	Calendar Year	OBC_PAYM	OBC_DATE3	OBC_DATE2	OBC_COUNT
			bank collection	6/3/04	6/3/04	1
			by bill	6/19/04	8/21/04	2
			check	4/3/04	4/3/04	1
	1	2004				

- Same settings as described in the example above, but the property *Maximum Width of Header Area* is set to 300px.



OBC_PERS1	OBC_VEND1__OBC_EVAL	Calendar Year	OBC_DATE3	OBC_DATE2	OBC_COUNT	OBC_TURN	OBC_ORDN
			6/3/04	6/3/04	1	54.04	4
			6/19/04	8/21/04	2	135.93	5
			4/3/04	4/3/04	1	73.27	3
	1	2004					

- Same settings as described in the example above, but shrunk horizontally. Note that, as if it was set by user interaction, the dimension area width is preserved according to the *Maximum Width of Header Area* setting.

			OBC_D
OBC_PERS1	OBC_VEND1__OBC_EVAL	Cale	
	1	2004	

- Shrinking the crosstab further beyond the *Maximum Width of Header Area* value. In that case, auto-layout (1:1) applies.

		OBC_DATE3
OBC_PERS1	OBC	
		6/3/04
		6/19/04
		4/3/04
	1	6/19/04

- Enlarging the crosstab again. Note that the *Maximum Width of Header Area* value is applied again when the overall crosstab width gets big enough.

			OBC_DAT
OBC_PERS1	OBC_VEND1__OBC_EVAL	Cale	
	1	2004	

- Enlarging the crosstab further. If the crosstab is big enough to accommodate everything so that scrolling is no longer necessary, *Maximum Width of Header Area* no longer applies.

OBC_PERS1	OBC_VEND1__OBC_EVAL	Calendar Year	OBC_PAYM	OBC_DATE3	OBC_DATE2	OBC_COUNT	OBC_TURN	OBC_ORDN	OBC_ORDW
			bank collection	6/3/04	6/3/04	1	54.04	4	1.460
			by bill	6/19/04	8/21/04	2	135.93	5	10.050
			check	4/3/04	4/3/04	1	73.27	3	2.220
			credit card	6/20/04	6/20/04	1	50.36	3	5.180
1	1	2004							

## 33 Working with Real-Time Dashboards



You can create visualizations with streaming data (push based), but also allow users to create visualizations, which have a near real-time connection to SAP HANA or SAP BW (pull based). This enables you to create and run your own complex event processing (CEP) applications to derive continuous intelligence from streaming event data in real time. There are many applications for real-time functionality including in the area of operational systems, where data may only be relevant as it occurs and where it may not be necessary to persist the data, for example utility companies monitoring smart buildings and the financial capital markets.

The real-time functionality is implemented through a connection to a streaming data source, such as SAP HANA Smart Data Streaming (HANA Streaming or SDS), or SAP ESP (Event Stream Processor). Within the *Outline* view in the designer, you can select from the context menu ► *Add Custom Data Source* ► *Streaming Data Source* . This streaming data can then be connected to any OOTB *Info Chart*, *Chart* or *Geo Map*, in order to visualize the data.

You can also use the *Timer* technical component to poll a data source at regular intervals to update a chart. This is applicable in cases where you do not need real-time streaming using WebSockets to push to the browser.

### Note

- Measures must be in the columns area when connecting your *Streaming Data Source* to a *Geo Map*.
- Before working with real-time feature, you should refer to the following SAP Notes:

SAP Note Number	Description
<a href="#">2228243</a> 	WebSocket support required when using Streaming Data Source in Design Studio 1.6
<a href="#">2238142</a> 	Working with the Design Studio Real-Time Data Source component

## Related Information

[How to Create Real-Time Dashboards with Streaming Data Sources \[page 263\]](#)

[Streaming Data Source Additional Properties \[page 267\]](#)

[Working with the Outline View \[page 54\]](#)

## 33.1 How to Create Real-Time Dashboards with Streaming Data Sources



### Context

A streaming data source such as SAP HANA SDS (Smart Data Streaming) or SAP ESP (Event Stream Processor) can connect to Design Studio charts to create visualizations of streams of real-time data. You need to have an instance within SAP ESP 5.1 (SP09 or SP10) or SAP HANA SDS and have a project running.

#### Note

- The authentication for SAP HANA SDS is done by the HANA server. Therefore, Design Studio does not support SAP BI platform authentication with SAP HANA SDS.
- For more details on the [Additional Properties](#) of the [Streaming Data Source](#), you can refer to the chapter in this guide called “Streaming Data Source Additional Properties”.

### Procedure

1. Within Design Studio go to  [Outline View](#) .
2. In the context menu, select the option [Add Custom Data Source](#).
3. Select [Streaming Data Source](#).
4. In the [Properties](#) view, select [Additional Properties](#) for the [Streaming Data Source](#).  
Three [Additional Properties](#) tabs appear. The first tab is called [General Configuration](#). This tab enables you to configure your [Streaming Data Source](#).
5. Configure your [Streaming Data Source](#) by entering the Host and REST port on which your SAP ESP or SAP HANA SDS project is running.
6. Enter your user SAP ESP or SAP HANA SDS password and username credentials. Use your ESP Studio or HANA Studio username and password.  
The [Connect](#) button will be enabled once you have entered values for all the configuration and credential fields.
7. Select [Connect](#).  
The [Credentials](#) icon will turn green indicating that a connection to the [Streaming Data Source](#) has been made.

#### Note

If credential values are incorrect, the [Credentials](#) icon will turn red. The tooltip over this icon will provide you with details of the status of the connection.

8. [Workspace](#), [Project](#) and [Topic](#) are automatically defined based on the projects running in SAP ESP or SAP HANA SDS. You can modify these values if required. Your [Workspace](#) refers to the location of all your stored projects. The projects stored in that workspace will be loaded into the [Projects](#) drop down list. A project can contain multiple topics.
9. If the [Topic](#) you selected is a stream, you must set the [Retention Policy](#). This option will allow you to specify the maximum number of rows that the [Chart](#) will display.

10. Configure the *Update Interval (ms)* property to set the frequency with which new data updates are displayed. The default setting is 1,000 milliseconds.
11. Select the second *Additional Properties* tab, which is called *Measure/Dimension Configuration*. This tab displays the measures and dimensions from your SAP ESP or SAP HANA SDS project.
12. Review the data types and update measure or dimension configuration if required.
13. Select the third *Additional Properties* tab, which is called *Data Selection for Rows and Columns*. This tab displays the data selection for your rows and columns.

14. **Note**

Measures must be in the columns area when connecting your *Streaming Data Source* to a *Geo Map*.

Select your measures, dimensions, rows and columns to configure your query for the *Streaming Data Source* initial view.

15. Select a *Chart*, *Info Chart*, or *Geo Map* component and add to your canvas.
16. Assign the *Streaming Data Source* to your selected component.
17. If you wish to view the data at design time, select the *Show Live Preview* checkbox in the *Data Selection for Rows and Columns* tab for the *Streaming Data Source*.
18. You can change the *Info Chart* or *Chart* type at any time to get a modified view of your data.

## Related Information

[Streaming Data Source Additional Properties \[page 267\]](#)

[Working with Real-Time Dashboards \[page 262\]](#)

## 33.2 How to Create Real-Time Dashboards with a Single Pull-Based Data Source

### Context

You wish to have your dashboards update at a set interval rather than every time there is a new event. You can create almost real-time visualizations when connected to SAP HANA or SAP BW (pull based) data sources by using the *Timer* technical component in conjunction with an OOTB *Info Chart*, *Chart* or *Geo Map* within Design Studio. The *Timer* technical component allows you to set an interval in milliseconds, at which point it runs a user-defined script. This script can be used to reload a single data source.

The *Timer* component is activated through the optional *Technical Component* called *Timer*. It must be added in the *Outline* view of the designer application before this functionality is available.



---

To reload a single data source, follow the steps below:

## Procedure

1. Add an *Info Chart*, *Chart* or *Geo Map* component onto your canvas and assign a data source. The data source can be either SAP BW or SAP HANA.
2. Add the *Timer* component, which is located in the *Outline* view under ► *Technical Components* ► *Create Child* ► *Timer* ►.
3. Within the *Timer* properties, change the *Interval in Milliseconds* property to your chosen interval value.
4. To make the *Timer* reload the data source, begin by opening the *Script Editor* for the *Timer* components event *On Timer*.
5. To reload the data source use the `reloadData()` ; scripting method in the *Script Editor*. For example,  
`DS_1.reloadData()` ;
6. To start the *Timer*, configure the *On Startup* property of the application. Add to the *Script Editor* for the *On Startup* event to start the timer component. For example, `TIMER_1.start()` ;  
Once executed, this dashboard will now update by the defined number of seconds you set for the *Interval in Milliseconds* component property.

## Related Information

[Assigning a Data Source to a Component \[page 97\]](#)

[Working with the Outline View \[page 54\]](#)

[Working with Real-Time Dashboards \[page 262\]](#)

## 33.3 How to Create Real-Time Dashboards with Multiple Pull-Based Data Sources

### Context

You have multiple data sources with multiple chart types in a dashboard that needs to update at regular intervals. It may not be necessary to load all the data at regular intervals as the data sources might not all contain new data. To increase performance you can implement a separate data source that executes quickly and determines if there is new data across the multiple data sources. For example, this "trigger" data source can return a single row, containing the Id of the last entry inserted into the database. This Id can be compared to a previous value. If they are different, the other data sources should be reloaded.

To reload two data sources, follow the steps below:

## Procedure

1. Add two *Info Chart* components, two *Chart* components or two *Geo Map* components onto your canvas and assign a data source to each. The data source can be either SAP BW or SAP HANA.
2. Set up and add a trigger data source, which has only a single measure containing the Id of the last inserted event.
3. To keep track of the last Id of the trigger data source, add a *Text* component to the *Outline* view. You can set the *Visible* property of this component to **false**.
4. Add the *Timer* technical component, from within the *Outline* view.
5. Within the *Timer* properties, change the *Interval in Milliseconds* property to your chosen interval value.
6. Reloading a slower performing data source may impede user interaction. To alleviate this, move the reloading to the background processing. To move the reloading to the background processing, open the *Script Editor* of the *On Timer* property within the *Timer* component and use the `doBackgroundProcessing()` scripting method. For example: `TIMER_1.stop(); APPLICATION.doBackgroundProcessing();`
7. To start the *Timer* component, configure the *On Startup* property of the application. Add to the *Script Editor* for the *On Startup*. For example: `TIMER_1.start();`
8. Configure the *On Background Processing* event for the application within the *Script Editor*.  
Once executed, this dashboard will now check if there are updates for the dashboard by the defined number of seconds you set for the *Timer* component property *Interval in Milliseconds* and make updates if changes have been found.

### Example

```
DS_TRIGGER.reloadData();

var oldID = TEXT_1.getText();

var newID = DS_TRIGGER.getDataAsString("MAX_EVENT_ID", {});

if(oldID != newID)
{
    DS_1.reloadData();

    DS_2.reloadData();

    TEXT_1.setText(newID);
}

TIMER_1.start();
```

## Related Information

[How to Create Real-Time Dashboards with a Single Pull-Based Data Source \[page 264\]](#)

[Working with Real-Time Dashboards \[page 262\]](#)

[Working with the Outline View \[page 54\]](#)

## 33.4 Streaming Data Source Additional Properties

The *Streaming Data Source* additional properties are described in the table below.

Property Name	Property Value	Property Description
configuration	<ul style="list-style-type: none"><li>host</li><li>port</li></ul>	Host and REST port on which your project is running.
security	<ul style="list-style-type: none"><li>HTTPS</li></ul>	The default value is <i>True</i>
credentials	<ul style="list-style-type: none"><li>username</li><li>password</li><li><i>Connect</i> button</li></ul>	<ul style="list-style-type: none"><li>ESP Studio or HANA Studio username and password.</li><li>Connects to SAP ESP or SAP HANA SDS instance.</li></ul>
data source	<ul style="list-style-type: none"><li>workspace</li><li>project</li><li>topic</li><li>retention policy</li><li>update interval (ms)</li></ul>	<ul style="list-style-type: none"><li>ESP or HANA SDS projects are stored within the workspace.</li><li>Projects within a workspace.</li><li>Retention Policy: In SAP ESP or SAP HANA SDS, a topic is either a stream or a window. If the topic you select is a stream, a <i>Text</i> field will appear. You will be required to specify the <i>Retention Policy</i> for a stream. This option will allow you to specify the maximum number of rows that the <i>Chart</i> or <i>Info Chart</i> will display. If a new event occurs when the chart is already displaying its maximum number of rows, the oldest row will be deleted in order to make room. If the topic you select is a window the <i>Retention Policy</i> text field will not appear as SAP ESP or SAP HANA SDS handles the retention policy of windows.</li><li>Update Interval (ms): The interval for displaying new events in milliseconds. By default the display</li></ul>

Property Name	Property Value	Property Description
		updates with new data every second.
data types	a list of data types	<p>Currently SAP ESP and SAP HANA SDS have no concept of a measure or a dimension. As a result of this, some default rules have been applied to decide if a column is a measure or dimension. These rules are:</p> <ul style="list-style-type: none"> <li>• If a Column is Primary Key in SAP ESP or SAP HANA SDS, it is a dimension.</li> <li>• If a column is a number in SAP ESP or SAP HANA SDS, it is a measure.</li> <li>• Any remaining columns are dimensions.</li> </ul> <p><b>i Note</b></p> <p>You will also be able to configure some columns.</p>
measures	primary values	A number or quantity that records a directly observable value or performance. They typically represent some time dependent measure of the stream being visualized, for example, stock price, temperature, pressure.
dimensions	<ul style="list-style-type: none"> <li>• rows</li> <li>• columns</li> </ul>	A collection of related data members, which represents one aspect of a business. In relation to streaming data sources, time would be a typical dimension. Select the Rows or Columns option to decide where to position the dimensions.
measures	<ul style="list-style-type: none"> <li>• rows</li> <li>• columns</li> </ul>	Select the Rows or Columns option to decide where to position the measures.
live preview	checkbox	<p>To preview the live stream you must select the <a href="#">Show Live Preview</a> checkbox. Until you select <a href="#">Show Live Preview</a>, any charts connected to the data source will show no data available. <a href="#">Show Live Preview</a> will not persist and if connection is lost or the chart is closed and reopened it must be selected again.</p> <p><b>i Note</b></p> <p>Show Live Preview is unchecked by default.</p>

---

## Related Information

[Working with Real-Time Dashboards \[page 262\]](#)

[How to Create Real-Time Dashboards with Streaming Data Sources \[page 263\]](#)

## 34 Advanced Design Tasks

### 34.1 Scripting for User Interaction Enablement

SAP BusinessObjects Design Studio is a design tool for creating interactive analysis applications. To enable interactivity, you write scripts that are executed when the user performs an action in the executed application. For example, you can place the *button* component in the application and assign a script to the button's *On Click* event.

Scripts consist of one or more statements written in a JavaScript-based language that follow a specific syntax. For writing scripts the design tool offers two approaches: the statement wizard and the script editor. All objects, methods and expression types are listed in the API reference.

#### Script language

You write scripts in the BI Action Language (BIAL). BIAL is a true subset of the JavaScript/ECMAScript standard described here: <http://www.ecma-international.org/publications/files/ECMA-ST/Ecma-262.pdf> 📄. The scripts are executed on the Analysis Application Design Service (unlike JavaScript that is executed in the Web browser). Scripts have a clear definition of the supported language constructs, objects and methods.

#### Scripts, statements and events

Scripts consist of statements. Each statement is typically written in a single line. The execution of scripts is triggered by user interaction with the component. This interaction is also referred to as an event and the script executed in response to this event is the event handler. Each component has its own set of one or more events that are displayed in the *Properties* view of the component.

##### ➔ Remember

Events on components are only triggered by user interaction and not implicitly by scripting. For example, when changing the selection in a list box by using the script `LB.setSelectedItem()`, the *On Select* event of the list box is **not** triggered. An exception to this rule is the *On Result Set Changed* event, which can be triggered implicitly by a script method like `setFilter()` or `reloadData()`.

SAP BusinessObjects Design Studio currently supports the following statement types:

- call statements
- conditional execution statements
- assignment statements

## Syntax of call statements

Call statements execute an API method of an object and have the following format:

```
<Component>.<method>(<arguments>) ;
```

<Component> is the name of a data source alias or component in your application, for example, `DS_1` or `Button_1`.

<method> (also referred to as method) is an operation that is applied to the object specified on the left of the period. The available functions depend on component type. `DataSourceAlias` objects, for example, provide functions to filter data, and visual components provide functions to modify visibility, enablement, etc.

<arguments> is a comma-separated list of expressions. The passed expressions must match the requirements of the method.

Each statement ends with a semicolon (";").

You can use functions as arguments for other methods as long as the return type matches the argument type.

## Syntax of conditional execution statements

Conditional execution statements have one of the following formats:

- first format

```
if (<condition>) {  
  <sequence of statements to execute when condition is met>  
}
```

- second format

```
if (<condition>) {  
  <sequence of statements to execute when condition is met>  
} else {  
  <sequence of statements to execute when condition is NOT met>  
}
```

<condition> is a Boolean expression - one of the following:

- true or false as value literals (constants)
- a method call statement returning a Boolean value
- a comparison for equality in the form `a == b` or
- a comparison for inequality in the form `a != b`
- multiple conditions combined with `&&` and `||`, parentheses are optional

## Assignment statements

Assignment statements have one of the following formats:

- `var <variable> = <expression>;`  
This format defines a script variable (see Script Variables in this chapter) and assigns the result of an expression to this script variable.
- `<variable> = <expression>;`  
This format assigns the result of an expression to a script variable that has been already defined.

## Methods and object types

Methods are operations that are applied to the object specified on the left of the period in the statement. The available methods depend on the object type. For example, DataSourceAlias objects provide methods to filter data. Visual components have methods to modify visibility, position and so on. SAP BusinessObjects Design Studio supports methods that depend on the following object types:

- data source alias (DataSourceAlias)
- application (Application)
- component (Component)
- individual components like button, text, image, etc. (Button, Text, Image, etc.)
- information objects (ApplicationInfo, DataCell, Convert, DataSourceInfo)

## Expressions

Expressions compute a result (also referred to as returned value), similar to a formula. Expressions can appear as method arguments and as conditions (if-statements or conditional execution statements). Results of an expression have a type. An expression may consist of literals (String constants, integer numbers, Arrays, JSONs), operators, method calls and parentheses.

Operators supported by BIAL :

Operator	Description	Argument Type	Result Type	Example
+	Concatenates strings	String, (Integer, Float, Boolean) (Boolean and Integer will be converted to String automatically)	String	"ab"+"cd" (= "abcd") "ab"+1 (= "ab1")
+	Adds two integer values or floating point	Integer, Float	Integer	1+2 (=3)
-	Subtracts two integer values or floating point	Integer, Float	Integer	3-2 (=1)
*	Multiplies two integer values or floating point	Integer, Float	Integer, Float	3*2 (=6)



Operator	Description	Argument Type	Result Type	Example
/	Divides one integer value by the other or one floating point by the other	Integer, Float	Integer, Float	8/2 (=4)
==	Checks if the two operands are equal	Any	Boolean	1 == 1 (= true) "a" == "b" (=false)
!=	Checks if the two operands are not equal	Any	Boolean	1 != 2 (= true) "a" != "a" (=false)
&&	Logical AND	Boolean	Boolean	true && false (=false) true && true (=true) if (<condition1> && <condition2>) { <statements> } Statements will be executed if both conditions are true.
	Logical OR	Boolean	Boolean	true    false (=true) false    false (=false) if (<condition1>    <condition2>) { <statements> } Statements will be executed if at least one of the conditions is true.
!	Logical NOT	Boolean	Boolean	!true (=false) !false (=true) If (! <condition>) { <statements> }

Operator	Description	Argument Type	Result Type	Example
				Statements will be executed if condition is not true.

## Expression type system

Expression results can be used as method arguments. The expression type must match the argument type. In some cases, BIAL will automatically convert between types. This is typically only done for certain strings, where BIAL can check that the string value is valid (see below). The +Operator automatically converts Integer and Boolean arguments to String. In all other cases, the type system is strict and error messages are displayed.

There are four different types:

- primitive types (String, Integer, Boolean, Float)

Type	Example
String	"Hello"
Integer	123
Float	123
Boolean	true, false
String Array	["A", "B"]
Integer Array	[1, 2]
JSON	{"key": "value"}

- BI types (DataSourceAlias, Dimension, Measure, ...) enable technically appropriate input help for API method arguments in the script editor. Such special types are sometimes written in Enum, Array, or JSON format (or even a combination of these). To view examples, refer to the API reference.
- component types (Button, Text, Pagebook...)

The visual component types correspond to the list of components in the [Components](#) view of the design tool

## Enums

In many cases, the input of a function can be a fixed set of values. This set of values is referred to as an enumeration or simply "Enum". Enums are BI types. An Enum value is written as <EnumType>.<EnumValue>, for example, "ChartType.PIE".

## Script Variables

Script variables store expression results. They are useful, for example, to store intermediate results that are used repeatedly later in a script. Script variables have a name composed of the characters "A"-"Z", "a"-"z", "0"-"9", and

"\_". The name must not begin with a digit character "0"-"9". Like expressions, script variables have a type. There are local and global script variables:

- Local script variables

Local script variables are script variables that can be used only in the script in which they were defined, but not in other scripts. To define a local script variable, open a script with the [Script Editor](#) and add a line of the following format:

```
var <variable> = <expression>;
```

The type of the script variable is automatically determined by the type of the expression.

- Global script variables

Global script variables are script variables that can be used in any script of your application. To define a global script variable, first click the application in the [Outline](#) view. Next click the item [Scripting](#) [Global Script Variables](#) in the [Property](#) sheet. Click the button .... This opens the [Edit Global Script Variable](#) dialog box. You can insert a new global script variable and define its name, type, and default value. In addition you can decide if the global script variable is a URL parameter. If you decide to make it a URL parameter, then you can set the value of this global script variable by adding the global script variable and its value as a URL parameter to the applications URL.

#### Note

Script variable names used as a URL parameter must start with a capital x (**x**) and must not end with an underscore and a digit (**\_<digit>**).

## Methods calling the event handler

To avoid event handlers calling each other an infinite number of times (which could result in runtime crashes), events on components are only triggered by user interaction on the component and not by scripting. For example, if a user swipes the pages of a pagebook, the On Select event is raised, and the relevant event script is executed. In contrast, if the selected page index is set by script using the method `setSelectedPageIndex` of the pagebook, the On Select event is not raised. Another example is if a user selects a value from a dropdown box, the On Select event is raised and the event script is executed. In contrast, if the selected item is set using the `setSelectedValue` method of the dropdown box, the event is not raised. However there are situations when you want to execute the relevant event script without duplicating script code. In these cases, each component with an event (On Click, On Select) has a corresponding method that allows the event handler to be called from another handler's code. For example, the statement `BUTTON_1.onClick()`; calls the [On Click](#) event handler of the button component with the name `BUTTON_1`. Another example is the statement `TABSTRIP_1.onSelect()`; that calls the [On Select](#) event handler of the tabstrip component with the name `TABSTRIP_1`.

#### Example

You have created an application with a dropdown box `DROPDOWN_1` and a button `BUTTON_1`. If the user selects a value in the dropdown box, the selected value filters the dimension `MYDIM` in the data source aliases `DS_1` and `DS_2`. For this scenario, the [On Select](#) event script of the dropdown box looks like this:

```
DS_1.setFilter("MYDIM", DROPDOWN_1.getSelectedValue());
```

```
DS_2.setFilter("MYDIM", DROPDOWN_1.getSelectedValue());
```

If the user pushes the button, a specific value in the dropdown box should be selected programmatically using the same logic defined for user interaction. In other words, the selected value filters the dimension MYDIM in the data source aliases DS\_1 and DS\_2. Instead of duplicating the script code of the *On Select* event of DROPDOWN\_1, you add the following statements in the *On Click* event handler of the button:

```
DROPDOWN_1.setSelectedValue("MYDIMVALUE");
```

```
DROPDOWN_1.onSelect();
```

The benefits of calling the `onSelect` method increase as you start to work with more use cases in the application (where you want to set the selected value programmatically) and more statements in the *On Select* event handler of the dropdown box.

## Member key format

The most frequently used API methods deal with dimension members in the context of setting filters or variable values. Often the filter or variable values that need to be set are returned from the selection of a UI component.

In SAP Business Information Warehouse (BW) there are multiple key types; the most important are the INTERNAL and the EXTERNAL key formats. The INTERNAL key is a unique identifier for all users (user-locale independent), whereas the EXTERNAL key can be user-locale dependent.

By default, all methods that deal with member keys (as returned values or as parameters) work with the INTERNAL key format. This keeps applications language-independent to ensure that they can work for multilingual user groups. However, you might want to choose the EXTERNAL key format due to various situations or reasons:

- The EXTERNAL key format is more compact and either the application is known to be only used by users sharing the same locale, or the EXTERNAL key format is known to be locale-independent.
- The number of complex selections is more than the single values that need to be passed as parameters for the API methods `setFilter` or `setVariableValue`. In this case, the SAP Business Explorer selection syntax ("INPUT\_STRING") can be used, which references individual members by their EXTERNAL key.

For these special use cases, SAP BusinessObjects Design Studio provides Ext variants of methods, for example `setFilterExt`, `setVariableValueExt`.

## Writing script statements

In the design tool, there are two different approaches for writing script statements:

- You can write scripts in a free form fashion with the script editor and activate the content assistance at any time by clicking `CTRL` + `Space`.
- If you activate the content assistance right from the beginning, you can also use the statement wizard. This guides you through the necessary steps and then creates a script statement.

The statement wizard is functionally limited compared to free form scripting. However, if you are unfamiliar with scripting, it might be helpful to use the statement wizard and learn from the generated script statement.

---

## API Reference

In the API reference, you will find all the methods and their descriptions.

## Related Information

[Using the Script Editor \[page 277\]](#)

[Using the Statement Wizard \[page 281\]](#)

### 34.1.1 Using the Script Editor

#### Prerequisites

Before you can use the script editor, you must add the components that enable user interaction (such as [button](#) or [pagebook](#)) to your application. Also, you might have to add the appropriate data sources and assign them to the components, if this is required by your scenario.

#### Context

#### Procedure

1. Click the [On Click](#) or [On Select](#) property of the corresponding component that enables user interaction.
2. Click the [Edit the script](#) button next to the [On Click](#) or [On select](#) property.  
The script editor opens and displays the name of the property (event) (example: "On Click") and the component to which the script will be assigned (example: "Button\_1").
3. Type in one or more statements in this supported format:  
`<ComponentVariable>.<function>(<arguments>);`

#### ➔ Tip

You can activate the content assistance at any place in the script by pressing `CTRL` + `Space` on your keyboard. If you activate the content assistance right from the beginning, you can also switch to the statement wizard that guides you through the statement creation process.

4. Click [OK](#) to close the script editor.

---

## Results

You have created a script for a component that enables the user to interact in the application.

## Related Information

[Content Assistance in the Script Editor \[page 278\]](#)

[Error Analysis in Scripting \[page 279\]](#)

[API Reference \[page 423\]](#)

### 34.1.1.1 Content Assistance in the Script Editor

The script editor enables you to easily create scripts and use them for user interaction enablement. It supports you by providing a variety of useful functions, hints and checks:

#### Opening the script editor

You open the script editor by editing the *On Click* or *On Select* property of the component that you have dragged and dropped into the editor.

#### Automatic syntax check

The system checks the syntax of your script in the background. Whenever there is a mistake, you will see an error marker on the left hand side and the bad code is underlined. You can hover over the error marker or the underlined text to see the error message.

#### Auto correction

In some cases you will find a "Quick fix" that fixes the error automatically. Just check the correction links in the error message popup.

#### Display of additional information

When you hover over the script text with the mouse and keep the mouse pointer still, you see additional information about the text below the mouse pointer. For example, if you hover over a data source alias name, you

---

see the corresponding query or query view data source. Or if you hover over a dimension, you see its display name.

## Intelligent proposal offering by content assistance

When entering the script text, you can press **CTRL** + **Space** to see proposals on how to continue. The editor is an intelligent tool and knows which continuations make sense and which do not. If you press **CTRL** + **Space** in an empty script editor, you will see all the components and data source aliases of the application as proposals. You can select one and continue typing the script or press **CTRL** + **Space** again to get the next proposal.

In some cases the editor will show you proposals without you pressing **CTRL** + **Space**. For example, if you type the name of a data source alias and press **."**, all available functions for the data source alias will be displayed automatically.

## Relevance-ordered proposals and contextsensitive documentation

The proposals are ordered by relevance. The most likely proposals are displayed first. Proposals that have the same relevance are sorted alphabetically. If you select a proposal with the mouse or with the up/down keys, you will see some documentation about the proposal. For example, if you show proposals for a data source alias "DS\_1.", you see all available functions. Each function displays documentation.

## Linked mode and value help for member selection

If you have chosen a proposal for a function, the editor will automatically insert the function name, the opening and closing brackets "(...)" and placeholders for all arguments. The editor is now in the "linked mode" that allows you to enter the parameters. You can use the **Tab** key to switch to the next parameter and the **Shift** key to go back to the previous parameter. You will also recognize the little green bar behind the closing bracket. Proceeding to this bar using the tab, arrow keys or the mouse will exit the "linked mode". To fill the parameters, you press **CTRL** + **Space** again to see proposals for parameters.

Proposals for parameters depend on the function. The function "setFilter" will propose you dimensions in place. Members will be shown as proposals if the available number is not too high. Otherwise there will be a proposal that allows you to open the value help for member selection.

## 34.1.1.2 Error Analysis in Scripting

### Script validation

Each script is validated before it is executed. If an error is found, for example, in the syntax, the script is not executed. Instead a log entry and a message are created and displayed. The log entry in the [Error Log](#) view

contains detailed error messages for the script. At runtime, the end user will see a message in the message view stating that there is a problem in the script. The message contains a log ID that allows you - the application designer - to find detailed error information.

For performance reasons, validation is performed only once per script. However, a script with errors is never executed.

As an application designer, you can use the following application methods to analyze the general behavior of your application:

- `log`
- `alert`
- `createErrorMessage`
- `createWarningMessage`
- `createInfoMessage`


If you use the methods `log` and `alert`, the results are displayed in the design tool. For example, if you want to find out whether or not a certain script has been aborted, you can add the `log` or `alert` call statements with a user-defined message as the last line of the script. If the message does not appear in a dialog box (application method `alert`) or in the [Error Log](#) view of the design tool (application method `log`), this means the script has been aborted. You can add more messages to the script, to find the exact statement that causes the script to be aborted.

If you use the methods `createErrorMessage`, `createWarningMessage` or `createInfoMessage` the results of the analysis are displayed in the executed application and can be viewed on external machines or mobile devices (for example, iPads).

#### Note

The `log` and `alert` methods only work if the design tool is running. Messages will only appear on the machine where the design tool is installed and running and where user confirmation is expected. Therefore, any applications executed on an external machine or a mobile device are blocked if they encounter an alert statement in a script (assuming that you have activated external access for these applications on the preferences page).

## Using the Problems view

The [Problems](#) view displays script errors encountered during script validation at design time, after the application was saved. The [Problems](#) view shows errors and warning markers for all script-related issues, even if the script editor is currently not open. To display the [Problems](#) view, click  in the menu of the design tool. To examine and resolve the errors displayed in the view, simply double-click an error. This automatically selects the component with the bad script and the script editor is opened.

The Problem view is refreshed when you open and save an application. Therefore a marker can remain even if the problem is already fixed. This marker disappears when you save the application the next time.

#### Note

Differences between working in local mode and with the Business Intelligence platform:



- If you work locally, the marker remains when you close the application window or the Design Studio. You can double-click the marker to open the application window and navigate to the error. The *Problems* view displays all problems for all analysis applications in the local workspace.  
If you work with the BI platform, the markers of an application disappear when the application is closed. They are displayed again when the application is opened again. Therefore the *Problems* view only shows markers of open applications.

## Related Information

[Log \(log\) \[page 437\]](#)

[Alert \(alert\) \[page 432\]](#)

[Create Error Message \(createErrorMessage\) \[page 433\]](#)

[Create Info Message \(createInfoMessage\) \[page 433\]](#)

[Create Warning Message \(createWarningMessage\) \[page 434\]](#)

## 34.1.2 Using the Statement Wizard

### Prerequisites

Before you can use the script editor or the statement wizard, you must add the components that enable user interaction (such as *button* or *pagebook*) to your application. Also, you might have to add the appropriate data sources and assign them to the components, if this is required by your scenario.

### Context

The statement wizard guides you through the script statement creation process, step by step.

### Procedure

1. Click the *On Click* or *On Select* property of the relevant component that enables user interaction.
2. Click the *Edit the script* button next to the *On Click* or *On select* property.  
The script editor opens and displays the name of the property (event) (for example, "On Click") and the component to which the script will be assigned (for example, "Button\_1").
3. Press `CTRL +Space` to activate the content assistance.
4. Double click *New Statement Wizard...* in the content assistance window.  
The *New Statement* dialog box opens.
5. Select a statement from the available statements list and click *Next*.

#### ➔ Tip

You can also filter for statements by typing in the first characters of the statement name in the [Available Statements](#) field. The system automatically filters the statements while you are typing. The matching parts of the statement name are highlighted in the list. Click the eraser icon on the right side of the filter field to clear the filter. The eraser icon is only displayed after you have typed in characters in the filter field.

6. Set the parameters for your chosen statement and click [Finish](#).

Click [Back](#), to return to the previous step.

The statement you have created is displayed in the script editor.

7. Click [OK](#) to close the script editor.

## Results

You have created a script statement using the wizard. Note the following: You can modify the script afterwards in the script editor. However, if you make changes to a wizard-generated script, which are not supported by the wizard, the wizard cannot modify the statement anymore.

### 34.1.3 Selecting Members of a Dimension

When you use statements like `setFilter` for a dropdown box, you need to select single members of a dimension. You can select the members in the content assistance of the [Script Editor](#) dialog box, or, if you use the statement wizard, in the [Select Member](#) dialog box. The capabilities and options available in the [Select Member](#) dialog box depend on the type of the member's dimension. Dimension types currently supported:

- flat dimensions
- dimensions with a related hierarchy

The capabilities and options available in the [Select Member](#) dialog box also depend on the number of members that a dimension has. You can display a small or large quantity of members by adjusting the maximum threshold number of members. You can set this (default) threshold in the [Preferences](#) dialog box in the design tool.

#### ⚠ Restriction

With input help for member selection, you can only select single members. Multiple selections or ranges are not supported at present.

## Filtering members in flat dimensions

If the number of members is less than or equal to the maximum threshold number defined in the Preferences dialog box, the [Select Member](#) dialog box displays the members list with each member's text and key. You can now filter the members by typing a part of a member's text or key into the filter /input field. The list is filtered

while you type. Matching parts of member names or keys are highlighted in the list. The number of matches is displayed below the list.

#### ➔ Tip

Filtering is case-insensitive. You can use the following wildcard characters for filtering:

- An asterisk (\*) matches any sequence of zero, one or multiple characters.
- A question mark (?) matches a single character.

If your filter entry matches several members, the first member is always selected. You can use the currently selected member by pressing **OK**, double-clicking the selected member, or pressing **ENTER** on your keyboard.

You can discard the currently selected member by pressing **Cancel** or by pressing **ESC** on your keyboard.

To clear the filter field, click the eraser symbol on the right side of the filter field. This will display all members again. The eraser symbol only appears if you have entered characters in the filter field.

## Changing the sorting display and the threshold

You can switch the display sequence for the member text and key in the members list by selecting **Sort by key** or **Sort by text** in the popup menu. This menu appears when you click the arrow symbol above the input field. The system sorts the list automatically according to your choice.

You can define and change the maximum number of members displayed in the content assistance for the **Script Editor** dialog box and in the **Select Member** dialog box. Enter the required number in the **Maximum number of members to fetch from backend in content assistance** checkbox and/or in the **Maximum number of members to fetch from backend in dialog**. The default threshold is 20. If the number of available members exceeds this threshold number, the content assistance does not list single members. Instead it offers the **Select Member...** entry, which opens the **Select Member** dialog box.

## Searching for members

If the number of members exceeds the maximum threshold number, the **Member Selection** dialog box displays the members list with each member's text and key. The number of displayed members is cut off at the maximum threshold number. A message below the list indicates that the maximum threshold number of members has been exceeded and provides a link to the **Preferences** menu in the design tool, where you can configure this threshold number.

To search for members, first enter a part of a member's text or key into the search field. This enables the **Search** button. Then click the **Search** button or press **Enter** to perform the search.

#### ➔ Tip

The case-sensitivity of the search functionality is dependent on the backend system.

---

## Filtering and searching for members in dimensions with related hierarchies

If the number of members and the hierarchy nodes is less than or equal to the maximum threshold number, the [Select Member](#) dialog box displays the members in a hierarchical tree. The initial expansion level is determined by the corresponding setting in the BW query. Depending on how you close the dialog (table view or hierarchical view), the dialog opens in the mode again the next time it is opened. The total number of members is displayed below the tree.

If the number of members assigned to the hierarchy is less than or equal to the threshold, but the total number of members exceeds the threshold, then only the assigned nodes are displayed in the tree. The unassigned members are suppressed, and a dummy node is displayed for them instead. You can only filter for assigned members. If you want to filter for unassigned members, you need to switch to the leaves/table view.

If the number of assigned members exceeds the threshold, the Select Member dialog box initially displays the collapsed members tree. A message below the tree indicates that the maximum threshold number has been exceeded. You can now collapse and expand the nodes.

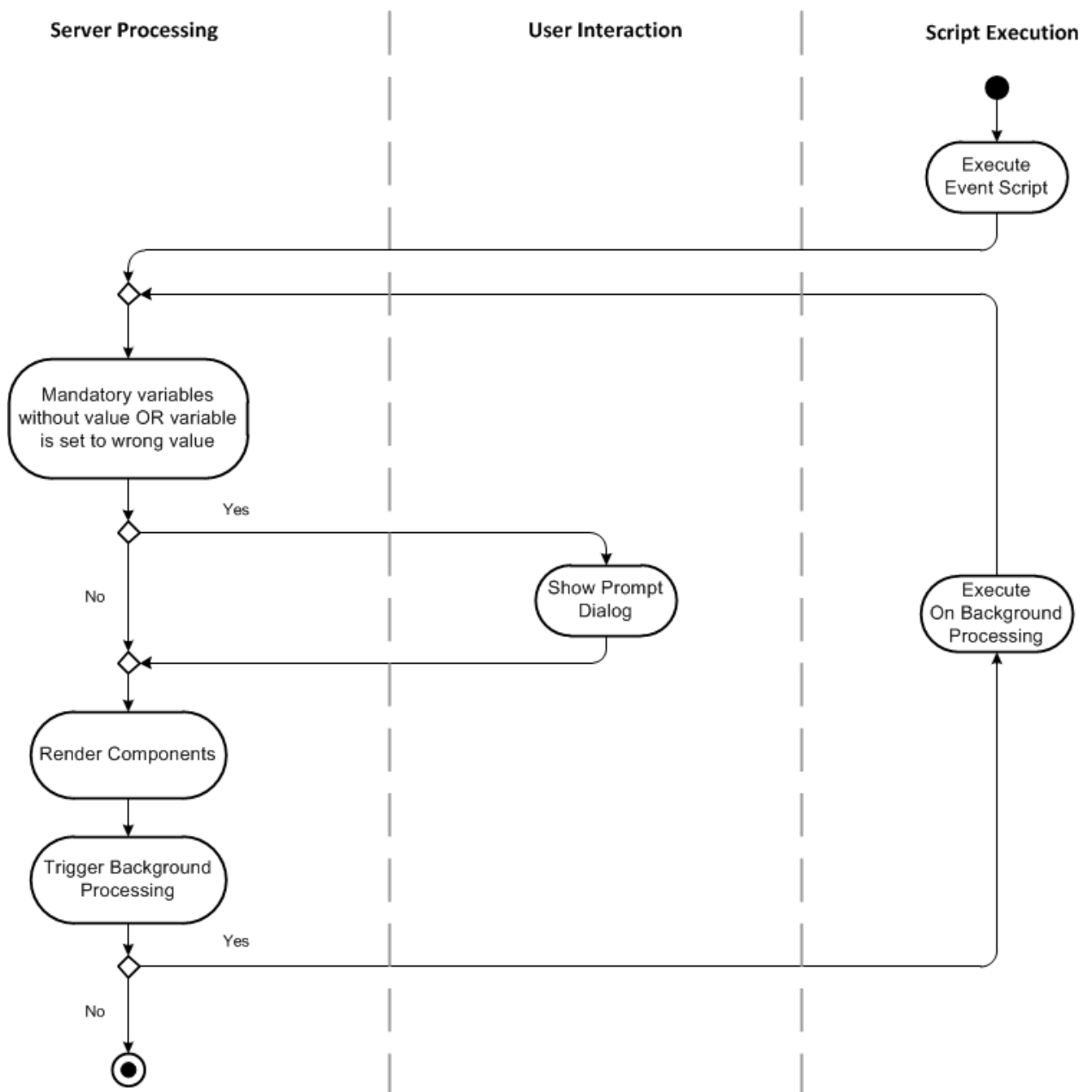
To expand or collapse the nodes and the entire hierarchy, press [Expand All](#) (+) or [Collapse All](#) (-).

You can now filter the members as described above. You can also change the sorting display of the member keys and texts, adjust the threshold number and display the hierarchy leaves as a table.

### 34.1.4 Script Processing Flow in Applications

The following graphic describes the processing flow for scripts that you create and insert into your analysis application.

**Figure 2: Script Processing Flow in Applications**



While executing an event script, the system checks on the server if there are mandatory variables in the application without values, or if a variable is set to a wrong value. If this is true for one of these cases, the prompt dialog box for entering the values appears. After the application user has entered the variable values, or if there are no mandatory variables or if the values are correct, the components are rendered before background processing is triggered. If there is a script in the application that should be processed in the background, the system checks again whether there are mandatory variables in the application without values or whether a variable is set to a wrong value. As a result, all the steps described above are repeated again. This can cause an endless loop. If there is no script to be executed in the background, the event script is finally executed.

#### **i** Note

If you use the method `APPLICATION.doBackgroundProcessing()` in the event *On Background Processing*, you create a loop. This can be useful, for example, when loading data from different data sources - one after the

---

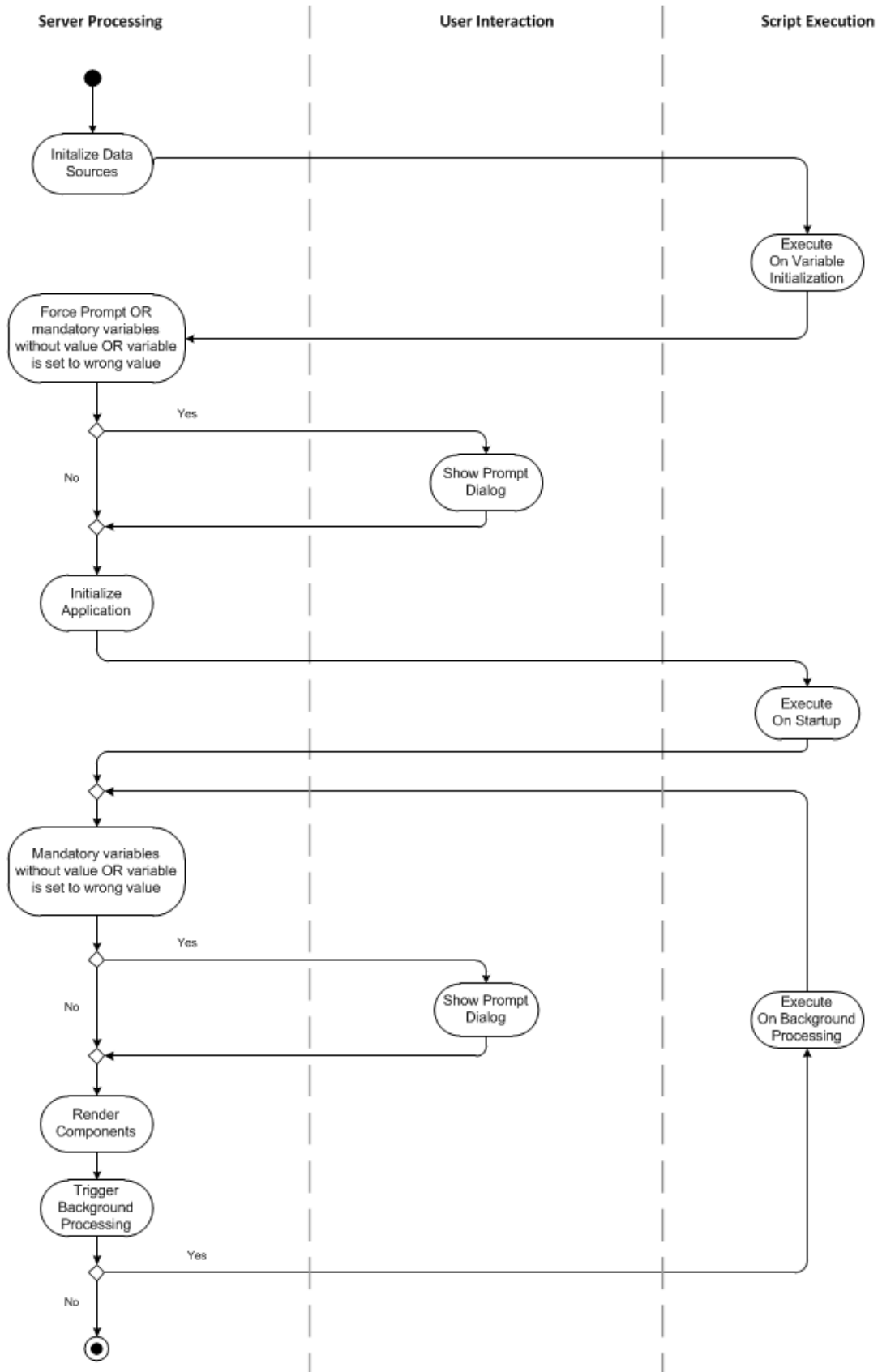
other. However, to avoid an endless loop, you must integrate a condition (in your script) that terminates the loop. For example, you can specify the number of calls by using a script like this: In this case, a global variable counter has been defined and initialized with 0. After the third call of the *On Background Processing* event, no further calls are executed. For more information, see *On Background Processing* event in [Properties of the Application \[page 325\]](#)

```
counter = counter + 1;
if (counter < 3 ) {      {
    APPLICATION.doBackgroundProcessing();
}
```

## 34.1.5 Process Flow at Application Start

This graphic describes the overall process flow in applications. It contains information about which application elements are initialized at a certain point in time, when scripts and dialog boxes are displayed and under what certain circumstances.

Figure 3: Process Flow at Application Start



---

First, the data sources are initialized on the server. This is followed by the event execution of *On Variable Initialization*. After executing *On Variable Initialization*, the system checks on the server if prompts have to be forced, if there are mandatory variables without values, or if a variable is set to a wrong value. If one of these cases is true, the prompt dialog box for entering values appears. After the application user has entered the variable values, or if there are no prompts to be forced, or if there are no mandatory variables, or if the values are correct, the application is initialized on the server and the *On Startup* event is executed. While executing the *On Startup* event, the system checks on the server if there are mandatory variables in the application without values, or if a variable is set to a wrong value. If one of these cases is true, the prompt dialog box for entering values appears. After the application user has entered the variable values, or if there are no mandatory variables, or if the values are correct, the components are rendered before background processing is triggered. If there is a script in the application that should be processed in the background, the system again checks whether there are mandatory variables in the application without values or whether a variable is set to a wrong value. As a result, all the steps described above are repeated again. This can cause an endless loop. If there is no script to be executed in the background, the event script is finally executed.

## 34.1.6 Business Cases

### 34.1.6.1 Setting a Crosstab Member as a Filter on Another Component

#### Context

The following business case example describes how to select a member from a crosstab, how to use the member as a filter for another component and how to jump from one tab to another tab in a tabstrip.

In this example, you create an application with two tabs in a tabstrip. In the first tab TAB\_1 you embed a crosstab MAIN\_CROSTAB with data source DS\_1. In the second tab TAB\_2 you embed a chart MAIN\_CHART with data source DS\_2. The two data sources should have common dimensions that can be used for selection. In this example, both data sources have the dimension "country".

In the MAIN\_CROSTAB, you allow users to select a member and set this member as a filter on the MAIN\_CHART. You also allow users to jump from TAB\_1 to TAB\_2.

For this scenario perform the following steps:

#### Procedure

1. Use a tabstrip component with two tabs (TAB\_1 and TAB\_2).
2. Embed a crosstab component in TAB\_1, name the crosstab MAIN\_CROSTAB and assign the data source DS\_1 to the crosstab. Embed a chart component in TAB\_2, name the chart MAIN\_CHART and assign the data source DS\_2 to the chart.
3. To enable user interaction in the crosstab MAIN\_CROSTAB by using events, set the property *Enable Selection* to *True* and write the following script for the On Select event of the  

```
crosstab:DS_2.setFilter( "country" , MAIN_CROSTAB.getSelectedMember( "country" ));  
TABSTRIP_1.setSelectedTabIndex(1);
```



## 34.1.6.2 Using Two Buttons with Toggle Function

### Context

In this example use case, you create an application with two different buttons - one below the other. Each button triggers a different function when the user pushes the button. The first button `BUTTON_FILTERON` allows the user to see the filter setting area, whereas the second `BUTTON_FILTEROFF` allows the user to hide the filter setting area. The buttons are embedded in the text component `TOOLBAR_BACKGROUND_1` that represents the background color of the buttons. A second text component `TOOLBAR_BACKGROUND_2` represents the background color for the filter area and is only displayed when the user clicks on `BUTTON_FILTERON`. The filter setting area itself is embedded in the grid layout component `TOOLBAR_LAYOUT`.

For this scenario perform the following steps:

### Procedure

1. Use a text component as the background color for the buttons (`TOOLBAR_BACKGROUND_1`).
2. Place two buttons, one above the other, inside the text component.

#### ➔ Tip

To match the exact size and position of the two buttons, you can copy `BUTTONFILTER_ON` (by using its context menu in the *Outline* view) and paste it under the *Layout* folder in the Outline view.

3. Use another text component for the background color of the filter setting area (`TOOLBAR_BACKGROUND_2`).
4. Create your filter settings area within the grid layout component `TOOLBAR_LAYOUT`.

#### i Note

For easier layout structuring, embed the two text components in a cell of the grid layout component (`TOOLBAR_LAYOUT`).

5. To enable user interaction, write a script for the *On Click* events of the buttons.

For `BUTTONFILTER_ON`:

```
TOOLBAR_BACKGROUND_2.setVisible(true);
TOOLBAR_LAYOUT.setVisible(true);
BUTTON_FILTEROFF.setVisible(true);
BUTTON_FILTERON.setVisible(false);
```

For `BUTTONFILTER_OFF`:

```
TOOLBAR_BACKGROUND_2.setVisible(false);
TOOLBAR_LAYOUT.setVisible(false);
BUTTON_FILTERON.setVisible(true);
BUTTON_FILTEROFF.setVisible(false);
```

### 34.1.6.3 Swiping in Pages

#### Context

In this example use case, you create an application containing a pagebook (PAGEBOOK\_1) with four pages (PAGE\_1, PAGE\_2, PAGE\_3, PAGE\_4). The user switches between the pages by swiping the pages horizontally. Set the relevant properties as described below:

Property	Property Value
selected page index	0 (specifies the visible page of the pagebook. In this case the visible page is the first page.)  <b>i Note</b> The first page of the pagebook has the page index 0, the second page has the index 1, the third page has the index 2, and so on. By setting the selected page index on 0, the pagebook starts with the first page.
transition effect	slide in
transition direction	horizontal
page caching	none

### 34.1.6.4 Switching Pages by Clicking Images

#### Context

In addition to the scenario described in *Swiping in pages*, you want to enable the user to switch between pages by clicking images. You also want to display a toolbar in the header of the application that changes the text (for example, from [Page 1](#) to [Page 2](#) or to [Page 3](#) or to [Page 4](#)) whenever the user clicks the image of a different page. Each image also has a special text (in this example, it is logical to use the texts [Page 1](#)... [Page 4](#)). Therefore you add four image components (PAGEICON\_1, PAGEICON\_2, PAGEICON\_3, PAGEICON\_4) to your application. Each component represents the corresponding page of the pagebook and each component has its own text component). You also add a text component to the header of the application. When the user clicks on an image, the relevant page is displayed and at the same time the text in the toolbar changes accordingly.

To configure this layout scenario proceed as follows:

#### Procedure

1. Prepare two images for each page (one for the selected image and one for the normal image). Save the images in the image folder or subfolder of the application directory.
2. Create a text component for the toolbar title (here TEXT\_TOOLBAR\_TITLE).

3. Create an image component for each page (here: PAGEICON\_n).
4. Create a text component (here: TEXT\_n) for each page. The text component and the image describe the page.
5. Write the following script statements for the On Click event of each image :

```
PAGEBOOK_1.setSelectedPageIndex(0);
TEXT_TOOLBAR_TITLE.setText(TEXT_1.getText());
PAGEICON_1.setImage("images/Icon_1_selected.png");
PAGEICON_2.setImage("images/Icon_2.png");
PAGEICON_3.setImage("images/Icon_3.png");
PAGEICON_4.setImage("images/Icon_4.png");
```

Adapt the script for each image to the corresponding page. For example, the script for the second page should look like this:

```
PAGEBOOK_1.setSelectedPageIndex(1);
TEXT_TOOLBAR_TITLE.setText(TEXT_2.getText());
PAGEICON_1.setImage("images/Icon_1.png");
PAGEICON_2.setImage("images/Icon_2_selected.png");
PAGEICON_3.setImage("images/Icon_3.png");
PAGEICON_4.setImage("images/Icon_4.png");
```

6. To enable the user to switch pages by clicking the images, you have to use the callable On Click event of the image component in the On Select event of the pagebook.

For this use case, the script for the *On Select* event of the pagebook should look like this:

```
if (PAGEBOOK_1.getSelectedPageIndex() == 0) {
    PAGEICON_1.onClick();
}
if (PAGEBOOK_1.getSelectedPageIndex() == 1) {
    PAGEICON_2.onClick();
}
if (PAGEBOOK_1.getSelectedPageIndex() == 2) {
    PAGEICON_3.onClick();
}
if (PAGEBOOK_1.getSelectedPageIndex() == 3) {
    PAGEICON_4.onClick();
}
```

## 34.1.6.5 getDataAsString for BW Queries with Structures

If you use BW queries with structures as data sources, you may want to read specific cells using the `getDataAsString` method. In the following business cases, this query is used (simplified presentation):

Table 47: BW query with a structure in the rows and the columns

Structure in the Rows		Structure in the Columns		
		Actual	Plan	Forecast
BW Key Figure 1 (Measure 1 = Sales)	Hierarchy Node 1 (= World)	Cell 1		
	Hierarchy Node 2 (= Europe)	Cell 2		
	Hierarchy Node 3 (= DE)	Cell 3		

BW Key Figure 2 (Measure 2 = Costs)	Hierarchy Node 1			
	Hierarchy Node 2			
	Hierarchy Node 3			

### **i** Note

As this business cases deal with BW queries, BW terminology is used. In BW, key figures are measures and characteristics are dimensions.

## Business Case 1: Read Cell 1

In the following business case, the key figure *Sales* is selected for a specific structure element in the columns (for example, Actual):

```
DS_3.getDataAsString("DCNFKHLVQTUD5AIFYGF4HQF4J",
{"DCNFKHLVQVGT4FC4BHDPM8JRN":"DCNFKHLVQSUZORHJ7MGXE26MR"});
```

In this business case a specific structure element is selected within a structure in the columns with "DCNFKHLVQVGT4FC4BHDPM8JRN=DCNFKHLVQSUZORHJ7MGXE26MR": "DCNFKHLVQSU ZORHJ7MGXE26MR" (for example, Actual). As no further specifications are made, the first cell (Cell 1) is read out.

Annotations:

- DCNFKHLVQTUD5AIFYGF4HQF4J is the UID (unique ID) of the key figure *Sales* in the BW key figure structure in the rows.
- DCNFKHLVQVGT4FC4BHDPM8JRN is the UID (unique ID) of the structure in the columns
- DCNFKHLVQSUZORHJ7MGXE26MR is the UID (unique ID) of a structure element in the columns (for example, Actual)

## Business Case 2: Read Cell 2

In the following business case the key figure *Sales* is selected and a specific structure element in the columns (for example, Actual) is filtered in combination with a hierarchy node (for example, Europe, which represents a text node):

```
DS_3.getDataAsString("DCNFKHLVQTUD5AIFYGF4HQF4J",
{"DCNFKHLVQVGT4FC4BHDPM8JRN":"DCNFKHLVQSUZORHJ7MGXE26MR"});
```

In this business case a specific structure element is selected within a structure in the columns with "DCNFKHLVQVGT4FC4BHDPM8JRN:DCNFKHLVQSUZORHJ7MGXE26MR" (for example, Actual). In addition, a specific hierarchy node (for example, EUROPE, which represents a text node in the hierarchy) is filtered. The value of Cell 2 is read.

Annotations:

- DCNFKHLVQTUD5AIFYGF4HQF4J is the UID (unique ID) of the key figure *Sales* in the BW key figure structure in the rows.

- DCFKHLVQVGT4FC4BHDPM8JRN is the UID (unique ID) of the structure in the columns
- DCFKHLVQSUZORHJ7MGXE26MR is the UID (unique ID) of a structure element in the columns (for example, Actual)
- "OPROFIT\_CTR": "HIERARCHY\_NODE/0HIER\_NODE/EUROPE" filters the hierarchy node "EUROPE" (which represents a text node in the hierarchy).

## Business Case 3: Read Cell 3

In the following business case, the key figure [Sales](#) is selected and a specific structure element in the columns (for example, Actual) is filtered in combination with a characteristic value (for example DE) which is part of a hierarchy:

```
DS_3.getDataAsString("DCFKHLVQTUD5AIFYGF4HQF4J",
{"DCFKHLVQVGT4FC4BHDPM8JRN": "DCFKHLVQSUZORHJ7MGXE26MR", "OPROFIT_CTR": "DE"});
```

In this business case, a specific structure element is selected within a structure in the columns with "DCFKHLVQVGT4FC4BHDPM8JRN" : "DCFKHLVQSUZORHJ7MGXE26MR" (for example, Actual). In addition, a specific characteristic value (for example, DE, which is part of the hierarchy) is filtered. The value of Cell 3 is read.

Annotations:

- DCFKHLVQTUD5AIFYGF4HQF4J is the UID (unique ID) of the key figure [Sales](#) in the BW key figure structure in the rows.
- DCFKHLVQVGT4FC4BHDPM8JRN is the UID (unique ID) of the structure in the columns
- DCFKHLVQSUZORHJ7MGXE26MR is the UID (unique ID) of a structure element in the columns (for example, Actual)
- OPROFIT\_CTR\_\_ZKBUSAREA=LS is the technical name of the characteristic value to be filtered (for example, DE, which is part of the hierarchy)

## 34.1.6.6 Export to Microsoft Excel

This feature enables the application user to specify and export a [Crosstab](#) from SAP BusinessObjects Design Studio into Microsoft Excel. If the [Crosstab](#) contains expanded hierarchies or expanded multiple hierarchies in either columns or rows, or both, the export to Microsoft Excel will represent the same expanded state as shown in the [Crosstab](#).

The following file formats are supported:

- CSV
- Microsoft Excel 2004 (.xls)
- Microsoft Excel 2007 (.xlsx)

You can add scripting to basic components to export data from a crosstab to a Microsoft Excel file.

### Example









To export crosstab\_1 to Microsoft Excel 2004, add the following script to a button:

```
APPLICATION.export(ExportType.EXCEL_xls, [CROSSTAB_1]).
```

### Note

- In Microsoft Excel 2000 and Microsoft Excel 2007, the metadata is separated from the crosstab data and displayed in a separate tab in the MS Excel spreadsheet. For example, the [Crosstab](#) itself is represented in sheet 1. Sheet 2 contains the metadata including runtime changes, applied filters or variables. Columns can also be resized depending on the length of the exported columns.
- Export to Microsoft Excel is not supported with SAP BusinessObjects Mobile.

You must read the following SAP Notes when exporting to Microsoft Excel.

SAP Note Number	Description
<a href="#">1917890</a> 	Listing the Microsoft Excel export feature as the last statement in the scripting.
<a href="#">1917891</a> 	Increasing jvm memory for large data export.
<a href="#">1917892</a> 	Using Microsoft Excel 2007 if a large column export is required.
<a href="#">1917943</a> 	Limiting the number of formatting elements in each crosstab.
<a href="#">1917944</a> 	The third parameter export setting currently not corresponding with the first parameter export type.
<a href="#">1917946</a> 	Microsoft Excel 2007 export feature not supported on BI platform 4.0.
<a href="#">2167615</a> 	Design Studio Microsoft Excel export cell size.
<a href="#">2239527</a> 	Design Studio Microsoft Excel hierarchy support.

## Related Information

[Naming Methods used for Exported Data and Metadata \[page 294\]](#)

### 34.1.6.6.1 Naming Methods used for Exported Data and Metadata

The following describes the naming method used for the exported data and its metadata.

Table 48: Exported Data






Name	Description	Additional Information
Excel file name	Composed of the application name and creation time.	None
Excel sheet name	Named according to the component name.	None
Excel sheet metadata	Export time	Format: dd:mm:yyyy hh:mm:ss

Name	Description	Additional Information
	Data Source	<ul style="list-style-type: none"> <li>• Data Source attached to the cross-tab user exported</li> <li>• Backend query name</li> <li>• Backend query description</li> </ul>
	Variables pre-set on the data source	<ul style="list-style-type: none"> <li>• Variable name</li> <li>• Variable Value Key</li> <li>• Variable Value Text</li> </ul>
	Static Filters pre-set to the data source	<ul style="list-style-type: none"> <li>• Dimension</li> <li>• Member Key</li> <li>• Member Text</li> </ul>
	Filters set on the data source at runtime by the user	<ul style="list-style-type: none"> <li>• Dimension</li> <li>• Member Key</li> <li>• Member Text</li> </ul>

## Parameters

Table 49:

Name	Type	Description
Export Type	String	<ul style="list-style-type: none"> <li>• To view all available export types, use <code>Ctrl+Space</code></li> <li>• The supported export types are CSV, Microsoft Excel 2004 (.xls) and Microsoft Excel 2007 (.xlsx)</li> </ul> <div> <p>➔ <b>Tip</b></p> <p>Always select the export types automatically displayed</p> </div>

Name	Type	Description
List of Crosstabs	Array	<p>The list and sequence of crosstabs that can be exported.</p> <ul style="list-style-type: none"> <li>Optional parameter.</li> <li>The sequence of elements in the array determines the sequence of sheets in the Excel file.</li> </ul> <div>  <b>Example</b>  [<code>cross_tab1</code>, <code>cross_tab1</code>] </div> <div>  <b>Example</b>  [<code>cross_tab7</code>, <code>cross_tab3</code>] </div> <div>  <b>Note</b>  All crosstabs will be exported if this parameter is not entered. </div>
Export Settings	String	<ul style="list-style-type: none"> <li>Optional parameter.</li> <li>Use <code>Ctrl+space</code> to view all the available options.</li> <li>CSV format is currently supported.</li> </ul> <div>  <b>Example</b>  <code>CSV_encoding_UTF8</code> </div> <div>  <b>Example</b>  <code>CSV_SEPARATOR_COMMA</code> </div>

## Related Information

[Export to Microsoft Excel \[page 293\]](#)



## 34.2 Working with Global Scripts Objects and Global Script Functions

You can create any number of *Global Scripts Objects*, a technical component type, which provide a grouping of global script functions.



On each global scripts object, you can create any number of script functions. Each script function has a configurable return type and any number of typed input parameters. The following types can be used for return value and input parameters:

- primitive types (int, Boolean, String, float)
- **<none>** for return values only
- DataSourceAlias
- all UI component types (Button, Tabstrip, ...) including SDK extension components

### Note

Array types are currently not supported.

### Creating new global scripts objects and script functions

To create a new scripts object, right-click the *Technical Components* type folder in the *Outline* view and choose  *Create Child* . The new object is displayed in the Technical Components folder.

To create a new global script function, right-click a global scripts object in the Outline view and choose *Create Global Script Function....* Enter the function name in the *Create Script Function* dialog box and click *OK*. The *Create Script Function* dialog box is displayed again. You can now enter a description for the function and enter the code for the global script function. Click *OK*, when you have finished the global script function. The function is now displayed under the corresponding global scripts object.

### Note

Due to security reasons, it is not possible to use `onClick` and other `on<ACTION>` methods in the script.

### Editing, renaming, deleting scripts objects and script functions

Right-click the scripts object or script function to:

- copy and paste scripts objects and script functions
- edit existing script functions by reopening the *Create Script Function* dialog box
- rename existing objects and functions (which automatically refactors each occurrence in scripts)
- delete scripts objects and script functions
- find all references to the scripts objects and script functions in scripts

## Example

- Example for return type and input parameters for the script function `computeAverage`

Script function: `computeAverage`

Return Type: `float`

Input Parameters: `value1` , `value2` and `value3` (all of type `float`)

Script code:

```
return (value1 + value2 + value3) / 3.0;
```

- Example for restyling a button using CSS classes. The style changes when the button is enabled or disabled

Script function: `styleButton`

Return Type: `<none>`

Input Parameters: `button` of type `Button`, `enabled` of type `boolean`

Script code:

```
if (enabled) {  
    button.setCSSClass("enabledCSSClass");  
} else {  
    button.setCSSClass("disabledCSSClass");  
}
```

## 34.3 Enabling Text Translation in Analysis Applications

### Context

In addition to the texts from the data sources, analysis applications can contain translatable texts, like labels on buttons or messages, that are created by you, the application designer. If you want to provide your analysis applications in different languages, you need to configure the analysis applications accordingly.

#### Note

Texts from the data sources are provided in localized form. You do not need to have these texts translated (for example, master data or metadata of the selected data source).

The translatable texts created by you can be divided into static and dynamic texts:

Table 50:

Text Type	Example
Static text	Button label that stays constant during application lifetime
Dynamic text	Message text that a script joins together from multiple parts

For both text types, you need to flag your application to be translatable by adding a `Text Pool` component to your application.

---

## Procedure

1. In the design tool, go to the [Outline](#) view.
2. Right-click [Technical Components](#) and choose [Create Child](#).
3. Choose [Text Pool](#).

## Results

The system collects all translation-relevant texts that you enter as property values in the [Properties](#) view for different components (like buttons or tooltips in this [Text Pool](#) component), and saves these texts for translation. The procedure for storing and translating the texts depends on the platform you use.

## Next Steps

In addition to the static texts you enter as property values in the [Properties](#) view for different components like buttons or tooltips, you can create dynamic texts that consist of different translation-relevant text parts. You create these dynamic texts with scripting. The dynamic texts are also collected in the [Text Pool](#) component.

## Related Information

[Using Scripts for Translatable Dynamic Texts \[page 299\]](#)

[Working With Translatable Texts in Analysis Applications \[page 300\]](#)

# 34.3.1 Using Scripts for Translatable Dynamic Texts

## Context

Application designers can write a script that joins together translatable texts from multiple parts. You can use this function to create message texts, for example.

## Procedure

1. After adding a *Text Pool* component to your application, select your *TEXT\_POOL* in the *Outline* view.
2. Go to the *Properties* view and under *Texts*, click the *Edit Texts...* button.
3. In the *Edit Text Pool Entries* dialog box, choose *Insert*.
4. Enter a key for your text.  
You need this key in your script to retrieve the text.
5. Enter your text in your original language.

### ➔ Tip

You can use placeholders (like {0} or {1}, for example) for dynamic parts of the text.

6. After inserting the required texts to the text pool, choose *OK*.
7. To retrieve the translated texts, replace the placeholders and display the whole text, create a script as follows:

```
var
    translated = TEXT_POOL.Key1;
var placeholdersReplaced = Convert.formatString(translated,
    [ "Replacement1" , "Replacement2" ]);
TEXT_1.setText(placeholdersReplaced);
```

## 34.3.2 Working With Translatable Texts in Analysis Applications

In analysis applications, texts created by the application designers (for example, button texts) are translatable. When the BI platform is used, these texts can be translated using the Translation Management Tool and stored in the applications InfoObject for each analysis application. When SAP NetWeaver is used as the platform, these texts are stored in the BW system in the TLOGO object table *RSAO\_T\_TEXT* for each analysis application. You can translate the texts with standard translation tools, for example, transaction *SE63*. For more information, see “Transaction SE63” on SAP Help Portal at <http://help.sap.com>.

At design time, with SAP NetWeaver, **application designers need to enter the texts in English**. When they save the analysis application, the design tool automatically saves the texts with language key *EN* in the TLOGO object table *RSAO\_T\_TEXT*. The texts are stored separately for each analysis application.

### i Note

If the text key changes, the system deletes all texts with the unused old key in all languages. The text key consists of the name of the component and the property.

Example of an analysis application with a button:

Table 51:

Text_Key	Text	Language
BUTTON_1.TEXT	Action	EN

Text_Key	Text	Language
BUTTON_1.TEXT	Aktion	DE

The application designer changes the text key of the button to BUTTON\_2.TEXT. The system deletes all BUTTON\_1.TEXT entries in EN and DE and saves the following new entry:

Table 52:

Text_Key	Text	Language
BUTTON_2.TEXT	Action	EN

**In the design tool, the original texts entered by the application designer are always displayed.**

At runtime, the texts of the analysis application appear in the BW logon language of the application user. If there are no translated texts available in this language, the texts of the analysis application (.biapp file) are displayed. These are the original texts that the application designer created in English.

With the BI platform, when transporting analysis applications from one system to another, the texts are automatically transported together with the analysis application.

With SAP NetWeaver, when transporting analysis applications from one BW system to another, the texts are automatically transported together with the analysis application (TLOGO object of type AZAP).

For more information about the required SAP NetWeaver Support Packages, see SAP Note [2051284](#).

## 34.4 Deploying SDK Extensions

In addition to the standard palette of components in SAP BusinessObjects Design Studio, you can install 3rd party components developed with the Design Studio SDK. Adding these 3rd party components, known as Design Studio SDK extensions, to your SAP BusinessObjects Design Studio installation enables you to create and execute local analysis applications containing these SDK extensions.

Furthermore, you can install new chart types developed with the SAP Lumira SDK. These SDK extensions, also known as CVOM chart extensions, are added to the list of additional chart types for the standard chart component. You can create and execute local analysis applications containing these new chart types.

### **i** Note

CVOM chart extensions are not supported in SAP BusinessObjects Design Studio if SAP HANA is used as the platform.

Before application users can launch analysis applications containing SDK extensions from one of the supported platforms (SAP NetWeaver, BI platform or SAP HANA), your administrator needs to deploy the SDK extensions to the required platform. For more information, see “Deploying SDK Extensions” in the corresponding *Administrator Guide: SAP BusinessObjects Design Studio* on SAP Help Portal at <http://help.sap.com/board>.

For more information about creating SDK extensions using the Design Studio SDK, see the *Developer Guide: Design Studio SDK* on the SAP Help Portal at <http://help.sap.com/board>.

---

For more information about creating CVOM chart extensions using the SAP Lumira SDK, see the *SAP Lumira Visualization Extensions Developer Guide* on SAP Help Portal at <http://help.sap.com/lumira>.

## Related Information

[Configuring Additional Chart Types \[page 211\]](#)

### 34.4.1 Installing Design Studio SDK Extensions to SAP BusinessObjects Design Studio

#### Context

You can add extensions developed with the Design Studio SDK to your SAP BusinessObjects Design Studio installation as new components.

#### Procedure

1. In SAP BusinessObjects Design Studio, choose **Tools** > *Install Extension to Design Studio...*.
2. Depending on where the SDK extension is located, proceed as follows:
  - For locally saved extensions, choose *Archive...* and select the archive file containing the SDK extension, under `C:\SampleExtension.zip`, for example.
  - For extensions stored on a Web server, enter the URL of the Web server.
3. Choose *OK*.
4. Select the required feature, for example, *SampleExtensionFeature*.
5. Select the Design Studio extensions that you want to install.
6. Choose *Finish* to proceed with the installation.
7. Choose *Next* and again *Next* to confirm the installation.
8. Accept the terms of the license agreement and choose *Finish*.
9. Choose *Yes* to allow SAP BusinessObjects Design Studio to restart.

#### Results

The SDK extension components appear in the *Components* view of SAP BusinessObjects Design Studio as new components.

The components are stored under `<user home directory>\Analysis-config`.

---

## 34.4.2 Installing CVOM Chart Extensions for SAP BusinessObjects Design Studio

### Context

You can add CVOM chart extensions, developed with the SAP Lumira SDK, to your SAP BusinessObjects Design Studio installation as new chart types for the standard chart component.

### Procedure

1. In SAP BusinessObjects Design Studio, choose **Tools** > **Install CVOM Chart Extension...**.
2. Depending on where the CVOM chart extension is located, proceed as follows:
  - For locally saved extensions, choose **Archive...** and select the archive file containing the extension, for example `C:\SampleChartExtension.zip`.
  - For extensions stored on a Web server, enter the URL of the Web server.
3. Choose **OK** and then choose **OK** again to confirm the installation.
4. To restart SAP BusinessObjects Design Studio, choose **Yes**.

### Results

When you create or open an analysis application with a chart component, the CVOM chart extensions you have added are listed as new chart types.

### Next Steps

In the **Properties** view of the chart component, under **Chart Type**, choose **Additional Chart Types...** The CVOM chart extensions are listed as new chart types in the **Additional Charts** dialog box.

### Related Information

[Configuring Additional Chart Types \[page 211\]](#)

---

## 34.4.3 Removing Extensions from SAP BusinessObjects Design Studio

### Context

You can remove SDK extensions that you have added to your SAP BusinessObjects Design Studio installation as follows:

### Procedure

1. In SAP BusinessObjects Design Studio, choose **Help > About...**.
2. Click the *Installation Details* button.
3. Select the feature containing the SDK extension, for example, *SampleExtensionFeature*.
4. Choose *Uninstall...*
5. In the *Uninstall* wizard, choose *Finish*.
6. Choose *Yes* to allow SAP BusinessObjects Design Studio to restart.

### Results

The SDK extension components are removed from the *Components* view of SAP BusinessObjects Design Studio. CVOM chart extensions are removed from the list in the *Additional Charts* dialog box.

## 34.5 Creating a Generic Analysis Template for RRI Jump Targets

### Context

Before application users can use the report-report interface (RRI) in analysis applications, you or your administrator need to configure the jump targets for the query that is used as data source in the analysis application. For more information, see

- “Using the Report-Report Interface” on SAP Help Portal at <http://help.sap.com>
- “BEx Query as Recipient” on SAP Help Portal at <http://help.sap.com>

If the jump target, which is configured for the data source query, is also a query, the target query is launched as a BEx Web application by default. In order to avoid this and ensure that the jump is handled by Design Studio and



the target query is displayed as an analysis application, you or your administrator need to specify a generic analysis template on your platform.

## Procedure

1. In the design tool, create an analysis application that can be used as the generic analysis template.

As this analysis application will be used for all query jumps, it should be a very generic application, which can handle basic analysis of an arbitrary query. In general, it must conform to the following constraints:

- Contains only one data source, which is loaded in script.
- Accepts query ID as URL parameter `XQUERY` (for SAP NetWeaver mode).
- Accepts query ID and system ID as URL parameters `XQUERY` and `XSYSTEM` respectively (for the BI platform mode).
- Loads the single data source (usually in the `On Startup` script) by calling `assignDataSource()` and passing the `XQUERY` and `XSYSTEM` parameters to the corresponding parameters of `assignDataSource()`.

### ➔ Tip

A good starting point, and an example of a valid generic analysis template, would be to select [Generic Analysis Template](#) when creating an application in the design tool. Note that you do not need to assign a data source to this [Generic Analysis Template](#). When application users jump to the target query, the necessary query information is automatically added to the [Generic Analysis Template](#) as URL parameters (`XQUERY`, `XSYSTEM`).

2. Save your generic analysis template on the platform.
3. To configure the runtime in order to use this analysis application as the generic analysis template for query jumps, you need to set this template accordingly on the platform. For more information, see "Configuring the Report-Report Interface for Analysis Applications" in the *Administrator Guide: SAP BusinessObjects Design Studio on SAP NetWeaver* or *Administrator Guide: SAP BusinessObjects Design Studio on BI Platform*.

## 34.6 Exporting Analysis Applications

### Context

Application designers can export analysis applications to their PC or a network share. You can use this function for the following purposes:

- It makes it possible to provide the original source code of analysis applications to SAP and helps us to resolve problems that you report to SAP as customer messages.
- You can export applications, in order to import them to a different platform.

### Note



For example, you can export an application from one BI platform and import it to a different BI platform. You can also use it for exporting an application from the SAP NetWeaver platform to the BI platform, or even from one local mode platform to another local mode platform.

- You can use the export for regular backups of your complex analysis applications.

### Note

Exporting analysis applications is available in all modes (BI platform, SAP NetWeaver, SAP HANA and local mode).

## Procedure

1. Choose  [Application](#)  [Export](#)  in the design tool.
2. In the [Export Application](#) dialog box, choose [Browse...](#) and select the required analysis application .

If an application is currently open, this application is automatically preselected, but can be changed using the [Browse...](#) button.

3. If you want to have the MIME objects (for example, images or CSS files), which are referenced in the application, exported, keep the [Export Referenced Files](#) checkbox activated.

The system analyzes the selected application and auto-detects all MIME objects referenced by the application (including MIME objects referenced by script coding). If you use string concatenation in a script to construct an image file name, or if the application uses a CSS file that internally references images, for example, the auto-detection might fail. In such cases, you can add the files manually to the export.

4. (optional) If you have MIME files that the system cannot auto-detect, choose [Add File...](#) and select the required files.

### Note

When exporting the same application on a regular basis (for backup purposes), you only need to fill this list once. For subsequent exports, the list is already pre-filled from the export history.

5. Choose [Next](#).
6. To select the target folder for the ZIP file (containing the application and MIME objects) on your local PC or on a network share, choose [Browse...](#)

For subsequent exports, the folder is already pre-filled from the export history.

7. (optional) If you want to change the default name of the ZIP file, enter the required name under [ZIP File Name](#).

The default name of the ZIP file is the name of the application suffixed with a timestamp.

### Note

The timestamp suffix prevents from overwriting previous ZIP file versions of the same application. For regular backups of the same application, we recommend that you use the default name with the timestamp.

8. (optional) Under *Export description*, you can enter a change list description.

For example, **First working version of data source filtering**.

This description is displayed in the *Import Application* dialog box and can help you identify the correct version to be imported.

9. Click *Finish*.

The system creates a ZIP file, which at least contains two files on root level: `content.biapp` and `export.properties`. The `export.properties` file contains information about the application (like contained MIME objects, used back end connections and the source platform), which will be used when importing the application.

10. (optional) To quickly access the created ZIP file, click *Open Export Folder* in the pop-up window on the right bottom corner.

## Results

You can use the ZIP file for backup purposes of your application, or you can attach it to your customer message for troubleshooting purposes. If you exported the application, in order to import it to a different platform, proceed with the import process under ► *Application* ► *Import...* ►.

## Related Information

[Importing Analysis Applications \[page 307\]](#)

# 34.7 Importing Analysis Applications

## Prerequisites

You have exported an application, which you have created on a different platform, and you want to import it to another platform or the local mode. Note that for importing (uploading) applications, you should always use the same Design Studio version that you used to export (downloading) the application. For example, to import a 1.4 biapp file, you should use design studio version 1.4.

### Note

Importing analysis applications is possible in all modes (BI platform, SAP NetWeaver, SAP HANA and local mode).

Once you have exported the application as a ZIP file to a folder of your choice, you are now logged on to another platform or the local mode.



## Context

You can import applications, which you have previously exported, to a different platform or mode.

### Note


For example, you can export an application from one BI platform and import it to a different BI platform. You can also use it for exporting an application from the SAP NetWeaver platform to the BI platform, or even from one local mode platform to another local mode platform.

## Procedure

1. Choose  **Application**  in the design tool.
2. In the **Import Application** dialog box, choose **Browse...** to select the ZIP file of the application you have exported. Once you have selected a valid ZIP file (the ZIP file must contain content.biapp and export.properties), information about the application is displayed. This enables you to check if it is the correct application to import.  
  
Choose **Next**.
3. Specify the **Name** and **Description** of the target application.  
  
Choose **Next**.
4. (optional) The next dialog box page is only displayed if the exported ZIP file contains referenced files (images, CSS). You can choose whether to import these files.

### Note

In local mode and HANA mode, the default option is to import the files (otherwise the application will be incomplete after the import), whereas in BI platform mode and the SAP NetWeaver mode, the default option is not to import the files. These modes only support shared global MIME objects. This means that an import without MIME objects can still result in a complete application, and you might not want to overwrite existing MIME objects because they could also be used by others. If you choose to import the files, additional options are displayed that depend on the capabilities of the target platform:

- In local mode, you can either keep the original folder structure of the files, or create a subfolder in the application's directory, into which all files are imported (in a flat way).
- In SAP HANA mode, the target folder for the files cannot be selected, and an information text is displayed, which indicates that all files are imported into the application's package.
- In BI platform mode, you are offered three options. The first option, to keep the original folder structure, might be disabled if the folders in the ZIP file are not BI platform compliant (in other words, if the folders do not start with **Root Folder/** or **User Folders/<user>/**). This is always the case if the source platform was not a BI platform. The second option is to import all files into a new or existing folder (in a flat way which means without keeping the folder structure). The third option is a checkbox where you can choose to overwrite existing files.
- In NetWeaver mode, the MIME import wizard page is displayed, but an information text may appear to indicate that the import of MIME objects is not supported, and refer you to SAP Note [2111546](#) . The MIME import is only enabled if the minimum support package is implemented in the SAP BW system. This Note contains a number of ABAP function modules that are required by Design Studio to upload

MIME objects, delete MIME objects, create new folders in the MIME repository, and so on. If the MIME object upload is supported, the same options are displayed as in BIP mode. The difference is that the option to import into the original location is always enabled.

Choose [Next](#).

5. (optional) The next dialog box page is only displayed if the exported application contains SAP BW, SAP HANA or Universe data sources (not SDK data sources).

On the [Map Backend Connections](#) page, you can map the backend connections used in the exported application to backend connections available in the target platform. This is especially useful if the application contains a lot of data sources from the same system, because on the mapping page only one entry needs to be mapped. There is no need to go through every data source and change the backend system manually using the property sheet. Normally the mapping page suggests an appropriate mapping. For each backend connection, only connections that are of the same type as the exported connection can be selected as mapping targets. This means if the exported connection is, for example, a SAP BW connection, it can only be mapped to a SAP BW connection in the target platform. To choose another connection than the proposed one, click on the proposed connection in the [Should Be Mapped To](#) field and open the dropdown box. Now you can choose another suitable connection for the platform/mode that you are logged on to. This wizard page is optional; you are not required to map anything.

Choose [Next](#).

6. The [Summary](#) page lists all changes that will be performed once you choose [Finish](#). Since the import might overwrite existing files, you should check the list before finishing the wizard.

Choose [Finish](#).

## Results

You have imported an application to another platform or local mode.

## 34.8 Exporting Applications As Templates

### Context

You can create analysis applications and provide them as templates for your fellow application designers.

### Procedure

1. Choose [Application](#) [Export as Template...](#) in the design tool.
2. In the [Export Application As Template](#) dialog box, choose [Browse...](#) and select the required analysis application.

If an application is currently open, this application is automatically preselected, but can be changed using the [Browse...](#) button.

3. If you want to have the MIME objects (for example, images or CSS files), which are referenced in the application, exported, keep the [Export Referenced Files](#) checkbox activated.

The system analyzes the selected application and auto-detects all MIME objects referenced by the application (including MIME objects referenced by script coding). If you use string concatenation in a script to construct an image file name, or if the application uses a CSS file that internally references images, for example, the auto-detection might fail. In such cases, you can add the files manually to the export.

4. (optional) If you have MIME files that the system cannot auto-detect, choose [Add File...](#) and select the required files.

#### Note

When exporting the same application again, you do not need to fill this file list again. For subsequent exports, the list is already pre-filled from the export history.

5. Choose [Next](#).
6. To select the template folder on your local PC or on a network share, choose [Browse...](#)

For subsequent exports, the folder is already pre-filled from the export history.

7. Enter a template name.
8. Select the required template category.

If you have not defined a template category, you need to jump to the [Preferences](#) page by clicking [Configure Categories...](#) Here you can add your template categories that indicate the target device types recommended for a specific template, for example.

9. Under [Caption](#), enter the name of the template to be displayed in the [New Application](#) dialog box.

Note that this name is case-sensitive.

10. Under [Icon](#), you can define the icon that symbolizes your template in the [New Application](#) dialog box.

You can select the required icon in the dropdown box or choose [Browse...](#) to select a new icon from your file repository.

#### Note

Images larger than 48x48 pixels will be scaled down when displayed in the [New Application](#) dialog box.

11. Under [Description](#), you can describe the content and purpose of your template.

This description is displayed in the [New Application](#) dialog box and can help other application designers identify which template to select.

12. Click [Finish](#).

In the selected template folder, the system creates a new folder with the `content.biapp` file and an `.info` file with the template description.

13. (optional) To access the created `content.biapp` file quickly, click [Open Template Folder](#) in the pop-up window on the right bottom corner.

## Results

When creating new applications, other application designers can choose from the templates provided by SAP BusinessObjects Design Studio, and can also select the template you have exported.

## Related Information

[Maintaining Settings in the Design Tool \[page 38\]](#)

## 34.9 Using the Backend Connection Component for Data Source Browsing

The technical component *Backend Connection* enables the application user to select a generic data source at runtime by means of a data source selection dialog box. You can either use the predefined data source selection dialog box for runtime and configure it in the properties of the Backend Connection, or you can use the API to create your own user interface for browsing data sources. Before you can use the Backend Connection, you have to assign a system at design time. This can be done in the *Property* view of the design tool or using the API.

### Note

When using the API, note that the Backend Connection Component relates to the API object type `CONNECTION`. For more information, see chapter “Connection” in the API reference.

## Adding a Backend Connection component to an application

To add a backend connection component to an application right-click on the *Technical Components* folder of the *Outline* view and choose ► *Create Child* ► *Backend Connection* ▾. The backend connection component is displayed in the *Technical Components* folder.

## Prerequisites: Assigning a backend system

Before you can use the Backend Connection, you have to assign a system at design time. This can be done in the *Properties* view of the design tool or using the API.

### Note

Each backend connection always relates to one backend system. If you need multiple systems in your application, we recommend that you create multiple backend connection components rather than reassigning the system each time at runtime.

The use of the backend connection component is not supported for DSL data sources.

## Configuring the Data Source Browser dialog box

Using the technical component Backend Connection in an application automatically provides you with a predefined data source browser dialog box for the runtime. You can configure the dialog box by setting the properties of the Backend Connection component, or by using the API:

- You can give the dialog box a user-defined name ([Title](#) property).
- You can specify which tab is shown by default when the dialog box is opened ([Default Tab](#) property). Decide whether the dialog box is opened with the Search, Roles, Workspaces or Folders/InfoAreas tab.
- You can specify which tabs are visible in the dialog box. By default all tabs (Search, Roles, Workspaces or Folders/InfoAreas tab) are visible, but you can change the visibility of the tabs as required.
- You can change the size of the dialog box ([Width](#) and [Height](#) properties). By default, the dialog box size is automatically adjusted to suit the screen size.

The UI of the data source browsing dialog box can have different appearances, depending on the chosen system for the Backend Connection component:

Table 53:

Data Source Browser dialog box for SAP HANA connections	Data Source Browser dialog box for SAP BW connections
For SAP HANA, the dialog box contains a maximum of two tabs, the <a href="#">Search</a> tab, where you can search for data sources, and the <a href="#">Folders</a> tab, which shows the folder/tree structure of the SAP HANA data repository.	For SAP BW the dialog box contains up to four tabs, the <a href="#">Search</a> tab (just like in HANA), the <a href="#">Roles</a> tab showing SAP BW roles, the <a href="#">Workspaces</a> tab showing SAP BW workspaces in a tree structure, and the <a href="#">InfoArea</a> tab, which shows the whole InfoArea tree of the SAP BW system

You can also set a configuration string to configure the dialog directly in the script, following a JSON syntax, for example `CONNECTION.showDataSourceBrowser('{"title": "JSON customized datasource browser", "width": "90%", "height": "90%", "defaultTab": "rolesTab", "hiddenTabs": ["foldersTab", "workspacesTab"]}');`

Make sure the JSON syntax follows the example (including ' " ') and that there are no line breaks in the JSON string.

## Opening the Data Source Browser dialog box

To open the dialog box in the application, you have to use the API method `connection.showDataSourceBrowser()`; for example, in the On Startup event of the application. The user interactions in the dialog box can be evaluated using two events - [On Data Source Browser Confirm](#) and [On Data Source Browser Cancel](#).

For further information on all methods relating to the object Backend Component, see the relevant section “Backend Connection” in the API documentation in this guide.



## Example

You use the Backend Connection component for an application, where the application user sets the system in a dropdown box and searches for a data source relating to the chosen data source, by opening the data source browsing dialog box with a click of a button.

For this example application, you have to add the following objects to your application:

- a crosstab
- a dropdown box with a tooltip (property *Tooltip*) **Select System**
- a button with a title (property *Text*) **Data Sources**

Perform the following steps:

1. Write the following script in the *On Startup* event of the application

```
var conns = CONNECTION_1.getConnections();
conns.forEach(function(element, index) {
    DROPDOWN_1.addItem(element.name, element.text);
});
```

2. Write the following script in the *On Select* event of the dropdown box with the tooltip **Select System**

```
var selected = me.getSelectedValue();
CONNECTION_1.setSystem(selected);
```

3. Write the following script in the *On Click* event of the button with the title **Data Sources**

```
CONNECTION_1.showDataSourceBrowser();
```

If you want to use the selected query for further purposes, proceed in the following way:

1. Write the following script in the *On Data Source Browser Confirmed* event of the backend connection component:

```
var ds = CONNECTION.getSelectedDataSource();
DS_2.assignDataSource(ds.connection, ds.type, ds.name, true);
```

### Note

As the script keeps running after the dialog box for data source browsing opens, you should use the script `CONNECTION.getSelectedDataSource()` in the event of the backend connection component and not in the event of the button component.

## 34.10 Using Processing Groups for Parallel Query Execution

Usually queries are executed sequentially (classic runtime). But even queries with short backend runtimes can add up to long server roundtrips when multiple queries are used in one application, and therefore you may need to execute queries in parallel. For this reason, with SAP BusinessObjects Design Studio 1.5 you can define groups of data sources, by using the data source property *Processing Group*. All of these groups can be executed in parallel.

---

With this property, you can assign each data source used in your application to one processing group. Each group is associated with a session. This means different groups are executed in parallel and all groups in one application run in parallel, whereas all data sources in one group still run sequentially.

Scripts are also executed sequentially even if parallel query execution is used in the application. To avoid the sequential execution of scripts, you can use the method `APPLICATION.loadDataSources()` or the data binding function. See [Load Data Sources \(loadDataSources\) \[page 437\]](#) and [Binding the Properties of Standard Components to Data Sources \[page 98\]](#)

## Prerequisites

When using parallel query execution, note the following points:

- Parallel query execution can only be used for a Design Studio deployed on the BI platform or locally.
- Parallel query execution can be used with all data sources (SAP BW, SAP HANA, DSL).
- Parallel query execution is only relevant for the following runtime phases:
  - during initialization of data sources on start-up
  - when fetching result sets during rendering
  - when submitting variables
  - during data binding
- Parallel query execution cannot be used with the following functions:
  - with planning-enabled queries:  
As planning-enabled queries need to run in one session and parallel query execution requires multiple sessions, these two functions cannot be used together. Therefore the system ignores planning-enabled queries as data sources if they are contained in any of the non-default processing groups.
  - with merged variables  
Variables used in the application need to be unmerged in order to use parallel query execution. If the application is set to use merged variables, the system ignores the processing group definition and an error message is displayed. If the application needs both parallel query execution and variable merging, then new Design Studio script methods can be used, which can emulate variable merging behavior.

## General considerations and guidelines for using parallel query execution

As an application designer you need to be aware that each processing group allocates additional resources. Therefore we recommend that you think about how and when to use additional processing groups. Take the following points into account when you want to run queries in parallel in your application:

- In general, you should not execute queries with a very short runtime in parallel as the overhead might be bigger than the performance improvement achieved by parallelization.
- When designing applications with SAP BW data sources, be aware that using additional processing groups causes additional load in the BW system.
- The decision about when processing groups are used and for which data sources should be mainly UI-based. This means that data sources, which should be visible at the same time on the UI, should potentially be categorized for different processing groups.
- When you use merged variables in the application and the variable processing takes a lot of time, it might not always be beneficial to change the application in order to use parallelization. In this case, the variables are not

merged anymore and the variable processing needs to run multiple times (for each query) instead of once for the merged container.

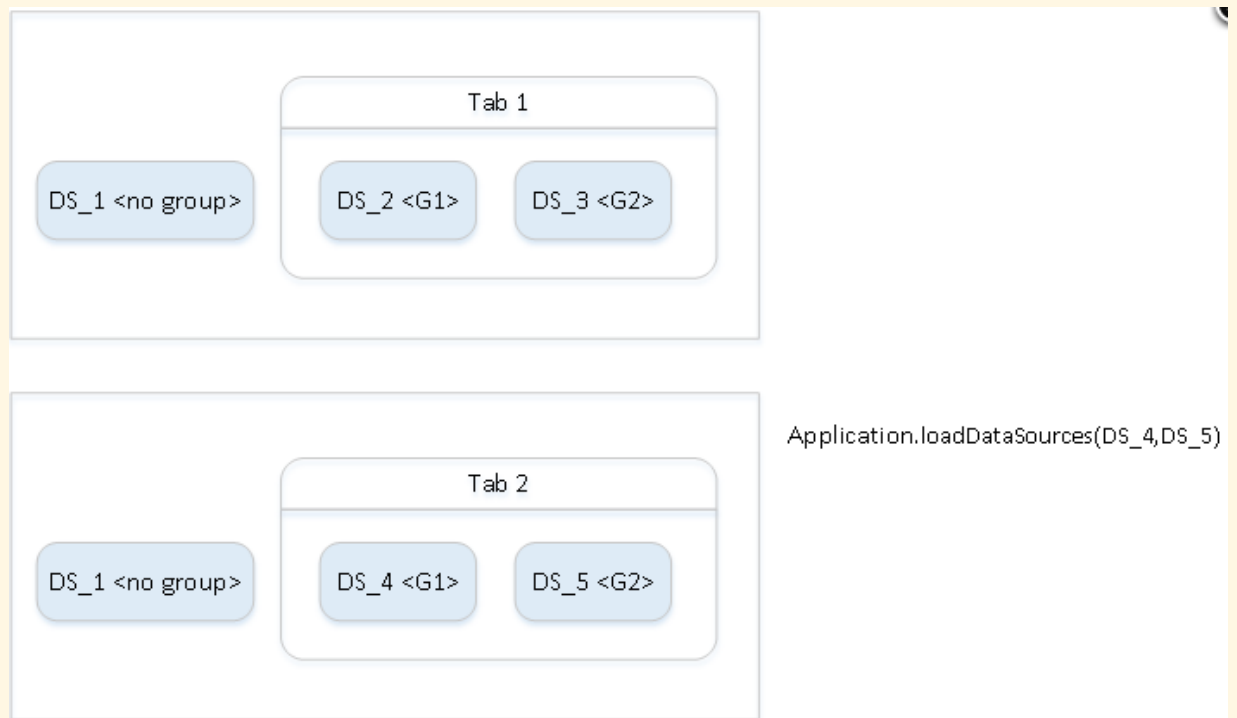
- We recommend that you always test different combinations of processing groups and design aspects to validate the best approach for each application.

For further information, see the respective blog at <http://scn.sap.com/community/businessobjects-design-studio/blog/2015/09/28/parallel-processing-and-scripting>.

### Example

The following image shows an example of a UI-driven configuration for an application with a tabstrip component and five data sources. The initial screen shows Tab 1 with two data sources (DS\_2 and DS\_3) plus an additional data source outside of the tabstrip which is always visible (DS\_1). For maximum parallelization, all three initially visible data sources were assigned to different groups (DS\_1 is contained in the default group, DS\_2 in G1 and DS\_3 in G2).

Neue Formulierung: The second tab contains two additional data sources. These data sources could reuse the groups from the data sources of the first tab, which means the number of parallel sessions would be kept low. As these additional data sources are not displayed initially, they should be initialized later by using the script method `Application.loadDataSources()`. Otherwise the initialization of these data sources slows down the start up time of the application.



---

## 34.11 Unmerging Prompts (Variables)

### Prerequisites

When unmerging prompts, note the following points:

- Unmerging prompts works only for a Design Studio deployment on SAP NetWeaver or the BI platform or for the local mode.
- Unmerging prompts for SAP HANA data sources only works if all data sources rely on the same analytic or calculation view.

### Technical Background

In SAP BW, variables (prompts) can either be merged or unmerged.

If variables are merged, the system creates one merged variable container for all data sources in an analysis application. This means, if the same variable is used in multiple queries, it can be entered only once.

If variables are unmerged, the system creates separate variable processors for each data source. This means, if the same variable is used in multiple queries, it has to be entered for each SAP BW query separately, and different values can be entered for different queries.

For further information on SAP BW variables, see “Variables” on SAP Help Portal at [Variables](#)

When creating analysis applications, you may have two reasons for unmerging variables:

- Business scenario: You want separate variable handling for the same variable (for example, you want to run the same query on different countries).
- Technical scenario: You want to unmerge variables for technical reasons (for example, performance requirements, parallel query execution). In this case, you might want to hide the unmerge option from the application user (for example, you can fill variables automatically in the background).

### Properties Setting in the Design Studio for (un)merging variables

In the Design Studio, there are two properties that relate to the functionality for unmerging variables:

- application property [Merge Prompts](#) with the values **true** and **false**; default: true
- data source properties [Text](#). With this property you can add a descriptive text that is displayed in the Prompts dialog box if prompts dialog box is filled (otherwise the data source alias specified in the name property will be used, for example, [DS\\_1](#))

## Prompts dialog box at runtime or design time (for mandatory variables)

The Prompts dialog box shows all variables for all queries in one application. In the Prompts dialog box, a prefix is added to the variable name, to enable you to identify which query the variable belongs to. By default this is the data source alias, for example *DS\_1*, or the text you entered in the *Text* property of the data source. For example, if you enter *myText1* in the Text property of the data source DS\_1, every instance of *DS\_1* in the dialog box is replaced with *myText1*.

*Prompt Settings / Select Prompts dialog box in the application properties*

This dialog box shows all variables for all queries in one application. Variables can be removed and filled automatically using the *On Before Prompts Submit* event of the application. Using this dialog box, you can remove (hide) mandatory variables. However, the mandatory variables will be displayed in the Prompts dialog box at runtime unless you set the *Merge Prompts* application property to *false*. This means mandatory variables will NOT be displayed in the prompts dialog box after they have been removed.

If mandatory variables are not filled, this will cause an error that you can avoid by filling the variable automatically using the *On Before Prompts Submit* event.

## Using the On Before Prompts Submit event

This event is called when the *OK* button in the Prompts dialog box is pressed and BEFORE variables are submitted. This allows users to copy variable values between data sources, which means application users do not necessarily have to enter the same variable for each query manually. For example,

```
DS_2.setVariableValueExt("0BC_HIER_MAND",  
DS_1.getVariableValueExt("0BC_HIER_MAND"));
```

```
DS_2.setVariableValueExt("0BC_HIER_MAND", DS_1.getVariableValueExt("0BC_HIER_MAND"));
```

## Using Load in Script property

When you use the *Load in Script* data source property in combination with the unmerge variables function, the prompts dialog box might be displayed multiple times, as in this case the variables are handled separately for each data source. To avoid this, you can copy variable values from another data source before loading a new data source. For example,

```
DS_2.setVariableValueExt("0BC_HIER_MAND",  
DS_1.getVariableValueExt("0BC_HIER_MAND"));  
DS_2.loadDataSource();
```

---

## 35 Troubleshooting

### 35.1 Auto-Recovering Applications

When you create an application in the Design Studio, the system automatically saves your application in the background as a snapshot at regular intervals. You can configure this interval in the [Preference](#) page under [Application Recovery](#). If the system crashes and you open the application you have worked on again, the dialog box [Application Recovery](#) appears. Here you can decide how to proceed with the auto-saved version. You can choose from the following options:

- [Save here:](#)  
Click this option, if you want to save the auto-saved version using a different name. Use the [Browse](#) button to choose or create a different name. This option opens both the last actively saved version and the auto-saved version. If you use an existing name for the auto-saved version only this application will be opened.
- [Overwrite last actively saved version](#)  
Click this option, if you want to overwrite the last actively saved version with the auto-saved version of the application.
- [Discard auto-saved version](#)  
Click this option, if you want to use the last actively saved version and discard the auto-saved version.



### 35.2 Using the Script Problems View

#### Context

To find, analyze and resolve script errors and problems at design time, you can use the [Script Problems](#) view in Design Studio. This view displays script errors in a list of lines, each connected to one single error. Each line provides information about the following:

- Error description
- Location of the script error
- Event script the error relates to
- Component the error relates to
- Application the error relates to
- Type of error

## Procedure

1. To display the *Script Problems* view, click  *View*  in the *View* design tool menu. The view is displayed below the layout editor.
2. Create the scripts for your application.  
If the scripts you write contain errors, these are displayed immediately in the *Script Problems* view.
3. Save the application. Any scripts errors that have occurred are now displayed in the view.
4. In the *Script Problems* view double-click the script error line you want to resolve. The system displays the *Script Editor* dialog box containing the script with the error.
5. Correct the script error and save the application. The relevant script error line is no longer displayed in the *Script Problems* view.

## 35.3 Working with Large eval() Statements

Sometimes Microsoft Internet Explorer® 9 has trouble with large `eval()` statements.

Microsoft Internet Explorer® 9 sometimes has trouble with large `eval()` statements in either of the following two cases:






- When the developer tools are open.
- When the one of the script debugging options is selected in the Microsoft Internet Explorer® 9 browser, on the *Advanced* tab.

To avoid this issue, we recommend not using the developer tools in the Microsoft Internet Explorer® 9 browser.

## 35.4 Setting Network Connections If Logon Problems Occur

### Context

After installing or deinstalling SDK extensions and restarting the design tool, you might not be able to log on to your BI platform. The *Authentication* dropdown box in the logon dialog box for the design tool is disabled.

This problem occurs when exceptions are specified with a "\*" wildcard character in the proxy settings of Internet Explorer under  *Internet options*  *Connections*  *LAN settings*  *Advanced*  *Exceptions*.

#### Example

The exceptions are defined as `localhost;127.0.0.1;*.mycompany.com`.

To communicate with the BI platform, SAP BusinessObjects Design Studio uses a third-party library, which can experience connectivity problems when "\*" wildcard characters are used.

---

To solve this issue, proceed as follows:

## Procedure

1. Start the design tool in local mode by pressing [Skip](#) in the logon dialog box.
2. Navigate to [Tools](#) > [Preferences](#) > [Application Design](#) > [Network Connections](#) .
3. Choose one of the following options:
  - Set the [Active Provider](#) to [Direct](#). This disables proxies completely.
  - Set the [Active Provider](#) to [Manual](#) and specify the proxy manually, omitting the problematic "\*" entries.
4. Restart the design tool.

## Results

This procedure only needs to be performed once.

For more information, see "Network Connection Preferences" in the online help. You can access this chapter by pressing F1 or the [Help](#) button in the [Preferences](#) dialog box.

# 35.5 Activating Runtime Traces

## Context

You can record traces to analyze problems in the design tool.

## Procedure

1. In the design tool, choose [Tools](#) > [Preferences](#) > [Application Design](#) > [Support Settings](#) .
2. Select the [Activate runtime trace](#) checkbox.
3. Click [OK](#).

## Results

The system will create a trace file on your local hard drive. The trace file is a text file that contains a log of the activities performed in the design tool's layout editor and in the executed analysis application. It is stored in the c : \<user>\Analysis-workspace\.metadata\.plugins\com.sap.ip.bi.zen\logs\RSTT folder.



---

The design tool also connects to the SAP BW system. This checks if the user requesting the trace has sufficient authorization to log traces (standard authorization object S\_RS\_RSTT). If this is the case, the SAP BW trace tool environment (transaction code RSTT in the connected BW system) is activated. With SAP NetWeaver as platform, a new trace containing your navigation steps will be created in the BW system. In BI platform mode or local mode (processing groups are used for parallel query execution in both modes), a new trace containing your navigation steps will be created in the BW system for each session (processing group). In addition, the main analysis application has its own trace.

**i Note**

For example, an analysis application with four processing groups starts five different traces in the BW system.

For information on how to replay and maintain the traces, see “Trace Tool Environment ” in the *SAP NetWeaver Library* , at <http://help.sap.com/netweaver>.

The recorded traces help us to resolve problems that you report to SAP as customer messages.

## 35.6 Activating SAP JCo Traces

### Context

You can record SAP JCo traces to analyze problems in the design tool.

### Procedure

1. In the design tool, choose **Tools** > **Preferences** > **Application Design** > **Support Settings** .
2. Select the **Activate SAP JCo Trace** checkbox.
3. Click **OK**.
4. Restart the design tool to enable the SAP JCo trace.

### Results

The SAP JCo trace is activated and the trace level is set to 8. You can collect all the SAP JCo trace files using the **Collect Support Information** function. The recorded traces help us to resolve problems that you report to SAP.

### Related Information

[Collecting Support Information \[page 323\]](#)

## 35.7 Viewing And Collecting Statistics Data At Runtime

### Context

If you encounter performance issues in your analysis applications, you can activate a runtime profiling function for your applications, which helps you to analyze the performance-critical processing steps.

### Procedure

1. In the design tool, go to ► [Tools](#) ► [Preferences](#) ► [Application Design](#) ► [Support Settings](#) ► and choose [Activate Runtime Profiling](#).
2. Execute the required analysis application either in local mode, or on your platform (SAP NetWeaver or BI platform).  
The system automatically adds the following parameter to the URL: `PROFILING=X`  
You can also add this parameter to the URL manually without activating the runtime profiling setting in the [Preferences](#) dialog box.
3. In the analysis application, a [Statistics](#) dialog box appears, where you can view the following statistics data:
  - Under [Runtime Statistics](#), you can view how much time is required to process each navigation step (and initial loading) on the server. This includes the processing time on the backend system (SAP BW or SAP HANA) and on the used platform (BI platform or SAP NetWeaver).  
With applications that use parallel query execution, the system displays the execution steps for each processing group separately. Whenever parallel execution starts, [Execute Processing Groups asynchronously](#) is displayed, followed by separate lines showing the execution of each processing group. The separation into processing groups is also reflected in the downloaded content.
  - Under [Rendering Statistics](#), you can view how much time is required to render the components of an analysis application in the Web browser for each navigation step (and initial loading).
  - Under [General Information](#), you can view the following information:
    - Timestamp of the application execution
    - Name and description of the application
    - Details about the data sources of the application. For each data source, the data source alias, the name of the object (for example, the query name in SAP BW systems or the view name in SAP HANA systems) representing the data source, the processing group (when parallel query execution is used), the connection type, and the initialization state are listed.
4. After reproducing the performance-critical navigation steps in your analysis application, choose [Refresh](#).  
All the relevant statistics data is downloaded and displayed.
5. To download the statistics data, choose [Download as Csv](#) or [Download as Text](#).

### Results

You can attach the recorded statistics data (\*.csv or \*.txt file) to customer messages that you send to SAP.

---

## 35.8 Collecting Support Information

### Context

If you encounter problems in the design tool, you can collect the relevant information to send to SAP in a zip file.

### Procedure

1. In the design tool, choose **Help > Support > Collect Support Information...**.
2. Select the target folder for the zip file.
3. Click **OK**.  
The support information is saved in the file `DS_Support.zip`.
4. To view the content of the zip file, click **View...**
5. Click **OK**.

### Results

You can attach the zip file to a customer message and send it to SAP.

## 35.9 Managing Logs in the Design Tool

### Defining the Log Level

You can specify how much information is stored in the log file. Under **Tools > Preferences > Support Settings**, the following options are available :

- **Warning**: The system stores exceptions, error messages, and warnings in the log file. This is the default option.
- **Error**: The system stores exceptions and error messages in the log file.
- **Information**: The system stores exceptions, error messages, warnings, and information messages in the log file.
- **Debugging Information**: The system stores exceptions, error messages, warnings, information messages and debugging information in the log file.

---

## Viewing Error Messages

You can view error messages in the design tool's [Error Log](#) view. To open this view, choose ► [View](#) > [Error Log](#) ►.


## Saving Error Messages

Press the [Export Log](#) button (in the upper right corner of the [Error Log](#) view) to save the error messages.

The exported error log files help us to resolve problems that you report to SAP in customer messages.

## 35.10 Best Practices

### 35.10.1 Improving Performance of Analysis Applications

As the topic **Improving Performance of Analysis Applications** is subject of regular and frequent changes, we recommend to read carefully the information in the SAP Note (Performance Hints for Design Studio Applications) [1931691](#) .

## 36 User Interface Reference

### 36.1 Properties of the Application

#### Syntax

Applications have the following properties:

Property	Property Value	Property Description
name	NameOfYourApplication	Displays the application name that you entered when creating the application.
type	NameOfObject	Specifies the object type, which the properties belong to. For an application, the type is <i>Analysis Application</i> .
description	DescriptionOfYourApplication	Displays the application description that you entered when creating the application. If you did not enter a description, you can do so now.
folder	YourFolderOnTheBIPlatform	Specifies the folder on the platform, where the application is stored (only relevant for deployment on the BI platform)
created by	UserNameForOperatingSystem	Displays the user name in the operating system.
creation time	none	Displays the time stamp when the application was created.
last modified by	none	Displays the user who made the last change.
last modification time	none	Displays the time stamp of the last change.
content version	number	Displays the version number of the application.
maximum number of steps back	integer	Specifies the number of undo steps permitted in an application. When set to a value more than zero, it also determines if the user can reset their application back to the original state. The default value is zero or disabled. The recommended maximum value is 20.

Property	Property Value	Property Description
drag and drop between components	true, false	<p>Specifies if drag and drop operations between different components are allowed or not. This property is set to <i>false</i> by default, meaning that drag and drop operations cannot be carried out between components. Drag and drop operations within one single component are still possible. Set this property to <i>true</i> if you want to enable drag and drop operations between components (for example, between the Navigation Panel and the Crosstab). For Drag&amp;Drop operations between the crosstab and other components, the crosstab property <i>Drag and Drop Enabled</i> has also to be set to <i>true</i>. For further information about drag and drop, see <a href="#">Crosstab [page 372]</a> and <a href="#">Working with Drag and Drop in Applications and Crosstabs [page 224]</a>.</p> <div> <p><b>i Note</b></p> <p>In Design Studio 1.6, drag and drop operations between different components are only possible between the crosstab and the navigation panel.</p> </div>
SAPUI5 m mode	true, false	<p>Specifies the rendering mode for the application and its components. This property is read-only and just for information purposes. You set the rendering mode in the <i>New Application</i> dialog box, when creating the application.</p> <p>Design Studio 1.6 now also supports the so-called <b>Main</b> part of the SAPUI5 library (sometimes just referred as “m”, these are controls in name space sap.m of SAPUI5). The Main (m) part was developed with the mobile use case in focus; these components are therefore specialized for mobile devices. For further information, see <a href="#">Using the SAPUI5 m Library [page 45]</a>.</p> <p>However, the usage of the SAPUI5 m library is not restricted to mobile scenarios; it also supports desktop applications. In order to adjust the visualization accordingly there are two</p>

Property	Property Value	Property Description
		form factors for the m mode: the <b>compact</b> form factor for the desktop and the <b>cozy</b> form factor with more spacing and padding for mobile use cases. For further information, see the application property <a href="#">Compact Form Factor</a> .
theme	SAP High Contrast Black, SAP Platinum, SAP Blue Crystal, SAP Mobile, SAP Gold Reflection (deprecated), SAP Mobile Black (deprecated)	Specifies the theme of the application. The SAP Platinum theme is recommended for desktop applications and the SAP Mobile theme is recommended for iPhone applications.  SAP Gold Reflection and SAP Mobile Black are deprecated themes that may have been used for applications created with a Design Studio prior to 1.5. If applications with these themes are opened in the design tool, only the technical keys (and not the text) are displayed for these themes in the <a href="#">Theme</a> property of the application.
compact form factor	true, false	This property is only relevant for applications and components that are rendered in the SAPUI5 m mode.  Specifies the form factor of the SAP UI5 m rendering mode. If set to <a href="#">true</a> , the components are rendered in desktop style of the SAPUI5 m mode. For further information, see <a href="#">Using the SAPUI5 m Library [page 45]</a>
custom CSS	none	Specifies the CSS style file for the application. You can change the default CSS style by entering the CSS style file of your choice. Using this function requires detailed knowledge of cascading style sheets (CSS) and Web design.  <b>⚠ Restriction</b>  If you use this function, you should be aware that SAP does not guarantee that custom CSS stylesheets will work properly. SAP does not guarantee that no UI-related and/or functionality-related problems will occur. SAP also does not guarantee that custom CSS

Property	Property Value	Property Description
		stylesheets will work properly after software upgrades.
position of message button	bottom right, bottom left, top right, top left	Specifies the position of the message button. The message button displays errors, warnings and information relating to the application at runtime.
loading indicator delay	default: 1000	Specifies the delay in milliseconds before the loading indicator is displayed.
position of message window	right, left	Specifies the position of the message window that is displayed, when the user clicks the message button at runtime.
displayed message types	none, errors, warnings and errors, all	Specifies the content in the message button.
merge prompts	true, false; default: true	Specifies if the same prompts (variables) that are used in different data sources in an analysis application are merged. If this property is set to true, the same variable used in multiple queries can be entered only once. If this property is set to false, the same variable used in multiple queries must be entered for each SAP BW query separately, and different values can be entered for different queries. For further information, see chapter <a href="#">Unmerging Prompts (Variables)</a> [page 316].
force prompts on start up	true, false	Specifies whether the prompts dialog box is displayed at runtime as soon as the application is displayed.
prompt settings		Specifies which prompts should be displayed in which order.
maximum number of members	<b>your number</b> , default: 100	Specifies the maximum number of members displayed in the value help for prompts. If the number of members is greater than the value for this property, no values are displayed. Instead, the application user has to limit the number by searching for the text or key of one or more values.
bookmark loading	hide and keep prompts, hide and clear prompts, show and keep prompts, show and clear prompts	Specifies the behavior of state and space prompts on loading a standard bookmark.
planning connection	none or listed planning system	Specifies the back end connection used for planning scenarios.



Property	Property Value	Property Description
planning model	none or listed planning models	Specifies the planning model (Business Planning and Consolidation (BPC) environment and model) of SAP Business Planning and Consolidation, version for SAP NetWeaver, Unified. This makes it possible to create integrated planning solutions in a BW system, thereby providing high flexibility and usability for specialist users. Using this property causes the Design Studio's planning functionality on the server to behave differently. For more information, see <a href="#">SAP Business Planning and Consolidation, version for SAP NetWeaver</a> on the SAP Help Portal.
global script variables	none	<p>Specifies global script variables that you can use for the whole application. To enter global script variables, choose <a href="#">Edit Global Script Variables</a>. A dialog box appears. Choose <a href="#">Insert</a>. You can now enter the name, type and default value of the global script variable and decide whether you want the variable to be used as a URL parameter.</p> <div> <p><b>i Note</b></p> <p>The name of a global script variable has to start with an <b>x</b> and must not end with <b>&lt;digit&gt;</b> when used as an URL parameter.</p> </div>
on variable initialization	none	<p>You can use this event to set query variables. If this event is used to set all mandatory variables, the variables will only be submitted once. The event is executed as follows:</p> <ul style="list-style-type: none"> <li>• before the event <a href="#">On Startup</a></li> <li>• before the first variable submit</li> </ul> <p>Opens the script editor. With this property/event, you can enable user interaction with the application by writing scripts. Click <code>CTRL +Space</code> to see the list of available methods for the application, data source alias and the components. Choose one of the following methods to use for this event:</p> <ul style="list-style-type: none"> <li>• for the object APPLICATION <ul style="list-style-type: none"> <li>◦ setVariableValue</li> <li>◦ setVariableValueExt</li> <li>◦ alert</li> </ul> </li> </ul>

Property	Property Value	Property Description
		<ul style="list-style-type: none"> <li>◦ createErrorMessage</li> <li>◦ createWarningMessage</li> <li>• global variables</li> <li>• all methods for the object CONVERT</li> </ul> <div> <b>i Note</b> <ul style="list-style-type: none"> <li>• If all mandatory variables are set by using the <i>On Variable Initialization</i> event, no variable screen is displayed.</li> <li>• If there are mandatory variables that are not set by using the <i>On Variable Initialization</i> event, the variable screen is displayed.</li> <li>• If the property <i>Force Prompts on Start Up</i> is set to true, the variable screen is displayed regardless of whether one or more mandatory variables are set. All variables set using event <i>On Variable Initialization</i> are set as default values in the variable screen.</li> </ul> </div>
on startup	none	<p>Opens the script editor. With this property/event, you can enable user interaction with the application by writing scripts. Click <code>CTRL+Space</code> to see the list of available methods for the application, data source alias and the components. Choose one of these methods or use the statement wizard to guide you through the statement creation process.</p> <p>For further information, see the method descriptions in the API reference.</p>
on background processing	none	<p>You can use this event to load data sources in the background. Although you can use any method in the API, it is recommended that you use this event with the methods for the data source alias.</p> <div> <b>i Note</b> <ul style="list-style-type: none"> <li>• To trigger this event, you must use the script method <code>doBackgroundProcessing</code>.</li> </ul> </div>

Property	Property Value	Property Description
		<p>This method executes the contents of the event.</p> <ul style="list-style-type: none"> <li>The execution is run after the script (which calls <code>doBackgroundProcessing</code>) has finished and the result has been sent to the front end.</li> </ul> <p>Click <code>CTRL</code> + <code>Space</code> to see the list of available methods for the application, data source alias and the components. Choose one of these methods (the most suitable methods are the methods for the data source alias) or use the statement wizard to guide you through the statement creation process.</p> <p>For further information, see the method descriptions in the API reference.</p> <p>There are two main scenarios in which the <i>On Background Processing</i> might be especially useful:</p> <ul style="list-style-type: none"> <li>guided navigation <p>A tree navigation refers to an application concept, where a user can only choose from a limited number of paths at a time, depending on what the user has selected previously. Depending on the measure that the user selects on the first page, three different pages can be shown afterwards. In this example, there are four data sources used, one initially selected and three others, depending on the user's selection. <i>On Background Processing</i> can be used here to load all three possible data sources in the background, while the user is still looking at the first data source to decide which path he/she will take.</p> </li> <li>tile effect <p>To use a tile effect, where data sources are loaded one after the other, the application designer can use the recursion functionality. If</p> </li> </ul>

Property	Property Value	Property Description
		<p>four different data sources are used in the application, and you want them to show their data as soon as one is loaded, the following script could be used:</p> <pre> if (Variable1 == 0) { DS_1.loadDataSource(); } if (Variable1 == 1) { DS_2.loadDataSource(); } if (Variable1 == 2) { DS_3.loadDataSource(); } if (Variable1 == 3) { DS_4.loadDataSource(); } Variable1 = Variable1 + 1; if (Variable1 &lt; 4) { APPLICATION.doBackgroundProcessing(); } </pre>
on before prompts submit	none	<p>You can use this event in combination with the Prompts Settings property. For more information, see chapter <a href="#">Unmerging Prompts (Variables) [page 316]</a></p>

## 36.2 Properties of the Data Source Alias

### Syntax

Data source aliases have the following properties:

Property	Property Value	Property Description
name	name of data source alias	Displays the name of the data source alias, for example, DS_1.
type	type of data source alias	Displays the type of the data source alias.
load in script	false, true	Specifies whether the data source alias is initialized in the script, instead of automatically loading the data source when the application is started.

Property	Property Value	Property Description
data source: name	name of underlying data source	Displays the name of the underlying data source, for example, the name of query or query view.
data source: connection	name of connection	Displays the connection of the data source.
data source: type	type of data source	Displays the type of the data source, for example, query view.
processing group		<p>This property can only be used if the Design Studio runs in BI platform mode or local. Also this property cannot be used for planning enabled queries and for merged variables.</p> <p>Specifies the processing group of a data source. Data sources in the same group are processed sequentially. Each group allocates additional resources ( for example, session in backend or thread on application server).</p> <p>As an application designer you need to decide which data sources should run in parallel and therefore need to be assigned to different processing groups:</p> <ul style="list-style-type: none"> <li>• For specifying a group at the beginning of design process, type in capital letters a group name, for example GROUP1.</li> <li>• The first typed in group name will be listed in the property if you want to set the property for another data source.</li> <li>• If you don't specify a group name, the so called default group is used. This default group does not allocate a new thread since it uses the current thread. Therefore the following configurations are behaving different: <ul style="list-style-type: none"> <li>◦ Configuration 1: DS_1 (&lt;no group&gt;), DS_2 (GROUP)</li> <li>◦ Configuration 2: DS_1 (GROUP1), DS_2 (GROUP2)</li> </ul> <p>Configurations like (2) where all data sources are assigned to a non-default group are not recommend</p> </li> </ul>

Property	Property Value	Property Description
		<p>since it unnecessary allocates an additional new thread.</p> <p>For more information on limiting the number of sessions and threads, see "Configuring the Number of Sessions for Parallel Query Execution" in the <i>Administrator Guide: SAP BusinessObjects Design Studio based on BI Platform</i></p> <p>For further background information on working the parallel query execution, see XXXX</p>
text	your Text	<p>This property enables you to add a descriptive text that is displayed in the Prompts dialog box. If no descriptive text is entered, the data source alias specified in the name property will be used, for example, <i>DS_1</i>. This property can be useful when unmerging variables. For further information, see <a href="#">Unmerging Prompts (Variables) [page 316]</a>.</p>
on result set changed	none	<p>Opens the script editor. With this property/event, you can enable user interaction with the application by writing scripts. Click <b>CTRL</b> + <b>+Space</b> to see the list of available methods for the application, data source alias and the components. Choose one of these methods or use the statement wizard to guide you through the statement creation process. For further information, see the method descriptions in the API reference.</p> <p>The script assigned to this event is executed if the result set for the data source is changed during a roundtrip. The script will be called in the following circumstances:</p> <ul style="list-style-type: none"> <li>• if a data source is initially assigned.</li> <li>• if the query model was changed (for example, by using a filter command) during the roundtrip.</li> </ul> <p>The script is called after all other scripts have been executed. That means that</p>

Property	Property Value	Property Description
		<p>multiple actions that modify this data source lead to one execution of the script.</p> <div> <b>Note</b> <p>An exception to this rule: if this data source is modified in an <i>On Result Set Changed</i> handler (of this data source or of a different data source), the script is called again.</p> </div>

## 36.3 General Properties for All Components

The following general properties are available for all components:

Property	Property Value	Property Description
name	String	Defines the unique name of a component. If you do not enter a name, the system takes the default name (for example CHART_1).
type	String	Displays the type of the component, for example, crosstab, chart, checkbox.
visible	true / false	Specifies whether a component is visible or not.
enabled	true / false	<p>Specifies whether a component is enabled. Disabled components do not allow user interaction.</p> <div> <b>Note</b> <p>This property is not available in all components.</p> </div>

### Example

#### Hidden buttons become visible

In an application with a dropdown box for filtering calendar years, the hidden buttons showing the quarters of the calendar year become visible. If the user selects 2011 for example, four buttons appear for filtering the data according to the four quarters of 2011.

## Example

### Disabled buttons

An application has a dropdown box for filtering calendar years and four buttons for the quarters of the calendar year. The buttons for quarters - for which no business data is available - are disabled. If the user selects 2012 at the beginning of July 2012 for example, only the first two quarters have business data available. Buttons Q1 and Q2 are enabled, and the user can filter the data for the first two quarters of 2012. Buttons Q3 and Q4 are disabled. The user sees that these buttons exist. However, the fact that these buttons are grayed out indicates that filtering for business data in Q3 and Q4 is not possible. Later on, in August or September for example, the Q3 button becomes active, as business data now exists for this period of time.

## 36.4 Display Properties for All Components

You use the display properties to specify the display of the component at runtime.

The following display properties are available for all components:

Table 54:

Property	Property Value	Property Description
CSS class	Your CSS class	Defines an additional CSS class for custom CSS. The CSS classes must have the format <b>myclass</b> (and not <b>.myclass</b> ). You do not have to use this property to be able to use <a href="#">Custom CSS</a> .

## 36.5 Layout Properties for All Components

You can define the layout properties by entering the values manually in the [Properties](#) view, or by dragging the borders of a component in the layout editor.

The following properties defining the layout of a component are available for all components:

Property	Property Value	Property Description
Top Margin	Numeric value in pixels or auto	Specifies the distance between the top margin of the component and the top margin of the application. Enter the numeric value in pixels, or set the value to auto.
Left Margin	Numeric value in pixels or auto	Specifies the distance between the left margin of the component and the left margin of the application. Enter the



Property	Property Value	Property Description
		numeric value in pixels, or set the value to <code>auto</code> .
Bottom Margin	Numeric value in pixels or <code>auto</code>	Specifies the distance between the bottom margin of the component and the bottom margin of the application. Enter the numeric value in pixels, or set the value to <code>auto</code> .
Right Margin	Numeric value in pixels or <code>auto</code>	Specifies the distance between the right margin of the component and the right margin of the application. Enter the numeric value in pixels, or set the value to <code>auto</code> .
Height	Numeric value in pixels or <code>auto</code>	Specifies the height of a component. Enter the numeric value in pixels, or set the value to <code>auto</code> .
Width	Numeric value in pixels or <code>auto</code>	Specifies the width of a component. Enter the numeric value in pixels, or set the value to <code>auto</code> .

#### Example

##### Crosstab with fixed margins

Top margin = 60  
 Left margin = 120  
 Bottom margin = 116  
 Right margin = 455

The values for width and height are set to `auto`. This means that the distances between the margins of the crosstab and the margins of the application are fixed, while the height and width of the crosstab are variable. The height and width of the application vary according to different screen or window sizes. In this case the height and width of the crosstab vary accordingly.

#### Example

##### Crosstab with fixed width and height

If you set the height and width of the crosstab to fixed values, one property of each axis is set to `auto`.

Top margin = 60  
 Left margin = 120  
 Bottom margin = `auto`  
 Right margin = `auto`  
 Width = 600  
 Height = 400

In this case, the left margin and the width of the crosstab are fixed, while the third property of the horizontal axis, namely the right margin, is variable. The top margin and the height of the crosstab are fixed, while the third property of the vertical axis, namely the bottom margin, is variable. With different screen or window sizes,

the height and width of the application vary. In this case, the bottom margin and the right margin vary accordingly.

#### **i** Note

It is not possible to set all three properties of one axis to fixed values. One property of each axis is always set to `auto`.

## 36.6 Analytic Components

### 36.6.1 Chart

Use the chart properties in the [Properties](#) and [Additional Properties](#) views to configure the settings of the [Chart](#) component. You can add charts to your application to present your data graphically. Charts can often emphasize irregularities or trends in your data, and help you focus your business analysis on those areas. When you drag and drop the [Chart](#) component into the layout editor, the component displays a graphic image of a generic chart. When you assign a data source to the [Chart](#) component, it then displays the data using the [Column](#) chart type, unless you have modified the chart type.

The properties of the [Chart](#) component include the following views:

- Properties
- Additional Properties:
  - Chart Area
  - Data Series
  - Chart CSS

The chart properties in the [Properties](#) view are described in the table below. The chart properties in the [Additional Properties](#) view are described in the Chart Area Additional Properties, Data Series Additional Properties and Chart CSS Additional Properties chapters, which are referenced below the Properties table.

## Properties

Table 55:

Property	Property Value	Property Description
data source	Name of data source alias, such as <a href="#">DS_1</a> .	<p>Displays all data source aliases. If you have created several data sources for the application, you can change the data source for the chart by choosing the corresponding data source alias.</p> <div><b>i Note</b> Because you can use the same data source several times within one application, you work in the layout editor using data source aliases as reference names.</div>
data selection	Selection string expressed in a JSON notation generated from the data selection.	<p>Dialog box allows you to select multiple rows or columns from the data result set to create a separate chart. Click the <a href="#">Add Selection</a> button to select the columns or rows that you want to appear in a separate chart.</p> <div><b>i Note</b> If your first data selection is in a row, your subsequent data selections must only be in rows. Similarly, if your first data selection is in a column, your subsequent data selections must only be in columns.</div>

Property	Property Value	Property Description
chart type	The property values as shown in the <a href="#">Property Description</a> column to the right.	<ul style="list-style-type: none"> <li>Specifies the type of chart in a drop-down list. Many chart types, such as column, bar, area, bubble, and water-fall, are available to help you visualize your data.</li> <li>The last element in the dropdown list is <a href="#">Additional Chart Types....</a> Selecting this option opens up an <a href="#">Additional Charts</a> dialog box from where you can assign measures and dimensions to additional chart types.</li> </ul> <div> <p><b>i Note</b></p> <p>If you have created and installed new chart extensions using the SAP Lumira SDK, these extensions will also appear in the <a href="#">Additional Chart Types...</a> dropdown list. For more information about creating SAP Lumira SDK extensions, see the <a href="#">SAP Lumira Visualization Extensions Developer Guide</a> on the SAP Help Portal at <a href="http://help.sap.com/lumira">http://help.sap.com/lumira</a>. Extensions developed with the SAP Lumira SDK can only be deployed to the BI platform and to SAP NetWeaver. They cannot be deployed to SAP HANA.</p> </div>
conditional formatting	Selection string expressed in a JSON notation generated from the conditional formatting applied to the selected chart.	Dialog box allows you to apply conditional formatting rules to measures or dimension members within a selected chart. These rules change the appearance of the chart when specific conditions are met.
swap axes	<ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul>	Select <b>true</b> to switch the column and row content. Select <b>false</b> to display the columns and rows as defined in the <a href="#">Edit Initial View...</a> dialog box. The default setting is <b>false</b> .
show totals	<ul style="list-style-type: none"> <li>true</li> <li>false</li> </ul>	When you select <b>true</b> , the totals for each dimension that you have added in the <a href="#">Edit Initial View...</a> dialog box are displayed in the chart. Select <b>false</b> to hide these totals. The default setting is <b>false</b> .

Property	Property Value	Property Description
show scaling factors	<ul style="list-style-type: none"> <li>• true</li> <li>• false</li> </ul>	<p>Select <b>true</b> to show scaling factors in the chart. The scaling factors displayed are the ones defined for the measures in the <i>Edit Initial View...</i> dialog box. Select <b>false</b> to hide scaling factors. The default setting is <i>false</i>.</p> <div> <p><b>Note</b></p> <ol style="list-style-type: none"> <li>1. If your measures are in the columns, the scaling factors will display in brackets in the axes, to the right of the measures. If your measures are in the rows, the scaling factors will display in brackets in the legend, to the right of the measures.</li> </ol> <p><b>Example</b></p> <p>Measure A (*1000)</p> <ol style="list-style-type: none"> <li>2. When you swap axes, the scaling factors will follow the measures.</li> <li>3. If there are mixed measures or units, they are not displayed on the chart, but are displayed in the crosstab.</li> </ol> <p><b>Example</b></p> <p>If a measure shows Net Sales in US\$ and Euro in the crosstab and <i>Show Scaling Factors</i> is set to <b>true</b>, in the chart, the scaling factor will appear beside the measures without the currency value.</p> <ol style="list-style-type: none"> <li>4. There are two ways to define scaling factors: <ul style="list-style-type: none"> <li>◦ using the context menu on the measure in the <i>Edit Initial View...</i> and selecting the required scale</li> <li>◦ scripting a component with an on-click event</li> </ul> </li> </ol> </div>

Property	Property Value	Property Description
dimension label	<ul style="list-style-type: none"> <li>initial view definitions</li> <li>text</li> <li>key</li> </ul>	<ul style="list-style-type: none"> <li>Select <a href="#">Initial View Definitions</a> to use the definitions that are set in the <a href="#">Edit Initial View...</a> dialog box.</li> <li>Select the <a href="#">Text</a> option to set all the members to be displayed as text.</li> </ul> <div> <p><b>Note</b></p> <p>This setting overrides the settings in the <a href="#">Edit Initial View...</a> dialog box.</p> </div> <ul style="list-style-type: none"> <li>Select <a href="#">Key</a> to use the <a href="#">Key</a> member display.</li> </ul> <div> <p><b>Note</b></p> <p>This setting overrides the settings in the <a href="#">Edit Initial View...</a> dialog box.</p> </div>
on select	none	<p>The <a href="#">On Select</a> event of the chart is triggered when a value is selected or deselected. This property enables you to assign a custom handler to the <code>OnSelect</code> event. To enable user interaction with the chart, use this property to assign a custom handler to the <code>OnSelect</code> event.</p> <ol style="list-style-type: none"> <li>Choose the <a href="#">Browse</a> button to open the <a href="#">Script Editor</a> dialog box.</li> <li>Press <code>CTRL</code> + <code>SPACEBAR</code> to see the list of available methods for the application, the data source alias, and the chart.</li> <li>Choose one of them or the <a href="#">New Statement Wizard</a> option. The <a href="#">New Statement</a> wizard guides you through the process of creating a statement.</li> </ol>

## Related Information

[Working with Charts \[page 182\]](#)

[Adding a Data Source \[page 68\]](#)

[Chart CSS Additional Properties \[page 351\]](#)

[Chart Area Additional Properties \[page 343\]](#)

[Chart Data Series Additional Properties \[page 350\]](#)

[Configuring Additional Chart Types \[page 211\]](#)

[Conditional Formatting for Charts \[page 212\]](#)

[Installing CVOM Chart Extensions for SAP BusinessObjects Design Studio \[page 303\]](#)

### 36.6.1.1 Chart Area Additional Properties

The *Chart Area* additional properties are described in the table below.

Table 56: Chart Area Additional Properties

Property	Property Value	Property Description
title	<ul style="list-style-type: none"> <li>• show title</li> <li>• title</li> <li>• align               <ul style="list-style-type: none"> <li>◦ left</li> <li>◦ centred</li> <li>◦ right</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Select the check box <i>Show Title</i> to show the chart title in the chart area.</li> <li>• Populate the <i>Title</i> textbox with a customized chart title.</li> <li>• Select from the three options available to align the chart title within the chart area.</li> </ul>
background	<ul style="list-style-type: none"> <li>• color:               <ul style="list-style-type: none"> <li>◦ solid fill</li> <li>◦ no fill</li> </ul> </li> <li>• color picker</li> <li>• reset default</li> </ul>	<ul style="list-style-type: none"> <li>• Select <i>Solid Fill</i> from the dropdown to make the chart background a solid color as defined by the color selection in the color picker.</li> <li>• Select <i>No Fill</i> from the dropdown to make the background transparent.</li> <li>• Select the <i>Color Picker</i> button to change the chart background color.</li> <li>• Choose the <i>Reset Default</i> button to reset the background color to the default theme color.</li> </ul> <div> <p><b>i Note</b></p> <p>If a chart has a custom background color, changing the theme will not update the color.</p> </div>

Property	Property Value	Property Description
axes labels and values	<ul style="list-style-type: none"> <li>• show x-axis</li> <li>• show title</li> <li>• title</li> <li>• show x-axis line</li> <li>• line size</li> <li>• show x-axis labels</li> <li>• format <ul style="list-style-type: none"> <li>◦ Default</li> <li>◦ None</li> <li>◦ <b>#,##0</b></li> <li>◦ <b>#,##0.00</b></li> <li>◦ <b>\$#,##0</b></li> </ul> </li> <li>• show gridline <ul style="list-style-type: none"> <li>◦ type <ul style="list-style-type: none"> <li>◦ line</li> <li>◦ dotted</li> <li>◦ incised</li> </ul> </li> <li>◦ color</li> <li>◦ size</li> </ul> </li> <li>• axis scaling: <ul style="list-style-type: none"> <li>◦ min value</li> <li>◦ max value</li> </ul> </li> <li>• Extend Label Capacity</li> </ul>	<ul style="list-style-type: none"> <li>• Select the <a href="#">Show X-Axis</a> or <a href="#">Show Y-Axis</a> checkbox to display an axis line with labels and title. Unchecking the checkbox removes the axis line, the labels and the title.</li> <li>• Select the <a href="#">Show Title</a> checkbox to display the axis title beside an axis line.</li> <li>• Populate the <a href="#">Title</a> textbox with a customized axis line title.</li> <li>• Select the <a href="#">Show X-Axis Line</a> or <a href="#">Show Y-Axis Line</a> checkbox to display an axis line in the chart. Unchecking the checkbox removes only the axis line. Axis line values and title remain visible.</li> <li>• Populate the <a href="#">Line Size</a> input field to determine the thickness of an axis line in the chart.</li> <li>• Select the <a href="#">Show X-Axis Labels</a> or the <a href="#">Show Y-Axis Labels</a> checkbox to display the measure names on an axis line.</li> <li>• Select one of the following suggested format options from the <a href="#">Format</a> dropdown list to define how the data in the chart is formatted: <div> <div><b>Default</b></div> <div>Applies the same formatting in the chart as applied in the crosstab.</div> </div> <div> <div><b>None</b></div> <div>Applies no formatting</div> </div> <div> <div><b>#,##0</b></div> <div>Formats the number with a thousands separator</div> </div> <div> <div><b>#,##0.00</b></div> <div>Formats the number with a thousands separator, period and two decimal places</div> </div> <div> <div><b>\$#,##0</b></div> <div>Formats the number as a dollar amount</div> </div> </li> </ul> <div> <p><b>i Note</b></p> <p>You can also define your own formatting style by manually editing the values displayed in the <a href="#">Format</a> dropdown list.</p> </div>



Property	Property Value	Property Description
		<ul style="list-style-type: none"> <li>Select the <a href="#">Show Gridline</a> checkbox to display the gridlines on the chart area. <ul style="list-style-type: none"> <li>Choose from the <a href="#">Type</a> drop-down list, the style of gridline required.</li> <li>Use the <a href="#">Color Picker</a> button to select the required grid line color. Alternatively, the HEX color value can be entered manually in the input field beside the <a href="#">Color Picker</a> button.</li> <li>The <a href="#">Size</a> input field determines the thickness of the gridline in the chart.</li> </ul> </li> <li>Select the <a href="#">Axis Scaling</a> checkbox to limit the value axis range displayed in your chart to the range defined in the <a href="#">Min Value</a> and the <a href="#">Max Value</a> input fields. The default value of the <a href="#">Min Value</a> and <a href="#">Max Value</a> is "0". Removing the default value "0" from the <a href="#">Min Value</a> input field and leaving it blank, ensures that your chart will display the minimum value of your data set. If the <a href="#">Axis Scaling</a> checkbox is not checked, the range reverts back to an automatic range for the axis.</li> </ul> <div> <p><b>i Note</b></p> <p>Axis scaling is available for all chart types except the following:</p> <ul style="list-style-type: none"> <li><a href="#">100% Stacked Chart</a></li> <li><a href="#">Pie Chart</a></li> <li>Select the <a href="#">Extend Label Capacity</a> checkbox to extend the maximum space taken by the axis labels from the default 25% to 75% of the entire chart area.</li> </ul> </div>

Property	Property Value	Property Description
legend	<ul style="list-style-type: none"> <li>• show legend</li> <li>• show title</li> <li>• title</li> <li>• hierarchical</li> <li>• position: <ul style="list-style-type: none"> <li>◦ top</li> <li>◦ bottom</li> <li>◦ right</li> <li>◦ left</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Select the <a href="#">Show Legend</a> checkbox to specify if a legend should be displayed in the chart area.</li> <li>• Select the <a href="#">Show Title</a> checkbox to specify that a legend title should be displayed in the chart area.</li> <li>• Populate the <a href="#">Title</a> textbox with a customized legend title.</li> <li>• Select the <a href="#">Hierarchical</a> checkbox to display the legend in a hierarchical way. Selecting this checkbox will display the chart dimensions in the legend in a horizontal list, with each item separated by a back-slash.</li> </ul> <div> <p><b>i Note</b></p> <p>The legend must be located to the right of the chart area for the hierarchical property to function correctly.</p> </div> <ul style="list-style-type: none"> <li>• Select one of the <a href="#">Position</a> buttons to select the default location of the legend in the chart area. <ul style="list-style-type: none"> <li>◦ Select <a href="#">Top</a> to display the legend above the chart.</li> <li>◦ Select <a href="#">Bottom</a> to display the legend below the chart.</li> <li>◦ Select <a href="#">Right</a> to display the legend to the right of the chart.</li> <li>◦ Select <a href="#">Left</a> to display the legend to the left of the chart.</li> </ul> </li> </ul>
data label	<ul style="list-style-type: none"> <li>• show data labels: <ul style="list-style-type: none"> <li>◦ position: <ul style="list-style-type: none"> <li>◦ inside</li> <li>◦ outside</li> </ul> </li> <li>◦ format</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Select the <a href="#">Show Data Labels</a> checkbox to display the measure names and dimension members in the chart area. <ul style="list-style-type: none"> <li>◦ Select <a href="#">Inside</a> from the dropdown list to set your preference to display the numeric values inside the chart bars or delineated areas.</li> </ul> </li> <li>• Select <a href="#">Outside</a> from the dropdown list to set your preference to display the numeric values outside the chart bars or delineated areas.</li> <li>• For a property description of <a href="#">Format</a> see the axes labels and values section above.</li> </ul>

Property	Property Value	Property Description
tooltip	<ul style="list-style-type: none"> <li>• show tooltips</li> <li>• format</li> <li>• always show within the chart area</li> </ul>	<ul style="list-style-type: none"> <li>• Select the <a href="#">Show Tooltips</a> checkbox to display tooltips in the analysis application during runtime.</li> <li>• For a property description of <a href="#">Format</a> see the axes labels and values section above.</li> <li>• Select the <a href="#">Always Show within the Chart Area</a> to set the position of the tooltip within the chart area.</li> </ul>
plot area	<ul style="list-style-type: none"> <li>• animate on data loading</li> <li>• animate on data updating</li> <li>• animate on data resizing</li> </ul>	<p>Select <a href="#">Plot Area</a> properties to allow application users to interact with the elements in the chart.</p> <div> <p><b>i Note</b></p> <p>Plot area properties differ according to the chart type selected.</p> </div>

Property	Property Value	Property Description
interaction	<ul style="list-style-type: none"> <li>• selection mode: <ul style="list-style-type: none"> <li>◦ inclusive</li> <li>◦ exclusive</li> <li>◦ single</li> <li>◦ multiple</li> <li>◦ none</li> </ul> </li> </ul>	<p>Define the selection mode the application user will apply at runtime to interact with a chart. Depending on the selection mode, the application user can make their member selections using the legend, the axes or the individual chart elements.</p> <div> <p><b>i Note</b></p> <p>It is recommended that the scripting method <code>getSelectedMembers()</code>; is used for all selection modes except for <i>single</i>.</p> </div> <p>Select one of the following selection modes:</p> <ul style="list-style-type: none"> <li>• Select <i>inclusive</i> to allow multiple members to be selected at the same time. The application user can set their member selections by using the click or lasso function in the legend, the axes or the individual chart elements. This persists the selection of previously selected members, whether the member is already selected or not. The default selection mode value is <i>inclusive</i>.</li> <li>• Select <i>exclusive</i> to limit the selection option to one member only. Using the lasso function it is also possible to select multiple members in the legend, the axes or the individual chart elements.</li> </ul> <div> <p><b>i Note</b></p> <p>It is recommended that the selection mode <i>exclusive</i> is used on mobile devices, as it guarantees that the member selected by the user really is the one that is selected. It also allows for legend selection if necessary.</p> </div> <ul style="list-style-type: none"> <li>• Select <i>single</i> to allow the user make one member selection only by selecting the individual chart elements. This selection mode does not allow legend or axis selection.</li> </ul>

Property	Property Value	Property Description
		<p><b>i Note</b></p> <p>It is recommended that the selection mode <i>single</i> is used with the scripting method <i>getSelected Member()</i>; only.</p> <ul style="list-style-type: none"> <li>• Select <i>multiple</i> to allow the user to add to their selection by selecting unselected items and to remove from their selection by clicking on selected items. This selection mode is a combination of the <i>inclusive</i> and the <i>exclusive</i> selection mode functionality.</li> <li>• Selecting <i>None</i> removes the option to allow the user to interact with the chart and select sets of data points.</li> </ul> <p><b>i Note</b></p> <p>A tooltip displays within the chart at runtime to show in detail what values are selected. The tooltip on a mobile device only displays how many items are selected, if more than one is selected.</p> <p><b>i Note</b></p> <p>To clear member selections, you can use the <i>CHART.clearSelection()</i>; scripting method. Alternatively, the user can click outside the data point area.</p>

### **i Note**

For more information on additional properties, see the *Charts: Additional Information* under ► [Help](#) ► [Help Contents](#) ► in the design tool.

## Related Information

[Chart \[page 338\]](#)

[Chart Data Series Additional Properties \[page 350\]](#)

[Chart CSS Additional Properties \[page 351\]](#)

[Clear Selection \(clearSelection\) \[page 474\]](#)

[Get Selected Members \(getSelectedMembers\) \[page 475\]](#)

[Get Selected Member \(getSelectedMember\) \[page 474\]](#)


## 36.6.1.2 Chart Data Series Additional Properties

The *Data series* additional properties are described in the table below.

Table 57: Data Series Additional Properties

Property	Property Value	Property Description
displayed series format	<ul style="list-style-type: none"><li>axis 1</li><li>axis 2</li><li>color picker</li><li>restore defaults</li></ul> <ol style="list-style-type: none"><li><b>Note</b> The property values Axis 1 and Axis 2 apply to all dual axis charts only. For dual axis charts the color picker applies to the axes only. The measures automatically get assigned a color based on the color range of the axis color</li><li><b>Note</b> For all other chart types, the color picker applies to the whole data series.</li></ol>	<ul style="list-style-type: none"><li>By default all the existing data series that have been added in the <i>Edit Initial View</i> window, are displayed in the <i>Displayed Series Format</i> panel.</li><li>Select the <i>button switch</i> to open the <i>color picker</i>, where the data series color changes can be specified. Alternatively, the HEX color value can be entered manually in the textbox beside the <i>color picker</i>.</li><li>Choose the <i>Restore Defaults</i> button to restore the chart theme settings in the <i>Displayed Series Format</i> panel.</li></ul>
displayed measures	<b>Note</b> This property and property value options appear only for the combination dual axis charts. <ul style="list-style-type: none"><li>bar</li><li>line</li><li>axis 1</li><li>axis 2</li><li>restore defaults</li></ul>	<ul style="list-style-type: none"><li>Use the drop-down box to select bar or line to display the measure as a bar or line (depending on the chart type selected) in the chart.</li><li>By default all measures that have been added in the <i>Edit Initial View</i> window are displayed.</li><li>Choose the <i>Restore Defaults</i> button to restore the default settings.</li></ul>
data series sequence	<b>Note</b> This property and property value options appear for waterfall charts only. <ul style="list-style-type: none"><li>cumulative</li><li>total</li><li>restore default</li></ul>	<ul style="list-style-type: none"><li>Choose <i>Cumulative</i> to display a cumulative total for the dimensions in your <i>Chart</i>.</li><li>Choose <i>Total</i> to display the total for the dimensions in your <i>Chart</i>.</li><li>Choose the <i>Restore Defaults</i> button to restore the settings in the <i>Data Series Sequence</i> panel.</li></ul>

### Note

For more information on additional chart properties, see the *Charts: Additional Information* under [Help -> Help Contents in the design tool.](#) 

## Related Information

[Chart CSS Additional Properties \[page 351\]](#)

[Chart Area Additional Properties \[page 343\]](#)

[Chart \[page 338\]](#)

## 36.6.1.3 Chart CSS Additional Properties

The [Chart CSS](#) tab contains the list of CSS tags that allow you to manipulate specific formats within the chart component. The table called *Chart CSS Property Groups* lists the available CSS property groups. The table called *Chart CSS Property Example* provides an example of a CSS property group, the properties within that group and the associated default values.

Table 58: Chart CSS Property Groups

Property Group	Property Description
.v-m-title .v-title	Defines the style for the title label
.v-m-xAxis .v-title	Defines the style for the x-axis title
.v-m-xAxis .v-label	Define the style of the y-axis label
.v-m-yAxis .v-title	Define the style of the x-axis title
.v-m-yAxis .v-label	Define the style of they y- axis title
.v-m-legend .v-title	Define style for the legend title
.v-m-legend .v-label	Define style for the legend label
.v-m-legend .v-scrollbarThumb	Define style for legend scrollbar thumb
.v-m-datalabel .v-datalabel	Define style for the data label

### Example

Table 59: Chart CSS Property Example

Property Group	Property	Default Value
.viz-title-label.v-title	font-family	<ul style="list-style-type: none"> <li>• Open Sans</li> <li>• Arial</li> <li>• Helvetica</li> <li>• sans-serif</li> </ul> <div> <i>i</i> <b>Note</b>  The default font value is determined by which of the above four options are available to the user. </div>
	font-size	16px
	font-weight	bold
	fill	#333333

### *i* Note

For more information on all the CSS chart properties available, see the *Charts: Additional Information* documentation under [Help > Help Contents](#) in the design tool.

## 36.6.2 Info Chart

Working with the *Info Chart* component to present your data graphically offers you improved functionality and ease of use when compared with the *Chart* component. The *Info Chart* component is supported when working in both *Commons* and *Main* (SAPUI5 Main) mode. For more information on SAPUI5 Main mode, you can refer to the chapter in this guide called [Using the SAPUI5 m Library \[page 45\]](#).

Use the info chart properties in the *Properties* and *Additional Properties* views to configure the appearance of *Info Chart* component in your application. When you drag and drop the *Info Chart* component into the layout editor, the component displays a graphic image of a generic chart. When you assign a data source to the *Info Chart* component, it then displays the data using the *Column* chart type, until you modify the chart type. The info chart type can be changed after assigning a data source, by using the *Chart Configuration* property.

The *Info Chart* component differs from the *Chart* component in the following ways:

- There is no requirement to apply conditional formatting directly in the *Info Chart*. The *Info Chart* automatically displays the same conditional formatting as it appears in your crosstab.
- The *Info Chart* offers a context menu on the category axis.
- A wide range of additional properties are divided into multiple tabs for ease of navigation.
- Dimensions can now be split into more than two axes. With the *Chart* component you could assign only crosstab rows and columns. With the *Info Chart* component, each dimension can be assigned a different axis independently.
- Changes to the query structure that do not add or remove measures or dimensions will have no impact on the *Info Chart*, so the swap axis functionality has no effect on the *Info Chart*.
- How you configure the *Info Chart* with measures and dimensions is defined using the *Chart Configuration* property at design time.



- When you select to use an *Info Chart* in conjunction with the *Info Chart Feeding Panel* and the *Chart Type Picker*, you can create an application that offers the application user increased flexibility to configure the chart themselves when they run their application. They can decide what chart type to select and where to put their measures and dimensions in the *Info Chart*.

#### **i** Note

- You cannot create new *Info Chart* extensions using the SAP Lumira SDK.
- When using the context menu, you must take note of the following:
  - If the data labels in the category axis are truncated, the filters in the context menu will not display correctly.
  - If you have two dimensions in the category axis and the data label is not fully extended, the filters in the context menu may not display correctly.
  - The changes made through the context menu are reflected in the data source and all other components connected to the same data source, not directly in the *Info Chart*.

The properties of the *Info Chart* component include the following views:

- Properties
- Additional Properties:
  - General
  - Plot Area
  - Axes
  - Legend
  - Data Series
  - Interaction

The chart properties in the *Properties* view are described in the table below. The chart properties in the *Additional Properties* view are described in their own individual chapters.

## Properties

Table 60:

Property	Property Value	Property Description
data source	Name of data source alias, such as <i>DS_1</i> .	Displays all data source aliases. If you have created several data sources for the application, you can change the data source for the chart by choosing the corresponding data source alias.  <b>i</b> Note As you can use the same data source several times within one application, you work in the layout editor using data source aliases as reference names.

Property	Property Value	Property Description
data selection	Selection string expressed in a JSON notation generated from the data selection.	<p>Dialog box allows you to select multiple rows or columns from the data result set to create a separate chart. Click the <a href="#">Add Selection</a> button to select the columns or rows that you want to appear in a separate chart.</p> <div> <p><b>i Note</b></p> <p>If your first data selection is in a row, your subsequent data selections must only be in rows. Similarly, if your first data selection is in a column, your subsequent data selections must only be in columns.</p> </div>
chart configuration	dialog	<p>Dialog box enables you to easily manipulate how your data is visualized, before you create your application. It comprises a <a href="#">Chart Type Picker</a>, <a href="#">Info Chart Feeding Panel</a> and a preview area of the info chart selected.</p>
conditional formatting visible	true, false	<p>Display the same conditional formatting in your chart as appears in your crosstab, set to true. It is set to false by default.</p> <div> <p><b>i Note</b></p> <p>Any conditional formatting configuration that needs to be done, must be done on the query. This is described in the chapter in this guide called <i>Crosstab</i>, within the <i>User Interface Reference</i> section.</p> </div>
show totals	true, false	<p>When you select true, the totals for each dimension that you have added in the <a href="#">Edit Initial View...</a> dialog box are displayed in the chart. Select false to hide these totals. The default setting is <i>false</i>.</p>

Property	Property Value	Property Description
on select	none	<p>The <i>On Select</i> event of the chart is triggered when a value is selected or deselected. This property enables you to assign a custom handler to the <code>OnSelect</code> event. To enable user interaction with the chart, use this property to assign a custom handler to the <code>OnSelect</code> event.</p> <ol style="list-style-type: none"> <li>1. Choose the <i>Browse</i> button to open the <i>Script Editor</i> dialog box.</li> <li>2. Press <code>CTRL</code> + <code>SPACEBAR</code> to see the list of available methods for the application, the data source alias, and the chart.</li> <li>3. Choose one of them or the <i>New Statement Wizard</i> option. The <i>New Statement</i> wizard guides you through the process of creating a statement.</li> </ol>

## Related Information

[Crosstab \[page 372\]](#)

[Using the SAPUI5 m Library \[page 45\]](#)

### 36.6.2.1 Info Chart General Additional Properties

The *General* additional properties are described in the table below.

Table 61: Info Chart General Additional Properties

Property	Property Value	Property Description
title	<ul style="list-style-type: none"> <li>• show title</li> <li>• text</li> <li>• alignment <ul style="list-style-type: none"> <li>◦ left</li> <li>◦ centred</li> <li>◦ right</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Select the check box <i>Show Title</i> to show the info chart title in the info chart area.</li> <li>• Populate the <i>Text</i> textbox with a customized info chart title.</li> <li>• Select from the three options available to align the info chart title within the info chart area.</li> </ul>

Property	Property Value	Property Description
font	<ul style="list-style-type: none"> <li>weight</li> <li>style</li> <li>size</li> <li>family</li> <li>color</li> </ul>	<ul style="list-style-type: none"> <li>Default value is bold.</li> <li>Default value is normal.</li> <li>Default value is 16px.</li> <li>Default value is 'Open Sans, Arial, Helvetica'.</li> <li>Select a font color from the <a href="#">Color Picker</a>.</li> </ul>
background	<ul style="list-style-type: none"> <li>show background</li> <li>color</li> <li>border width</li> <li>border color</li> <li>top border</li> <li>right border</li> <li>left border</li> <li>bottom border</li> </ul>	<ul style="list-style-type: none"> <li>Select the checkbox <a href="#">Show Background</a> to make the info chart background visible.</li> <li>Select a color from the <a href="#">Color Picker</a> to make the info chart background a solid color as defined by the color selection in the color picker.</li> <li>Select <a href="#">Border Width</a> to determine the width of the info chart border.</li> <li>Select <a href="#">Border Color</a> to determine the color of the info chart border.</li> <li>Select one of the border positions to determine which borders of the info chart area are highlighted.</li> </ul>
layout	padding top, padding right, padding left, padding bottom	Select a padding value to determine the amount of space between the outside of the info chart and the info chart area.

## 36.6.2.2 Info Chart Plot Area Additional Properties

The [Plot Area](#) additional properties are described in the table below.

Table 62: Info Chart Plot Area Additional Properties

Property	Property Value	Property Description
general	<ul style="list-style-type: none"> <li>fixed data point size</li> <li>show line (line and radar)</li> <li>smooth line (line)</li> <li>show negative values (bubble)</li> <li>marker size (scatter and line)</li> <li>inner radius ratio (doughnut)</li> <li>width (line and radar)</li> </ul>	<ul style="list-style-type: none"> <li>Select the <i>Fixed Data Point Size</i> checkbox to fix the size of the columns and then display a scrollbar if required.</li> <li>Select the <i>Show Line</i> checkbox to display lines in your line and radar info charts.</li> <li>Select <i>Smooth Line</i> to smooth into a curve, the lines in your line and info charts.</li> <li>Select <i>Show Negative Values</i> to display negative values in your bubble info chart.</li> <li>The <i>Marker Size</i> property determines the size of the marker used in your scatter or line info chart.</li> <li>The <i>Inner Radius Ratio</i> property determines the ratio of the size of the inner radius of a doughnut info chart versus the outer radius.</li> <li>The <i>Width</i> property allows you to determine the width of the lines displayed in your line and radar info chart.</li> </ul>
data point	opacity (bubble and scatter)	<ul style="list-style-type: none"> <li><i>Opacity</i> indicates the transparency of the bubbles, ranging from 0 to 1.</li> </ul>

Property	Property Value	Property Description
data label	<ul style="list-style-type: none"> <li>• show total (stacked bar, stacked column)</li> <li>• show data labels</li> <li>• type: (pie, doughnut) <ul style="list-style-type: none"> <li>◦ color</li> <li>◦ value</li> <li>◦ percentage</li> <li>◦ color and percentage</li> </ul> </li> <li>• hide when overlap</li> <li>• position <ul style="list-style-type: none"> <li>◦ inside</li> <li>◦ outside</li> <li>◦ outside first</li> </ul> </li> <li>• format <ul style="list-style-type: none"> <li>◦ default</li> <li>◦ none</li> <li>◦ #,##0</li> <li>◦ #,##0.00</li> <li>◦ \$#,##0</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Select <a href="#">Show Total</a> to display totals in your stacked bar and column info charts.</li> <li>• Select the <a href="#">Show Data Labels</a> checkbox to display the measure names and dimension members in the chart area.</li> <li>• Choose from the <a href="#">Type</a> dropdown list, the information to be displayed on the data label.</li> <li>• Select the <a href="#">Hide When Overlap</a> checkbox to select not to display the label when it is too long for the space available.</li> <li>• Select an option from the <a href="#">Position</a> dropdown to decide the position of your label: <ul style="list-style-type: none"> <li>◦ Select <a href="#">Inside</a> from the dropdown list to set your preference to display the numeric values inside the chart bars or delineated areas. If there is not enough space it will display the label outside.</li> <li>◦ Select <a href="#">Outside</a> from the dropdown list to set your preference to display the numeric values outside the chart bars or delineated areas.</li> <li>◦ Select <a href="#">Outside First</a> from the dropdown list to display the label outside. If there is not enough space it will display the label inside.</li> </ul> </li> </ul>

Property	Property Value	Property Description
		<ul style="list-style-type: none"> <li>Select one of the following suggested format options from the <a href="#">Format</a> dropdown list to define how the data in the chart is formatted:</li> </ul> <p><b>Default</b> Applies the same formatting in the chart as applied in the crosstab.</p> <p><b>None</b> Applies no formatting</p> <p><b>#,##0</b> Formats the number with a thousands separator</p> <p><b>#,##0.00</b> Formats the number with a thousands separator, period and two decimal places</p> <p><b>\$#,##0</b> Formats the number as a dollar amount</p> <div> <p><b>i Note</b></p> <p>You can also define your own formatting style by manually editing the values displayed in the <a href="#">Format</a> dropdown list.</p> </div>

Property	Property Value	Property Description
font	<ul style="list-style-type: none"> <li>• weight</li> <li>• style</li> <li>• size</li> <li>• family</li> <li>• color</li> </ul>	<ul style="list-style-type: none"> <li>• Default value is bold.</li> <li>• Default value is normal.</li> <li>• Default value is 16px.</li> <li>• Default value is 'Open Sans, Arial, Helvetica'.</li> <li>• Select a font color from the <a href="#">Color Picker</a>.</li> </ul>
gridline	<ul style="list-style-type: none"> <li>• show gridline</li> <li>• type: <ul style="list-style-type: none"> <li>◦ line</li> <li>◦ dotted</li> <li>◦ incised</li> </ul> </li> <li>• size</li> <li>• color</li> </ul>	<ul style="list-style-type: none"> <li>• Select the <a href="#">Show Gridline</a> checkbox to display the gridlines on the info chart area.</li> <li>• Select from the <a href="#">Type</a> dropdown list, the style of gridline required.</li> <li>• The <a href="#">Size</a> input field determines the thickness of the gridline in the info chart.</li> <li>• Use the <a href="#">Color Picker</a> button to select the required grid line color. Alternatively, the HEX color value can be entered manually in the input field beside the <a href="#">Color Picker</a> button.</li> </ul>
trellis grid	<ul style="list-style-type: none"> <li>• show trellis grid</li> <li>• color</li> <li>• size</li> </ul>	<ul style="list-style-type: none"> <li>• Select the <a href="#">Show Trellis Grid</a> checkbox to display the grid within trellis info charts.</li> <li>• The <a href="#">Color</a> property allows you to select the color of your trellis grid.</li> <li>• Select the <a href="#">Size</a> property to select the size of your trellis grid.</li> </ul>



Property	Property Value	Property Description
zero line	<ul style="list-style-type: none"> <li>highlight zero line</li> <li>color</li> </ul>	<ul style="list-style-type: none"> <li>Select the <a href="#">Highlight Zero Line</a> when you wish to highlight a line on your chart that displays, for example, data between -10 and 10</li> <li>Use the <a href="#">Color Picker</a> button to select the required grid line color. Alternatively, the HEX color value can be entered manually in the input field beside the <a href="#">Color Picker</a> button.</li> </ul>
marker	<ul style="list-style-type: none"> <li>show marker</li> <li>size</li> <li>shape <ul style="list-style-type: none"> <li>circle</li> <li>diamond</li> <li>triangle up</li> <li>triangle down</li> <li>triangle left</li> <li>triangle right</li> <li>cross</li> <li>intersection</li> </ul> </li> <li>hover only mode (time series line)</li> </ul>	<ul style="list-style-type: none"> <li>Select <a href="#">Show Marker</a> to display a marker in your line and radar info chart.</li> <li>Select <a href="#">Size</a> to determine the size of your marker.</li> <li>Use the <a href="#">Shape</a> property to determine the shape of your marker.</li> <li>Select the <a href="#">Hover Only Mode</a> when using the time series line info chart.</li> </ul>

Property	Property Value	Property Description
primary scale	<ul style="list-style-type: none"> <li>• fixed range</li> <li>• min value</li> <li>• max value</li> </ul>	<ul style="list-style-type: none"> <li>• Select the <i>Fixed Range</i> checkbox to set whether the axis range is limited to the range between the Min Value and Max Value. The default value of the <i>Min Value</i> and <i>Max Value</i> is "0". Removing the default value "0" from the <i>Min Value</i> input field and leaving it blank, ensures that your chart will display the minimum value of your data set. If the <i>Fixed Range</i> checkbox is not checked, the range reverts back to an automatic range for the axis.</li> </ul>
secondary scale	<ul style="list-style-type: none"> <li>• fixed range</li> <li>• min value</li> <li>• max value</li> </ul>	<ul style="list-style-type: none"> <li>• The <i>Secondary Scale</i> represents the scale for the second axis in a dual axis info chart. Similar to the <i>Primary Scale</i> property, you must select the <i>Fixed Range</i> checkbox to set whether the axis range is limited to the range between the Min Value and Max Value. The default value of the <i>Min Value</i> and <i>Max Value</i> is "0". Removing the default value "0" from the <i>Min Value</i> input field and leaving it blank, ensures that your chart will display the minimum value of your data set. If the <i>Fixed Range</i> checkbox is not checked, the range reverts back to an automatic range for the axis.</li> </ul>

Property	Property Value	Property Description
area (radar)	<ul style="list-style-type: none"> <li>• show area</li> <li>• opacity</li> <li>• color</li> </ul>	<ul style="list-style-type: none"> <li>• The <a href="#">Show Area</a>, <a href="#">Opacity</a> and <a href="#">Color</a> properties allow you to display, fill and define the transparency of the shapes in the radar info chart.</li> </ul>
polar axis (radar)	<ul style="list-style-type: none"> <li>• show category axis</li> </ul>	<ul style="list-style-type: none"> <li>• Select <a href="#">Show Category Axis</a> to display the category axis.</li> </ul>
value axis (radar)	<ul style="list-style-type: none"> <li>• show value axis</li> <li>• show labels</li> <li>• label format</li> <li>• show value axis title</li> <li>• font               <ul style="list-style-type: none"> <li>◦ weight</li> <li>◦ style</li> <li>◦ size</li> <li>◦ family</li> <li>◦ color</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Select <a href="#">Show Value Axis</a> to display the value axis in your radar info chart.</li> <li>• Select <a href="#">Show Labels</a> to display labels on your radar info chart.</li> <li>• The <a href="#">Label Format</a> property allows you to select a format for your radar info chart label.</li> <li>• To display a value axis title, select the <a href="#">Show Value Axis Title</a>.</li> <li>• Select from the <a href="#">Font</a> property to determine how the value axis title will be displayed.</li> </ul>

### 36.6.2.3 Info Chart Axes Additional Properties

The [Axes](#) additional properties are described in the table below.

Table 63: Info Chart Axes Additional Properties

Property	Property Value	Property Description
value axis	<ul style="list-style-type: none"> <li>• show value axis</li> <li>• color</li> </ul>	<ul style="list-style-type: none"> <li>• Select the <a href="#">Show Value Axis</a> checkbox to display a value axis line in the info chart.</li> <li>• Use the <a href="#">Color Picker</a> button to select the required value axis line color. Alternatively, the HEX color value can be entered manually in the input field beside the <a href="#">Color Picker</a> button.</li> </ul>

Property	Property Value	Property Description
(value axis) axis line	<ul style="list-style-type: none"> <li>show axis line</li> <li>size</li> </ul>	<ul style="list-style-type: none"> <li>Select the <a href="#">Show Axis Line</a> checkbox to display a value axis line in the chart. Unchecking the checkbox removes only the value axis line. Axis line values and title remain visible.</li> <li>The <a href="#">Size</a> input field determines the thickness of the value axis line in the info chart.</li> </ul>
(value axis) axis tick	show axis tick	Select the <a href="#">Show Axis Tick</a> to display a tick along the value axis.
(value axis) label	<ul style="list-style-type: none"> <li>show labels</li> <li>format: <ul style="list-style-type: none"> <li>default</li> <li>none</li> <li><b>#,##0</b></li> <li><b>#,##0.00</b></li> <li><b>\$#,##0</b></li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Select the <a href="#">Show Labels</a> checkbox to display labels on the value axis.</li> <li>Select one of the following suggested format options from the <a href="#">Format</a> dropdown list to define how the data in the value axis labels is formatted: <div> <p><b>Default</b> Determined by your locale</p> <p><b>None</b> Applies no formatting</p> <p><b>#,##0</b> Formats the number with a thousands separator</p> <p><b>#,##0.00</b> Formats the number with a thousands separator, period and two decimal places</p> <p><b>\$#,##0</b> Formats the number as a dollar amount</p> </div> </li> </ul> <div> <p><b>i Note</b></p> <p>You can also define your own formatting style by manually editing the values displayed in the <a href="#">Format</a> dropdown list.</p> </div>

Property	Property Value	Property Description
(value axis label) font	<ul style="list-style-type: none"> <li>weight</li> <li>style</li> <li>size</li> <li>family</li> <li>color</li> </ul>	<ul style="list-style-type: none"> <li>Default value is bold.</li> <li>Default value is normal.</li> <li>Default value is 16px.</li> <li>Default value is 'Open Sans, Arial, Helvetica'.</li> <li>Select a font color from the <a href="#">Color Picker</a>.</li> </ul>
(value axis) title	<ul style="list-style-type: none"> <li>show title</li> <li>text</li> <li>apply axis line color</li> </ul>	<ul style="list-style-type: none"> <li>Select the <a href="#">Show Title</a> checkbox to display the value axis title beside a value axis line.</li> <li>Populate the <a href="#">Text</a> textbox with a customized value axis line title.</li> <li>Select the <a href="#">Apply Axis Line Color</a> to apply the same color to the title as you have set for the value axis.</li> </ul>
(value axis title) font	<ul style="list-style-type: none"> <li>weight</li> <li>style</li> <li>size</li> <li>family</li> <li>color</li> </ul>	<ul style="list-style-type: none"> <li>Default value is bold.</li> <li>Default value is normal.</li> <li>Default value is 16px.</li> <li>Default value is 'Open Sans, Arial, Helvetica'.</li> <li>Select a font color from the <a href="#">Color Picker</a>.</li> </ul>
category axis	<ul style="list-style-type: none"> <li>show category axis</li> <li>color</li> </ul>	<ul style="list-style-type: none"> <li>Select the <a href="#">Show Category Axis</a> checkbox to display a category axis line in the chart.</li> <li>Use the <a href="#">Color Picker</a> button to select the required category axis color. Alternatively, the HEX color value can be entered manually in the input field beside the <a href="#">Color Picker</a> button.</li> </ul>
(category axis) axis line	<ul style="list-style-type: none"> <li>show axis line</li> <li>size</li> </ul>	<ul style="list-style-type: none"> <li>Select the <a href="#">Show Axis Line</a> checkbox to display a category axis line in the chart. Unchecking the checkbox removes only the category axis line. Axis line values and title remain visible.</li> <li>The <a href="#">Size</a> input field determines the thickness of the category axis line in the info chart.</li> </ul>
(category axis) axis tick	show axis tick	Select the <a href="#">Show Axis Tick</a> to display a tick along the category axis.

Property	Property Value	Property Description
(category axis) label	<ul style="list-style-type: none"> <li>• show labels</li> <li>• hide sub levels</li> <li>• rotation: <ul style="list-style-type: none"> <li>◦ auto</li> <li>◦ fixed</li> </ul> </li> <li>• angle: <ul style="list-style-type: none"> <li>◦ 0</li> <li>◦ 30</li> <li>◦ 45</li> <li>◦ 60</li> <li>◦ 90</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Select the <a href="#">Show Labels</a> checkbox to display labels on the category axis.</li> <li>• Select the <a href="#">Hide Sub Levels</a> checkbox to hide all dimensions apart from the first one in the category axis.</li> <li>• Select the <a href="#">Rotation</a> value to switch between manual and automatic angle calculation for the category axis labels.</li> <li>• Select the <a href="#">Angle</a> value to change the display angle of the labels in the category axis.</li> </ul>
(category axis label) font	<ul style="list-style-type: none"> <li>• weight</li> <li>• style</li> <li>• size</li> <li>• family</li> <li>• color</li> </ul>	<ul style="list-style-type: none"> <li>• Default value is bold.</li> <li>• Default value is normal.</li> <li>• Default value is 16px.</li> <li>• Default value is 'Open Sans, Arial, Helvetica'.</li> <li>• Select a font color from the <a href="#">Color Picker</a>.</li> </ul>
(category axis) title	<ul style="list-style-type: none"> <li>• show title</li> <li>• text</li> <li>• apply axis line color</li> </ul>	<ul style="list-style-type: none"> <li>• Select the <a href="#">Show Title</a> checkbox to display the value axis title beside a value axis line.</li> <li>• Populate the <a href="#">Text</a> textbox with a customized value axis line title.</li> <li>• Select the <a href="#">Apply Axis Line Color</a> to apply the same color to the title as you have set for the category axis.</li> </ul>
(category axis title) font	<ul style="list-style-type: none"> <li>• weight</li> <li>• style</li> <li>• size</li> <li>• family</li> <li>• color</li> </ul>	<ul style="list-style-type: none"> <li>• Default value is bold.</li> <li>• Default value is normal.</li> <li>• Default value is 16px.</li> <li>• Default value is 'Open Sans, Arial, Helvetica'.</li> <li>• Select a font color from the <a href="#">Color Picker</a>.</li> </ul>

## 36.6.2.4 Info Chart Legend Additional Properties

The [Legend](#) additional properties are described in the table below.

Table 64: Info Chart Legend Additional Properties

Property	Property Value	Property Description
legend	<ul style="list-style-type: none"> <li>• show legend</li> <li>• show full label</li> <li>• scrollable</li> <li>• always show</li> </ul>	<ul style="list-style-type: none"> <li>• Select the <a href="#">Show Legend</a> checkbox to specify if a legend should be displayed in the chart area.</li> <li>• Select the <a href="#">Show Full Label</a> checkbox to specify that the complete legend labels should be displayed in the chart area.</li> <li>• Select the <a href="#">Scrollable</a> checkbox to allow the legend to scroll when the legend list increases in size.</li> <li>• Select the <a href="#">Always Show</a> checkbox to ensure that the legend remains visible in all cases.</li> </ul>
layout	<ul style="list-style-type: none"> <li>• position <ul style="list-style-type: none"> <li>◦ top</li> <li>◦ bottom</li> <li>◦ right</li> <li>◦ left</li> </ul> </li> <li>• alignment</li> </ul>	<ul style="list-style-type: none"> <li>• Select one of the <a href="#">Position</a> buttons to select the default location of the legend in the chart area. <ul style="list-style-type: none"> <li>◦ Select <a href="#">Top</a> to display the legend above the chart.</li> <li>◦ Select <a href="#">Bottom</a> to display the legend below the chart.</li> <li>◦ Select <a href="#">Right</a> to display the legend to the right of the chart.</li> <li>◦ Select <a href="#">Left</a> to display the legend to the left of the chart.</li> </ul> </li> <li>• Select one of the alignment options to align the legend.</li> </ul>
marker	<ul style="list-style-type: none"> <li>• square with radius</li> <li>• square</li> <li>• rectangle</li> </ul>	<ul style="list-style-type: none"> <li>• Select <a href="#">Square With Radius</a> to display a square-shaped marker with a radius.</li> <li>• Select <a href="#">Square</a> to display a square-shaped marker.</li> <li>• Select <a href="#">Rectangle</a> to display a rectangle-shaped marker.</li> </ul>

Property	Property Value	Property Description
labels	<ul style="list-style-type: none"> <li>font: <ul style="list-style-type: none"> <li>weight</li> <li>style</li> <li>size</li> <li>family</li> <li>color</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Select <i>Solid Fill</i> from the dropdown to make the chart background a solid color as defined by the color selection in the color picker.</li> <li>Select <i>No Fill</i> from the dropdown to make the background transparent.</li> <li>Select the <i>Color Picker</i> button to change the chart background color.</li> <li>Choose the <i>Reset Default</i> button to reset the background color to the default theme color.</li> </ul> <div> <b>Note</b>  If a chart has a custom background color, changing the theme will not update the color. </div>
title	<ul style="list-style-type: none"> <li>show title</li> <li>text</li> </ul>	<ul style="list-style-type: none"> <li>Select the <i>Show Title</i> checkbox to display the legend title.</li> <li>Populate the <i>Text</i> textbox with a customized legend title.</li> </ul>
(title) font	<ul style="list-style-type: none"> <li>weight</li> <li>style</li> <li>size</li> <li>family</li> <li>color</li> </ul>	<ul style="list-style-type: none"> <li>Default value is bold.</li> <li>Default value is normal.</li> <li>Default value is 16px.</li> <li>Default value is 'Open Sans, Arial, Helvetica'.</li> <li>Select a font color from the <i>Color Picker</i>.</li> </ul>

## 36.6.2.5 Info Chart Data Series Additional Properties

The *Data Series* additional properties are described in the table below.

Table 65: Info Chart Data Series Additional Properties

Property	Property Value	Property Description
displayed measures  <div> <b>Note</b>  This property and property value options appear only for the combination dual axis info charts. </div>	<ul style="list-style-type: none"> <li>bar</li> <li>line</li> </ul>	<ul style="list-style-type: none"> <li>Use the drop-down box to select bar or line to display the measure as a bar or line (depending on the chart type selected) in the chart.</li> <li>By default all measures that have been added in the <i>Edit Initial View</i> window are displayed.</li> </ul>



Property	Property Value	Property Description
color palette	15 different options	Select the color palette you wish to use in your info chart.
additional color (bullet)	<ul style="list-style-type: none"> <li>• default value</li> <li>• color picker</li> </ul>	<ul style="list-style-type: none"> <li>• The <i>Additional Color</i> property for bullet info charts, allows you to set an additional color for your info chart.</li> </ul>
forecast color (bullet)	<ul style="list-style-type: none"> <li>• default value</li> <li>• color picker</li> </ul>	The <i>Forecast Color</i> property for bullet info charts, allows you to set a forecast color for your info chart.
target (bullet)	<ul style="list-style-type: none"> <li>• value color</li> </ul>	<ul style="list-style-type: none"> <li>• Select <i>Value Color</i> to set the color of the target tick.</li> </ul>

## 36.6.2.6 Info Chart Interaction Additional Properties

The *Interaction* additional properties are described in the table below.

Table 66: Info Chart Interaction Additional Properties

Property	Property Value	Property Description
interaction	<ul style="list-style-type: none"> <li>• non-interaction mode</li> </ul>	<ul style="list-style-type: none"> <li>• Select the <i>Non-interactive Mode</i> checkbox to switch off interaction mode.</li> </ul>

Property	Property Value	Property Description
selection	<ul style="list-style-type: none"> <li>• axis labels</li> <li>• plot lasso</li> <li>• legend</li> <li>• mode <ul style="list-style-type: none"> <li>◦ inclusive</li> <li>◦ exclusive</li> <li>◦ single</li> <li>◦ multiple</li> <li>◦ none</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Allows the user to select where on the info chart they can interact with it</li> <li>• Select <i>Axis Labels</i> checkbox to interact with the info chart in the axis labels.</li> <li>• Select <i>Plot Lasso</i> checkbox to select to interact with the info chart with a lasso.</li> <li>• Select <i>Legend</i> checkbox to interact with the info chart in the legend area.</li> <li>• Select <i>Mode</i> option to define the mode the application user will apply at runtime to interact with an info chart. Depending on the mode selected, the application user can make their member selections using the legend, the axes or the individual chart elements.</li> </ul> <div> <p><b>i Note</b></p> <p>It is recommended that the scripting method <i>getSelectedMembers()</i> is used for all selection modes except for <i>single</i>.</p> </div> <p>Select one of the following selection modes:</p> <ul style="list-style-type: none"> <li>◦ Select <i>inclusive</i> to allow multiple members to be selected at the same time. The application user can set their member selections by using the click or lasso function in the legend, the axes or the individual chart elements. This persists the selection of previously selected members, whether the member is already selected or not. The default selection mode value is <i>inclusive</i>.</li> <li>◦ Select <i>exclusive</i> to limit the selection option to one member only. Using the lasso function it is also possible to select multiple members in the leg-</li> </ul>

Property	Property Value	Property Description
		<p>end, the axes or the individual chart elements.</p> <div> <p><b>i Note</b></p> <p>It is recommended that the selection mode <i>exclusive</i> is used on mobile devices, as it guarantees that the member selected by the user really is the one that is selected. It also allows for legend selection if necessary.</p> </div> <ul style="list-style-type: none"> <li>○ Select <i>single</i> to allow the user make one member selection only by selecting the individual chart elements. This selection mode does not allow legend or axis selection.</li> </ul> <div> <p><b>i Note</b></p> <p>It is recommended that the selection mode <i>single</i> is used with the scripting method <i>getSelectedMember()</i>; only.</p> </div> <ul style="list-style-type: none"> <li>○ Select <i>multiple</i> to allow the user to add to their selection by selecting unselected items and to remove from their selection by clicking on selected items. This selection mode is a combination of the <i>inclusive</i> and the <i>exclusive</i> selection mode functionality.</li> <li>○ Selecting <i>None</i> removes the option to allow the user to interact with the chart and select sets of data points.</li> </ul> <div> <p><b>i Note</b></p> <p>A tooltip displays within the chart at runtime to show in detail what values are selected. The tooltip on a mobile device only displays how many items are selected, if more than one is selected.</p> </div>

Property	Property Value	Property Description
		<p><b>i Note</b></p> <p>To clear member selections, you can use the <a href="#">CHART.clearSelection()</a>; scripting method. Alternatively, the user can click outside the data point area.</p>
zoom	<ul style="list-style-type: none"> <li>• enablement <ul style="list-style-type: none"> <li>◦ auto</li> <li>◦ enabled</li> <li>◦ disabled</li> </ul> </li> <li>• direction <ul style="list-style-type: none"> <li>◦ all</li> <li>◦ category axis</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Select from the <a href="#">Enablement</a> property options to determine whether the zoom functionality is enabled by default, enabled or disabled.</li> <li>• Select from the <a href="#">Direction</a> property options to determine if the user can select to zoom in all directions or only on the category axis.</li> </ul>

### 36.6.3 Crosstab

The crosstab displays multi-dimensional data in a grid with analytic functions. After you have dragged and dropped a crosstab in the layout editor, the crosstab is initially displayed with scrollbars and dummy data. As soon as you assign a data source to the crosstab, the data of the data source is displayed in the crosstab.

#### **i Note**

- The scrollbars of the crosstab displayed in the editor are not intended to be used during design time. They are only intended for layout and design purposes.
- The scrollbars are only displayed if the crosstab has a minimum size (row headers, column headers and a specific number of data cells).

The [Crosstab](#) component has the following specific properties:

## Data Binding Properties

Table 67:

Property	Property Value	Property Description
data source	name of data source alias (for example, "DS_1")	<p>Displays all data source aliases. If you have created multiple data sources for the application, you can change the data source for the crosstab by choosing the relevant data source alias.</p> <div><p><b>i Note</b></p><p>As you can use the same data source several times within one application, you work in the design tool with data source aliases as reference names.</p></div>

## Properties: Optimization for Low Data Volume

Table 68:

Property	Property Value	Property Description
pixel-based scrolling	true, false	<p>Enables smooth scrolling experience. We recommend using this property when creating mobile applications and/or applications with low data volume. If you do not activate pixel-based scrolling, then row-based/column-based scrolling will be applied by default.</p> <div> <b>i Note</b>            This property is mandatory for setting the following properties of row and column limits:           <ul style="list-style-type: none"> <li>For iPad applications, SAP recommends that you limit the total number of cells (the product of rows*columns) to 500, to improve performance and user experience.</li> <li>For desktop browser applications with pixel-based scrolling, the total number of cells can be set higher, depending on the performance of the client PC (recommended maximum number of cells 5000).</li> </ul> </div>
row limit	numeric value for rows; default: 20 (max)	Specifies the maximum number of rows for pixel-based scrolling. With this property, you can set the number of rows to be displayed. For iPad applications, SAP recommends a maximum of 20 rows.
column limit	numeric value for columns; default: 20 (max)	Specifies the maximum number of columns for pixel-based scrolling. With this property, you can set the number of columns to be displayed. For iPad applications, SAP recommends a maximum of 20 columns.

## Properties User Interactivity

Table 69:

Property	Property Value	Property Description
selection type	none (default), single, multi, data cell	<p>Specifies if and which kind of data selection is offered for the application user at runtime:</p> <ul style="list-style-type: none"> <li>If set to value <i>None</i>, selection is completely disabled. The user cannot select anything in the crosstab. There is no hovering effect even if the property <i>Enable Hover Effect</i> is set to <i>true</i>.</li> <li>If set to value <i>Single</i>, the user can select rows and columns in the crosstab by hovering over and clicking the members of the required dimensions.</li> </ul> <div> <p><b>i Note</b></p> <p>If you open an application with the formerly used property <i>Selection Enabled</i> set to <i>true</i> on the crosstab but the application was created with an older version of the design tool (prior to SAP BusinessObjects Design Studio 1.5), the formerly used <i>Enable Selection</i> property automatically transferred to <i>Enable Selection</i> with value <i>Single</i>.</p> </div> <ul style="list-style-type: none"> <li>If set to value <i>Multi</i>, multiple selection is enabled. This means you can select multiple rows and columns by pressing <b>Ctrl</b> + <b>Left</b> mouse button on the required rows and columns. To remove a single selected row or column from multiple selected rows or columns, press <b>Ctrl</b> + <b>Left</b> mouse button on the selected row or column again. You can also achieve multi selection by choosing a member of an outer dimension with children.</li> </ul> <div> <p><b>i Note</b></p> <ul style="list-style-type: none"> <li>Multiple selection is not possible on mobile devices.</li> <li>It is not possible to specify multiple selection by clicking the left mouse key on a start row/column and then clicking <b>Shift</b> + <b>Left</b> mouse button on the end row/column.</li> <li>Each time the user performs an individual selection, an <i>On Select</i> event is triggered.</li> <li>You can only select multiple members that have a parent dimension (for example all re-</li> </ul> </div>

Property	Property Value	Property Description
		<p>gions of Germany). If you change the parent dimension (for example, from Germany to Cuba), the already selected members get de-selected as they do not belong to this group.</p> <ul style="list-style-type: none"> <li>• If set to value <i>Data cell</i>, the selection of a single data cell is enabled. This means all members on the row and column axes that specify this data cell are highlighted and selected as well. The application designer can get the full selection with the API method <code>getSelection</code> or get the relevant members using <code>getSelectedMember</code>. It is not possible to select more than one data cell.</li> </ul>
selectable area	all (default), rows, columns	<p>Specifies if the application user can select the rows or columns in the header area. However, a selection cannot exist at the same time for both rows and columns. If rows are selected, all previously made column selections are undone (same applies for the other way round).</p> <p><b>i Note</b></p> <p>This property is only useful, if you have set the property <i>Enable Selection</i> to <i>Multi</i> or <i>Single</i>.</p>
enable hover effect	true (default), false	<p>This property does not have any effect if selection is disabled in the crosstab. If selection is generally enabled, the property specifies if hover effects should be displayed when the mouse pointer is moved over a selectable cell.</p> <p><b>i Note</b></p> <p>On mobile devices, there are no hover effects as there is no mouse pointer.</p>



Property	Property Value	Property Description
single on select event	false (default), true	<p>Specifies the behavior of the <a href="#">On Select</a> event when multiple rows or columns in the crosstab are selected with the <code>CTRL</code> key pressed (also see property <a href="#">Selection Type</a>, value <a href="#">Multi</a>). When selecting multiple rows or columns in the crosstab with the <code>CTRL</code> key pressed, the <a href="#">On Select</a> event will be fired on each single selection click, thus executing any script that is set for the On Select event. This may not always be desired. Instead, you might want to first finish the multi-selection and then have the On Select event executed only once. This parameter controls exactly this behavior.</p> <p>When set to <a href="#">false</a> (default), each click with the <code>CTRL</code> key pressed will fire an On Select event. When set to <a href="#">true</a>, each click with the CTRL key pressed will NOT fire an On Select event, but there will be only one single On Select event fired when the user releases the <code>CTRL</code> key. This means that the user can finish multi-selection before an event is fired.</p>
hierarchy navigation enabled	false, true	<p>Specifies whether the application user can expand or collapse hierarchy nodes, if the crosstab contains one or more hierarchies or hierarchical structures. If this property is set to <a href="#">false</a>, the hierarchy expand/collapse icons (plus and minus symbols) are not shown in the crosstab. However, the end user can still see that there is a hierarchy because of the indentation of the nodes and leaves.</p>
sorting enabled	true, false	<p>Specifies whether the end user can sort columns. If the property is set to <a href="#">false</a>, the icons for sorting are not shown in the column headers.</p>
column resizing enabled	true, false	<p>Specifies whether the end user can resize columns by double-clicking on the right border of the column header. If this property is set to <a href="#">false</a>, the hover spot for the double-click resize will not be available.</p>

Property	Property Value	Property Description
horizontal header resizing enabled	true, false	<p>Specifies whether the application user can move the vertical border between the dimension area and measure area by dragging and dropping this border horizontally and thus altering the width ratio between dimension area and measure area. The default value is <i>false</i>.</p> <div> <p><b>i Note</b></p> <p>If you have specified a maximum header width in pixels, the user cannot extend the width of the dimension area beyond this value.</p> </div> <p>For further information, see <a href="#">Using Header Scrolling in Crosstabs [page 248]</a></p>
horizontal scrolling for header enabled	true, false	<p>Specifies whether the header area will have a horizontal scrollbar. When set to <i>false</i> (default), the crosstab will not have a scrollable header area at runtime. Only if this property is set to <i>true</i>, the other properties relating to the header scrolling function in the crosstab (<i>Horizontal Header Resizing Enabled</i> and <i>Maximum Width of Header Area</i>) are taken into account to enable finer control of the way the dimension area behaves. For further information, see <a href="#">Using Header Scrolling in Crosstabs [page 248]</a>.</p>
context menu enabled	true, false	<p>Specifies whether a context menu can be displayed on the crosstab. The context menu can only be displayed on the crosstab, if the property is set to <i>true</i> and the technical component CONTEXT_MENU is available in the application (in the <i>Outline</i> view in the <i>Technical Components</i> folder). If the property is set to <i>false</i>, the context menu cannot be displayed regardless of whether the technical component CONTEXT_MENU is part of the application or not. The entries of the context menu depend on the element of the data source the user clicks on. For more information, see <a href="#">Using the Context Menu (Technical Component) [page 60]</a></p>

Property	Property Value	Property Description
drag and drop enabled	true, false	<p>Specifies if the crosstab is drag and drop enabled. This property is set to <i>false</i> by default. If set to true, the crosstab</p> <ul style="list-style-type: none"> <li>enables internal drag and drop operations. Thus the application user can drag and drop dimensions and members within the crosstab and remove dimensions and members by dragging and dropping them outside the area of the crosstab.</li> <li>accepts external drops of dimensions from other components (for example the navigation panel). This only works if the application property <i>Drag and Drop between Components</i> is set to <i>true</i> as well.</li> </ul> <p>For further information about Drag&amp;Drop, see <a href="#">Working with Drag and Drop in Applications and Crosstabs [page 224]</a>.</p>

## Properties: Display

Table 70:

Property	Property Value	Property Description
CSS class	Your CSS class	Defines an additional CSS class for custom CSS. The CSS classes must have the format <b>myclass</b> (and not <b>.myclass</b> ). You do not have to use this property to be able to use <i>Custom CSS</i>
units and scaling factors	display both in header, display units in data cells, do not display	Specifies how units and scaling factors are shown in the crosstab. Measures can have scaling factors and units or currencies. If you have chosen <i>Display both In Header</i> , an additional header row is added to the row or column header. This additional header row contains information about the scaling factor and the unit or currency - provided that the information is consistent (for example, EUR 1,000).
always fill	true, false	Specifies whether the component space should be filled as defined by the layout properties. If you set the property value to <i>false</i> , the crosstab is displayed in its actual size, but within a frame that has the size defined in the layout properties. If you set the property to <i>true</i> , the crosstab is adjusted and displayed in the exact size defined in the layout properties.

Property	Property Value	Property Description
conditional formatting visible	true, false	<p>Specifies whether conditional formats (exceptions) that have been defined in the BEx Query Designer or in SAP BusinessObjects Analysis, edition for Microsoft Office (Analysis), are shown in the crosstab.</p> <div> <p><b>i Note</b></p> <ul style="list-style-type: none"> <li>To return the names of conditional formats, this property must be set to true</li> <li>In the BEx Query Designer you can define threshold values (exceptions) for a query. Data that deviates from these exceptions is marked in different colors. You can use these exceptions to spot deviations from expected results straight away. The exception visualization is based on nine alert levels. For each alert level, the affected cells are displayed in the corresponding background color. For more information, see the documentation for the BEx Query Designer on SAP Help Portal at <a href="http://help.sap.com">http://help.sap.com</a></li> <li>In Analysis, any results that fall outside a set of predetermined threshold values (rules for conditional formatting) are highlighted in color or designated with symbols. For more information, see the <i>SAP BusinessObjects Analysis, edition for Microsoft Office User's Guide</i> on SAP Help Portal at <a href="http://help.sap.com/boaa">http://help.sap.com/boaa</a>.</li> </ul> </div>

Property	Property Value	Property Description
column widths	numeric values $\geq 0$	<p>Sets the column widths of the crosstab. If you select this property, the <a href="#">Edit Columns Widths</a> dialog box is displayed. Choose <a href="#">Insert</a> to set the column widths.</p> <p>A valid column width setting consists of two parts:</p> <ul style="list-style-type: none"> <li>• <a href="#">Column Index</a>: The index of the addressed column. The index must be a number <math>\geq 0</math>. The leftmost column of the crosstab is column 0, the second is column 1 and so on. If the column index addresses a column outside the total number of columns in the crosstab, the setting will be ignored.</li> <li>• <a href="#">Column Width</a>: The column width must be a number <math>\geq 0</math>. The width unit is pixels. There is a minimum column width for each column even when a column width is set to 0, depending on the theme and the applied custom CSS styles. The minimum width depends on the settings for borders, padding, margin and similar style attributes of the crosstab cells that can be changed by the customer. Therefore it is not possible to specify the exact minimum width.</li> </ul> <div style="background-color: #fff9c4; padding: 10px; margin: 10px 0;"> <p><b>i Note</b></p> <p>It is not possible to set a column width that is smaller than this minimum width. You cannot use this feature to hide columns by trying to set a width of 0 pixels for a column in the crosstab.</p> </div> <p>If the application user is allowed to resize columns, the resulting width will take precedence over any width set using the API method <code>setColumnWidth</code> or by the crosstab property <a href="#">Column Widths</a>. Subsequent API calls to change the column width of a user-resized column will have no effect. If you do not want the user to be able to resize the crosstab columns at runtime, you can disable this feature by setting the crosstab property <a href="#">Column Resizing Enabled</a> to false.</p>

Property	Property Value	Property Description
maximum width of header area	auto, integer values > 0	<p>Specifies the header area width of the crosstab. If the property is set to <a href="#">auto</a> (default setting), the width of the dimension area can take any value within the horizontal boundaries of a fully rendered dimension area. If set to a pixel value, the width of the dimension area can only be lower than or equal to that value. The automatic dimension area width determination and the user's possibility to alter the width of the dimension area if the property <a href="#">Horizontal Header Resizing Enabled</a> is set to <a href="#">true</a>, take a non-auto pixel width setting into account. If that pixel value is set, the crosstab will try to render the dimension header area width in this size. However, if the property's pixel value is larger than the fully rendered width of the dimension area, the dimension area will only take up as much horizontal space as it actually needs. The dimension area width is not artificially enlarged (cells made wider) to stretch out to the given pixel width. For further information, see <a href="#">Using Header Scrolling in Crosstabs [page 248]</a></p> <div> <p><b>i Note</b></p> <p>This property cannot be used to hide the header area.</p> </div>
display repeated texts	true, false	Specifies if multiple header cells with the same texts are merged into single cells with row spans or column spans.

## Properties: Planning

Table 71:

Property	Property Value	Property Description
number of new rows	numeric value	Specifies the number of new rows for manual planning applications.
position of new rows	bottom, top	Specifies the position of new rows for planning applications.
maximum number of value help members	numeric value (default: 100)	Specifies the maximum number of members displayed in a value help for non-hierarchical dimensions.

## Events

Table 72:

Property	Property Value	Property Description
on select	none	<p>Opens the script editor. With this property/event, you can enable user interaction with the crosstab by writing scripts. Click <code>CTRL</code> <code>+Space</code> to see the list of available methods for the application, the data source alias and the crosstab. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process. The following specific methods are currently available for the crosstab:</p> <p>For further information, read the relevant method description in the API reference.</p> <div><p><b>i Note</b></p><p>The <i>On Select</i> event of the crosstab is triggered when a value is selected and also when a value is deselected.</p></div>

## 36.6.4 Dimension Filter

### Syntax

Using the dimension filter, you can add a generic filter for one dimension to an application, without the need to use scripting. The dimension filter provides an easy way for the end user to define a filter for a single dimension of a data source. Click on the dimension filter at runtime to open a popup with the "selection view" of the dimension.

The *Dimension Filter* component has the following specific properties:

Property	Property Value	Property Description
data source	name of data source alias (for example "DS_1")	<p>Displays all data source aliases. If you have created several data sources for the application, you can change the data source for the filter panel by choosing the relevant data source alias.</p> <div><p><b>i Note</b></p><ul style="list-style-type: none"><li>As you can use the same data source several times within one application, you need to work in</li></ul></div>

Property	Property Value	Property Description
		<p>the design tool using data source aliases as reference names.</p> <ul style="list-style-type: none"> <li>Keep in mind that the data source is filtered automatically and you do not have to add it to the <a href="#">Target Data Sources</a> property.</li> <li>If you change the data source, the system automatically refreshes the list of the <a href="#">Target Data Sources</a>.</li> </ul>
target data sources (optional)	names of target data source aliases (for example, "DS_1; DS_2" )	If you want the filter to be applied to other, additional data sources, you can also define target data sources. If the target data sources contain the same dimension as the (source) data source, the filter applies the dimension to the target data sources in the same way as in the (source) data source. The target data sources must be existing data sources aliases added to the application and separated by a semicolon.
dimension	dimension	Specifies the dimension to be filtered. You can select a dimension using the <a href="#">Dimension Selection</a> dialog box or search for a dimension.
dimension name	true, false	Specifies whether the dimension name of the selected dimension is displayed in the dimension filter.
display mode	filter list, filter count	Specifies the display mode of the dimension members. If you choose <a href="#">Filter List</a> , the applied members (= filter values) are displayed as a comma-separated list (for example, <a href="#">Fax</a> , <a href="#">Internet</a> , <a href="#">Phone</a> ). If you choose <a href="#">Filter Count</a> , the number of the applied members (=filter values) is displayed in parentheses (for example, <a href="#">(3)</a> ).
member display	key, text, key and text, text and key	Specifies the dimension member display.
popup width	numeric value in pixel	Specifies the width of the popup displayed when the user clicks on the dimension filter at runtime.
popup height	numeric value in pixel	Specifies the height of the popup displayed when the user clicks on the dimension filter at runtime.



Property	Property Value	Property Description
popup position	bottom right, bottom left, center right, center left, top right, top left, bottom center, top center	Specifies the position of the popup displayed when the user clicks on the dimension filter at runtime.
popup is modal	true, false (default)	Specifies whether the value help popup dialog box of the dimension filter is modal. If the property is set to <i>true</i> , the user cannot interact with other UI elements in the background of the application (for example, click a button or dropdown box). The user can only interact with the elements in the value help popup dialog box. If the property is set to <i>false</i> , users can interact with other UI elements.
remove redundant selections	true, false	Specifies whether the system removes redundant selections in hierarchical dimensions. If the property is set to <i>true</i> and if the application user selects a node and a subnode, the system removes the subnode because it is automatically selected when the parent node is selected.
auto apply	true, false (default)	Specifies if changes made by the application user at runtime are applied automatically. If set to <i>false</i> , the application user has to click on <i>Apply</i> after making changes in the filter panel. The filtering changes are also applied if the application user uses the direct input and then clicks on the <i>Back</i> button or <i>OK</i> button
maximum number of members	<i>your number</i> , default: 100	Specifies the maximum number of members displayed in the value help for non-hierarchical dimensions. If the number of members is greater than the value for this property, no values are displayed. Instead, the application user has to limit the number by searching for the text or key of one or more values.
desktop style for popup	true, false	Specifies the style of the value help popup and enables you to switch between mobile style and desktop style (changes layout of buttons). If the property is set to true, the value help popup is displayed in desktop style.
on apply	none	Opens the script editor. With this property/event, you can enable user interaction with the dimension filter by writing scripts. The <i>On Apply</i> event is triggered when the application user presses the <i>Apply</i> button of the

Property	Property Value	Property Description
		<p>dimension filter at runtime. In the script editor, click <code>CTRL</code> + <code>Space</code> to see the list of available methods for the application, the data source alias and the dimension filter. Choose one of the methods or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the relevant method description in the API reference.</p>

## Additional, changed or removed properties in SAPUI5 m Mode

Due to the different rendering in SAPUI5 m mode, the dimension filter has a different properties sheet in the m mode:

- Property *Display Mode* has been removed.
- Property *Popup Width* has been removed.
- Property *Popup Height* has been removed.
- Property *Popup Position* has been removed.
- Property *Popup is Modal* has been removed.
- Property *Auto Apply* has been removed.

### 36.6.5 Filter Line

The *Filter Line* analytic component offers the user a list of filters that are applied to the assigned data source. It behaves in a similar way to the *Filter Panel*. With the *Filter Line*, the user can add, remove, view and edit the dimensions and measures to which they can apply filters. When the user selects to add or edit a filter, the *Dimension Filter* will appear. The filter on the selected dimension and member(s) is added to the row of filters in the *Filter Line* and the filter is applied to your data source.

Filter items can be directly removed from the *Filter Line*, without having to drill down into the *Dimension Filter*.

On SAP HANA the measure dimension will be *Measures*. On SAP BW the measure dimension will be *Key Figures*.

You can use a combination of scripting methods and the *Filter Line* component, to create a global filter that allows the application user to apply a filter across multiple data sources. This global filter can be dynamically added to their application. For more information on using the filter line to apply a global filter, you can refer to the chapters in this guide called *Applying Global Filters* and *Working with Online Composition*.

The *Filter Line* component has the following specific property:

Property	Property Value	Property Description
measure visible	true, false	Select <i>true</i> to include measures in the list of data source filters in the <i>Filter Line</i> menu.

## Related Information

[Filter Panel \[page 387\]](#)

[Dimension Filter \[page 383\]](#)

[Applying Global Filters \[page 156\]](#)

[Working with Online Composition \[page 150\]](#)

## 36.6.6 Filter Panel


### Syntax

With the filter panel, you can add a generic filter to an application without using scripting. The filter panel makes it easy for the end user to define a filter for the data source. To use the filter panel, you have to add at least one data source to the application and assign it to the filter panel.

#### Note

If a dimension has more than 100 members, the members are not listed at runtime. Instead the user is asked to use the search function.

The *Filter Panel* component has the following specific properties:

Property	Property Value	Property Description
data source	name of data source alias (for example "DS_1")	<p>Displays all data source aliases. If you have created several data sources for the application, you can change the data source for the filter panel by choosing the relevant data source alias.</p> <div>  <b>Note</b> <ul style="list-style-type: none"> <li>As you can use the same data source several times within one application, you need to work in the design tool using data source aliases as reference names.</li> <li>Keep in mind that the data source is filtered automatically</li> </ul> </div>

Property	Property Value	Property Description
		<p>and you do not have to add it to the <a href="#">Target Data Sources</a> property.</p> <ul style="list-style-type: none"> <li>If you change the data source, the system automatically refreshes the list of the <a href="#">Target Data Sources</a>.</li> </ul>
target data sources (optional)	names of target data source aliases (for example "DS_1; DS_2" )	<p>If you want the filter to be applied to other, additional data sources, you can also define target data sources. If the target data sources contain the same dimension as the source data source, the filter applies the dimension to the target data sources in the same way as in the source data source. The target data sources have to be existing data sources aliases that have been added to the application. After you have selected the target data sources in the <a href="#">Select Target Data Sources</a> dialog box, the target data sources are displayed, separated by semicolons.</p>
mode	filtering, navigation, filtering and navigation	<p>Enables you to set the function scope of the filter panel:</p> <ul style="list-style-type: none"> <li><a href="#">Filtering</a> enables the application user to filter data.</li> <li><a href="#">Navigation</a> enables the application user to add or remove dimensions from rows or columns of the result set by using the corresponding buttons.</li> <li><a href="#">Filtering and Navigation</a> enables both functions explained above.</li> </ul>
dimensions	dimensions	<p>Opens the <a href="#">Select Dimensions</a> dialog box. Here you can select and sort dimensions of the data source you have assigned to the filter panel.</p>
display mode	filter list, filter count	<p>Specifies the display mode of the dimension members. If you choose <a href="#">Filter List</a>, the applied members (= filter values) are displayed as a comma-separated list (for example, <a href="#">Fax</a>, <a href="#">Internet</a>, <a href="#">Phone</a>). If you choose <a href="#">Filter Count</a>, the number of the applied members (=filter values) is displayed in parentheses (for example, <a href="#">(3)</a>).</p>
member display	key, text, key and text, text and key	<p>Specifies the display mode of the dimension members.</p>

Property	Property Value	Property Description
title	<code>yourTitle</code>	You can set a title for the filter panel. The title is displayed in the middle of the filter panel header.
remove redundant selections	true, false	Specifies whether the system removes redundant selections in hierarchical dimensions. If the property is set to <i>true</i> and if the application user selects a node and a subnode, the system removes the subnode as this one is automatically selected if the the parent node is selected.
drag & drop	true, false	Specifies whether the application user can change the axis of the dimension in navigation mode by using drag & drop.
direct input for filter	enabled, disabled for measures, disabled	Specifies whether the application user can use direct input for dimension or measure keys. If set to <i>Disabled for Measures</i> , the application user cannot use direct input for measures but only for dimensions.
auto apply	true, false	Specifies if all changes made by the application user at runtime are applied automatically. If set to <i>false</i> , the application user has to click on <i>Apply</i> after making changes in the filter panel. all changes in navigation will be applied immediately. The changes in the filter panel are also applied if the application user uses the direct input and then clicks on the <i>Back</i> button or <i>OK</i> button.
maximum number of members	<code>your number</code> , default: 100	Specifies the maximum number of members displayed in the value help for non-hierarchical dimensions. If the number of members is greater than the value for this property, no values are displayed. Instead, the application user has to limit the number by searching for the text or key of one or more values.
on apply	none	Opens the script editor. With this property/event, you can enable user interaction with the filter panel by writing scripts. Click <code>CTRL +Space</code> to see the list of available methods for the application, the data source alias and the filter panel. Choose one of the methods or use the statement wizard. The wizard guides you through the statement creation process. The <i>On Apply</i> event is rendered when the end user presses the <i>Apply</i> button at runtime. The filter value is set first.

Property	Property Value	Property Description
		For further information, read the relevant method description in the API reference.
on cancel	none	<p>Opens the script editor. With this property/event, you can enable user interaction with the filter panel by writing scripts. Click <b>CTRL +Space</b> to see the list of available methods for the application, the data source alias and the filter panel. Choose one of the methods or use the statement wizard. The wizard guides you through the statement creation process. The <i>On Cancel</i> event is rendered when the end user presses the <i>Cancel</i> button at runtime.</p> <p>For further information, read the relevant method description in the API reference.</p>

## Additional, changed or removed properties in SAPUI5 m Mode

Due to the different rendering in SAPUI5 m mode, the filter panel has a different properties sheet in the m mode:

- Property *Display Mode* has been removed.
- Property *Drag and Drop* has been removed.
- Property *Auto Apply* has been removed.
- Property *Direct Input for Filter* has been removed.
- Property *Desktop Style for Popup* has been removed.
- Event *On Cancel* has been removed.



## 36.6.7 Geo Map

The *Geo Map* analytic component allows you to display different layers of geographical information on a map and allow users to drilldown through the different layers to reveal data in a variety of ways. You can apply three different types of layers to your *Geo Map* - shapes (polygons, multi-polygons, lines and multi-lines), points, bubble charts and pie charts.

You can assign a different data source to each layer.

### Note

You must read the following SAP Note before using the *Geo Map* component:

SAP Note Number	Description
2238090 	Working with SAP BusinessObjects Design Studio Geo Maps.
2239530 	Design Studio RTL support for Geo Maps.


The properties of the *Geo Map* component include the following views:

- Properties
- Additional Properties

The *Geo Map Properties* and the *Additional Properties* views are described in the tables below.

## Properties

Table 73:

Property	Property Value	Property Description
basemap url	url	<p>The URL template which is used to provide the basemap. The parameters LOD, X and Y are expanded to provide a URL for each individual basemap. For example, <b>my.map/{LOD}/{X}/{Y}.png</b>. Selecting to have a basemap is optional.</p> <div>  <b>Note</b>  LOD indicates 'level of detail'. It is also sometimes referred to as 'Z'. </div>
basemap copyright	text	Text which is displayed at the bottom right of the map. The basemap provider may require you to specify a reference to the copyright text for the basemap url you have selected to use.
map legend visible	true, false	Specifies whether the application user can set the map legend to visible.

## Additional Properties

Table 74:

Property	Property Value	Property Description
id	unique layer id	<p>Enables you to assign a unique identifier to each layer of your <a href="#">Geo Map</a>. This ID is then referenced by the <a href="#">Geo Map</a> scripting methods.</p> <div> <p><b>i Note</b></p> <p>IDs must be valid CSS selectors. A name must begin with an underscore, a hyphen, or a letter, followed by any number of hyphens, underscores, letters, or numbers. If the first character is a hyphen, the second character must be a letter or underscore. The name must be at least two characters long.</p> </div>
show layer	checkbox	Sets the visibility of the layer. When this checkbox is selected, the layer is visible on the <a href="#">Geo Map</a> . This checkbox is selected by default.
type	<ul style="list-style-type: none"> <li>• shape</li> <li>• point</li> <li>• bubble chart</li> <li>• pie chart</li> </ul>	Specifies the type of layer to display data on the <a href="#">Geo Map</a> . The type selected determines which other properties become available. The <a href="#">Shapes</a> layer is the default layer type.
data source	drop down list of available data sources beside an ellipsis button	Enables you to select a data source to assign to the current layer in the <a href="#">Geo Map</a> .
measure	drop down list of the available measures	Enables you to assign a measure to a shape, bubble chart or pie chart layer.
start color	hex, color picker	You can use a color picker to specify the start color of a shape layer color gradient. The default start color is white.
end color	hex, color picker	You can use a color picker to specify the end color of a shape layer color gradient. The default end color is black.
classification type	quantile, equal interval	Defines the type of classification to be applied to a shape layer. Select the quantile option to distribute a set of values into groups that contain an equal number of values. Select the equal interval option to arrange a set of values into groups that contain an equal range of values.



Property	Property Value	Property Description
custom geoJSON file	text	Specifies the file path to the GeoJSON file. If you are on the BI platform or the <a href="#">SAP NetWeaver</a> platform, you can select the ellipsis button within the <a href="#">Custom GeoJSON File</a> property to select the location of your custom GeoJSON file. Selecting the <a href="#">Custom GeoJSON File</a> ellipsis button in local mode, opens a <a href="#">Select Local GeoJSON</a> dialog box, from where you can select your locally stored custom GeoJSON file.
geoJSON mapping property	drop down list	Provides a list of all available properties in the GeoJSON file. To assist you with your selection, a text value is available beside the first property within the GeoJSON file. Select the drop down arrow to view it.
geoJSON mapping type	key, text	Specifies whether key or text of the geo dimension member is used when mapping to the GeoJSON.
latitude	drop down	Enables you to assign a latitude measure name to points and charts layers.
longitude	drop down	Enables you to assign a longitude measure name to points and charts layers.
color	hex, color picker	Assigns a color to points and charts layers.
custom marker file	text	File path to the location of the custom marker file.
vertical anchor	<ul style="list-style-type: none"> <li>• top left</li> <li>• top center</li> <li>• top right</li> <li>• center left</li> <li>• center</li> <li>• center right</li> <li>• bottom left</li> <li>• bottom center</li> <li>• bottom right</li> </ul>	Defines the anchor point for the selected custom marker. For example, if you select bottom left option, the bottom left corner of the marker will indicate the anchor point.

## Related Information

[Working with Geo Maps \[page 215\]](#)

## 36.6.8 Navigation Panel

Using the navigation panel, you can easily change the drilldown of the data and see at a glance the navigation state of the data source at runtime. To change the drilldown, the application user can drag and drop the dimensions from the list into the rows or columns area. The application user can also rearrange the drilldown by dragging and moving dimensions from the rows into the columns area and/or the other way round. To remove dimensions from the drilldown, the application user can drag a dimension and drop it anywhere in the application.

The *Navigation Panel* component has the following specific properties:

Property	Property Value	Property Description
dimensions	dimensions	Opens the <i>Select Dimensions</i> dialog box. Here you can select and sort the dimensions of the data source that you have assigned to the navigation panel.
mode	navigation, list only	Specifies the mode used to display the different parts of the component. If you choose <i>Navigation</i> , the component is displayed with the rows and columns area (navigation area). If you choose <i>List Only</i> , the dimensions are displayed in a list, without the rows and columns area. In this case, no drag and drop is possible; the application user can use the context menu instead.
list measures separately	true, false; default: false	Specifies how the measures are displayed in the list. If set to <i>true</i> , the measures (key figures) are listed separately above the dimensions.
pause refresh	true, false, default: true	Specifies if a refresh is paused initially.

## Additional, changed or removed properties in SAPUI5 m Mode

Due to the different rendering in SAPUI5 m mode, the navigation panel has a different properties sheet in the m mode:

- Property *Pause/Refresh* has been removed.

## 36.6.9 Spreadsheet

The *Spreadsheet* component displays multi-dimensional data in a grid with analytic functions. In addition, the Spreadsheet component offers an easy-to-use copy function for copying cells and sheet areas. The spreadsheet is mainly intended for use with planning applications with input-ready queries.

## Restrictions

When using the spreadsheet component, note the following restrictions:

- The universal display hierarchy of SAP BW data sources.
- Only one hierarchy per axis is supported.
- The Key and Text of dimensions are only displayed in one cell/column and not in separate cells/columns.
- Attributes are not supported
- In SAP UI5 m rendering mode, the Spreadsheet component is only supported in the Compact Form Factor.
- The Spreadsheet component can only be used with the Blue Crystal theme
- The Spreadsheet component does not allow any interaction in mobile applications.

## Restrictions for Design Studio based on SAP HANA as platform

If your Design Studio is based on SAP HANA as platform, note the following additional restrictions when using the spreadsheet component:

- Bottom up hierarchies (parent node below child node/member) are not supported.
- New lines cannot be inserted.
- Dynamic calculations and calculations based on the menu entry *Calculate Totals As...* are not supported.

The Crosstab component has the following specific properties:

## Properties User Interactivity

Table 75:

Property	Property Value	Property Description
dimension toggles visible	false (default), true	Specifies if the dimension toggles are visible. With the dimension toggles, the user can hide one or more dimension rows or columns. This can be useful if the query has many dimensions and the user wants to restrict which dimensions are displayed.
hierarchy navigation enabled	false, true (default)	Specifies if the application user can expand or collapse hierarchy nodes, if the spreadsheet contains one or more hierarchies or hierarchical structures. If this property is set to <i>false</i> , the hierarchy expand/collapse icons (plus and minus symbols) are not shown in the crosstab. However, the end user can still see that there is a hierarchy because of the indentation of the nodes and leaves.

Property	Property Value	Property Description
context menu enabled	true (true), false	Specifies if a context menu can be displayed on the cross-tab. The context menu can only be displayed on the spreadsheet, if the property is set to <i>true</i> and the technical component CONTEXT_MENU is available in the application (in the <i>Outline</i> view in the <i>Technical Components</i> folder). If the property is set to <i>false</i> , the context menu cannot be displayed, regardless of whether the technical component CONTEXT_MENU is part of the application. The entries displayed in the context menu depend on the data source element that the user clicks on. For more information, see <a href="#">Using the Context Menu (Technical Component) [page 60]</a>

## Properties: Display

Table 76:

Property	Property Value	Property Description
CSS class	Your CSS class	Defines an additional CSS class for custom CSS. The CSS classes must have the format <b>myclass</b> (and not <b>.myclass</b> ). You do not have to use this property to use <a href="#">Custom CSS</a>
conditional formatting visible	true, false	<p>Specifies if conditional formats (exceptions), which have been defined in the BEx Query Designer or in SAP BusinessObjects Analysis, edition for Microsoft Office (Analysis), are shown in the spreadsheet.</p> <div> <p><b>i Note</b></p> <ul style="list-style-type: none"> <li>To return the names of conditional formats, this property must be set to true</li> <li>In the BEx Query Designer you can define threshold values (exceptions) for a query. Data that deviates from these exceptions is marked in different colors. You can use these exceptions to spot deviations from expected results straight away. The exception visualization is based on nine alert levels. For each alert level, the affected cells are displayed in the corresponding background color. For more information, see the documentation for the BEx Query Designer on SAP Help Portal at <a href="http://help.sap.com">http://help.sap.com</a></li> <li>In Design Studio, any results that fall outside a set of predetermined threshold values (rules for conditional formatting) are highlighted in color or designated with symbols. For more information, see the <i>SAP BusinessObjects Analysis, edition for Microsoft Office User's Guide</i> on SAP Help Portal at <a href="http://help.sap.com/boaa">http://help.sap.com/boaa</a>.</li> </ul> </div>

Property	Property Value	Property Description
display repeated texts	true, false	Specifies if multiple header cells with the same texts are merged into single cells with row spans or column spans.

## Properties: Planning

Table 77:

Property	Property Value	Property Description
maximum number of value help members	numeric value (default: 100)	Specifies the maximum number of members displayed in a value help for non-hierarchical dimensions.
paste new lines enabled	false (default), true	Specifies if the application user can insert new empty rows or paste copied rows or areas of the spreadsheet below the current last line of the spreadsheet.

## 36.7 Basic Components

### 36.7.1 Button

Buttons enable the user to interact within the application. To configure this interaction, you add a script to the button's On Click property. The script is triggered when the user clicks the button in the application. You can insert any text for the button label and add an icon.

The [Button](#) component has the following specific properties:

Property	Property Value	Property Description
text	default text: button	Specifies the text to be displayed on the button. You can overwrite the default text and enter your own text. The text is centered on the button.
icon	name of icon; save location	Specifies the icon to be displayed on the button. Insert the name of the image you want to add. Use one of the following methods: <ul style="list-style-type: none"> <li>If your Design Studio is in Local Mode and the icon is stored in the application directory (&lt;userhome&gt;</li> </ul>

Property	Property Value	Property Description
		<p> \Analysis_workspace  \com.sap.ip.bi.zen\repository  \MyApplication), type in the name of the icon (<b>MyIcon.jpg</b>) . </p> <ul style="list-style-type: none"> <li>• If your Design Studio is in Local Mode and the icon is stored in a subfolder of the application directory, type in the relative path, for example, <b>MySubFolder\MyImage.jpg</b>.</li> <li>• If the icon is stored on the Internet/intranet, type in the URL, for example, <b>http://www.myWebServer.com/myImage.jpg</b>.</li> <li>• If your Design Studio is connected to a SAP NetWeaver platform, click on the <i>Browse</i> button. The <i>Open Image</i> dialog box opens, where you can select the image or icon from the MIME repository. After the dialog box is closed, the path to the icon is inserted as the property value.</li> </ul> <div> <p><b>i Note</b></p> <p>In the MIME repository, you can select images and icons uploaded by the system administrator. For more information, see the <i>Administrator Guide: SAP BusinessObjects Design Studio</i> at <a href="http://help.sap.com/boad">http://help.sap.com/boad</a></p> </div> <p>If you select a text and an icon for the button, the icon is aligned left of the text. If you only select an icon, it is centered on the button.</p>
on click	script	<p>Opens the script editor. Using this property, you can enable user interaction with the button by writing scripts. The <i>On Click</i> event is triggered when the application user clicks on the button. In the script editor, click <b>CTRL</b> + <b>Space</b> to see the list of available methods for the application, the data source alias and the button. Choose one of them or use the statement wizard.</p>

Property	Property Value	Property Description
		<p>The wizard guides you through the statement creation process.</p> <p>For further information, read the relevant method description in the API reference.</p>

## 36.7.2 Chart Type Picker

The *Chart Type Picker* basic component allows the application user to select an alternative visualization type to display their data, while actually running their application. You can manipulate the *Chart Type Picker* properties to add additional visualization types to the existing palette of types. You can decide, through scripting, how the visualizations behave when the user runs the application. A dropdown list beside each group of visualizations, displays a list of each visualization type within each group. If you add an additional type, for example, a crosstab or an image component, to the *Chart Type Picker*, they will be grouped together in a new group.

Property	Property Value	Property Description
chart reference	dropdown list	Specifies which chart component in the application is changed when a different chart type is selected in the <i>Chart Type Picker</i> . The chart component type will change to reflect this chart type selection once the selection made is different to the chart component's current type.
additional types	ellipsis button that pops up a dialog box	<p>Dialog is displayed to allow you to add or remove additional visualization types. Each additional visualization type requires the following:</p> <ul style="list-style-type: none"> <li>• name</li> <li>• type</li> <li>• icon</li> </ul> <div> <p><b>Note</b></p> <ul style="list-style-type: none"> <li>• The additional visualization type entered must be unique. It must not be one of the standard chart types already available</li> <li>• All additional visualization types added, will be grouped together in a new group within the <i>Chart Type Picker</i> component.</li> <li>• If you have created and installed new chart extensions using the</li> </ul> </div>

Property	Property Value	Property Description
		<p>SAP Lumira SDK, these extensions will also appear in the <i>Chart Type Picker</i> component under the <i>Others</i> chart type category. For more information about creating SAP Lumira SDK extensions, see the <i>SAP Lumira Visualization Extensions Developer Guide</i> on the SAP Help Portal at <a href="http://help.sap.com/lumira">http://help.sap.com/lumira</a>.</p> <p>Extensions developed with the SAP Lumira SDK can only be deployed to the BI platform and to SAP NetWeaver. They cannot be deployed to SAP HANA.</p>

### 36.7.3 Info Chart Feeding Panel

The *Info Chart Feeding Panel* is a basic component that you can bind to a data bound info chart, through the *Chart Reference* property in the designer, or by using the `setChartReference` scripting method. The *Info Chart Feeding Panel* offers the application user a representation of all the measures and dimensions bound to that info chart in the *Edit Initial View....* It also allows the user to move measures and dimensions up and down within their respective areas in the *Info Chart Feeding Panel* component. Any changes they make to the position of measures and dimensions in the *Info Chart Feeding Panel* are then directly reflected in the info chart. It is important to note that you cannot drag measures into the dimensions area or vice versa. If the user makes a change to the position of measures and dimensions in the *Info Chart Feeding Panel* and then changes the chart type using the *Chart Type Picker*, the measures and dimensions remain in their new position. This ability to modify how data is fed to the info chart, gives the application user greater freedom when running their application.

Property	Property Value	Property Description
chart reference	dropdown list	Specifies which chart component is referenced by the <i>Info Chart Feeding Panel</i> .

## Related Information

[Working with Info Charts \[page 177\]](#)



## 36.7.4 Checkbox

Checkboxes enable the user to interact within the application. To do this, add a script to the On Click property of the checkbox. The script is triggered when the user activates the checkbox in the application. You can use any text for the checkbox.

The [Checkbox](#) component has the following specific properties:

Property Type	Property Value	Property Description
text	default text: checkbox	Specifies the text to be displayed on the right of the checkbox. You can overwrite the default text and enter your own text.
checked	false/true	Specifies whether the checkbox is initially selected.
on click	script	<p>Opens the script editor. With this property, you can enable user interaction with the checkbox by writing scripts. The <a href="#">On Click</a> event is triggered when the application user activates the checkbox. In the script editor, press <code>CTRL +Space</code> to see the list of available methods for the application, the data source alias and the checkbox. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the method description in the API reference.</p>

## 36.7.5 Checkbox Group

The checkbox group displays several checkboxes, each for one item. All checkboxes are aligned in a vertical fashion. Checkbox groups enable the user to interact within the application. To do this, add a script to the [On Select](#) property of the checkbox group. The script is triggered when the user activates or deactivates one checkbox in the application.

The [Checkbox Group](#) component has the following specific properties:

Property Type	Property Value	Property Description
items	list of items	<p>With the items property, you can open the <a href="#">Edit items</a> dialog. Here, you can insert items, change the order of items or remove them. Each item has the following properties:</p> <ul style="list-style-type: none"><li>• value</li></ul>

Property Type	Property Value	Property Description
		<p>The value property is mandatory for all items.</p> <ul style="list-style-type: none"> <li>text (optional)</li> </ul> <p>The text property is optional. If you enter a text for an item, this text is displayed right of the checkbox. If no text is entered, the value for the item is displayed.</p> <p>If you want a specific item to be displayed as default, select the relevant item and click <a href="#">Set as Default</a>. This item is then shown as the default entry in the radio button group.</p>
on select	script	<p>Opens the script editor. With this property, you can enable user interaction with the checkbox group by writing scripts. The <a href="#">On Select</a> event is triggered when the application user activates one checkbox. In the script editor, press <code>CTRL +Space</code> to see the list of available methods for the application, the data source alias and the checkbox group. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the method description in the API reference.</p>

## 36.7.6 Date Field

The date field component enables the user to select a date. To activate user interaction with the date field, you need to add a script to the On Select property. The script is triggered when the user picks a date in the date field or when the user enters a date manually at runtime.

The [Date Field](#) component has the following specific properties:

Property Type	Property Value	Property Description
date	date	Specifies the initial date in the format <code>yyyymmdd</code> .
on select	script	<p>Opens the script editor. With this property/event, you can enable user interaction with the date field by writing scripts. Click <code>CTRL +Space</code> to see the</p>

Property Type	Property Value	Property Description
		<p>list of available methods for the application, the data source alias and the date field. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the relevant method description in the API reference.</p>

## 36.7.7 Dropdown Box

Dropdown boxes enable the user to select items, for example, to set a filter. To enable user interaction, you need to add a script to the On Select property of the dropdown box. The script is triggered when the user selects an item in the dropdown box.

The *Dropdown Box* component has the following specific properties:

Property Type	Property Value	Property Description
items	list of items	<p>With the items property, you can open the <i>Edit items</i> dialog box. Here, you can insert items, change the order of items or remove them. Each item has the following properties:</p> <ul style="list-style-type: none"> <li>value The value property is mandatory for all items.</li> <li>text (optional) The text property is optional. If you enter a text for an item, the text is displayed. If no text is displayed, the value for the item is displayed.</li> </ul> <p>If you want a specific item to be displayed by default, select the relevant item and click <i>Set as Default</i>. This is shown as the default entry in the dropdown box.</p>
on select	script	<p>Opens the script editor. With this property/event, you can enable user interaction with the dropdown box by writing scripts. The <i>On Select</i> event is triggered when the application user selects an item from the dropdown box. In the script editor, click <span>CTRL +Space</span> to see the list of available methods for</p>

Property Type	Property Value	Property Description
		<p>the application, the data source alias and the dropdown box. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the relevant method description in the API reference.</p>

### Example

You have created an application with a crosstab. You have assigned the data data source with the data source alias DS\_1 to the crosstab. In addition, you have added a dropdown box to the application. This dropdown box contains items that represent the values of the dimension DEMO\_CHA2 of DS\_1. Using the on select event of the dropdown box, enter the following script:

```
DS_1.setFilter("DEMO_CHA2", DROPDOWN_1.getSelectedValue());
```

This event triggers the following action: the crosstab is filtered by the value selected in the dropdown box.

## 36.7.8 Formatted Text View

The *Formatted Text View* component enables you to apply text formatting within one control. The set of tags and attributes that can be used within this control are listed in the table below. To use the *Formatted Text View* component, you must drag the component onto the *Layout Panel*. The *Formatted Text View* editor appears in the *Additional Properties* panel.

The *Formatted Text View* editor supports the following input:

Table 78: Formatted Text View Editor

Input Type	Input Options
functions	<ul style="list-style-type: none"> <li>• cut</li> <li>• copy</li> <li>• paste</li> <li>• paste as plain text</li> <li>• undo</li> <li>• redo</li> </ul>
font	<ul style="list-style-type: none"> <li>• bold</li> <li>• italic</li> <li>• remove format</li> </ul>

Input Type	Input Options
styles	<ul style="list-style-type: none"> <li>• block styles <ul style="list-style-type: none"> <li>◦ address</li> </ul> </li> <li>• inline styles <ul style="list-style-type: none"> <li>◦ cited work</li> <li>◦ inline quotation</li> </ul> </li> </ul>
paragraph format	<ul style="list-style-type: none"> <li>• normal</li> <li>• h1</li> <li>• h2</li> <li>• h3</li> <li>• h4</li> </ul>
list	<ul style="list-style-type: none"> <li>• insert/remove numbered list</li> <li>• insert/remove bulleted list</li> </ul>
layout	<ul style="list-style-type: none"> <li>• decrease indent</li> <li>• increase indent</li> </ul>

## Supported HTML Tags

The following table lists all the supported HTML Tags for this component.

HTML Tag	Description
abbr	abbreviation
acronym	acronym
address	format as address
blockquote	blockquote
br	line break
cite	cite
dd	definition item
dfn	definition
div	div
dl	definition list
dt	definition topic
em	emphasis
h1	header level 1
h2	header level 2
h3	header level 3
h4	header level 4

HTML Tag	Description
kbd	text to be entered via keyboard
li	list item for "ol" and "ul"
code	inline code
ol	ordered list (numbered list)
p	section symbol
pre	pre-formatted text
q	quotation
samp	sample(block)
span	span
strong	strong
ul	unordered list
var	text that is variable

#### **i** Note

- Header tags must be on the top level.
- If styled accordingly in your chosen theme, you can either use `<em>` for italic font and `<strong>` for bold font. Alternatively, use a span tag with a class carrying the desired styling.
- The attributes `class` and `id` can be used with the following tags:
  - `div`
  - `span`

For more information on [Formatted Text View](#), see the following SAPUI5 documentation. [SAPUI5 documentation](#)

## Related Information

[Get HTML Text \(getHTMLText\) \[page 634\]](#)

[Set HTML Text \(setHTMLText\) \[page 635\]](#)

## 36.7.9 Fragment Gallery

The [Fragment Gallery](#) is a basic component that is used to store portable fragment bookmarks for the user while they are working with them online to create customized dashboards. The [Fragment Gallery](#) along with the [Split Cell](#), plays an integral role in the online composition feature. For more information on using the [Fragment Gallery](#), you can refer to the chapter called [Using the Online Composition Feature](#).

Property Type	Property Value	Property Description
item dimension	default value: 100 pixels	Specifies the base value in pixels from which the width or height of the items in the component is calculated. When the orientation is horizontal, the width is based on this value. When vertical, the height is based on this value.
display mode	image; image/text; text	Specifies the display mode of the items within the component. There are 3 modes for each item: Text, Image, or ImageText
orientation	horizontal; vertical	Specifies the orientation of the component. There are two orientation modes: horizontal and vertical.

## Related Information

[Working with Fragment Gallery Components \[page 153\]](#)

[Working with Online Composition \[page 150\]](#)

[Working with Fragment Gallery Components \[page 153\]](#)

[Working with Split Cell Containers \[page 155\]](#)

[Using the SAPUI5 m Library \[page 45\]](#)

## 36.7.10 Icon

The *Icon* component displays icons. The icons can either be chosen from a set of built-in icons delivered by SAP or from a custom font in TTF or OTF format.

The *Icon* component has the following specific properties:

Table 79:

Property	Property Value	Description
CSS class	Your CSS class	Defines an additional CSS class for custom CSS. The CSS classes must have the format <b>myclass</b> (and not <b>.myclass</b> ). You do not have to use this property to use <i>Custom CSS</i> .

Property	Property Value	Description
custom font	font	Specifies the custom font. With the file picker you can specify the font file in the <code>.ttf</code> or <code>.otf</code> format. Ensure that the font file's <i>Font embeddability</i> property has been set to <i>Installable</i> , otherwise the icons cannot be rendered in Internet Explorer. This property can be found in the <i>Details</i> tab of the files properties using Windows™ Explorer. The corresponding font icons are displayed on the <i>Additional Properties</i> sheet on the <i>Custom Font</i> tab in design studio and are available for selection.
color	color	Specifies the color of the icon itself.
background color	color	Specifies color of the icon background.
background shape	Rectangle, Ellipsis, None	Specifies the shape of the icon background. The shapes are <i>RECTANGLE</i> , <i>ELLIPSIS</i> or <i>NONE</i> (transparent).
size factor	numeric value	Specifies the size of the icon compared to the bounding rectangle (valid range from 0.0 to 1.0).
icon URI	URI	Specifies the icon URI. The icon URI must have the following structure <b>sap-icon://name</b> or <b>sap-icon://collection/name</b>
tooltip	text	Specifies the tooltip that is displayed when the user hovers over the component at runtime. Type in the text that should be displayed as a tooltip.



Property	Property Value	Description
on click	script	<p>Opens the script editor. With this property, you can enable user interaction with the icon component by writing scripts. The On Click event is triggered when the application user clicks on the icon. In the script editor, press <b>CTRL</b> + <b>Space</b> to see the list of available methods for the application, the data source alias, the icon and other components of the application. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the method descriptions in the API reference.</p>

## 36.7.11 Image

Using the image component, you can enhance applications by adding images.

The *Image* component has the following specific properties:

Property Type	Property Value	Property Description
image	image location/name	<p>Specifies the image that is displayed.</p> <p>There are several possible locations where you can store the images and icons that you want to use in your application:</p> <ul style="list-style-type: none"> <li>in the application directory  <code>&lt;userhome&gt;\Analysis-workspace\com.sap.ip.bi.zen\repository\MyApplication\MyImage.jpg</code>  Type in the image name, for example, <b>MyImage.jpg</b>.</li> <li>in a subfolder of the application directory, for example,  <code>&lt;userhome&gt;\Analysis-workspace</code></li> </ul>

Property Type	Property Value	Property Description
		<p>\com.sap.ip.bi.zen \repository \MyApplication \MySubFolder\MyImage.jpg Type in the image name, for example, <b>MyImage.jpg</b>.</p> <ul style="list-style-type: none"> <li>• Internet or intranet</li> <li>• If your Design Studio is connected to a SAP NetWeaver platform, click on the <a href="#">Browse</a> button that is displayed. The <a href="#">Open Image</a> dialog box opens where you can select the image or icon from the MIME repository.</li> </ul> <div> <p><b>i Note</b></p> <p>In the MIME repository, you can select images and icons uploaded by the system administrator. For more information, see the <i>Administrator Guide: SAP BusinessObjects Design Studio</i> at <a href="http://help.sap.com/boad">http://help.sap.com/boad</a></p> </div>
hover image	image location/name	Specifies the image that is displayed when the user hovers with the mouse over the component. Type in the image name, for example, <b>MyImage.jpg</b> or use the <a href="#">Open Image</a> dialog box to select the image or icon from the MIME repository (if your Design Studio is connected to SAP NetWeaver).
click image	image location/name	Specifies the image that is displayed when the user clicks the image. Type in the image name, for example, <b>MyImage.jpg</b> or use the <a href="#">Open Image</a> dialog box to select the image or icon from the MIME repository if your Design Studio is connected to SAP NetWeaver.
opacity	Opacity percentage value (0 - 100: 0 = transparent, 100 = solid).	Specifies the image opacity.
on click	none	Opens the script editor. With this property/event, you can enable user interaction with the image by writing scripts. Click <b>CTRL</b> + <b>Space</b> to see the list of available methods for the

Property Type	Property Value	Property Description
		<p>application, the data source alias and the image. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the relevant method description in the API reference.</p>

## 36.7.12 Input Field

The input field enables the user to write user-specific content in the input field at runtime.

The *Input Field* component has the following specific properties:

Property Type	Property Value	Property Description
value	string	Specifies the initial value that the application user can overwrite during runtime.
on change	none	<p>Opens the script editor. With this property/event, you can enable user interaction with the input field by writing scripts. Click <b>CTRL</b> + <b>Space</b> to see the list of available methods for the application, the data source alias and the input field. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the relevant method description in the API reference.</p>

## 36.7.13 List Box

List boxes enable the user to select items, for example, to set a filter. To do this, you need to add a script to the On Select property of the list box. The script is triggered when the user selects an item from the list box.

The *List Box* component has the following specific properties:

Property Type	Property Value	Property Description
items	list of items	Opens the <i>Edit items</i> dialog box. Here, you can insert items, change the order

Property Type	Property Value	Property Description
		<p>of items or remove them. Each item has the following properties:</p> <ul style="list-style-type: none"> <li>value The value property is mandatory for all items.</li> <li>text (optional) The text property is optional. If you enter a text for an item, this text is displayed. If no text is entered, the value of the item is displayed.</li> </ul> <p>If you want a specific item to be displayed by default, select the relevant item and click <a href="#">Set as Default</a>. This item is shown as the default entry in the list box.</p>
multiple selection	false, true	Allows multiple values to be selected. This property can be used to filter data sources by multiple members.
on select	script	<p>Opens the script editor. With this property/event, you can enable user interaction with the listbox by writing scripts. The <a href="#">On Select</a> event is triggered when the application user selects an item from the list box. In the script editor, click <code>CTRL+Space</code> to see the list of available methods for the application, the data source alias and the list box. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the relevant method description in the API reference.</p>

## Additional, changed or removed properties in SAPUI5 m Mode

Due to the different rendering in SAPUI5 m mode, the list box has a different properties sheet in the m mode:

- Property [Multi Selection](#) has been removed.
- Property [Selection Mode](#) has been added.  
This property specifies the behavior as well as the visualizations for the list box. This includes a mode which supports multiple selection.

## 36.7.14 Radio Button Group

Radio button groups enable the user to select items, to set a filter for example. To do this, you need to add a script to the radio button group's On Select property. The script is triggered when the user selects an item from the radio button group.

The *Radio Button Group* component has the following specific properties:

Property Type	Property Value	Property Description
items	list of items	<p>With the items property, you can open the <a href="#">Edit items</a> dialog. Here, you can insert items, change the order of items or remove them. Each item has the following properties:</p> <ul style="list-style-type: none"><li>• value The value property is mandatory for all items.</li><li>• text (optional) The text property is optional. If you enter a text for an item, this text is displayed. If no text is entered, the value for the item is displayed.</li></ul> <p>If you want a specific item to be displayed as default, select the relevant item and click <a href="#">Set as Default</a>. This item is then shown as the default entry in the radio button group.</p>
columns	numeric value for columns, default: 1	<p>Specifies the number of columns for the radio button group. The specified columns are filled by the system with the specified items.</p> <p>Example: You have specified five items and two columns. The items are displayed as follows:</p> <ul style="list-style-type: none"><li>• line 1: item 1 and item 2</li><li>• line 2: item 3 and item 4</li><li>• line 3: item 5</li></ul> <div><p>➔ Remember</p><p>If you specify several columns, you need to enlarge the width of the component. Otherwise the columns will not be fully or correctly displayed.</p></div>
on select	script	<p>Opens the script editor. With this property/event, you can enable user</p>


Property Type	Property Value	Property Description
		<p>interaction with the radio button group by writing scripts. The <a href="#">On Select</a> event is triggered when the application user selects an item in the radio button group. In the script editor, click <code>CTRL</code> <code>+Space</code> to see the list of available methods for the application, the data source alias and the radio button group. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the relevant method description in the API reference.</p>

## 36.7.15 Text

You use the [Text](#) component to add user-defined text to your application. By setting the relevant properties, you can change the style and size of the text.

The [Text](#) component has the following specific properties:

Property	Property Value	Property Description
Text	Some Text	Here you can enter the text that you want to add to your application.
Style	Standard, Heading1, Heading 2, Heading 3, Heading 4	Specifies a predefined style.
CSS Style	YourCssStyle	<p>Defines the CSS style that is applied for this component. Click the browse button to open the <a href="#">CSS Style Editor</a> dialog box and write your CSS style into the white area, for example, <code>color: pink;</code>. You can specify one of the following CSS attributes:</p> <ul style="list-style-type: none"> <li>• font size</li> <li>• font color</li> <li>• weight (bold)</li> <li>• style (italic)</li> <li>• text decoration (underline)</li> </ul> <p>This property provides advanced options for text formatting. You can type in the CSS style code directly. Alternatively, you can use any external</p>

Property	Property Value	Property Description
		<p>CSS style editor (such as cssmate or any other) to generate code for the CSS attributes you want to use. Copy this code into the CSS style editor of the design tool.</p> <div>  <b>Restriction</b> <p>Only the CSS attributes listed above are supported for both desktop browser and iPad/iPhone applications. If you use any other CSS attributes in the external CSS style editor, run the application on your target device to check whether the text is displayed as required and as defined by the attributes.</p> </div>

#### Example

An example of CSS code for formatted text.

```
font-size: 15px; weight: bold; font-style: italic; text-decoration: underline;
color: red;
```

## Additional, changed or removed properties in SAPUI5 m Mode

Due to the different rendering in SAPUI5 m mode, the text component has a different properties sheet in the m mode:

- Property *Style* has been removed.

## 36.7.16 Tree

The *Tree* basic component can be bound to a hierarchical dimension in a data source. If the dimension has no hierarchy, the *Tree* component will display a simple list. If the dimension has a hierarchy, the *Tree* component will reflect the actual result set as it appears in the crosstab. If a change is made within the *Tree* component, the result set in the crosstab and other components bound to the same data source will change accordingly. Similarly, if you change the result set in the crosstab, the *Tree* component will also change. The columns of the *Tree* component reflect the display settings of the dimension. For example, if you change the dimensions display settings from Text and Key to Key in the context menu on the crosstab, the tree will display only one column. Your selection will persist if you select a dimension member in the *Tree* component and then change the order of your dimension

members, for example, by sorting. If you select a child element in the hierarchy and then collapse the parent node, your selection is not persisted.

#### **i** Note

- The [Tree](#) component does not support line breaks when displaying members. It only supports single line texts.
- It does not support hierarchies that open upwards.
- It can be used when assigning bookmarks to folders in the online composition scenario.

Property	Property Value	Property Description
display column headers	false,true	Specifies whether column headers are displayed in the <a href="#">Tree</a> component. The default value is false.
tree items	bound, not bound	Specifies whether the component is bound to a data source.
(source) type	hierarchical dimension members binding	Specifies the binding type.
(source) dimension	dropdown list	Specifies the dimension on the data source bound to the component.
(source) maximum number of entries	integer	Specifies how many entries are displayed. The default value is 100.
selection mode	single, multi	Specifies how many members can be selected.

## Related Information

[Adding a Data Source \[page 68\]](#)

[Binding the Properties of Standard Components to Data Sources \[page 98\]](#)

[Assigning Bookmarks to Folders \[page 173\]](#)

[Working with Online Composition \[page 150\]](#)

## 36.8 Container Components

### 36.8.1 Grid Layout

You use the grid layout component to group and order the content of your application in a grid that is not displayed at runtime.

The [Grid Layout](#) component has the following specific properties:




Property Type	Property Value	Property Description
number of rows	numeric value, default: 1	Specifies the number of rows in the grid.
number of columns	numeric value, default: 1	Specifies the number of columns in the grid.

## 36.8.2 Pagebook

The pagebook component enables the user to switch between the different views of an application. You group the components or contents of a view on a separate page. For example, you want to show different data on different pages or you want to display the data in different ways (in a chart on one page and in a crosstab on another page). The pagebook only shows one page at time.

### Note

- The pages are shown in the [Outline](#) view .
- If you drag and drop a pagebook component in the layout editor, the system automatically creates two pages for the pagebook. If you want to add further pages to the pagebook, use the context menu of the pagebook in the [Outline](#) view and click .

The [Pagebook](#) component has the following specific properties:

Property	Property Value	Property Description
selected page index	index	Specifies the visible page of the pagebook. The index 0 represents the first page of the pagebook, 1 represents the second page, 2 represents the third page, etc.
transition effect	none, slide in, fade, flip, cube	Specifies the transition effect when the user switches between the pages of the pagebook.
transition direction	horizontal, vertical	Specifies the transition direction when the user switches between the pages of the pagebook.
page caching	none, all, adjacent	<p>Specifies the caching behavior for the pagebook component.</p> <ul style="list-style-type: none"> <li>• If you set the property to <a href="#">none</a>, no pages are cached. This property setting is recommended for applications with high data volume.</li> <li>• If you set the property to <a href="#">all</a>, every page of the pagebook is cached. This property setting is useful when the data volume is low.</li> <li>• If you set the property to <a href="#">adjacent</a>, the previous page and next page are cached.</li> </ul>

Property	Property Value	Property Description
show page indicator	true, false	<p>Specifies whether a page indicator is displayed in the pagebook component. The page indicator displays how many pages the pagebook consists of and indicates which page is active. With the page indicator, you can also change between pages by clicking the indicators.</p> <div> <b>i Note</b> <p>On the iPad, you can only change to the page right or left of the selected page. It is not possible, for example, to change from page two to page five. If you are on page two, you can only change to page one or three.</p> </div>
enable swiping	true, false	<p>Specifies whether the application user can swipe the pages of the pagebook at runtime. If <i>Enable Swiping</i> is set to <i>true</i> and <i>Transition Effect</i> is set to <i>None</i>, the application user can swipe the pages but without any transition effect.</p>
on select	none	<p>Opens the script editor. With this property/event, you can enable user interaction with the pagebook by writing scripts. Click <b>CTRL+Space</b> to see the list of available methods for the application, the data source alias and the pagebook. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the relevant method description in the API reference.</p>

## Additional, changed or removed properties in SAPUI5 m Mode

Due to the different rendering in SAPUI5 m mode, the filter panel has a different properties sheet in the m mode:

- Property *Transition Effect* has been removed.
- Property *Transition Direction* has been removed.
- Property *Enable Swiping* has been removed.

## 36.8.3 Panel

You use the panel component to group other components in your application. The panel component is a very simple container component.

The *Panel* component has the following specific properties:

Table 80:

Property	Property Value	Property Description
CSS style	CSS style	<p>Defines the CSS style that is applied to the component. Click the browse button to open the <i>CSS Style Editor</i> dialog box and write your CSS style into the white area, for example, <code>color: pink;</code>. You can specify one of the following CSS attributes:</p> <ul style="list-style-type: none"><li>• font size</li><li>• font color</li><li>• weight (bold)</li><li>• style (italic)</li><li>• text decoration (underline)</li></ul> <p>This property provides advanced options for text formatting. You can type in the CSS style code directly. Alternatively, you can use any external CSS style editor (such as <i>cssmate</i>) to generate code for the CSS attributes you want to use. Copy this code into the CSS style editor of the design tool.</p> <div><p><b>Note</b></p><p>Restriction: Only the CSS attributes listed above are supported for both desktop browser and iPad/iPhone applications. If you use any other CSS attributes in the external CSS style editor, run the application on your target device, to check whether the text is displayed as required and as defined by the attributes.</p></div>

Property	Property Value	Property Description
on click	none	<p>Opens the script editor. With this property/event, you can enable user interaction with the panel by writing scripts. Click <code>CTRL +Space</code> to see the list of available methods for the application, the data source alias and the panel. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the relevant method description in the API reference.</p>

## 36.8.4 Popup

### Syntax

The popup component helps users to quickly enter information, perform configurations or make selections. In addition, popups can also be useful for displaying more specific data for a selected item displayed on the main page of the application. As the popup is a container component, you can put any other component into the popup (for example, crosstab, button or checkbox).

### Restriction

- The popup can only be nested in the root layout and not within another container component.
- The popup can only be positioned relative to the root layout.

The popup component is initially invisible. It is displayed (or closed) when the user interacts in the application either with another visible component (for example, a button) or a script (for example, at startup).

The *Popup* component has the following specific properties:

Property	Property Value	Property Description
modal	true, false	Specifies whether the popup is modal. If the popup is set to modal, the user can only navigate and perform interactions within the popup. Otherwise the user can also interact within the whole application.
animation	no animation, flip animation, pop animation, horizontal slide animation, vertical slide animation	Specifies the animation effect when the popup is displayed or closed.
autoclose	true, false	Specifies whether the component is automatically closed, when the user

Property	Property Value	Property Description
		interacts outside the popup but inside the application.

## 36.8.5 Tabstrip

You use the tabstrip component to group and order the content of your application in tabs. You can also use it to enable user interaction within the application. To do this, you add a script to the tabstrip's On select property. The script is triggered when the user selects one of the tabs on the tabstrip in the application.

The *Tabstrip* component has the following specific properties:

Property Type	Property Value	Property Description
selected tab index	index	Specifies the initially displayed tab of the tabstrip. The index 0 represents the first tab of the tabstrip, 1 represents the second tab, 2 represents the third tab etc.
on select	script	<p>Opens the script editor. With this property/event, you can enable user interaction with the tabstrip by writing scripts. The <i>On Select</i> event is triggered when the application user selects one tab. In the script editor, click <b>CTRL</b> <b>+Space</b> to see the list of available methods for the application, the data source alias and the tabstrip. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the relevant method description in the API reference.</p>

## 36.8.6 Split Cell

You use the *Split Cell* container component to enable the user to place and arrange portable fragment bookmarks, dragged from the fragment gallery, in a tabular format. Thus the split cell container can only be used with the *Fragment Gallery* component and the portable fragment bookmarks which can be created by using the corresponding API methods.

The *Split Cell* container component has the following specific properties:

## Note

When portable fragment bookmarks are inserted into a split cell container, the components contained in them may be renamed. In general, script references will be adapted to the new names. However, in case the name is given as a string, this renaming will not take place and could result in unexpected behaviour.

- working example: `PAGEBOOK_1.setSelectedPageIndex(0)`
- non-working example: `PAGEBOOK_1.setSelectedPageByName("PAGE_1")`

Property Type	Property Value	Property Description
on drop	script	<p>Opens the script editor. With this property/event, you can enable user interaction with the splitt cell by writing scripts. The <i>On Drop</i> event is triggered after a drop from the Fragment Gallery has been made. It is not triggered after moving or resizing cells. In the script editor, click <code>CTRL</code> + <code>Space</code> to see the list of available methods for the application, the data source alias and the splitt cell. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the relevant method description in the API reference.</p>
on delete	script	<p>Opens the script editor. With this property/event, you can enable user interaction with the splitt cell by writing scripts. The <i>On Delete</i> event is triggered when the application user deletes a cell. It is not triggered after moving or resizing cells. In the script editor, click <code>CTRL</code> + <code>Space</code> to see the list of available methods for the application, the data source alias and the splitt cell. Choose one of them or use the statement wizard. The wizard guides you through the statement creation process.</p> <p>For further information, read the relevant method description in the API reference.</p>

## Related Information

[Working with Split Cell Containers \[page 155\]](#)

## 37 API Reference

SAP BusinessObjects Design Studio is a design tool to create interactive analysis applications. To enable interactivity, you write scripts that are executed when the user performs an action in the executed application. For example, you can place the *button* component in the application and assign a script to the button's *On Click* event.

Scripts consist of one or more statements written in a JavaScript-based language that follow a specific syntax. For writing scripts the design tool offers two approaches: the statement wizard and the script editor.

All objects, methods and expression types are listed in this API reference. The methods are grouped by the object types they refer to. These object types are:

- data source alias (DataSourceAlias)
- application (Application)
- component (Component)
- single components like button, text, image (for example, Button, Text, Image)
- information objects (ApplicationInfo, DataCell, Convert)

### Script language

You write scripts in the BI Action Language (BIAL). BIAL is a true subset of the JavaScript/ECMAScript standard described here: <http://www.ecma-international.org/publications/files/ECMA-ST/Ecma-262.pdf>. The scripts are executed on the Analysis Application Design Service (unlike JavaScript that is executed in the Web browser). Scripts have a clear definition of the supported language constructs, objects and methods.

### Scripts, statements and events

Scripts consist of statements. Each statement is typically written in a single line. The execution of scripts is triggered by user interaction with the component. This interaction is also referred to as an event and the script executed in response to this event is the event handler. Each component has its own set of one or more events that are displayed in the *Properties* view of the component.

#### ➔ Remember

Events on components are only triggered by user interaction and not implicitly by scripting. For example, when changing the selection in a list box by using the script `LB.setSelectedItem()`, the *On Select* event of the list box is **not** triggered. An exception to this rule is the *On Result Set Changed* event, which can be triggered implicitly by a script method like `setFilter()` or `reloadData()`.

SAP BusinessObjects Design Studio currently supports the following statement types:

- call statements
- conditional execution statements

- assignment statements

## Syntax of call statements

Call statements execute an API method of an object and have the following format:

```
<Component>.<method>(<arguments>);
```

<Component> is the name of a data source alias or a component in your application, for example, DS\_1 or Button\_1.

<method> is an operation that is applied to the object specified on the left of the period. The available methods depend on component type. DataSourceAlias objects, for example, provide methods to filter data, and visual components provide methods to modify visibility, enablement and so on.

<arguments> is a comma-separated list of expressions. The passed expressions must match the requirements of the method.

Each statement ends with a semicolon (";").

You can use methods as arguments for other methods as long as the return type matches the argument type.

## Syntax of conditional execution statements

Conditional execution statements have one of the following formats:

- first format

```
if (<condition>) {  
  <sequence of statements to execute when condition is met>  
}
```

- second format

```
if (<condition>) {  
  <sequence of statements to execute when condition is met>  
} else {  
  <sequence of statements to execute when condition is NOT met>  
}
```

<condition> is a Boolean expression - one of the following:

- true or false as value literals (constants)
- a method call statement returning a Boolean value
- a comparison for equality in the form a == b or
- a comparison for inequality in the form a != b
- multiple conditions joined using && and ||, parentheses are optional



## Assignment statements

Assignment statements have one of the following formats:

- `var <variable> = <expression>;`  
This format defines a script variable (see Script Variables in this chapter) and assigns the result of an expression to this script variable.
- `<variable> = <expression>;`  
This format assigns the result of an expression to a script variable, which has been already defined.

## Methods and object types

Methods are operations that are applied to the object specified on the left of the period in the statement. The available methods depend on the object type. For example, `DataSourceAlias` objects provide methods to filter data. Visual components have methods to modify visibility, enablement and so on. SAP BusinessObjects Design Studio supports methods that depend on the following object types:

- data source alias (`DataSourceAlias`)
- application (`Application`)
- component (`Component`)
- individual components like button, text, image (for example, `Button`, `Text`, `Image`)
- information objects (`ApplicationInfo`, `DataCell`, `DataSourceInfo`)

## Expressions

Expressions compute a result (also referred to as returned value) in a similar way to a formula. Expressions can appear as method arguments and as conditions ("if" statements or conditional execution statements). Results of an expression have a type. An expression may consist of literals (String constants, integer numbers, Arrays, JSONs), operators, method calls and parentheses.

Operators supported by BIAL :

Operator	Description	Argument Type	Result Type	Example
+	Concatenates strings	String, (Integer, Float, Boolean) (Boolean and Integer will be converted to String automatically)	String	"ab"+"cd" (= "abcd") "ab"+1 (= "ab1")
+	Adds two integer values or floating point	Integer, Float	Integer	1+2 (=3)
-	Subtracts two integer values or floating point	Integer, Float	Integer	3-2 (=1)
*	Multiplies two integer values or floating point	Integer, Float	Integer, Float	3*2 (=6)

Operator	Description	Argument Type	Result Type	Example
/	Divides one integer value by the other or one floating point by the other	Integer, Float	Integer, Float	8/2 (=4)
==	Checks if the two operands are equal	Any	Boolean	1 == 1 ( = true) "a" == "b" (=false)
!=	Checks if the two operands are not equal	Any	Boolean	1 != 2 ( = true) "a" != "a" (=false)
&&	Logical AND	Boolean	Boolean	true && false (=false) true && true (=true) if (<condition1> && <condition2>) { <statements> } Statements will be executed if both conditions are true.
	Logical OR	Boolean	Boolean	true    false (=true) false    false (=false) if (<condition1>    <condition2>) { <statements> } Statements will be executed if at least one of the conditions is true.
!	Logical NOT	Boolean	Boolean	!true (=false) !false (=true) If (! <condition>) { <statements> }

Operator	Description	Argument Type	Result Type	Example
				Statements will be executed if condition is not true.

## Expression type system

Expression results can be used as method arguments. The expression type must match the argument type. In some cases, BIAL will automatically convert between types. This typically only happens for certain strings, where BIAL can check that the string value is valid (see below). The +Operator automatically converts Integer and Boolean arguments to String. In all other cases, the type system is strict and error messages are displayed if there is a mismatch.

There are four different types:

- primitive types (String, Integer, Boolean, Float)

Type	Example
String	"Hello"
Integer	123
Float	123
Boolean	true, false
String Array	["A", "B"]
Integer Array	[1, 2]
JSON	{"key": "value"}

- BI types (DataSourceAlias, Dimension, Measure, ...) enable context-relevant input help for API method arguments in the script editor. These special types are sometimes written in Enum, Array, or JSON format (or even a combination of those ). To view examples, refer to the API reference.
- component types (Button, Text, Pagebook...)

The visual component types correspond to the list of components in the [Components](#) view of the design tool

## Enums

In many cases, the input for a function can be one of a fixed set of values. This set of values is referred to as an enumeration or simply "Enum". Enums are BI types. An Enum value is written as <EnumType>.<EnumValue>, for example, "ChartType.PIE".

## Script Variables

Script variables store expression results. They are useful, for example, for storing intermediate results that are used repeatedly in a script. Script variables have a name composed of the characters "A"-"Z", "a"-"z", "0"-"9", and "\_". The name cannot begin with any of the digits "0"-"9". Like expressions, script variables have a type. There are local and global script variables:

- Local script variables

Local script variables are script variables that can only be used in the script in which they were defined, and not in any other scripts. To define a local script variable, open a script with the [Script Editor](#) and add a line with the following format:

```
var <variable> = <expression>;
```

The type of the script variable is automatically determined by the type of the expression.

- Global script variables

Global script variables are script variables that can be used in any script of your application. To define a global script variable click the Application in the [Outline](#) view, then click in the [Property](#) sheet the item [Scripting](#) [Global Script Variables](#). Click the button .... This opens the [Edit Global Script Variable](#) dialog box. You can insert a new global script variable and define its name, type, and default value. In addition you can specify whether the global script variable is a URL parameter. If you make it a URL parameter, then you can set the value of this global script variable by adding the global script variable and its URL parameter value to the applications URL.

### Note

Script variable names used as a URL parameter must start with a capital x (**x**) and must not end with an underscore and a digit (`_<digit>`).

## Methods calling the event handler

To avoid event handlers calling each other infinitely which could result in runtime crashes, events on components are only triggered by user interaction on the component and not by scripting. For example, if a user swipes the pages of a pagebook, the On Select event is raised, and the relevant event script is executed. In contrast, if the selected page index is set by script using the method `setSelectedPageIndex` of the pagebook, the On Select event is not raised. Another example is, if a user selects a value from a dropdown box, the On Select event is raised and the event script is executed. In contrast, if the selected item is set using the `setSelectedValue` method of the dropdown box, the event is not raised. However there are situations when you want to execute the relevant event script without duplicating script code. In these cases, each component with an event (On Click, On Select) has a corresponding method that allows the event handler to be called from another handler's code. For example, the statement `BUTTON_1.onClick()`; calls the [On Click](#) event handler of the button component with the name `BUTTON_1`. Another example is the statement `TABSTRIP_1.onSelect()`; that calls the [On Select](#) event handler of the tabstrip component with the name `TABSTRIP_1`.

## Example

You have created an application with a dropdown box DROPDOWN\_1 and a button BUTTON\_1. If the user selects a value in the dropdown box, the selected value filters the dimension MYDIM in the data source aliases DS\_1 and DS\_2. For this scenario, the *On Select* event script of the dropdown box looks like this:

```
DS_1.setFilter("MYDIM", DROPDOWN_1.getSelectedValue());
```

```
DS_2.setFilter("MYDIM", DROPDOWN_1.getSelectedValue());
```

If the user pushes the button, a specific value in the dropdown box should be selected programmatically using the same logic defined for user interaction. In other words, the selected value filters the dimension MYDIM in the data source aliases DS\_1 and DS\_2. Instead of duplicating the script code of the *On Select* event of DROPDOWN\_1, you add the following statements in the *On Click* event handler of the button:

```
DROPDOWN_1.setSelectedValue("MYDIMVALUE");
```

```
DROPDOWN_1.onSelect();
```

The benefits of calling the `onSelect` method increase as you start to work with more use cases in the application (where you want to set the selected value programmatically) and more statements in the *On Select* event handler of the dropdown box.

## Member key format

The most frequently used API methods deal with dimension members in the context of setting filters or variable values. Often the filter or variable values to set are returned from the selection of a UI component.

In SAP Business Information Warehouse (BW) there are multiple key types; the most important are the INTERNAL and the EXTERNAL key formats. Whereas the INTERNAL key is a unique identifier for all users (user-locale independent), the EXTERNAL key can be user-locale dependent.

By default, all methods dealing with member keys as returned values or as parameters, work with the INTERNAL key format. This keeps applications language-independent to ensure, that they can work for multilingual user groups. However, you might want to choose the EXTERNAL key format due to various situations or reasons:

- The EXTERNAL key format is more compact and either the application is known to be used only by users sharing the same locale, or the EXTERNAL key format is known to be locale-independent.
- The number of complex selections is more than the single values that need to be passed as parameters for the API methods `setFilter` and/or `setVariableValue`. In this case, the SAP Business Explorer selection syntax ("INPUT\_STRING") can be used, which references individual members by their EXTERNAL key.

For these special use cases, SAP BusinessObjects Design Studio provides Ext variants of methods, for example `setFilterExt`, `setVariableValueExt`.

## 37.1 Action Sheet

### 37.1.1 Add Item (addItem)

Adds an item to the component.

#### Parameters

Name	Type	Description
value	String	Unique key (or technical value) of the item. The method call is ignored if an item with this key already exists in the component.
(optional) text	String	Display text of the item.
(optional) icon	Url	Path of image file to use.
(optional) index	Integer	Index where to insert the item. If the index is -1 or greater than the number of items then the item is added after the last item (default: -1).

#### Returned Value

None

### 37.1.2 Get Selected Text (getSelectedText)

Returns the selected items text.

#### Parameters

None

## Returned Value

String. It contains the selected text.

### 37.1.3 Get Selected Value (getSelectedValue)

Returns the selected items value.

## Parameters

None

## Returned Value

String. It contains the selected value.

### 37.1.4 Open (open)

Opens the Action Sheet next to another component.

## Parameters

Name	Type	Description
component	Component	Specifies the component next to which the Action Sheet will open. The component must be currently visible.

## Returned Value

None

---

## 37.1.5 Remove All Items (removeAllItems)

Removes all items from the component.

### Parameters

None

### Returned Value

None

## 37.1.6 Remove Item (removeItem)

Removes an item from the component.

### Parameters

Name	Type	Description
value	String	Unique key (or technical value) of the item. The method call is ignored if no such key exists in the component.

### Returned Value

None

## 37.2 Application (Application)

### 37.2.1 Alert (alert)

Opens a message dialog on the computer where the design tool is installed.



### **i** Note

This method is intended for debugging a locally executed application. It works if the application is executed locally, but will be ignored if the application is executed directly on the supported platforms.

## Parameters

Name	Type	Description
message	String	Message text

## Returned value

None

## 37.2.2 Create Error Message (createErrorMessage)

Creates an error message that is displayed in the message view.

## Parameters

Name	Type	Description
message	String	Message text

## Returned value

None

## 37.2.3 Create Info Message (createInfoMessage)

Creates an info message that is displayed in the message view.

## Parameters

Name	Type	Description
message	String	Message text

## Returned value

None

## 37.2.4 Create Warning Message (createWarningMessage)

Creates a warning message that is displayed in the message view.

## Parameters

Name	Type	Description
message	String	Message text

## Returned value

None

## 37.2.5 Do Background Processing (doBackgroundProcessing)

Triggers the script written in the [On Background Processing](#) property of the application. This script is executed in the background when the UI is rendered.

## Parameters

None

## Returned value

None

## 37.2.6 Export (export)

Exports selected components. Currently only the Crosstab component is supported.

### Parameters

Name	Type	Description
exportType	ExportType	Type of export format
(optional) crosstab	Array of Crosstab	Type of export component (Currently only the Crosstab component is supported)
(optional) exportSettings	ExportSettings	Setting for export output

## Returned Value

None

## 37.2.7 Get Info (getInfo)

Returns analysis application information, for example, the current date or the name of the analysis application.

### Parameters

None

## Returned value

Returns object providing information about the application.

## 37.2.8 Get Resource String (getResourceString)

Returns a resource string using its id.

### Parameters

Name	Type	Description
id	String	ID of the resource string

### Returned value

String. It contains the resource string.

## 37.2.9 Get Tick Count (getTickCount)

Returns a point in time in milliseconds. This method is useful for performance measurements.

### Parameters

None

### Returned value

Integer

#### Example

In the following example, the time for executing method `setFilter` is displayed:

```
var startTime = APPLICATION.getTickCount();
DS_MYDATA.setFilter(...);
var endTime = APPLICATION.getTickCount();
APPLICATION.alert("Elapsed time: " + (endTime - startTime) + " ms");
```

## 37.2.10 Get User Agent (getUserAgent)

Returns the user agent of the browser executing the Design Studio script methods.

### Parameters

None

### Returned value

String. It contains the user agent.

## 37.2.11 Load Data Sources (loadDataSources)

Loads the specified data sources.

This method is equivalent to `loadDataSource` on a data source but allows loading several data sources at once.

This method respects the [Processing Group](#) property of data sources: Data sources can be loaded in parallel if contained in such a group. Therefore it might be faster to call this method instead of calling `loadDataSource` on each data source.

### Parameters

Name	Type	Description
dataSourceAliases	Array of DataSourceAlias	Data sources to load

### Returned value

None

## 37.2.12 Log (log)

Creates a message in the design tool's [Error Log](#) view for script error analysis.

## Parameters

Name	Type	Description
message	String	Message text

## Returned value

None

### 37.2.13 Open New Window (openNewWindow)

Opens a new browser window with the specified URL.

If you want to open an external domain, you have to specify the protocol explicitly, for example `http://www.sap.com`. If you do not specify the protocol, the window will open the URL on the same domain as the invoking Analysis Application.

## Parameters

Name	Type	Description
newUrl	String	URL

#### Restriction

This method does not open in a new window with SAP BusinessObjects Mobile 4.1 (navigation continues in the same window).

## Returned value

None

#### Example

##### Using an external domain

In this example, the Web page `sap.com` is opened in a new browser window:

```
APPLICATION.openNewWindow("http://www.sap.com");
```

## 37.2.14 Open Prompt Dialog Box (openPromptDialog)

Opens prompt dialog box.

### Note

Be aware that the `openPromptDialog` method must be the **last** method in the script that uses this method.

## Parameters

Table 81:

Name	Type	Description
width	Integer	Width of the prompt dialog box in pixels
height	Integer	Height of the prompt dialog box in pixels

## Returned value

None

### Example

In the following example, the prompt dialog box is opened:

```
APPLICATION.openPromptDialog(400, 500);
```

## 37.2.15 Set Variable Value (setVariableValue)

Sets data source variable values in the internal key format and executes the data source query again. Variable values are set in the application for all data source aliases containing the variable.

Variable values are set in the application for all data source aliases containing the variable. With this method you can set only single members or hierarchy nodes in the internal key format as variable values.

With this method you can set variable values of input-enabled characteristic value variables (only single values) or hierarchy node variables. If you want to set variable values for the following cases, use

`setVariableValueExt` instead:


- formula variables
- text variables
- hierarchy variables
- variables representing precalculated values sets (buckets)

- multiple single values
- intervals
- variables representing a selection option
- using external format

For hierarchy nodes, prefix the internal non-compounded presentation of the hierarchy node with HIERARCHY\_NODE/NodeType/, for example HIERARCHY\_NODE/OHIER\_NODE/ROOT.

## Parameters

Name	Type	Description
variable	Variable	Data source variable to set
value	VariableValue	Variable value to set in internal key format


**Restriction**


It is not possible to use the crosstab and chart method `getSelectedMember` for setting variables for compounded characteristics.

## Returned value

None

## Restrictions

It is not possible to use the Crosstab component and the Chart component method `getSelectedMember` for setting variables for compounded dimensions.


**Example**

In the following example a fixed string value is set to a variable:

```
APPLICATION.setVariableValue("0VAR", "4711");
```

In the following example the value is set to a variable returned from a component selection:

```
APPLICATION.setVariableValue("0VAR",
DROPDOWN_1.getSelectedValue());
```



---

## 37.2.16 Print (print)

Prints an application.

### Parameters

None

### Returned value

None

## 37.2.17 Search Data Sources (searchDataSources)

Searches for all data sources matching a given pattern.

### Parameters

Name	Type	Description
dataSourceConnection	DataSourceConnection	Name of data source connection
searchFor	SearchExpression	Pattern to search for

### Returned value

Array of DataSourceDescriptor. It contains information about the matching data sources. None

## 37.2.18 Set Variable Value Ext (setVariableValueExt)

Sets data source variable values in external key format, then executes the data source query again.

Variable values are set in the application for all data source aliases containing the variable. Use this method if other selections than single members or hierarchy nodes need to be set as variable value.

## **i** Note

External key values might be dependent on the user's regional settings. If the application is intended to be used by a multi-lingual group of users, it is therefore recommended to use the `setVariableValue` method when possible.

## Parameters

Name	Type	Description
variable	Variable	Data source variable to set
value	InputString	Variable value to set in external key format

## Syntax for entering values

You can specify the input string as variable value. This syntax allows you to specify more complex selections (such as intervals and excluding selections). In contrast to the method `setVariableValue`, the members are specified in the external format (for example, 001.2012).

Description	Example
equal to	15
multiple	15;18;20
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
less than	<12
exclude values less than <value limit>	!<12
less than or equal to	<=12
exclude values less than or equal to <value limit>	!<=12
complex combination	15;10-15;20-25;1-5;>8; etc.
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)

## Returned value

None

### Example

In the following example a fixed string value is set to a variable:

```
APPLICATION.setVariableValue("OVAR", "4711");
```

In the following example the value returned from a component selection is set to a variable:

```
APPLICATION.setVariableValue("OVAR", DROPDOWN_1.getSelectedValue());
```

## 37.3 ApplicationInfo

An object providing information about the application.

### **dateNow**

Current date in the user's locale.

### **dateNowInternalFormat**

Current date in the format "YYYYMMDD".

### **isRightToLeft**

Specifies whether the application is in right-to-left (RTL) mode.

### **name**

Name of the application.

## 37.4 Array

The object Array has the following sub types:

- DataSourceConnectionArray
- DataSourceDescriptorArray
- DimensionArray
- HierarchyArray
- MemberArray
- MeasureFilterIdArray
- StringArray
- VariableArray
- ComponentArray
- CrosstabArray
- DataSourceAliasArray
- DataSourceDescriptorArray
- DimensionArray
- FilterArray
- FragmentBookmarkArray
- HierarchyArray
- KeyValueArray
- ListValueArray
- MemberArray
- PanelArray
- SingleMemberFilter
- StringArray
- BooleanArray
- floatArray
- intArray

### 37.4.1 Accessing Array Elements

You can access array elements of an array by index. Indexes start with 0. If the index is outside the proper range of the array then a value of undefined is returned.

#### Example

In the following example, the third array element of an array of members is retrieved and its external key value is displayed:

```
var array = DS_1.getMembers("0D_CUSTOMER", 100);  
APPLICATION.alert("Element with index " + 2 + " is \"" + array[2] + "\".");
```

## 37.4.2 For Each (forEach)

Iterates through the elements of an array.

### Parameters

Name	Type	Description
callback	Function	The function that is executed with each iteration.

### Returned value

None

#### Example

In the following example, the final value of sum is 6:

```
var sum = 0; var array = [1, 2, 3]; array.forEach(function(element, index) { sum
= sum +
    element; });
```

## 37.4.3 Length (length)

The array's number of elements.

## 37.4.4 Pop (pop)

Removes the last element from the array and returns it.

### Parameters

None

## Returned Value

T. It contains the removed element of the array.

### Example

In the following example, the last value of the array (30) is removed from the array and stored in `lastValue`:

```
var array = [10, 20, 30];  
var lastValue = array.pop();
```

## 37.4.5 Push (push)

Adds an element to the end of the array and returns the array's new number of elements.

### Parameters

Name	Type	Description
element	T	Element to add

## Returned Value

Integer. It contains the array's new number of elements.

### Example

In the following example, the value 30 is added to the array:

```
var array = [10, 20];  
array.push(30);
```

## 37.5 AttributeMember

Provides access to the representations of an attribute member.

---

## **externalKey**

The member's representation as external key.

## **externalNoncompoundedKey**

The member's representation as external non-compounded key.

## **internalKey**

The member's representation as internal key.

## **internalNoncompoundedKey**

The member's representation as internal non-compounded key.

## **text**

The member's representation as text.

# **37.6 Bookmark**

## **37.6.1 Assign To Folder (assignToFolder)**

Assigns a standard bookmark to a folder. Both BookmarkFolderId and BookmarkId must be provided to identify the target folder directory and target bookmark.

### **Parameters**

Name	Type	Description
folderId	BookmarkFolderId	Bookmark folder id

Name	Type	Description
bookmarkId	BookmarkId	Bookmark id

## Returned Value

None

### Note

Supported platforms - BI platform and local.

## 37.6.2 Bookmark With Title Exists (bookmarkWithTitleExists)

Returns true if a bookmark with specified title exists and false if not.

## Parameters

Name	Type	Description
title	String	Bookmark title

## Returned value

Boolean. True if a bookmark with the specified title exists and false if not.

### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.6.3 Delete All Bookmarks (deleteAllBookmarks)

Deletes all standard bookmarks in the analysis application.



## Parameters

Name	Type	Description
(optional) applIdentifier	String	Application identifier  <b>i Note</b> This parameter is not supported on SAP NetWeaver.

## Returned value

None

### **i Note**

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.6.4 Delete Bookmark (deleteBookmark)

Deletes a standard bookmark with specified id.

## Parameters

Name	Type	Description
id	BookmarkId	BookmarkId id

## Returned Value

None

### **i Note**

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.6.5 Get All Bookmarks (getAllBookmarks)

Gets all standard bookmarks for the current version of an analysis application.

### Parameters

Name	Type	Description
(optional) applIdentifier	String	Application identifier  <b>i Note</b> This parameter is not supported on SAP NetWeaver.

### Returned value

Array of BookmarkInfo. It contains a list of standard bookmarks for the current version of an analysis application.

#### **i Note**

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.6.6 Get All Bookmarks By Folder (getAllBookmarksByFolder)

Gets all standard bookmarks for the current version of an analysis application by folder.

### Parameters

Name	Type	Description
id	BookmarkFolderId	Bookmark folder id

---

## Returned Value

Array of BookmarkInfo. It contains standard bookmarks located within a specified folder for the current version of an analysis application. If the folder identifier is invalid or there are no bookmarks found, an empty array is returned.

### Note

Supported platforms - BI platform and local.

## 37.6.7 Get Bookmark Folders (getBookmarkFolders)

Gets all bookmark folders which have been configured on the platform.

## Parameters

None

## Returned Value

Array of BookmarkFolderInfo. It contains a list of folders which have been configured on the platform.

### Note

Supported platforms - BI platform and local.

## 37.6.8 Get Bookmark Folders Tree Model (getBookmarkFoldersTreeModel)

Gets a tree model of all bookmark folders. Optional folder Id is used to return a specific bookmark folder tree model by folder Id.

## Parameters

Name	Type	Description
includeBookmarksAsLeaves	Boolean	Specifies whether to include bookmarks as leaves in the tree model.
(optional) folderId	BookmarkFolderId	Bookmark folder id

## Returned Value

TreeModel. It contains a tree model of bookmark folders.

### Note

Supported platforms - BI platform and local.

## 37.6.9 Get Bookmark Info (getBookmarkInfo)

Returns a standard bookmark specified by Bookmark Id.

## Parameters

Name	Type	Description
id	BookmarkId	Bookmark id

## Returned Value

BookmarkInfo. BookmarkInfo

### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.6.10 Get Bookmark Url (getBookmarkUrl)

Gets the bookmark Url for a given standard bookmark.

## Parameters

Name	Type	Description
id	BookmarkId	Bookmark id

## Returned value

String. Bookmark Url.

### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.6.11 Load Bookmark (loadBookmark)

Loads the state of an analysis application from a standard bookmark.

## Parameters

Name	Type	Description
id	BookmarkId	Bookmark id

## Returned value

None

### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.6.12 Save Bookmark (saveBookmark)

Creates a new bookmark with an optional title for the current state of an analysis application. If the title is not specified, then a default title will be generated automatically.

## Parameters

Name	Type	Description
(optional) title	String	Bookmark title
(optional) applIdentifier	String	Application identifier <div><b>i Note</b> This parameter is not supported on SAP NetWeaver.</div>

## Returned value

BookmarkId. It contains the saved bookmark.

### **i Note**

Supported platforms - BI platform, SAP NetWeaver and local mode.

## 37.6.13 Share Bookmark (shareBookmark)

Shares an analysis application Url from a standard bookmark.

## Parameters

Name	Type	Description
url	String	Bookmark Url

## Returned value

None

### **i Note**

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.7 Fragment Bookmark

### 37.7.1 Assign To Folder (assignToFolder)

Assigns a fragment bookmark to a folder. Both BookmarkFolderId and BookmarkId must be provided to identify the target folder directory and target bookmark.

#### Parameters

Name	Type	Description
folderId	BookmarkFolderId	Bookmark folder id
bookmarkId	BookmarkId	Bookmark id

#### Returned Value

None

#### Note


Supported platforms - BI platform and local.

### 37.7.2 Delete All Bookmarks (deleteAllBookmarks)

Deletes all fragment bookmarks of the analysis application.

#### Parameters

Name	Type	Description
(optional) appIdIdentifier	String	Application identifier

 **Note**  
This parameter is not supported on SAP NetWeaver.

## Returned Value

None

### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.7.3 Delete Bookmark (deleteBookmark)

Deletes a fragment bookmark with specified id.

### Parameters

Name	Type	Description
id	BookmarkId	BookmarkId id

## Returned Value

None

### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.7.4 Get All Bookmark Infos (getAllBookmarkInfos)

Gets a list of all fragment bookmarks for an analysis application.

### Parameters

Name	Type	Description
(optional) applIdentifier	String	Application identifier



Name	Type	Description
		<b>Note</b> This parameter is not supported on SAP NetWeaver.

## Returned Value

Array of FragmentBookmarkInfo. FragmentBookmarkArray

**Note**  
Supported platforms - BI platform, SAP NetWeaver, Local

## 37.7.5 Get All Bookmarks By Folder (getAllBookmarksByFolder)

Gets all fragment bookmarks for the current version of an analysis application by folder.

## Parameters

Name	Type	Description
id	BookmarkFolderId	Bookmark folder id

## Returned Value

Array of FragmentBookmarkInfo. It contains fragment bookmarks located within a specified folder for the current version of an analysis application. If the folder identifier is invalid or there are no bookmarks found, an empty array is returned.

**Note**  
Supported platforms - BI platform and local.

## 37.7.6 Get Bookmark Info (getBookmarkInfo)

Returns a fragment bookmark specified by Bookmark Id.

### Parameters

Name	Type	Description
id	BookmarkId	Bookmark id

### Returned Value

FragmentBookmarkInfo. FragmentBookmarkInfo

#### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.7.7 Get Bookmark Url (getBookmarkUrl)

Gets the bookmark Url for a given fragment bookmark.

### Parameters

Name	Type	Description
id	BookmarkId	Bookmark id

### Returned Value

String. Bookmark Url

#### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.7.8 Load Bookmark (loadBookmark)

Loads the state of an analysis application from a fragment bookmark.

### Parameters

Name	Type	Description
id	BookmarkId	Bookmark id

### Returned Value

None

#### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.7.9 Save Bookmark (saveBookmark)

Creates a new fragment bookmark using a container component as a selector. Selection is recursive to any depth in the specified container. If the title is not specified then a unique default title will be generated for the current application.

### Parameters

Name	Type	Description
selection	Container	Container selection
(optional) title	String	Bookmark title
(optional) description	String	Bookmark description
(optional) toOverwrite	BookmarkId	Bookmark to overwrite
(optional) appIdIdentifier	String	Application identifier

Name	Type	Description
		<p><b>i Note</b></p> <p>This parameter is not supported on SAP NetWeaver.</p>

## Returned Value

FragmentBookmarkInfo.FragmentBookmarkInfo

### **i Note**

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.7.10 Share Bookmark (shareBookmark)

Shares an analysis application Url from a fragment bookmark.

## Parameters

Name	Type	Description
url	String	Bookmark Url

## Returned Value

None

### **i Note**

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.8 Portable Fragment Bookmark

### 37.8.1 Assign To Folder (assignToFolder)

Assigns a portable fragment bookmark to a folder. Both BookmarkFolderId and BookmarkId must be provided to identify the target folder directory and target bookmark.

#### Parameters

Name	Type	Description
folderId	BookmarkFolderId	Bookmark folder id
bookmarkId	BookmarkId	Bookmark id

#### Returned Value

None

#### Note

Supported platforms - BI platform and local.

### 37.8.2 Delete All Bookmarks (deleteAllBookmarks)

Deletes all portable fragment bookmarks of all analysis applications specified by group identifier.

#### Parameters

Name	Type	Description
groupIdentifier	String	Group Identifier
(optional) applIdentifier	String	Application identifier

Name	Type	Description
		 <b>Note</b> This parameter is not supported on SAP NetWeaver.

## Returned Value

None

### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.8.3 Delete Bookmark (deleteBookmark)

Deletes a portable fragment bookmark with specified id.

## Parameters

Name	Type	Description
id	BookmarkId	BookmarkId id

## Returned value

None

### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.8.4 Get All Bookmark Infos (getAllBookmarkInfos)

Gets a list of all portable fragment bookmarks for a group identifier.

## Parameters

Name	Type	Description
groupIdentifier	String	Group Identifier
(optional) applIdentifier	String	Application identifier  <b>i Note</b> This parameter is not supported on SAP NetWeaver.

## Returned Value

Array of PortableFragmentBookmarkInfo. PortableFragmentBookmarkInfoArray.

### **i Note**

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.8.5 Get All Bookmark Infos For Application (getAllBookmarkInfosForApplication)

Gets a list of all portable fragment bookmarks for the current analysis application.

## Parameters

Name	Type	Description
groupIdentifier	String	Group Identifier
(optional) applIdentifier	String	Application identifier  <b>i Note</b> This parameter is not supported on SAP NetWeaver.

## Returned Value

Array of PortableFragmentBookmarkInfo. PortableFragmentBookmarkInfoArray.

### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.8.6 Get All Bookmarks By Folder (getAllBookmarksByFolder)

Gets all portable fragment bookmarks for the current version of an analysis application by folder.

## Parameters

Name	Type	Description
folderId	BookmarkFolderId	Bookmark folder id
groupIdentifier	String	Group Identifier

## Returned Value

Array of PortableFragmentBookmarkInfo. It contains portable fragment bookmarks located within a specified folder for the current version of an analysis application. If the folder identifier is invalid or there are no bookmarks found, an empty array is returned.

### Note

Supported platforms - BI platform and local.

## 37.8.7 Get BookmarkInfo (getBookmarkInfo)

Returns a portable fragment bookmark specified by Bookmark Id.



## Parameters

Name	Type	Description
id	BookmarkId	Bookmark id

## Returned Value

PortableFragmentBookmarkInfo. PortableFragmentBookmarkInfo

### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.8.8 Get Bookmark Url (getBookmarkUrl)

Gets the bookmark Url for a given portable fragment bookmark.

## Parameters

Name	Type	Description
id	BookmarkId	Bookmark id

## Returned Value

String. Bookmark Url

### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.8.9 Load Bookmark (loadBookmark)

Loads the state of an analysis application from a portable fragment bookmark.

## Parameters

Name	Type	Description
id	BookmarkId	Bookmark id

## Returned Value

None


### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.8.10 Save Bookmark (saveBookmark)

Creates a new portable fragment bookmark using the specified groupIdentifier. The group identifier must adhere to the following rules: minimum of eight characters, including a minimum of 4 alphabetical characters are required, numbers and underscores are allowed, special characters and spaces are not allowed. Selection is recursive to any depth in the specified container. Title is optional, and if not present, will be generated automatically. Portable fragment bookmarks are group id and user specific and can be accessed across applications.

## Parameters

Name	Type	Description
groupIdentifier	String	Group Identifier
selection	Container	Container selection
(optional) title	String	Bookmark title
(optional) description	String	Bookmark description
(optional) image	String	Bookmark image <div> <b>Note</b> On the BI platform, if you wish to over-write the optional image applied when saving a portable fragment bookmark, you will need to specify in your scripting, a fully qualified path</div>

Name	Type	Description
		to where the required image is stored. An example of such a path would be the following: "User Folders/Administrator/GEN (MIMEs) / vizgal_column.svg".
(optional) toOverwrite	BookmarkId	Bookmark to overwrite
(optional) appIdIdentifier	String	Application identifier <div> <b>i Note</b>  This parameter is not supported on SAP NetWeaver. </div>

## Returned Value

PortableFragmentBookmarkInfo. It contains a portable fregment bookmark info.

### **i Note**

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.8.11 Share Bookmark (shareBookmark)

Shares an analysis application Url from a portable fragment bookmark.

## Parameters

Name	Type	Description
url	String	Bookmark Url

## Returned Value

None

---

### **i** Note

Supported platforms - BI platform, SAP NetWeaver, Local

## **37.9 Button**

### **37.9.1 Get Icon (getIcon)**

Returns the path to the image used for the button's icon.

#### **Parameters**

None

#### **Returned Value**

Url. It contains the path to the used image.

### **37.9.2 Get Text (getText)**

Returns the text displayed on the button.

#### **Parameters**

None.

#### **Returned value**

String. It contains the text displayed on the button.

---

### 37.9.3 Get Tooltip (getTooltip)

Returns the tooltip of the component.

#### Parameters

None.

#### Returned value

String. It contains the tooltip text.

### 37.9.4 Is Enabled (isEnabled)

Returns whether component is enabled.

#### Parameters

None

#### Returned value type

Boolean. True if component is enabled or false if component is disabled.

### 37.9.5 On Click (onClick)

Calls the script that is executed when the user clicks the button.

#### Parameters

None

## Returned value

None

## 37.9.6 Set Enabled (setEnabled)

Enables or disables component.

Disabled components do not allow user interaction.

## Parameters

Name	Type	Description
isEnabled	Boolean	Specifies whether to enable or disable the component

## Returned value

None

## 37.9.7 Set Icon (setIcon)

Sets the path to the image to use for the button's icon.

## Parameters

Name	Type	Description
imageURI	Url	Path of image file to use

## Returned Value

None

## 37.9.8 Set Text (setText)

Sets the text displayed on the button.

### Parameters

Name	Type	Description
text	String	Button text

### Returned value

None

## 37.9.9 Set Tooltip (setTooltip)

Sets the tooltip of the component.

### Parameters

Name	Type	Description
tooltip	String	Tooltip text

### Returned value

None.

## 37.10 Chart

### 37.10.1 Get Chart Type (getChartType)

Returns name of chart type.

## Parameters

None

## Returned value

String. It contains the name of chart type.

## 37.10.2 Get Axis Scaling Max (getAxisScalingMax)

Gets the maximum value for an axis.

## Parameter

Name	Type	Description
axis	ChartAxisScaling	Property axis number

## Returned Value

Float. It contains the maximum value of the specified axis.

## 37.10.3 Remove Axis Scaling (removeAxisScaling)

Disables axis scaling for an axis.

## Parameters

Name	Type	Description
axis	ChartAxisScaling	Property axis number



## Returned Value

None

### 37.10.4 Get Axis Scaling Min (getAxisScalingMin)

Gets the minimum value for an axis.

## Parameter

Name	Type	Description
axis	ChartAxisScaling	Property axis number

## Returned Value

Float. It contains the maximum value of the specified axis.

### 37.10.5 Set Axis Scaling (setAxisScaling)

Sets a minimum and maximum value for an axis.

## Parameter

Name	Type	Description
axis	ChartAxisScaling	Property axis number
minValue	Float	Property min value
maxValue	Float	Property max value

## Returned Value

None

---

## 37.10.6 Clear Selection (clearSelection)

Clears the selection from the chart.

### Parameters

None

### Returned Value

None

## 37.10.7 Get Legend Position (getLegendPosition)

Returns the position of the chart legend.

### Parameters

None

### Returned Value

LegendPosition. It contains the position of the chart legend.

## 37.10.8 Get Selected Member (getSelectedMember)

Returns the member that is contained in the current selection of the chart.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the selected members

## Returned value

Member. It is the member that has been selected in the chart.

## 37.10.9 Get Selected Members (getSelectedMembers)

Returns the members that are contained in the current selections of the chart.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the selected members

## Returned value

Array of Member. It contains the members that have been selected in the chart.

## 37.10.10 Set Chart Type (setChartType)

Sets the chart type.

## Parameters

Name	Type	Description
chartType	ChartType	Chart type to set.

---

## Returned value

None

### 37.10.11 Get Style (getStyle)

Returns name of chart style.

## Parameters

None.

## Returned value

String. It contains the name of chart style.

### 37.10.12 Is Visible (isVisible)

Returns whether component is visible.

## Parameters

None

## Returned value

Boolean. True if component is shown or false if component is hidden.

### 37.10.13 On Select (onSelect)

Calls the script that is executed when the user selects an area in the chart.

## Parameters

None

## Returned value

None

### 37.10.14 Set Legend Position (setLegendPosition)

Sets the position of the chart legend.

## Parameters

Name	Type	Description
position	LegendPosition	Position of the chart legend.

## Returned Value

None

### 37.10.15 Set Style

Sets chart style.

#### **i** Note

As of release 1.1 this method has no effect in the application. You can still use this method but no error warning is displayed in the design studio.

## Parameters

Name	Type	Description
style	ChartStyle	Chart style to set

## Returned value

None

### 37.10.16 Set Visible (setVisible)

Shows or hides component.

## Parameters

Name	Type	Description
isVisible	Boolean	Specifies whether to show or hide component

## Returned value

None

### 37.10.17 Show Data Labels (showDataLabels)

Shows or hides the data labels.

## Parameter

Name	Type	Description
isVisible	Boolean	Specifies whether to show or hide the data labels

## Returned Value

None

## 37.10.18 Show Scaling Factors (showScalingFactors)

Shows the scaling factors on axis and tooltip.

## Parameters

Name	Type	Description
isVisible	Boolean	Specifies whether to show or hide scaling factors on axis and tooltip.

## Returned value

None

## 37.10.19 Show Totals (showTotals)

Shows or hides totals and subtotals.

## Parameters

Name	Type	Description
isVisible	Boolean	Specifies whether to show or hide totals and subtotals

## Returned value

None

## 37.10.20 Swap Axes (swapAxes)

Swaps the result set axes of the assigned data source.

## Parameters

Name	Type	Description
swapAxes	Boolean	Specifies whether to swap axes

## Returned value

None

## 37.10.21 Set Data Selection (setDataSelection)

Sets the subset of the data of the source to be shown by the chart.

## Parameters

Name	Type	Description
selection	ResultSetSelectionByString	Data selection to show by the chart



Name	Type	Description

## Returned Value

None

## 37.10.22 Set Data Source (setDataSource)

Assigns a data source.

## Parameters

Table 82:

Name	Type	Description
dataSourceAlias	DataSourceAlias	Data source alias

## Returned value

None.

## 37.11 Chart Type Picker

### 37.11.1 Get Selected Value (getSelectedValue)

Gets the value selected in the Chart Type Picker.

## Parameters

None

## Returned Value

String. It contains the selected value.

### 37.11.2 Set Selected Value (setSelectedValue)

Sets the value selected by the ChartTypePicker component. The value must be either a Chart type (passed as a string, for example "PIE", or a Chart constant, for example ChartType.PIE), or an additional type added to Chart Type Picker, or an additional type added to Chart Type Picker via the Additional Types property (passed as a string).

## Parameters

Name	Type	Description
value	String	Value to select

## Returned Value

None

## 37.12 Info Chart

### 37.12.1 Clear Selection (clearSelection)

Clears the selection from the chart.

## Parameters

None

## Returned Value

None

### 37.12.2 Get Axis Scaling Max(getAxisScalingMax)

Gets the maximum value for an axis.

#### Parameter

Name	Type	Description
axis	InfoChartAxisScaling	Property axis number

## Returned Value

Float. It contains the maximum value of the specified axis.

### 37.12.3 Get Axis Scaling Min (getAxisScalingMin)

Gets the minimum value for an axis.

#### Parameter

Name	Type	Description
axis	InfoChartAxisScaling	Property axis number

## Returned Value

Float. It contains the maximum value of the specified axis.

---

## 37.12.4 Get Bottom Margin (getBottomMargin)

Returns the bottom margin of the component.

### Parameters

None

### Returned value

Integer. It contains the bottom margin of component in pixels. It is -1 if the current value is *auto*.

## 37.12.5 Get Chart Type (getChartType)

Returns name of chart type.

### Parameters

None

### Returned value

InfoChartType. It contains the name of chart type.

## 37.12.6 Get CSS Class (getCSSClass)

Returns the Cascading Style Sheet (CSS) class name of the component.

### Parameters

None

---

## Returned value

String. It contains the CSS class name of the component.

### 37.12.7 Get Data Source (getDataSource)

Returns the assigned data source.

## Parameters

None

## Returned value

DataSourceAlias. It contains a data source alias.

### 37.12.8 Get Height (getHeight)

Returns the height of the component.

## Parameters

None

## Returned value

Integer. It contains the height of component in pixels. It is -1 if the current value is *auto*.

### 37.12.9 Get Left Margin (getLeftMargin)

Returns the left margin of the component.

---

## Parameters

None

## Returned value

Integer. It contains the left margin of component in pixels. It is -1 if the current value is *auto*.

### 37.12.10 Get Legend Position (getLegendPosition)

Returns the position of the chart legend.

## Parameters

None

## Returned Value

LegendPosition. It contains the position of the chart legend.

### 37.12.11 Get Right Margin (getRightMargin)

Returns the right margin of the component.

## Parameters

None

## Returned value

Integer. It contains the right margin of component in pixels. It is -1 if the current value is *auto*.

## 37.12.12 Get Selected Member (getSelectedMember)

Returns the member that is contained in the current selection of the chart.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the selected member.

### Returned Value

Member. It contains the member that has been selected in the chart.

## 37.12.13 Get Selected Members (getSelectedMembers)

Returns the members that are contained in the current selections of the chart.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the selected members

### Returned value

Array of Member. It contains the members that have been selected in the chart.

## 37.12.14 Get Top Margin (getTopMargin)

Returns the top margin of the component.

---

## Parameters

None

## Returned value

Integer. It contains the top margin of the component in pixels. It is -1 if the current value is *auto*.

### 37.12.15 Get Width (getWidth)

Returns the width of the component.

## Parameters

None

## Returned value

Integer. It contains the width of component in pixels. It is -1 if the current value is *auto*.

### 37.12.16 Hide Loading State (hideLoadingState)

Hides the loading indicator on the component.

## Parameters

None

## Returned value

None



## 37.12.17 Is Visible (isVisible)

Returns whether component is visible.

### Parameters

None

### Returned value

Boolean. True if component is shown or false if component is hidden.

## 37.12.18 Remove Axis Scaling (removeAxisScaling)

Disables axis scaling for an axis.

### Parameters

Name	Type	Description
axis	InfoChartAxisScaling	Property axis number

### Returned Value

None

## 37.12.19 Set Axis Scaling (setAxisScaling)

Sets a minimum and maximum value for an axis.

## Parameter

Name	Type	Description
axis	InfoChartAxisScaling	Property axis number
minValue	Float	Property min value
maxValue	Float	Property max value

## Returned Value

None

### 37.12.20 Set Bottom Margin (setBottomMargin)

Sets the bottom margin of the component.

#### Note

This function has no effect, if the current value is *auto*.

## Parameters

Table 83:

Name	Type	Description
bottomMargin	Integer	Bottom margin of component in pixels

## Returned value

None

### 37.12.21 Set CSS Class (setCSSClass)

Sets the Cascading Style Sheet (CSS) class name of the component.

## Parameters

Table 84:

Name	Type	Description
className	String	Name of the CSS class

## Returned value

None

## 37.12.22 Set Chart Type (setChartType)

Sets info chart type.

## Parameters

Name	Type	Description
chartType	InfoChartType	Chart type to set.

## Returned value

None

## 37.12.23 Show Data Labels (showDataLabels)

Shows or hides the data labels.

## Parameter

Name	Type	Description
isVisible	Boolean	Specifies whether to show or hide the data labels

## Returned Value

None

## 37.12.24 Set Data Selection (setDataSelection)

Sets the subset of the data of the source to be shown by the chart.

## Parameters

Name	Type	Description
selection	ResultSetSelectionByString	Data selection to show by the chart

## Returned Value

None

## 37.12.25 Set Data Source (setDataSource)

Assigns a data source.

## Parameters

Table 85:

Name	Type	Description
dataSourceAlias	DataSourceAlias	Data source alias

## Returned value

None.

### 37.12.26 Set Height (setHeight)

Sets the height of the component.

#### Note

This function has no effect if the current value is *auto*.

## Parameters

Table 86:

Name	Type	Description
height	Integer	Height of the component in pixels

## Returned value

None

### 37.12.27 Set Left Margin (setLeftMargin)

Sets left margin of the component.

### **i** Note

This function has no effect if the current value is *auto*.

## Parameters

Table 87:

Name	Type	Description
leftMargin	Integer	Left margin of component in pixels

## Returned value

None

## 37.12.28 Set Legend Position (setLegendPosition)

Sets the position of the chart legend.

## Parameters

Name	Type	Description
position	LegendPosition	Position of the chart legend.

## Returned Value

None

## 37.12.29 Show Loading State (showLoadingState)

Shows a loading indicator on the component.

## Parameters

Name	Type	Description
(optional) text	String	Text to show in the loading indicator

## Returned value

None

### 37.12.30 Set Right Margin (setRightMargin)

Sets right margin of the component.

#### Note

This function has no effect if the current value is *auto*.

## Parameters

Table 88:

Name	Type	Description
rightMargin	Integer	Right margin of component in pixels

## Returned value

None

### 37.12.31 Set Top Margin (setTopMargin)

Sets top margin of the component.

#### Note

This function has no effect if the current value is *auto*.

## Parameters

Table 89:

Name	Type	Description
topMargin	Integer	Top margin of component in pixels

## Returned value

None

### 37.12.32 Show Totals (showTotals)

Shows or hides totals and subtotals.

## Parameters

Name	Type	Description
isVisible	Boolean	Specifies whether to show or hide totals and subtotals

## Returned value

None

### 37.12.33 Set Visible (setVisible)

Shows or hides component.



## Parameters

Name	Type	Description
isVisible	Boolean	Specifies whether to show or hide component

## Returned value

None

### 37.12.34 Set Width (setWidth)

Sets width of the component.

#### Note

This function has no effect if the current value is *auto*.

## Parameters

Table 90:

Name	Type	Description
width	Integer	Width of component in pixels

## Returned value

None

## 37.13 Info Chart Feeding Panel

### 37.13.1 Get Bottom Margin (getBottomMargin)

Returns the bottom margin of the component.

---

## Parameters

None

## Returned value

Integer. It contains the bottom margin of component in pixels. It is -1 if the current value is *auto*.

### 37.13.2 Get CSS Class (getCSSClass)

Returns the Cascading Style Sheet (CSS) class name of the component.

## Parameters

None

## Returned value

String. It contains the CSS class name of the component.

### 37.13.3 Get Height (getHeight)

Returns the height of the component.

## Parameters

None

## Returned value

Integer. It contains the height of component in pixels. It is -1 if the current value is *auto*.

---

## 37.13.4 Get Left Margin (getLeftMargin)

Returns the left margin of the component.

### Parameters

None

### Returned value

Integer. It contains the left margin of component in pixels. It is -1 if the current value is [auto](#).

## 37.13.5 Get Right Margin (getRightMargin)

Returns the right margin of the component.

### Parameters

None

### Returned value

Integer. It contains the right margin of component in pixels. It is -1 if the current value is [auto](#).

## 37.13.6 Get Top Margin (getTopMargin)

Returns the top margin of the component.

### Parameters

None

---

## Returned value

Integer. It contains the top margin of the component in pixels. It is -1 if the current value is *auto*.

### 37.13.7 Get Width (getWidth)

Returns the width of the component.

## Parameters

None

## Returned value

Integer. It contains the width of component in pixels. It is -1 if the current value is *auto*.

### 37.13.8 Hide Loading State (hideLoadingState)

Hides the loading indicator on the component.

## Parameters

None

## Returned value

None

### 37.13.9 Is Visible (isVisible)

Returns whether component is visible.

## Parameters

None

## Returned value

Boolean. True if component is shown or false if component is hidden.

### 37.13.10 Set Bottom Margin (setBottomMargin)

Sets the bottom margin of the component.

#### **i** Note

This function has no effect, if the current value is *auto*.

## Parameters

Table 91:

Name	Type	Description
bottomMargin	Integer	Bottom margin of component in pixels

## Returned value

None

### 37.13.11 Set CSS Class (setCSSClass)

Sets the Cascading Style Sheet (CSS) class name of the component.

## Parameters

Table 92:

Name	Type	Description
className	String	Name of the CSS class

## Returned value

None

### 37.13.12 Set Chart Reference (setChartReference)

Sets the chart to be referenced by the chart feeding panel.

## Parameters

Name	Type	Description
chartReference	InfoChart	The chart to be referenced.

## Returned Value

None

### 37.13.13 Set Height (setHeight)

Sets the height of the component.

#### Note

This function has no effect if the current value is *auto*.

## Parameters

Table 93:

Name	Type	Description
height	Integer	Height of the component in pixels

## Returned value

None

### 37.13.14 Set Left Margin (setLeftMargin)

Sets left margin of the component.

#### Note

This function has no effect if the current value is *auto*.

## Parameters

Table 94:

Name	Type	Description
leftMargin	Integer	Left margin of component in pixels

## Returned value

None

### 37.13.15 Set Right Margin (setRightMargin)

Sets right margin of the component.

#### Note

This function has no effect if the current value is *auto*.

## Parameters

Table 95:

Name	Type	Description
rightMargin	Integer	Right margin of component in pixels

## Returned value

None

### 37.13.16 Set Top Margin (setTopMargin)

Sets top margin of the component.

#### Note

This function has no effect if the current value is *auto*.

## Parameters

Table 96:

Name	Type	Description
topMargin	Integer	Top margin of component in pixels

## Returned value

None

### 37.13.17 Set Visible (setVisible)

Shows or hides component.



## Parameters

Name	Type	Description
isVisible	Boolean	Specifies whether to show or hide component

## Returned value

None

### 37.13.18 Set Width (setWidth)

Sets width of the component.

#### **i** Note

This function has no effect if the current value is *auto*.

## Parameters

Table 97:

Name	Type	Description
width	Integer	Width of component in pixels

## Returned value

None

### 37.13.19 Show Loading State (showLoadingState)

Shows a loading indicator on the component.

## Parameters

Name	Type	Description
(optional) text	String	Text to show in the loading indicator

## Returned value

None

## 37.14 Checkbox

### 37.14.1 Get Text (getText)

Returns checkbox text.

## Parameters

None

## Returned value

String. It contains the text of the checkbox.

### 37.14.2 Get Tooltip (getTooltip)

Returns the tooltip of the component.

## Parameters

None.

---

## Returned value

String. It contains the tooltip text.

### 37.14.3 Is Checked (isChecked)

Returns whether checkbox is selected. Returns true, if checkbox is selected, or false, if checkbox is unselected.

## Parameters

None

## Returned value

Boolean. True if checkbox is selected or false if checkbox is unselected.

### 37.14.4 Is Enabled (isEnabled)

Returns whether component is enabled.

## Parameters

None

## Returned value type

Boolean. True if component is enabled or false if component is disabled.

### 37.14.5 Is Visible (isVisible)

Returns whether component is visible.

---

## Parameters

None

## Returned value

Boolean. True if component is shown or false if component is hidden.

### 37.14.6 On Click (onClick)

Calls the script that is executed when the user clicks the checkbox.

## Parameters

None

## Returned value

None

### 37.14.7 Set Checked (setChecked)

Selects or unselects checkbox.

## Parameters

Name	Type	Description
isChecked	Boolean	Specifies whether to select or unselect checkbox .

## Returned value

None

## 37.14.8 Set Enabled (setEnabled)

Enables or disables component.

Disabled components do not allow user interaction.

## Parameters

Name	Type	Description
isEnabled	Boolean	Specifies whether to enable or disable the component

## Returned value

None

## 37.14.9 Set Text (setText)

Sets checkbox text.

## Parameters

Name	Type	Description
text	String	Checkbox text

## Returned value

None

---

## 37.14.10 Set Tooltip (setTooltip)

Sets the tooltip of the component.

### Parameters

Name	Type	Description
tooltip	String	Tooltip text

### Returned value

None.

## 37.14.11 Set Visible (setVisible)

Shows or hides component.

### Parameters

Name	Type	Description
isVisible	Boolean	Specifies whether to show or hide component

### Returned value

None

## 37.15 Checkbox Group

### 37.15.1 Add Item (addItem)

Adds an item to the component.

#### Parameters

Name	Type	Description
value	String	Unique key (or technical value) of the item. The method call is ignored if an item with this key already exists in the component.
text	String	Display text of the item.
(optional) index	Integer	Index where to insert the item (default: -1). If the index is greater than the number of items or less than 0 then the item is added as the last item. If the index is 0 then it is inserted before the first item.

#### Returned value

None

### 37.15.2 Get Selected Texts (getSelectedTexts)

#### Note

This method is only valid for the [List Box](#) component.

Returns the selected item's text. Can be used if multiple selection is enabled.

#### Parameters

None

---

## Returned value

StringArray. It contains the selected texts as an array.

### 37.15.3 Get Selected Values (getSelectedValues)

#### **i** Note

This method is only valid for the [List Box](#) component.

Returns the selected item's value. Can be used if multiple selection is enabled.

## Parameters

None

## Returned value

StringArray. It contains the selected values as an array.

### 37.15.4 Is Enabled (isEnabled)

Returns whether component is enabled.

## Parameters

None

## Returned value type

Boolean. True if component is enabled or false if component is disabled.



---

## 37.15.5 Remove All Items (removeAllItems)

Removes all items from the component.

### Parameters

None

### Returned value

None

## 37.15.6 Remove Item (removeItem)

Removes an item from the component.

### Parameters

Name	Type	Description
value	String	Unique key (or technical value) of the item. The method call is ignored if no such key exists in the component.

### Returned value

None

## 37.15.7 Set Enabled (setEnabled)

Enables or disables component.

Disabled components do not allow user interaction.

## Parameters

Name	Type	Description
isEnabled	Boolean	Specifies whether to enable or disable the component

## Returned value

None

## 37.15.8 Set Items (setItems)

Assigns a new list of items to the component.

The old content of the component will be deleted.

## Parameters

Name	Type	Description
value	ValueTextList	List of value-text pairs

## Returned value

None

### Example

In the following example, method `setItems` sets the first 100 members from the **Customer** dimension to the checkbox group / list box. Each list item's value is represented by the customer's key, each list item's text is represented by the customer's text and key. No item is added to represent all members.

```
LISTBOX_1.setItems(DS_1.getMemberList("OD_CUSTOMER", MemberPresentation.KEY,
MemberDisplay.TEXT_KEY, 100));
```

or

```
CHECKBOXGROUP_1.setItems(DS_1.getMemberList("OD_CUSTOMER",
MemberPresentation.KEY, MemberDisplay.TEXT_KEY, 100));
```

## 37.15.9 Set Selected Values (setSelectedValues)

Selects the items with the specified values.

### Parameters

Name	Type	Description
values	listValue Array	Values of the items to select

### Returned value

None

## 37.15.10 Sort (sort)

Sorts values in alphabetical order.

### Parameters

Table 98:

Name	Type	Description
(optional) isSortAscending	Boolean	Specifies whether values are sorted in ascending or descending alphabetical order

### Returned value

None

---

## 37.16 Constants

### 37.16.1 Axis

A set of constants to specify an axis.

#### **COLUMNS**

Columns axis

#### **FREE**

Free axis

#### **ROWS**

Rows axis

### 37.16.2 ChartType

A set of constants to specify the type of chart.

#### **BAR\_COMBINATION**

Bar combination

#### **BUBBLE**

Bubble Chart

---

## **COLUMN\_COMBINATION**

Column combination

## **DUAL\_BAR**

Bar - Dual Axis

## **DUAL\_COLUMN**

Dual Column Chart

## **DUAL\_COMBINATION**

Dual Combination Chart

## **DUAL\_HORIZONTAL\_COMBINATION**

Horizontal Dual Combination Chart

## **DUAL\_LINE**

Dual Line Chart

## **HORIZONTAL\_AREA**

Horizontal area

## **HORIZONTAL\_BAR**

Horizontal bar

---

## **HORIZONTAL\_LINE**

Horizontal line

## **HORIZONTAL\_STACKED\_100\_BAR**

100% stacked bar

## **HORIZONTAL\_STACKED\_BAR**

Stacked bar

## **HORIZONTAL\_WATERFALL**

Horizontal waterfall

## **LINE**

Line

## **MULTIPIE**

Multiple pie

## **MULTIRADAR**

Multiple radar

## **PIE**

Pie

---

## **RADAR**

Radar

## **SCATTER**

Stacked waterfall

## **STACKED\_WATERFALL**

Stacked waterfall

## **SURFACE**

Area

## **VERTICAL\_BAR**

Column

## **VERTICAL\_STACKED\_100\_BAR**

100% stacked column

## **VERTICAL\_STACKED\_BAR**

Stacked column

## **WATERFALL**

Waterfall

---

### 37.16.3 ClickArea

A set of constants to specify the area of the click.

#### DATA

Data cell has been clicked

#### DIMENSION

Dimension cell has been clicked

#### MEMBER

Member cell has been clicked

### 37.16.4 ContextMenuItemId

A set of constants to specify the item.

#### ATTRIBUTES

Configure the displayed attributes

#### BACK\_ONE\_STEP

Undo previous step

#### BACK\_TO\_START

Back to start



---

## **CALCULATION\_ADD**

Add calculation when selecting multiple measures

## **CALCULATION\_DYNAMIC**

Add dynamic calculations

## **CALCULATION\_EDIT\_NAME**

Edit name for calculations

## **CALCULATION\_REMOVE**

Remove a calculation

## **CALCULATION\_TOTALS**

Change the totals calculation mode

## **CELL\_LOCKING**

Lock cell when selecting an input enabled data cell

## **DISPLAY**

Configure the displayed presentations

## **DRILL\_DOWN\_BY\_DIMENSION**

Drilldown by dimension that allows selecting a dimension for drilling down deeper

---

## **DRILL\_DOWN\_REMOVE**

Remove from drilldown

## **DRILL\_DOWN\_SWAP\_AXES**

Swap axes

## **DRILL\_DOWN\_SWAP\_WITH**

Swap with another dimension

## **FILTER\_CLEAR**

Select all members that will clear the filter

## **FILTER\_KEEP**

Keep member

## **FILTER\_KEEP\_AXIS**

Keep member on axis

## **FILTER\_SELECT**

Filter members that open a selector for defining the filter

## **FILTER\_SWAP**

Filter member and swap

---

## **HIERARCHY\_COLLAPSE**

Collapse hierarchy

## **HIERARCHY\_EXPAND**

Expand hierarchy

## **HIERARCHY\_EXPAND\_LEVEL**

Expand hierarchy to a selected level

## **HIERARCHY\_SELECT**

Select hierarchy

## **JUMP**

Jump

## **MEASURE\_FILTER\_DEFINE**

Filter by measure that allows to define, edit, or remove a measure-based filter

## **MEASURE\_FILTER\_PREDEFINED**

Predefined measure filter that allows to activate or deactivate measure-based filters created in another tool

## **NUMBER\_FORMAT**

Format number

---

## **SORT**

Sort

## **SUPPRESS\_ZEROS**

Suppress zeros

## **TOTALS\_DISPLAY**

Configure the displayed totals

## **37.16.5 DataSourceBrowseType**

A set of constants to specify the type of a data source browser tree.

## **INFO\_AREA\_OR\_FOLDER**

Info Areas (BW) or Folders (HANA)

## **ROLES**

SAP BW Roles

## **WORKSPACES**

SAP BW Workspaces

## **37.16.6 DataSourceConnectionType**

A set of constants to specify the type of a data source connection.

---

## **BW**

SAP BW backend

## **HANA**

SAP HANA backend

### **37.16.7 DataSourceExportType**

A set of constants to specify the export type.

## **LUMIRA\_lums**

Export to Lumira (lums)

### **37.16.8 DataSourceType**

A set of constants to specify the type of a data source.

## **FOLDER**

Folder (not a data source)

## **INFOPROVIDER**

Info Provider

## **QUERY**

BEx query

---

## VIEW

BEx query view

### 37.16.9 IconBackgroundShape

A set of constants to specify the background of the icon.

#### ELLIPSIS

Elliptical background

#### NONE

No background (transparent)

#### RECTANGLE

Rectangular background

### 37.16.10 InfoChartType

A set of constants to specify the type of Info Chart.

#### INFO\_100\_DUAL\_STACKED\_BAR

Dual Stacked Bar Info Chart

#### INFO\_100\_DUAL\_STACKED\_COLUMN

100% Dual Stacked Column Info Chart

---

## **INFO\_100\_STACKED\_BAR**

100% Dual Stacked Bar Info Chart

## **INFO\_100\_STACKED\_COLUMN**

100% Stacked Column Info Chart

## **INFO\_AREA**

Area Info Chart

## **INFO\_BAR**

Bar Info Chart

## **INFO\_BUBBLE**

Bubble Info Chart

## **INFO\_BULLET**

Bullet Info Chart

## **INFO\_COLUMN**

Column Info Chart

## **INFO\_COMBINATION**

Combination Info Chart

---

## **INFO\_DONUT**

Doughnut Info Chart

## **INFO\_DUAL\_BAR**

Dual Bar Info Chart

## **INFO\_DUAL\_COLUMN**

Dual Column Info Chart

## **INFO\_DUAL\_COMBINATION**

Dual Combination Info Chart

## **INFO\_DUAL\_HORIZONTAL\_COMBINATION**

Dual Horizontal Combination Info Chart

## **INFO\_DUAL\_HORIZONTAL\_LINE**

Dual Horizontal Line Info Chart

## **INFO\_DUAL\_HORIZONTAL\_STACKED\_COMBINATION**

Dual Horizontal Stacked Combination Info Chart

## **INFO\_DUAL\_LINE**

Dual Line Info Chart



---

## **INFO\_DUAL\_STACKED\_BAR**

Dual Stacked Bar Info Chart

## **INFO\_DUAL\_STACKED\_COLUMN**

Dual Stacked Column Info Chart

## **INFO\_DUAL\_STACKED\_COMBINATION**

Dual Stacked Combination Info Chart

## **INFO\_HEATMAP**

Heatmap Info Chart

## **INFO\_HORIZONTAL\_AREA**

Horizontal Area Info Chart

## **INFO\_HORIZONTAL\_COMBINATION**

Horizontal Combination Info Chart

## **INFO\_HORIZONTAL\_LINE**

Horizontal Line Info Chart

## **INFO\_HORIZONTAL\_STACKED\_COMBINATION**

Horizontal Stacked Combination Info Chart

---

## **INFO\_LINE**

Line Info Chart

## **INFO\_PIE**

Pie Info Chart

## **INFO\_RADAR**

Radar Info Chart

## **INFO\_SCATTER**

Scatter Info Chart

## **INFO\_STACKED\_BAR**

Stacked Bar Info Chart

## **INFO\_STACKED\_COLUMN**

Stacked Column Info Chart

## **INFO\_STACKED\_COMBINATION**

Stacked Combination Info Chart

## **INFO\_TIMESERIES\_LINE**

Time Series Line Info Chart

---

## **INFO\_TREEMAP**

Tree Map Info Chart

## **INFO\_TRELLIS\_AREA**

Trellis Area Info Chart

## **INFO\_TRELLIS\_BAR**

Trellis Bar Info Chart

## **INFO\_TRELLIS\_COLUMN**

Trellis Column Info Chart

## **INFO\_TRELLIS\_HORIZONTAL\_AREA**

Trellis Horizontal Area Info Chart

## **INFO\_TRELLIS\_HORIZONTAL\_LINE**

Trellis Horizontal Line Info Chart

## **INFO\_TRELLIS\_LINE**

Trellis Line Info Chart

## **INFO\_VERTICAL\_BULLET**

Vertical Bullet Info Chart

---

## 37.16.11 Layout

A set of constants to specify layout information.

### AUTO

The value should be auto

## 37.16.12 LegendPosition

A set of constants to specify the position of the chart legend.

### BOTTOM

Position the legend at the bottom of the chart

### LEFT

Position the legend to the left of the chart

### RIGHT

Position the legend to the right of the chart

### TOP

Position the legend at the top of the chart

## 37.16.13 MemberDisplay

A set of constants to specify how members are displayed.

---

## **KEY**

Member is displayed as external key

## **KEY\_TEXT**

Member is displayed as external key and text

## **NO\_DISPLAY**

Member is not displayed

## **TEXT**

Member is displayed as text

## **TEXT\_KEY**

Member is displayed as text and external key

## **37.16.14 MemberPresentation**

A set of constants to specify which aspect of a member to present.

## **EXTERNAL\_KEY**

Member is represented by its external key

## **EXTERNAL\_NONCOMPOUNDED\_KEY**

Member is represented by its external non-compounded key

---

## INTERNAL\_KEY

Member is represented by its internal key

## INTERNAL\_NONCOMPOUNDED\_KEY

Member is represented by its internal non-compounded key

## TEXT

Member is represented by its text

### 37.16.15 NegativeNumberDisplay

A set of constants to specify how a negative number is displayed.

#### LEADING\_MINUS

Negative values are displayed as -*X*

#### PARENTHESES

Negative values are displayed as (*X*)

#### TRAILING\_MINUS

Negative values are displayed as *X*-

### 37.16.16 Scaling

A set of constants to specify a scaling factor applied to numbers.

---

## **FACTOR\_1**

Scaling factor 1 (no scaling)

## **FACTOR\_10**

Scaling factor 10

## **FACTOR\_100**

Scaling factor 100

## **FACTOR\_1000**

Scaling factor 1000

## **FACTOR\_100000**

Scaling factor 10000

## **FACTOR\_1000000**

Scaling factor 100000

## **TEXT**

Scaling factor 1000000

## **FACTOR\_10000000**

Scaling factor 10000000

---

## **FACTOR\_100000000**

Scaling factor 100000000

## **FACTOR\_1000000000**

Scaling factor 1000000000

## **FACTOR\_DEFAULT**

Default scaling factor

## **37.16.17 TotalsDisplay**

A set of constants to specify whether totals are displayed in the result set.

### **HIDE**

Totals are always hidden

### **HIDE\_IF\_ONLY\_ONE\_MEMBER**

Totals are hidden if only one member is available

### **SHOW**

Totals are always shown

## **37.16.18 TotalsPosition**

A set of constants to specify where totals are displayed in the result set.



---

## AFTER

Display totals after members

## BEFORE

Display totals before members

## DEFAULT

Use default display

### 37.16.19 UnitsAndScalingFactorsDisplay

A set of constants to specify how units and scaling factors are displayed in the Crosstab.

#### BOTH\_IN\_HEADER

Display both units and scaling factors in the header

#### NO\_DISPLAY

Do not display units and scaling factors

#### UNITS\_IN\_DATA\_CELLS

Display units in the data cells

### 37.16.20 ZeroDisplay

A set of constants to specify how the number zero is displayed.

---

## CUSTOM

Zero values are displayed as a custom text

## DEFAULT

Default value

## EMPTY\_CELL

Zero values are displayed as empty strings

## WITHOUT\_CURRENCY\_UNIT

Zero values are displayed without currency and unit information

## 37.17 Component

Base class for all visual components, for example, button and crosstab components. The methods in this class are available for all components.

### 37.17.1 Get Bottom Margin (getBottomMargin)

Returns the bottom margin of the component.

#### Parameters

None

---

## Returned value

Integer. It contains the bottom margin of component in pixels. It is -1 if the current value is [auto](#).

### 37.17.2 Get CSS Class (getCSSClass)

Returns the Cascading Style Sheet (CSS) class name of the component.

## Parameters

None

## Returned value

String. It contains the CSS class name of the component.

### 37.17.3 Get Height (getHeight)

Returns the height of the component.

## Parameters

None

## Returned value

Integer. It contains the height of component in pixels. It is -1 if the current value is [auto](#).

### 37.17.4 Get Left Margin (getLeftMargin)

Returns the left margin of the component.

---

## Parameters

None

## Returned value

Integer. It contains the left margin of component in pixels. It is -1 if the current value is *auto*.

### 37.17.5 Get Right Margin (getRightMargin)

Returns the right margin of the component.

## Parameters

None

## Returned value

Integer. It contains the right margin of component in pixels. It is -1 if the current value is *auto*.

### 37.17.6 Get Top Margin (getTopMargin)

Returns the top margin of the component.

## Parameters

None

## Returned value

Integer. It contains the top margin of the component in pixels. It is -1 if the current value is *auto*.

---

## 37.17.7 Get Width (getWidth)

Returns the width of the component.

### Parameters

None

### Returned value

Integer. It contains the width of component in pixels. It is -1 if the current value is *auto*.

## 37.17.8 Hide Loading State (hideLoadingState)

Hides the loading indicator on the component.

### Parameters

None

### Returned value

None

## 37.17.9 Is Visible (isVisible)

Returns whether component is visible.

### Parameters

None

## Returned value

Boolean. True if component is shown or false if component is hidden.

## 37.17.10 Set Bottom Margin (setBottomMargin)

Sets the bottom margin of the component.

### Note

This function has no effect, if the current value is *auto*.

## Parameters

Table 99:

Name	Type	Description
bottomMargin	Integer	Bottom margin of component in pixels

## Returned value

None

## 37.17.11 Set CSS Class (setCSSClass)

Sets the Cascading Style Sheet (CSS) class name of the component.

## Parameters

Table 100:

Name	Type	Description
className	String	Name of the CSS class

Returned value

None

37.17.12 Set Height (setHeight)

Sets the height of the component.

*i* Note

This function has no effect if the current value is *auto*.

Parameters

Table 101:

Name	Type	Description
height	Integer	Height of the component in pixels

Returned value

None

37.17.13 Set Left Margin (setLeftMargin)

Sets left margin of the component.

*i* Note

This function has no effect if the current value is *auto*.

## Parameters

Table 102:

Name	Type	Description
leftMargin	Integer	Left margin of component in pixels

## Returned value

None

### 37.17.14 Set Right Margin (setRightMargin)

Sets right margin of the component.

#### Note

This function has no effect if the current value is *auto*.

## Parameters

Table 103:

Name	Type	Description
rightMargin	Integer	Right margin of component in pixels

## Returned value

None

### 37.17.15 Set Top Margin (setTopMargin)

Sets top margin of the component.

#### Note

This function has no effect if the current value is *auto*.



## Parameters

Table 104:

Name	Type	Description
topMargin	Integer	Top margin of component in pixels

## Returned value

None

### 37.17.16 Set Visible (setVisible)

Shows or hides component.

## Parameters

Name	Type	Description
isVisible	Boolean	Specifies whether to show or hide component

## Returned value

None

### 37.17.17 Set Width (setWidth)

Sets width of the component.

#### Note

This function has no effect if the current value is *auto*.

## Parameters

Table 105:

Name	Type	Description
width	Integer	Width of component in pixels

## Returned value

None

### 37.17.18 Show Loading State (showLoadingState)

Shows a loading indicator on the component.

## Parameters

Name	Type	Description
(optional) text	String	Text to show in the loading indicator

## Returned value

None

## 37.18 Connection

### 37.18.1 Get Children (getChildren)

Returns the child elements of data source descriptor.

The specific kind of child elements depends on the type of the data source descriptor. For example, a data source descriptor of type

`DataSourceType.FOLDER` can contain anything, while a data source descriptor of type `DataSourceType.INFOPROVIDER` usually contains queries, which can contain query views.

## Parameters

Name	Type	Description
parent	DataSourceDescriptor	Data source descriptor containing child elements

## Returned value

Array of DataSourceDescriptor. It contains the child elements of a data source descriptor as an array of data source descriptors.

## 37.18.2 Get Connections (getConnections)

Returns all available connections for the currently running platform.

## Parameters

None

## Returned value

Array of DataSourceConnection. It contains an array of data source connections of all available connections.

## 37.18.3 Get Root Folders (getRootFolders)

Returns the folders of the first hierarchy level of the data source tree.

## Parameters

Name	Type	Description
browse type	DataSourceBrowseType	Browse Type

## Returned value

Array of `DataSourceDescriptor`. It contains an array of data source descriptors that comprise the folders of the first hierarchy level of the data source tree. Each data source descriptor is of type `DataSourceType.FOLDER`. The child elements of a folder can be examined with method `getChildren`.

## 37.18.4 Get Selected Data Source (`getSelectedDataSource`)

Gets the selected data source of the data source browser.

### Note

To ensure dialog processing is completed, call this method in the script of the *On Data Source Browser Confirmed* event property only.

## Parameters

None

## Returned value

`DataSourceDescriptor`. It contains the data source descriptor of the selected data source.

### Example

In the following example, a data source that was selected in the data source browser is assigned to a data source:

```
var ds = CONNECTION_1.getSelectedDataSource();
DS_1.assignDataSource(ds.connection, ds.type, ds.name, true);
```

## 37.18.5 Get System (`getSystem`)

Returns the backend system of the connection.

## Parameters

None

## Returned value

DataSourceConnection. It contains a data source connection of the backend system.

## 37.18.6 Search Data Sources (searchDataSources)

Searches the backend system for data sources that match the search pattern.

### Note

Replaces method `APPLICATION.searchDataSources`.

## Parameters

Name	Type	Description
searchPattern	String	String pattern to search for

## Returned value

Array of DataSourceDescriptor. It contains an array of data source descriptors that match the search pattern.

## 37.18.7 Set System (setSystem)

Sets the backend system for the connection.

## Parameters

Name	Type	Description
system	DataSourceConnection	Backend system. Use either a DataSourceConnection object or a system name.

## Returned value

None

## 37.18.8 Show Data Source Browser (showDataSourceBrowser)

Displays the data source browser for the backend system.

The backend system is configured in the [Backend Connection](#) component.

### Note

This method opens the data source browser, but the script that called this method does not wait for the data source browser to be closed. See also method `getSelectedDataSource`.

## Parameters

Name	Type	Description
(optional) config.Json	String	Configuration JSON

You can also set a configuration string to configure the dialog directly in the script, following a JSON syntax, for example `CONNECTION.showDataSourceBrowser('{ "title": "JSON customized datasource browser", "width": "90%", "height": "90%", "defaultTab": "rolesTab", "hiddenTabs": ["foldersTab", "workspacesTab"]}');`

Make sure the JSON syntax follows the example (including ' ' ') and that there are no line breaks in the JSON string.

## Returned value

None

---

## 37.19 Context Menu (ContextMenu)

### 37.19.1 Get Click Area (getClickArea)

Returns the click area of what has been clicked.

#### Parameters

None

#### Returned value

ClickArea. It contains the click area of what has been clicked.

### 37.19.2 Get Component (getComponent)

Returns the component for which the context menu was opened.

#### Parameters

None

#### Returned Value

Component. It contains the component for which the context menu was opened.

### 37.19.3 Get Context (getContext)

Returns the data context of what has been clicked.

## Parameters

None

## Returned value

ResultSetSelectionByString. It contains the data context of what has been clicked.

## 37.19.4 Get Data Source (getSource)

Returns the data source for which the context menu was opened.

## Parameters

None

## Returned Value

DataSourceAlias. It contains the data source for which the context menu was opened.

## 37.19.5 Is Item Visible (isItemVisible)

Returns whether item is visible.

## Parameters

Name	Type	Description
itemName	ContextMenuItemId	Constant that specifies the item

## Returned Value

Boolean. True if item is visible, false if item is not visible.



## 37.19.6 Set Item Visible (setItemVisible)

Shows or hides an item.

### Parameters

Name	Type	Description
itemName	ContextMenuItemId	Constant that specifies the item
isVisible	Boolean	Specifies whether to show or hide the item

### Returned Value

None

## 37.20 Convert

A global object providing functions to convert data from one data type to another.

### 37.20.1 Float to String (floatToString)

Converts a floating-point number to a string by applying a formatting pattern.

The conversion uses the default locale (US English).

### Parameters

Table 106:

Name	Type	Description
number	Float	Floating-point number
(optional) formatString	String	Formatting pattern, for example "###,###,##0.00 "

## Returned value

String. It contains a string representation of the given floating-point number.

### Example

In the following example, the float number with the value 12345.678 is converted to the string "12,345.68 EUR":

```
Convert.floatToString(12345.678, "###,###,##0.00 EUR");
```

### Note

Keep the following points in mind:

- values are rounded to two places after the separator
- the hash characters (number signs) in the formatting pattern are filled according to the length of the number

### Example

In the following example, the floating-point with the value 0.6 is converted to the string "\$ 0.60":

```
Convert.floatToString(0.6, "$ ###,###,##0.00");
```

## 37.20.2 Float to String Using Locale (floatToStringUsingLocale)

Converts a floating-point number to a string using the locale of the current user.

## Parameters

Table 107:

Name	Type	Description
number	Float	Floating-point number
(optional) numberOfDecimals	Integer	Number of decimals used for rounding and formatting. If no number is specified, the default setting is used.

## Returned value

String. It contains a string representation of the given floating-point number.

### Example

In the following example, the floating-point number with the value 1234.56 is converted to the string "1,234.56" (the current user uses the US English locale) :

```
Convert.floatToStringUsingLocale (1234.56) ;
```

In the following example, the floating-point number with the value 1234.56 is converted to the string "1.234,56" (the current user uses the German locale) :

```
Convert.floatToStringUsingLocale (1234.56) ;
```

## 37.20.3 Index Of (indexOf)

Returns the index of the first occurrence of the specified substring within a string. Optionally, you can provide an index from where to start the search.

### Parameters

Table 108:

Name	Type	Description
string	String	String in which to search
searchFor	SearchExpression	Substring to search for
(optional) startIndex	Integer	Index from where to start searching (default:0)

### Returned value

Integer. It contains the index of the first occurrence of the specified substring within a string. The first character of that string has index 0. The returned value is -1 if the substring was not found.

### Example

In the following example, the returned index is 6:

```
Convert.indexOf ("Hello world!", "world");
```

In the following example, the returned index is -1:

```
Convert.indexOf ("Hello world!", "sailor");
```

## 37.20.4 Replace All (replaceAll)

Replaces each substring within a string that matches the given search expression with the given replacement.

### Parameters

Table 109:

Name	Type	Description
string	String	String in which to search
searchFor	SearchExpression	Substring to search for
replaceWith	String	String to be replaced for each match

### Returned value

String. It contains the given string with the applied replacements.

#### Example

In the following example, in the string `We stayed three days and three nights`, the substrings `three` are replaced with `four`. The returned string is `We stayed four days and four nights`.

```
Convert.replaceAll("We stayed three days and three nights","three", "four");
```

In the following example, the returned index is -1:

```
Convert.indexOf("Hello world!", "sailor");
```

## 37.20.5 String Length (stringLength)

Returns the length (number of characters) of the specified string.

## Parameters

Table 110:

Name	Type	Description
string	String	String to calculate the length of

## Returned value

Integer. It contains the number of characters in the specified string.

## 37.20.6 String to Float (stringToFloat)

Converts a string to a floating-point number.

The conversion uses the default locale (US English).

## Parameters

Table 111:

Name	Type	Description
string	String	String representing a floating-point number

## Returned value

Float. It contains the specified string converted to a floating-point number. If the conversion is not possible, NaN is returned.

### Example

In the following example, the string "1234.56" is converted to a floating-point number with the value 1234.56:

```
Convert.stringToFloat("1234.56");
```

## 37.20.7 String to Float Using Locale (stringToFloatUsingLocale)

Converts a string to a floating-point number using the locale of the current user.

### Parameters

Table 112:

Name	Type	Description
floatLiteral	String	String representing a floating-point number in the locale of the current user

### Returned value

Float. It contains the specified string converted to a floating-point number. If the conversion is not possible, NaN is returned.

#### Example

In the following example, the string "1,234.56" is converted to a floating-point number with the value 1234.56 (the current user is using the US English locale):

```
Convert.stringToFloatUsingLocale("1,234.56");
```

In the following example, the string "1.234,56" is converted to a floating-point number with the value 1234.56 (the current user is using the German locale):

```
Convert.stringToFloatUsingLocale("1.234,56");
```

## 37.20.8 String to Int (stringToInt)

Converts a string to an integer number.

## Parameters

Table 113:

Name	Type	Description
intLiteral	String	String representing an integer number

## Returned value

Integer. It contains the specified string converted to an integer number. If the conversion is not possible, NaN is returned.

### Example

In the following example, the string "1234" is converted to an integer number with the value 1234:

```
Convert.stringToInt("1234");
```

## 37.20.9 Substring (substring)

Returns a new string that is a substring of the specified string.

The substring is composed of the characters in the specified string between the specified startIndex and specified endIndex - 1. If endIndex is omitted, then the substring includes the characters up to the last character of the specified string.

## Parameters

Table 114:

Name	Type	Description
string	String	String from which a substring is extracted
startIndex	Integer	Start index of the substring, inclusive
(optional) endIndex	Integer	End index of the substring, exclusive

## Returned value

String. It contains the substring of the specified string.

## 37.20.10 URL Encode (urlEncode)

Encodes a string into `x-www-form-urlencoded` format. This is useful if the string is used as a parameter of an URL. Octets are encoded using the UTF-8 character encoding, for example `%C3%80`.

### Parameters

Table 115:

Name	Type	Description
string	String	String to encode

### Returned value

String. It contains the `x-www-form-urlencoded` encoded string.

#### Example

In the following example, the returned string is `Hello+world%21`:

```
Convert.urlEncode("Hello world!");
```

## 37.21 Crosstab

### 37.21.1 Get Selected Member (getSelectedMember)

Returns the member that is contained in the current selection of the crosstab.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the selected member.



## Return value

Member. It is the member that has been selected in the crosstab.

### Example

The following script statement sets all selected products in CROSSTAB\_1 as a filter to the same dimension in another data source alias DS\_2:

```
DS_2.setFilter("0PRODUCT", CROSSTAB_1.getSelectedMembers("Product"));
```

## 37.21.2 Get Selected Members (getSelectedMembers)

Returns the members that are contained in the current selection of the crosstab.

### Parameters

Table 116:

Name	Type	Description
dimension	Dimension	Dimension

### Returned value

Array of Member. It contains an array of members that have been selected in the crosstab.

## 37.21.3 Get Selection (getSelection)

Returns the selection.

### Parameters

None

---

## Returned value

ResultSetSelectionByString. It contains the selection

### 37.21.4 Is Column Resizing Enabled (isColumnResizingEnabled)

Returns whether column resizing is enabled.

## Parameters

None

## Returned value

Boolean

### 37.21.5 Is Conditional Formatting Visible (isConditionalFormattingVisible)

Returns whether the crosstab shows conditional formatting or not

## Parameters

None

## Returned value

Boolean

---

## 37.21.6 Is Hierarchy Navigation Enabled (isHierarchyNavigationEnabled)

Returns whether hierarchy navigation is enabled

### Parameters

None

### Returned value

Boolean

## 37.21.7 Is Sorting Enabled (isSortingEnabled)

Returns whether columns sorting is enabled .

### Parameters

None

### Returned value

Boolean

## 37.21.8 On Select (onSelect)

Calls the script that is executed when the user selects a row or column in the crosstab.

---

## Parameters

None

## Returned value

None

### 37.21.9 Remove Selection (removeSelection)

Removes the selection.

## Parameters

None

## Returned value

None

### 37.21.10 Reset All Column Widths (resetAllColumnWidths)

Resets all previously set column widths. All columns are rendered with automatic width calculation.

## Parameters

None

## Returned value

None

## 37.21.11 Reset Column Width (resetColumnWidth)

Resets any previously set column width. If there is still a default width over a width for a specific column that is removed, the default setting will be applied. Calling this method again will then remove the default width for that specific column, thus rendering the column with auto size again.

### Parameters

Name	Type	Description
columnIndex	Integer	The zero-based index of the column

### Returned value

None

## 37.21.12 Set Column Resizing Enabled (setColumnResizingEnabled)

Specifies if columns can be resized in the crosstab

### Parameters

Name	Type	Description
isEnabled	Boolean	Specifies if columns can be resized in the crosstab.

### Returned value

None

## 37.21.13 Set Column Width (setColumnWidth)

Sets the width for the specified column.

### Parameters

Name	Type	Description
columnIndex	Integer	The zero-based index of the column
width	Integer	The width in pixels

### Returned value

None

#### Example

In the following example, the width of column 2 (zero-based) is set to 200 pixels.

```
setColumnWidth(2, 200);
```

## 37.21.14 SetConditionalFormattingVisible (setConditionalFormattingVisible)

Sets if conditional formatting shall be displayed in the crosstab or not

### Parameters

Name	Type	Description
isVisible	Boolean	Specifies if conditional formatting shall be displayed in the crosstab or not.

## Returned value

None

## 37.21.15 Set Default Column Width (setDefaultColumnWidth)

Sets the width for all columns that do not have a column-specific setting. This default width can be overwritten for each column by calling the setColumnWidth function.

## Parameters

Name	Type	Description
width	Integer	The width in pixels

## Returned value

None

### Example

In the following example, the width of all columns is set to 50 pixels, except for column 2 that has a width of 200 pixels.

```
setColumnWidth(2, 200);
```

```
setDefaultColumnWidth(50);
```

## 37.21.16 Set Hierarchy Navigation Enabled (setHierarchyNavigationEnabled)

Specifies if hierarchies can be expanded/collapsed in the crosstab

## Parameters

Name	Type	Description
isEnabled	Boolean	Specifies if hierarchies can be expanded/collapsed in the crosstab.

## Returned value

None

### 37.21.17 Set Sorting Enabled (setSortingEnabled)

Specifies if columns can be sorted in the crosstab.

## Parameters

Name	Type	Description
isEnabled	Boolean	

## Returned value

None

### 37.21.18 Set Units and Scaling Factors Display (setUnitsAndScalingFactorsDisplay)

Specifies how units and scaling factors are displayed.



## Parameters

Name	Type	Description
unitsAndScalingFactorsDisplay	unitsAndScalingFactorsDisplay	Display mode for units and scaling factors

## Returned value

None

## 37.22 DataBoundComponent

### 37.22.1 Get Data Source (getDataSource)

Returns the assigned data source.

## Parameters

None

## Returned value

DataSourceAlias. It contains a data source alias.

### 37.22.2 Set Data Source (setDataSource)

Assigns a data source.

## Parameters

Table 117:

Name	Type	Description
dataSourceAlias	DataSourceAlias	Data source alias

## Returned value

None.

## 37.23 Data Cell (dataCell)

An object providing information about a data cell of a result set.

### formattedValue

Formatted value of this cell. The scaling factor has been already applied to this value. This value also reflects the user locale and number settings of the BEx Query Designer.

### scalingFactor

Scaling factor in powers of ten that has been applied to the value of this cell.

### unitOfMeasure

Unit of measure of the value of this cell, for example, a currency.

### value

Raw data value of this cell. The scaling factor has been already applied to this value.

## 37.24 Data Source Alias (DataSourceAlias)

### 37.24.1 Activate Hierarchy (activateHierarchy)

Activates the currently inactive hierarchy. If no hierarchy is inactive, no action will be performed.

#### Parameters

Name	Type	Description
dimension	Dimension	Dimension whose hierarchy is activated

#### Returned value

None

### 37.24.2 Assign Data Source (assignDataSource)

Assigns a data source.

This method can be used in combination with the data source alias property [Load in Script](#) to load data sources on demand.

#### Parameters

Name	Type	Property
dataSourceConnection	DataSourceConnection	Connection alias
dataSourceType	DataSourceType	Type of data source
dataSourceName	DataSourceName	Name of data source
(optional) isLoadNow	Boolean	Specifies whether data source is loaded after assignment.

## Returned value

None

### Example

In the following example a new data source is assigned and the data is loaded immediately:

```
DS_1.assignDataSource("Q99", "QUERY", "AM_SALES");
```

### Example

In the following example a new data source is assigned and loading the data is deferred (see also method `loadDataSource`):

```
DS_1.assignDataSource("Q99", "QUERY", "AM_SALES", false);
```

This prompts all components which are linked to it to signal, that the data was not loaded yet. The data source is "greyed out" as if no data source has been assigned.

## 37.24.3 Assign Hierarchy (assignHierarchy)

Assigns hierarchy to dimension.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source alias to be displayed as a hierarchy
hierarchy	Hierarchy	Hierarchy to be assigned

## Returned value

None

## 37.24.4 Clear All Filters (clearAllFilters)

Removes filters for all dimensions.

## Parameters

None

## Returned value

None

### 37.24.5 Clear Filter (clearFilter)

Removes the filter for a dimension.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source alias with filter to be removed.

## Returned value

None

### 37.24.6 Collapse Node (collapseNode)

Collapses the specified node. The node has to be visible at the time of execution or no action will be performed.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the hierarchy whose node is collapsed

Name	Type	Description
value	HierarchyNode	Node which is collapsed

## Returned value

None

## 37.24.7 Configure Input Readiness (configureInputReadiness)

Configures the input readiness (if supported) of a data source used for planning.

## Parameters

Name	Type	Description
isInputReady	Boolean	If true then the data source is set to <code>input_ready</code> if it has been loaded and no query locks are in place. If false then the data source is set to display mode.

## Returned value

None

## 37.24.8 Copy Filter (copyFilter)

Copies a filter value of a dimension from a data source.

### Note

Values of a static filter will not be copied, even if the dimension is available as a filter in the target data source. Therefore, values of a static filter need to be retrieved from the data source and applied to the target data source separately. Values of static filters can be retrieved with `getStaticFilterExt` and `getStaticFilterText` from dimensions of a data source.

## Parameters

Name	Type	Description
targetDimension	Dimension	Dimension to copy filters to
dataSourceAlias	DataSourceAlias	DataSourceAlias to copy filters from
(optional) sourceDimension	Dimension	Dimension to copy filters from

## Returned Value

None

### 37.24.9 Copy Filters (copyFilters)

Copies filter values of common dimensions from a data source.

#### **i** Note

Values of static filters will not be copied, even if the dimension is available as a filter in the target data source. Therefore, values of static filters need to be retrieved from the data source and applied to the target data source separately. Values of static filters can be retrieved with `getStaticFilterExt` and `getStaticFilterText` from dimensions of a data source.

## Parameters

Name	Type	Description
dataSourceAlias	DataSource Alias	Data source alias to copy filters from
isCopyMeasuresFilter (optional)	Boolean	Specifies whether the filter values for measures are copied as well (default: false).

## Returned value

None

## 37.24.10 Deactivate Hierarchy (deactivateHierarchy)

Deactivates the currently active hierarchy. If no hierarchy is active, no action will be performed.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension whose hierarchy is deactivated

### Returned value

None

## 37.24.11 Expand Node (expandNode)

Expands the specified node up to a defined level. The node has to be visible at the time of execution or no action will be performed.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the hierarchy whose node is expanded
value	HierarchyNode	Node that is expanded
(optional) levels	Integer	Number of levels by which the node is expanded

#### Note

Use the following syntax to specify a node: **HIER\_NODE/NodeType/Node**

- for hierarchy nodes in BW systems

You can find the relevant hierarchy node type in the hierarchy maintenance in the BW system. It is either `0HIER_NODE` for text nodes or the technical name of the external InfoObject in the hierarchy. The node will be the internal fully compounded member key; for example, K42013001 for the fiscal year period 1 in 2013 (fiscal year variant K4). If the value consists of a `/`, remember to escape the `/` with a backslash (`\`)



- for hierarchy nodes in SAP HANA systems

The node type for parent child hierarchies is the alias name of the child attribute. The node type for level-based hierarchies is the alias name of the corresponding level. If the value consists of a /, remember to escape the / with a backslash (\)

## Returned value

None

## 37.24.12 Export (export)

Exports the resultSet of the data source.

## Parameters

Name	Type	Description
exportType	DataSourceExportType	Type of export format

## Returned Value

None

## 37.24.13 Get Assigned Hierarchy (getAssignedHierarchy)

Returns the assigned hierarchy.

### **i** Note

Before calling this method, check with method `isHierarchyAssigned()` if the hierarchy is assigned.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension if of the hierarchy

## Returned value

Hierarchy. It contains the assigned hierarchy.

## 37.24.14 Get Conditional Formats (getConditionalFormats)

Returns the names of the conditional formats.

## Parameters

None

## Returned Value

Array of ConditionalFormatId. It contains the names of the conditional formats.

## 37.24.15 Get Conditional Format Name (getConditionalFormatName)

Returns the name of a conditional format.

## Parameters

Name	Type	Description
id	ConditionalFormatId	id of the conditional format

## Returned Value

String. It contains the name of the conditional format.

### 37.24.16 Get Conditional Format Value (getConditionalFormatValue)

Returns the conditional format value applied to a data cell at runtime

## Parameters

Name	Type	Description
measure	Measure	Measure corresponding to returned value
selection	MultiDimFilter	Combination of dimension members in internal key format that specifies the data cell selection

## Returned value

Integer. A value of 0 indicates that no conditional formatting is applied. Values of 1 to 9 indicate the priority level of the highest applied rule (see Conditional Formatting documentation).

### 37.24.17 Get Conditional Format Value Ext (getConditionalFormatValueExt)

Returns the conditional format value applied to a data cell at runtime using external member keys.

## Parameters

Name	Type	Description
measure	Measure	Measure corresponding to returned value

Name	Type	Description
selection	MultiDimFilterExt	Combination of dimension members in external key format that specifies the data cell selection

## Returned value

Integer. A value of 0 indicates that no conditional formatting is applied. Values of 1 to 9 indicate the priority level of the highest applied rule (see Conditional Formatting documentation).

## 37.24.18 Get Data (getData)

Returns a single data cell from the result set.

### Note

- This method specifies members in the internal format.
- For dimensions that are not specified, their aggregates will be used, provided that the corresponding aggregates exist in the result set.
- The requested data must be part of the retrieved result set. This is especially important when requesting hierarchy nodes. The hierarchy must be expanded in advance to show the required hierarchy node.

## Parameters

Table 118:

Name	Type	Description
measure	Measure	Measure corresponding to returned value
selection	MultiDimFilter	Combination of dimension members in internal key format that specifies the data cell selection

## Returned value

DataCell - an object providing information about a data cell of a result set.

### Example

In the following example, the method returns the data cell containing the sales revenue for product P4711 of the first quarter of 2012:

```
DS_1.getData("Sales", {"OMATERIAL":"P4711", "OCALYEAR":"2012",  
"OQUARTER":"01.2012"});
```

In the following example, the method returns the data cell containing the aggregate sales revenue for product P4711 in 2012:

```
DS_1.getData("Sales", {"OMATERIAL":"P4711", "OCALYEAR":"2012"});
```

The value for dimension OQUARTER was not specified,. Therefore the aggregate value is returned, provided that it exists in the result set. You can only omit dimensions from right to left, but not in-between. For example, you can omit the value for dimension OCALYEAR in the first example.

## Related Information

[Data Cell \(dataCell\) \[page 570\]](#)

## 37.24.19 Get Data as String (getDataAsString)

Returns a single data cell value from the query result set.

### Format of parameter selection

- To specify dimensions, use the JSON-format: You specify the dimension name and the value, separated by a colon, for example {"DIM1": "Member1", "DIM2": "Member2"}.
- To specify hierarchy nodes, use the internal fully compounded format of the hierarchy node in the form **HIERARCHY\_NODE/Node type/**, for example **HIERARCHY\_NODE/OHIER\_NODE/ROOT**.
- If you want to specify the members with the external format, use the method `getDataAsStringExt` instead.

#### Note

- For dimensions that are not specified, the result will used - provided that the corresponding result line exists in the result set.
- The data needs to be part of the retrieved result set. This is especially important when requesting hierarchy nodes. The hierarchy needs to be expanded in advance to show the required hierarchy node.

## Parameters

Name	Type	Description
measure	Measure	Measure corresponding to returned value
selection	MultiDimFilter	Combination of dimension members in internal key format that specifies the data cell selection

## Returned value type

String. It is formatted based on the user's regional settings, scaling factors and unit of measurement.

### Example

In the following example the method returns the sales revenue for product P4711 of the first quarter of 2012:

```
DS_1.getDataAsString("Sales", {"OMATERIAL":"P4711", "OCALYEAR":"2012",  
    "OQUARTER":"01.2012"});
```

In the following example the method returns the aggregate sales revenue for product P4711 in 2012:

```
DS_1.getDataAsString("Sales", {"OMATERIAL":"P4711", "OCALYEAR":"2012"});
```

The value for dimension OQUARTER was not specified, thus the aggregate value is returned, provided it exists in the result set. When retrieving values, make sure that the dimension values specify an existing value of the result set. For example, when you omit the value for dimension OCALYEAR (thus indicating an aggregate value for this dimension) in the first example then no value is returned, because there is no cell that matches the specified dimension values.

## 37.24.20 Get Data as String Ext (getDataAsStringExt)

Returns a single data cell value from the query result set using external member keys. The returned value is a string that is formatted based on the user's regional settings, scaling factors and unit of measurement.

To specify dimensions, use the JSON-format: You specify the dimension name and the value, separated by a colon, for example {"DIM1": "Member1", "DIM2": "Member2"}.

### Note

- For dimensions that are not specified, their aggregates will be used " provided that the corresponding aggregates exists in the result set.
- The requested data needs to be part of the retrieved result set. This is especially important when requesting hierarchy nodes. The hierarchy needs to be expanded in advance to show the required hierarchy node.

## Parameters

Name	Type	Description
measure	Measure	Measure corresponding to returned value
selection	MultiDimFilterExt	Combination of dimension members in external key format that specifies the data cell selection

## Returned value

String. It is formatted based on the user's regional settings, scaling factors and unit of measurement.

### Example

In the following example the method returns the sales revenue for product P4711 of the first quarter of 2012:

```
DS_1.getDataAsStringExt("Sales", {"OMATERIAL":"P4711", "OCALYEAR":"2012",  
    "OQUARTER":"01.2012"});
```

In the following example the method returns the aggregate sales revenue for product P4711 in 2012:

```
DS_1.getDataAsStringExt("Sales", {"OMATERIAL":"P4711", "OCALYEAR":"2012"});
```

The value for dimension OQUARTER was not specified, thus the aggregate value is returned, provided it exists in the result set. When retrieving values, make sure that the dimension values specify an existing value of the result set. For example, when you omit the value for dimension OCALYEAR (thus indicating an aggregate value for this dimension) in the first example then no value is returned, because there is no cell that matches the specified dimension values.

## 37.24.21 Get Decimal Places (getDecimalPlaces)

Returns the number of decimal places displayed for a measure.

## Parameters

Name	Type	Description
measure	Measure	Measure for which the number of decimal places is returned

## Returned value

Integer. It contains the number of decimal places of the measure.

### 37.24.22 Get Dimension Text (getDimensionText)

Returns the localized text of a dimension.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension with the text to be returned

## Returned value

String. It contains the localized text of the dimension.

### 37.24.23 Get Dimensions (getDimensions)

Returns an array of dimensions of an axis.

## Parameters

Name	Type	Description
(optional) axis	Axis	Axis for which to get the dimensions

## Returned value

DimensionArray. It contains the dimensions of the axis. If no axis is specified, returns all dimensions.



## 37.24.24 Get Filter Ext (getFilterExt)

Returns the filter value of a data source dimension in external key format.

### Note

Use this method to pass the filter value to other methods.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source

### Returned value

String. It contains the filter value in external key format.

## 37.24.25 Get Filter Text (getFilterText)

Returns the filter value of a data source dimension.

### Note

Use this method to display the filter value.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source

### Returned value

String. It contains the filter value.

---

## 37.24.26 Get Hierarchies (getHierarchies)

Returns an array of available hierarchies for a dimension.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the hierarchies

### Returned value

HierarchyArray. It contains the available hierarchies.

## 37.24.27 Get Info (getInfo)

Returns data source information, for example, the key date or the technical name.

### Parameters

None

### Returned value

DataSourceInfo. It contains information about the data source.

## 37.24.28 Get Measures Dimension (getMeasuresDimension)

Returns the measures dimension.

## Parameters

None

## Returned value

Dimension. It contains the measures dimension.

## 37.24.29 Get Measure Filters (getMeasureFilters)

Returns the names of the measures filters.

## Parameters

None

## Returned Value

Array of MeasureFilterId. It contains the names of the measures filters.

## 37.24.30 Get Measure Filter Name (getMeasureFilterName)

Returns the name of a measures filter.

## Parameters

Name	Type	Description
id	MeasureFilterId	id of the measures filter

## Returned Value

String. It contains the name of the measures filter.

## 37.24.31 Get Members (getMembers)

Returns an array of members of a dimension.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension for which to get the members
maxNumber	Integer	Maximum number of members to return (should not be larger than 100). If maxNumber is 0 or negative, then maxNumber is set to a default value of 50.
(optional) attributes	Array of DimensionAttribute	Attributes to be fetched along with the members. Avoids backend access when calling <code>getAttributeMember</code> on the returned members - at the cost of more data being processed and retrieved during this call.

### Returned value

MemberArray. It contains the members of the dimension.

## 37.24.32 Get Member Display (getMemberDisplay)

Returns the member display for the dimension of the data source.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source for which to retrieve the member display.

## Returned value

MemberDisplay


### 37.24.33 Get Member List (getMemberList)

Retrieves a list of dimension members.

#### Note

If you define *allMemberText* for this method and use **INTERNAL\_KEY** as the *memberPresentation*, and if you want to select allMemberText by using the method setSelected value, the value for setSelectedValue is "(ALL\_MEMBERS)". If you choose a different presentation type, the value is an empty string.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source alias
memberPresentation	MemberPresentation	Presentation of member keys
memberDisplay	MemberDisplay	Textual display of members <div> <b>Caution</b> It is not supported to choose NO_DISPLAY as member display type.</div>
maxNumber	Integer	Maximum number of members to be returned (should not be larger than 100). If maxNumber is 0 or negative, then maxNumber is set to a default value of 50.
(optional) allMemberText	String	Text of the item that represents all members. If no text is specified, the item is not added to the list

## Returned value

ValueTextList. It contains a list of dimension members that can be used to set the items list of a dropdown or another selection component.

### Example

In the following example, a list box is populated with members from a data source:

```
LISTBOX_1.setItems(DS_1.getMemberList("0D_CUSTOMER",  
MemberPresentation.INTERNAL_KEY, MemberDisplay.TEXT_KEY, 100));
```

## 37.24.34 Get Negative Number Display (getNegativeNumberDisplay)

Returns how negative numbers are displayed.

### Parameters

None

### Returned value

NegativeNumberDisplay. It contains the display mode for negative numbers of the data source.

## 37.24.35 Get Scaling Factor (getScalingFactor)

Returns the scaling factor for a measure.

### Parameters

Name	Type	Description
measure	Measure	Measure for which the scaling factor is returned

### Returned value

Scaling. It contains the scaling factor of the measure.

## 37.24.36 Get Static Filter Ext (getStaticFilterExt)

Returns the static filter value of a data source dimension in external key format.

### Note

Use this method to pass the filter value to other methods.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source

### Returned value

String. It contains the static filter value in external key format.

## 37.24.37 Get Static Filter Text (getStaticFilterText)

Returns the static filter value of a data source dimension.

### Note

Use this method to display the static filter value.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source

### Returned value

String. It contains the static filter value.

## 37.24.38 Get Text (getText)

Returns the description of the data source as specified by the application developer.

This text will, for example, be displayed in the prompt dialog if the application runs in unmerged prompts mode.

This allows you to specify different descriptions for multiple instances of the same query: If DS\_1 and DS\_2 both refer to the same query, they can still have different texts assigned.

### Note

Do not confuse this text with the backend text of the underlying query.

### Parameters

None

### Returned value

String.

## 37.24.39 Get Totals Display (getTotalsDisplay)

Returns how the totals of a dimension are displayed.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension for which the totals display is returned

### Returned value

TotalsDisplay. It contains the totals display mode of the dimension.



## 37.24.40 Get Totals Position (getTotalsPosition)

Returns the position of the totals in rows or columns, relative to the members.

### Parameters

Name	Type	Description
axis	Axis	Axis for which the totals position is returned

### Returned value

TotalsPosition. It contains the totals position of the axis.

## 37.24.41 Get Variables (getVariables)

Returns the available variables.

### Parameters

None

### Returned value

VariableArray. It contains the available variables.

## 37.24.42 Get Variable Value (getVariableValue)

Returns the variable value of a data source variable in internal key format.

#### Note

Use this method to pass the variable value to other methods.

## Parameters

Name	Type	Description
variable	Variable	Data source variable

## Returned value

String. It contains the variable value in internal key format.

### 37.24.43 Get Variable Value Ext (getVariableValueExt)

Returns the variable value of a data source variable in external key format.

#### Note

Use this method to pass the variable value to other methods.

## Parameters

Name	Type	Description
variable	Variable	Data source variable

## Returned value

String. It contains the variable value in external key format.

### 37.24.44 Get Variable Value Text (getVariableValueText)

Returns the variable value of a data source variable.

#### Note

Use this method to display the variable.

## Parameters

Name	Type	Description
variable	Variable	Data source variable

## Returned value

String. It contains the variable value.

## 37.24.45 Get Zero Display (getZeroDisplay)

Returns how zero values are displayed.

## Parameters

None

## Returned value

ZeroDisplay. It contains the display mode for zero values of the data source.

## 37.24.46 Get Zero Display Custom Text (getZeroDisplayCustomText)

Returns the custom text that represents zero values.

## Parameters

None

## Returned value

String. It contains the custom text that represents zero values. Is an empty string ("" ) if the display mode for zero values is not `CUSTOM`.

### 37.24.47 Is Conditional Format Active (isConditionalFormatActive)

Returns true if the conditional format is active and false if it is not active.

## Parameters

Name	Type	Description
id	ConditionalFormatId	id of the conditional format

## Returned Value

Boolean. True if the conditional format is active and false if it is not active.

### 37.24.48 Is Hierarchy Active (isHierarchyActive)

Returns true if a hierarchy is active and false if not.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the hierarchy

## Returned value

Boolean. True if a hierarchy is active and false if it is not.

---

## 37.24.49 Is Initialized (isInitialized)

Returns true if a data source is assigned and loaded.

### Parameters

None

### Returned value

None

## 37.24.50 Is Input Ready (isInputReady)

Returns whether the data source is input ready.

### Note

The returned value may differ from the configured input readiness if the data source has not been loaded or query locks are in place.

### Parameters

None

### Returned value

Boolean. True if the data source is input ready, false if it is not.

## 37.24.51 Is Measure Filter Active (isMeasureFilterActive)

Returns true if the measures filter is active and false if it is not active.

## Parameters

Name	Type	Description
id	MeasureFilterId	id of the measures filter

## Returned Value

Boolean. True if the measures filter is active and false if it is not active.

## 37.24.52 Is Result Set Empty (isResultSetEmpty)

Returns true if a data source is not initialized or the result set is empty or too large (size restriction for result sets).

## Parameters

None

## Returned value

None

## 37.24.53 Load Data Source (loadDataSource)

Loads the assigned data source.

This method can be used in combination with the data source property *Load in Script* to load data sources on demand with the method `assignDataSource`.

## Parameters

None

## Returned value

None

### Example

#### loadDataSource in combination with assignDataSource

In the following example the assignment of a data source and the loading of its data are separated; thus the loading of the data can be deferred. The data source is assigned with method `assignDataSource(..., false)`, yet the data are actually loaded with method `loadDataSource`:

```
DS_1.assignDataSource("Q99", DataSourceType.QUERY, "SAMPLE_QUERY", false);  
DS_1.loadDataSource();
```

### Example

#### loadDataSource in combination with the property Load in Script

In the following example the data of a data source are loaded with the property *Load in Script*. You use the data source alias DS\_2. The property Load in Script of this data source has been set to true. To load DS\_2 you use method `loadDataSource`:

```
DS_2.loadDataSource();
```

## 37.24.54 Move Dimension After (moveDimensionAfter)

Adds a dimension after another dimension.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source alias to be added to the axis.
otherDimension	Dimension	The above dimension is added after this dimension on the rows or columns axis. If otherDimension is not on the rows or columns axis, the method call is ignored.

## Returned value

None

## 37.24.55 Move Dimension Before (moveDimensionBefore)

Adds a dimension in front of another dimension.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source alias to be added to the axis.
otherDimension	Dimension	The above dimension is added in front of this dimension on the rows or columns axis. If otherDimension is not on the rows or columns axis, the method call is ignored.

### Returned value

None

## 37.24.56 Move Dimension to Columns (moveDimensionToColumns)

Moves a dimension to a specific position on the columns axis (drill-down in columns).

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source alias to be moved to the axis.
(optional) position	Integer	Position on the axis (position 0 is the first position on the axis). If there is already a dimension at this position, it is replaced. If the parameter is not specified, the dimension is moved to the last position on the axis. If the specified position index is higher than the number of dimensions on the axis, the



Name	Type	Description
		dimension is also moved to the last position on the axis.

## Returned value

None

## 37.24.57 Move Dimension to Rows (moveDimensionToRows)

Moves a dimension to a specific position on the rows axis (drill-down in rows).

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source alias to be moved to the axis.
(optional) position	Integer	Position on the axis (position 0 is the first position on the axis). If there is already a dimension at this position, it is replaced. If the parameter is not specified, the dimension is moved to the last position on the axis. If the specified position index is higher than the number of dimensions on the axis, the dimension is also moved to the last position on the axis.

## Returned value

None

## 37.24.58 Open Prompt Dialog (openPromptDialog)

Opens Prompt dialog box.

## Parameters

Table 119:

Name	Type	Description
width	Integer	Width of Prompt dialog in pixels
height	Integer	Height of Prompt dialog in pixels

## Returned value

None

### Example

In the following example, the Prompt dialog is opened:

```
DS_1.openPromptDialog(400, 500);
```

## 37.24.59 Reload Data (reloadData)

Reloads current data from data source

This method is useful with real-time data sources where the actual data is updated during the user's session; the method enables the refresh of the data at runtime although nothing else besides the data has changed, that is, neither filters and drill-down have changed, nor the data source structure.

## Parameters

None

## Returned value

None

## 37.24.60 Remove Dimension (removeDimension)

Removes a dimension from the rows or columns axis.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source alias to be removed from the rows or columns axis. If the dimension is not on the rows or column axis, the method call is ignored.

### Returned value

None

## 37.24.61 Set Conditional Format Active (setConditionalFormatActive)

Activates or deactivates a conditional format.

### Parameters

Name	Type	Description
id	ConditionalFormatId	id of the conditional format
isActive	Boolean	Specifies whether the conditional format is active

### Returned Value

None

## 37.24.62 Set Decimal Places (setDecimalPlaces)

Configures how many decimal places are displayed for a measure.

### Parameters

Name	Type	Description
measure	Measure	Measure to be configured
decimals	Integer	Number of decimal places displayed (possible values: -1 (use default), 0 - 9)

### Returned value

None

## 37.24.63 Set Drill Level (setDrillLevel)

Sets the drill level for the hierarchy. All data is reloaded.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the hierarchy, whose drill level is set
level	Integer	Number of levels to drill down

### Returned value

None

## 37.24.64 Set Filter (setFilter)

Sets a filter for a dimension in the internal key format.

The existing filter is replaced.

You can specify filter values in the following ways:

- For a single member, specify the internal key, for example "US".
- For multiple single members, specify an array of internal keys, for example ["US", "DE"].
- For an interval, use the JSON-format with "low" and "high", for example {"low": "0", "high": "10"}.
- For an open range , use the JSON-format with either "low" or "high", for example {"high": "10"}.
- For a combination of multiple single members, intervals, and ranges , use for example ["0", { "low": "10", "high": "50"}, { "low": "100" } ] .
- For hierarchy nodes, prefix the internal, fully compounded format with HIERARCHY\_NODE/Node Type/, for example HIERARCHY\_NODE/OHIER\_NODE/ROOT.

If you want to specify filter values with the external key format use the method `setFilterExt` instead.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source alias to be filtered
value	Array of Filter	Filter to be set

### Returned value

None.

## 37.24.65 Set Filter Ext (setFilterExt)

Sets a filter for a dimension in the external key format.

The existing filter is replaced.

You can specify the input string as a filter value. The input string syntax allows you to specify complex selections, such as intervals and excluding selections. In contrast to method `setFilter`, the members are specified in the external fully compounded key member presentation format, for example **K4/001.2012**.

## Note

External key values might be dependent on the user's regional settings. If the application is intended to be used by a multi-lingual group of users, or if you want to specify filter values with the internal key (for example, to avoid localization problems), use the method `setFilter` instead.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source alias to be filtered
value	InputString	Filter to be set

You can specify the input string as a filter value. This syntax allows you to specify more complex selections (such as intervals and excluding selections). In contrast to the method `setFilter`, the members are specified in the external, fully compounded key member presentation format (for example, **K4/001.2012**).

## Syntax for entering values

Follow these syntax rules when you enter members for filtering.

Description	Example
equal to	15
multiple	15;18;20
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
less than	<12
exclude values less than <value limit>	!<12
less than or equal to	<=12
exclude values less than or equal to <value limit>	!<=12
complex combination	15;10-15;20-25;1-5;>8; etc.
dimension hierarchy node	+<Dimension Attribute>(<Technical Name of Dimension>), e.g. +ELEMENT1(WBS_ELEMENT)

Description	Example
text hierarchy node	+<Technical Name of Hierarchy Node>(OHIER_NODE) , e.g. +EUROPE(OHIER_NODE)

## Returned value

None

## 37.24.66 Set Measure Filter Active (setMeasureFilterActive)

Activates or deactivates a measures filter.

## Parameters

Name	Type	Description
id	MeasureFilterId	id of the measures filter
isActive	Boolean	Specifies whether the measures filter is active

## Returned Value

None

## 37.24.67 Set Member Display (setMemberDisplay)

Sets the member display for the data source dimension.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source to which a new member display is set.
memberDisplay	MemberDisplay	Display mode of the members

## Returned value

None

### 37.24.68 Set Negative Number Display (setNegativeNumberDisplay)

Specifies how negative numbers are displayed.

## Parameters

Name	Type	Description
negativeNumberDisplay	NegativeNumberDisplay	Display mode for negative values

## Returned value

None

### 37.24.69 Set Scaling Factor (setScalingFactor)

Configures the scaling factor applied to a measure.



## Parameters

Name	Type	Description
measure	Measure	Measure to be configured
scaling	Scaling	Scaling factor applied

## Returned value

None

### 37.24.70 Set Totals Display (setTotalsDisplay)

Configures how the totals of a dimension are displayed.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension for which the totals display is configured
totalsDisplay	TotalsDisplay	Totals display

## Returned value

None

### 37.24.71 Set Totals Position (setTotalsPosition)

Configures the position of the totals in rows or columns, relative to the members.

## Parameters

Name	Type	Description
axis	Axis	Axis for which the totals position is configured
totalsPosition	TotalsPosition	Position where the totals are displayed

## Returned value

None

### 37.24.72 Set Variable Value (setVariableValue)

Sets data source variable values in internal key format, then executes the data source query again.

When the merge flag is set, variable values are set in the application for all data sources containing the variable, when merge is not set, variable values are set for the data source.

With this method you can set only single members or hierarchy nodes in the internal key format as variable values.

If you want to set variable values for the following cases, use `setVariableValueExt` instead:

- formula variables
- text variables
- hierarchy variables
- variables representing precalculated values sets (buckets)
- multiple single values
- intervals
- variables representing a selection option
- using external format

For hierarchy nodes, prefix the internal non-compounded presentation of the hierarchy node with `HIERARCHY_NODE/nodeType/`, for example `HIERARCHY_NODE/0HIER_NODE/ROOT`.

#### Note

It is not possible to use the Crosstab component and the Chart component method `getSelectedMember` for setting variables for compounded dimensions.

## Parameters

Table 120:

Name	Type	Description
variable	Variable	Data source variable
value	VariableValue	Variable value in internal key format

## Returned value

None

### Example

In the following example, a fixed string value is set to a variable:

```
DS_1.setVariableValue("0VAR", "4711");
```

In the following example the value is set to a variable returned from a component selection:

```
DS_1.setVariableValue("0VAR", DROPDOWN_1.getSelectedValue());
```

## 37.24.73 Set Variable Value Ext (setVariableValueExt)

Sets data source variable values in external key format, then executes the data source query again.

When the merge flag is set, variable values are set in the application for all data sources containing the variable, when merge is not set, variable values are set for the data source.

Use this method if other selections than single members or hierarchy nodes need to be set as variable value.

### Note

External key values might be dependent on the user's regional settings. If the application is Intended to be used by a multi-lingual group of users, it is therefore recommended to use the `setVariableValue` method when possible.

## Parameters

Table 121:

Name	Type	Description
variable	Variable	Data source variable
value	InputString	Variable value in external key format

## Returned value

None

### Example

In the following example, a fixed string value is set to a variable:

```
DS_1.setVariableValueExt("OVAR", "4711");
```

In the following example the value returned from a component selection is set to a variable:

```
DS_1.setVariableValueExt("OVAR", DROPDOWN_1.getSelectedValue());
```

## 37.24.74 Set Zero Display (setZeroDisplay)

Specifies how zero values are displayed.

## Parameters

Name	Type	Description
zeroDisplay	ZeroDisplay	Display mode for zero values
(optional) userDefinedText	String	Specifies the text if the display mode for zero values is CUSTOM (default: "")

## Returned value

None

## 37.24.75 Sort by Attribute (sortByAttribute)

Sorts the result set by a specified attribute of a specified dimension in ascending or descending order.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension whose attribute is used to sort the results
attribute	DimensionAttribute	Dimension attribute used to sort the results
isSortAscending	Boolean	Specifies the sort order. If true then sort in ascending order, if false then sort in descending order.

### Returned value

None

## 37.24.76 Sort by Hierarchy (sortByHierarchy)

Sorts the result set by the natural sort order of the assigned hierarchy of a dimension.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension with the assigned hierarchy used to sort the results

### Returned value

None

## 37.24.77 Sort by Measure (sortByMeasure)

Sorts the result set by a specified measure in ascending or descending order.

### Parameters

Name	Type	Description
measure	Measure	Measure used to sort the results
isSortAscending	Boolean	Specifies the sort order. If <code>true</code> then sort in ascending order, if <code>false</code> then sort in descending order.

### Returned value

None

## 37.24.78 Sort By Member (sortByMember)

Sorts the result set by the members of a specified dimension in ascending or descending order.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension whose members are used to sort the results
isSortAscending	Boolean	Specifies the sort order. If <code>true</code> then sort in ascending order, if <code>false</code> then sort in descending order..
(optional) isSortByKey	Boolean	If the member display of the dimension is set to <code>KEY_TEXT</code> or <code>TEXT_KEY</code> , the sort order is refined: If <code>true</code> then sort by key, if <code>false</code> then sort by text (default: <code>false</code> ).

## Returned value

None

## 37.24.79 Swap Dimensions (swapDimensions)

Swaps two dimensions.

At least one of the dimensions must be on the rows or columns axis, otherwise the method call is ignored.

## Parameters

Name	Type	Description
dimension1	Dimension	Dimension of the data source alias to be swapped with dimension2.
dimension2	Dimension	Dimension of the data source alias to be swapped with dimension1.

## Returned value

None

## 37.24.80 Unassign Hierarchy (unassignHierarchy)

Unassigns hierarchy from dimension.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source alias to be not displayed as a hierarchy anymore

---

## Returned value

None

## 37.25 Data Source Info (dataSourceInfo)

An object providing information about a data source.

### **createdBy**

Name of the user who created the data source.

### **dataSourceName**

Name of the data source.

### **infoProviderDescription**

Description of the InfoProvider.

### **infoProviderTechnicalName**

Technical name of the InfoProvider.

### **keyDate**

Key date.

### **lastChangedAt**

Date of the last change made to the data source.



---

## **lastChangedBy**

Name of the user who made the last change to the data source.

## **lastDataUpdate**

Date of the last data update.

## **lastDataUpdateMaximum**

Date of the last data update of all InfoProviders in a MultiProvider. Example: A MultiProvider contains three InfoProviders. The last date when all three InfoProviders were updated is the LastDataUpdateMaximum.

## **lastRefreshedAt**

Date of the last data source refresh.

## **queryDescription**

Description of the query.

## **queryTechnicalName**

Name of the query.

## **system**

Name of the system.

## **user**

Name of the current user.

## 37.26 DataSourceConnection

An object representing an available backend connection.

### Fields

Table 122:

Name	Type	Description
name	String	Name, alias, or key of the backend connection
text	String	Description text
type	DataSourceConnectionType	Connection type

## 37.27 DataSourceDescriptor

### connection

The data source connection.

### description

The description of the data source.

### hasChildren

Specifies whether this data source descriptor has child elements

---

## **name**

The name of the data source.

## **type**

The type of the data source (INFOPROVIDER, QUERY or VIEW).

## **37.28 DataSourceName**

Name of an existing data source. A data source could be, for example, a query, a query view, or an InfoProvider.

## **37.29 Date Field**

### **37.29.1 Get Date (getDate)**

Returns the selected date.

#### **Parameters**

None

#### **Returned value**

String. It contains the date in the format "yyyymmdd".

### **37.29.2 Is Enabled (isEnabled)**

Returns whether component is enabled.

---

## Parameters

None

## Returned value type

Boolean. True if component is enabled or false if component is disabled.

## 37.29.3 On Select (onSelect)

Calls the script that is executed when the user selects a date.

## Parameters

None

## Returned value

None

## 37.29.4 Set Date (setDate)

Sets selected date.

If the date is invalid then the Date Field component is empty.

## Parameters

Name	Type	Description
date	String	Date to be set in the format "yymmdd"

## Returned value

None

## 37.29.5 Set Enabled (setEnabled)

Enables or disables component.

Disabled components do not allow user interaction.

## Parameters

Name	Type	Description
isEnabled	Boolean	Specifies whether to enable or disable the component

## Returned value

None

## 37.30 Dimension

A dimension groups data of a multi-dimensional cube in categories such as, for example, company code, product, customer group, fiscal year, period, or region.

## isMeasuresDimension

Indicates if this is the measures dimension.

## name

The dimension's name.

---

## **text**

The dimension's text.

## **37.31 Dimension Filter**

### **37.31.1 Cancel (cancel)**

Discards the filter values that have not been applied yet.

#### **Parameters**

None

#### **Returned value**

None

### **37.31.2 Get Dimension Key (getDimensionKey)**

Returns the technical key of the dimension.

#### **Parameters**

None

#### **Returned value**

String. It contains the technical key of the dimension.

---

### 37.31.3 Get Dimension Name (getDimensionName)

Returns the name of the dimension.

#### Parameters

None

#### Returned value

String. It contains the name of the dimension.

### 37.31.4 Set Dimension (setDimension)

Set the dimension.

#### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the data source to be set

#### Returned value

None

### 37.31.5 Show Filter Dialog (showFilterDialog)

Shows the filter dialog.

## Parameters

None.

## Returned value

None.

## 37.31.6 Submit (submit)

Applies the filter values.

## Parameters

Table 123:

Name	Type	Description
(optional) executeOnApply	Boolean	Specifies whether to execute the script of the On Apply property after the execution of submit (default: false)

## Returned value

None

## 37.32 Dropdown Box, List Box, Radio Button Group

### 37.32.1 Add Item (addItem)

Adds an item to the component.



## Parameters

Name	Type	Description
value	String	Unique key (or technical value) of the item. The method call is ignored if an item with this key already exists in the component.
text	String	Display text of the item.
(optional) index	Integer	Index where to insert the item (default: -1). If the index is greater than the number of items or less than 0 then the item is added as the last item. If the index is 0 then it is inserted before the first item.

## Returned value

None

## 37.32.2 Get Selected Text (getSelectedText)

Returns the selected item's text.

## Parameters

None

## Returned value

StringArray. It contains the selected text. If no text is defined, the selected item's value is returned instead.

## 37.32.3 Get Selected Texts (getSelectedTexts)

### Note

This method is only valid for the *List Box* component.

---

Returns the selected item's text. Can be used if multiple selection is enabled.

## Parameters

None

## Returned value

StringArray. It contains the selected texts as an array.

### 37.32.4 Get Selected Value (getSelectedValue)

Returns the selected item's value.

## Parameters

None

## Returned value

String. It contains the selected value.

### 37.32.5 Get Selected Values (getSelectedValues)

#### Note

This method is only valid for the [List Box](#) component.

Returns the selected item's value. Can be used if multiple selection is enabled.

---

## Parameters

None

## Returned value

StringArray. It contains the selected values as an array.

## 37.32.6 Dropdown Box: Get Tooltip (getTooltip)

Returns the tooltip of the dropdown box..

## Parameters

None.

## Returned value

String. It contains the tooltip text.

## 37.32.7 Is Enabled (isEnabled)

Returns whether component is enabled.

## Parameters

None

## Returned value type

Boolean. True if component is enabled or false if component is disabled.

---

## 37.32.8 On select (onSelect)

Calls the script that is executed when the user selects an item.

### Parameters

None

### Returned value

Boolean. True if component is enabled or false if component is disabled.

## 37.32.9 Remove All Items (removeAllItems)

Removes all items from the component.

### Parameters

None

### Returned value

None

## 37.32.10 Remove Item (removeItem)

Removes an item from the component.

## Parameters

Name	Type	Description
value	String	Unique key (or technical value) of the item. The method call is ignored if no such key exists in the component.

## Returned value

None

### 37.32.11 Set Enabled (setEnabled)

Enables or disables component.

Disabled components do not allow user interaction.

## Parameters

Name	Type	Description
isEnabled	Boolean	Specifies whether to enable or disable the component

## Returned value

None

### 37.32.12 Set Items (setItems)

Assigns a new list of items to the component.

The old content of the component will be deleted.

## Parameters

Name	Type	Description
value	ValueTextList	List of value-text pairs

## Returned value

None

### Example

In the following example, method `setItems` sets the first 100 members from the **Customer** dimension to the checkbox group / list box. Each list item's value is represented by the customer's key, each list item's text is represented by the customer's text and key. No item is added to represent all members.

```
LISTBOX_1.setItems(DS_1.getMemberList("OD_CUSTOMER", MemberPresentation.KEY,  
MemberDisplay.TEXT_KEY, 100));
```

or

```
CHECKBOXGROUP_1.setItems(DS_1.getMemberList("OD_CUSTOMER",  
MemberPresentation.KEY, MemberDisplay.TEXT_KEY, 100));
```

## 37.32.13 Set Selected Value (setSelectedValue)

Selects the item with the specified value.

## Parameters

Name	Type	Description
value	ListValue	Value of the item to select

## Returned value

None

## 37.32.14 Set Selected Values (setSelectedValues)

### **i** Note

This method is only valid for the *List Box* component.

Selects the items with the specified values.

### Parameters

Name	Type	Description
values	listValue Array	Values of the items to select

### Returned value

None

## 37.32.15 Dropdown Box: Set Tooltip (setTooltip)

Sets the tooltip of the dropdown box.

### Parameters

Name	Type	Description
tooltip	String	Tooltip text

### Returned value

None.

## 37.32.16 Sort (sort)

Sorts values in alphabetical order.

### Parameters

Table 124:

Name	Type	Description
(optional) isSortAscending	Boolean	Specifies whether values are sorted in ascending or descending alphabetical order

### Returned value

None

## 37.33 Fragment Gallery

### 37.33.1 Add Item (addItem)

Add a PortableFragmentBookmark to the component.

### Parameters

Name	Type	Description
info	PortableFragmentBookmarkInfo	An object representing an existing PortableFragmentBookmark. The operation is ignored if no such bookmark exists.

### Returned Value

None



---

## 37.33.2 Add Items (addItem)

Add each PortableFragmentBookmark in a PortableFragmentArray to the component.

### Parameters

Name	Type	Description
array	Array of PortableFragmentBookmarkInfo	An object representing an existing PortableFragmentArray. The operation is ignored if no such collection exists.

### Returned Value

None

## 37.34 Filter Panel

### 37.34.1 Cancel (cancel)

Discards the filter values that have not been applied yet.

### Parameters

None

### Returned value

None

### 37.34.2 Set Dimensions (setDimensions)

Sets the dimensions.

## Parameters

Table 125:

Name	Type	Description
dimensionArray	Array of Dimension	Array of dimensions

## Returned value

None

### 37.34.3 Submit (submit)

Applies the filter values.

## Parameters

Table 126:

Name	Type	Description
(optional) executeOnApply	Boolean	Specifies whether to execute the script of the On Apply property after the execution of submit (default: false)

## Returned value

None

## 37.35 Formatted Text View

### 37.35.1 Get HTML Text (getHTMLText)

Returns the HTML text.

## Parameters

None

## Returned value

String. It contains the HTML text.

## 37.35.2 Set HTML Text (setHTMLText)

Sets the HTML text.

## Parameters

Name	Type	Description
htmlText	String	HTML text

## Returned value

None

## 37.36 FragmentBookmarkInfo

A fragment bookmark info object which contains a description of the fragment bookmark.

## 37.37 Geo Map

### 37.37.1 Center Map (centerMap)

Centers the map on a layer.

## Parameters

Name	Type	Description
layerId	String	Layer id

## Returned Value

None

### 37.37.2 Get Copyright Text (getCopyrightText)

Returns the basemap copyright text.

## Parameters

None

## Returned Value

String

### 37.37.3 Get Selected Layer (getSelectedLayer)

Returns the selected layer.

## Parameters

None

## Returned Value

String

## 37.37.4 Is Layer Visible (isLayerVisible)

Returns whether the map layer is visible.

### Parameters

Name	Type	Description
layerId	String	Layer id

### Returned Value

Boolean. True if the layer is visible, false if the layer is not visible.

## 37.37.5 Set Copyright Text (setCopyrightText)

Sets the basemap copyright text.

### Parameters

Name	Type	Description
copyright	String	Copyright text

### Returned Value

None

## 37.37.6 Set Layer Visible (setLayerVisible)

Shows or hides a map layer.

## Parameters

Name	Type	Description
layerId	String	Layer id
isVisible	Boolean	Specifies whether to show or hide layer.

## Returned Value

None

### 37.37.7 Set Map Url (setMapUrl)

Sets the basemap URL.

## Parameters

Name	Type	Description
url	String	Basemap Url

## Returned Value

None

### 37.37.8 Get Selected Member (getSelectedMember)

Returns the member that is contained in the current selection of the chart.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the selected member.

## Returned Value

Member. It contains the member that has been selected in the chart.

## 37.38 Hierarchy

A hierarchy organizes members of a dimension into a tree structure such as, for example, a hierarchy for cost centers that are combined in cost center groups.

### name

The hierarchy's name.

### text

The hierarchy's text.

## 37.39 Icon

### 37.39.1 Get Background Color (getBackgroundColor)

Returns the background color of the icon.

## Parameters

None

---

## Returned Value

String. It contains the background color of the icon.

### 37.39.2 Get Background Shape (getBackgroundShape)

Returns the shape of the icon.

## Parameters

None

## Returned Value

IconBackgroundShape. It contains the shape of the icon.

### 37.39.3 Get Color (getColor)

Returns the color of the icon.

## Parameters

None

## Returned Value

String. It contains the color of the icon.

### 37.39.4 Get Icon Uri (getIconUri)

Returns the icon URI of the icon.



---

## Parameters

None

## Returned Value

String. It contains the icon URI of the icon.

### 37.39.5 Get Size Factor (getSizeFactor)

Returns the size factor of the icon.

## Parameters

None

## Returned Value

Float. It contains the size factor of the icon. This is a float value between 0 and 1, which indicates how much the icon image is scaled in the icon component's area. For example, a value of 1.0 scales the icon image to fill all of the icon component's area, a value of 0.5 scales the icon image to fill half of the icon component's area.

### 37.39.6 Get Tooltip (getTooltip)

Returns the tooltip of the icon.

## Parameters

None

## Returned Value

String. It contains the tooltip text of the icon.

## 37.39.7 Set Background Color (setBackgroundColor)

Sets the background color of the icon.

## Parameters

Name	Type	Description
backgroundColor	String	Background color

## Returned Value

None

## 37.39.8 Set Background Shape (setBackgroundShape)

Sets the shape of the icon.

## Parameters

Name	Type	Description
backgroundShape	IconBackgroundShape	Shape

## Returned Value

None

---

## 37.39.9 Set Color (setColor)

Sets the color of the icon.

### Parameters

Name	Type	Description
color	String	Color

### Returned Value

None

## 37.39.10 Set Icon Uri (setIconUri)

Sets the icon URI of the icon.

### Parameters

Name	Type	Description
iconUri	String	Icon URI

### Returned Value

None

## 37.39.11 Set Size Factor (setSizeFactor)

Sets the size factor of the icon.

## Parameters

Name	Type	Description
sizeFactor	Float	Size factor of the icon. This is a float value between 0 and 1, which indicates how much the icon image is scaled in the icon component's area. For example, a value of 1.0 scales the icon image to fill all of the icon component's area, a value of 0.5 scales the icon image to fill half of the icon component's area.

## Returned Value

None

### 37.39.12 Set Tooltip (setTooltip)

Sets the tooltip of the icon.

## Parameters

Name	Type	Description
tooltip	String	Tooltip text

## Returned Value

None

## 37.40 Image

### 37.40.1 Get Click Image (getClickImage)

Returns path of the click image file.

---

## Parameters

None

## Returned value

Url. It contains the path of the click image file in the format returned by the [Open Image](#) dialog of the [Image](#) property.

## 37.40.2 Get Hover Image (getHoverImage)

Returns path of the hover image file.

## Parameters

None

## Returned value

Url. It contains the path of the hover image file in the format returned by the [Open Image](#) dialog of the [Image](#) property.

## 37.40.3 Get Image (getImage)

Returns path of the default image file.

## Parameters

None

---

## Returned value

Url. It contains the path of the default image file in the format returned by the [Open Image](#) dialog of the [Image](#) property.

### 37.40.4 Get Opacity (getOpacity)

Returns opacity value.

## Parameters

None

## Returned value

Integer. A value between 0 (transparent) and 100 (opaque).

### 37.40.5 Get Tooltip (getTooltip)

Returns the tooltip of the component.

## Parameters

None.

## Returned value

String. It contains the tooltip text.

### 37.40.6 On Click (onClick)

Calls the script that is executed when the user clicks the image.

---

## Parameters

None

## Returned value

None

# 37.40.7 Set Click Image (setClickImage)

Sets image displayed on mouse click.

## Parameters

Name	Type	Description
imageURI	Url	Path of image file in the format returned by the <a href="#">Open Image</a> dialog of the <a href="#">Image</a> property

## Returned value

None

# 37.40.8 Set Hover Image (setHoverImage)

Sets image displayed on mouse hover.

## Parameters

Name	Type	Description
imageURI	Url	Path of image file in the format returned by the Open Image dialog of the Image property

## Returned value

None

### 37.40.9 Set Image (setImage)

Sets default image.

If the default image is not set, the Image component appears transparent.

## Parameters

Name	Type	Description
imageURI	Url	Path of image file in the format returned by the <i>Open Image</i> dialog of the <i>Image</i> property

## Returned value

None

### 37.40.10 Set Opacity (setOpacity)

Sets opacity value.



## Parameters

Name	Type	Parameter
opacity	Integer	Opacity value between 0 (transparent) and 100 (opaque)

## Returned value

None

### 37.40.11 Set Tooltip (setTooltip)

Sets the tooltip of the component.

## Parameters

Name	Type	Description
tooltip	String	Tooltip text

## Returned value

None.

### 37.41 Input Field

#### 37.41.1 Get Value (getValue)

Returns value of input field.

---

## Parameters

None

## Returned value

String. It contains the string entered into the input field.

### 37.41.2 Get Tooltip (getTooltip)

Returns the tooltip of the component.

## Parameters

None.

## Returned value

String. It contains the tooltip text.

### 37.41.3 Is Editable (isEditable)

Returns whether editing is enabled.

## Parameters

None.

## Returned value

Boolean. True if editing is enabled or false if editing is disabled.

## 37.41.4 Is Enabled (isEnabled)

Returns whether component is enabled.

### Parameters

None

### Returned value type

Boolean. True if component is enabled or false if component is disabled.

## 37.41.5 Set Editable (setEditable)

Enables or disables editing in the Inputfield.

### Parameters

Name	Type	Description
isEditable	Boolean	Specifies whether to enable editing

### Returned value

None

## 37.41.6 Set Enabled (setEnabled)

Enables or disables component.

Disabled components do not allow user interaction.

## Parameters

Name	Type	Description
isEnabled	Boolean	Specifies whether to enable or disable the component

## Returned value

None

## 37.41.7 Set Tooltip (setTooltip)

Sets the tooltip of the component.

## Parameters

Name	Type	Description
tooltip	String	Tooltip text

## Returned value

None.

## 37.41.8 Set Value (setValue)

Sets value of input field.

## Parameters

Name	Type	Description
value	String	Input field value

**Returned value**

None

**37.42 KeyValuePair**

Generic key value pair structure

**key**

The key.

**value**

The value.

**37.43 JSON**

A generic untyped JSON.

**37.43.1 For Each (forEach)**

Iterates through the elements of a JSON.

**Parameters**

Name	Type	Description
callback	Function	Function that is called with each iteration

## Returned value

None.

### Example

In the following example, the final value of result is Key: key1 Value: 100 Key: key2 Value: 200.

```
var result = "";
var sample = {"key1": 100, "key2": 200};
sample.forEach(function(value, key) {
  result = result + "Key: " + key + " Value: " + value;
});
```

## 37.44 Math

### E

Constant number value for e, the base of the natural logarithms, which is approximately 2.7182818284590452354.

### LN10

Constant number value for the natural logarithm of 10, which is approximately 2.302585092994046.

### LN2

Constant number value for the natural logarithm of 2, which is approximately 0.6931471805599453.

### LOG10E

Constant number value for the base-10 logarithm of e, the base of the natural logarithms, which is approximately 0.4342944819032518.

## LOG2E

Constant number value for the base-2 logarithm of e, the base of the natural logarithms, which is approximately 1.4426950408889634.

## P1

Constant number value for pi, which is approximately 3.1415926535897932.

## SQRT1\_2

Constant number value for the square root of 1/2, which is approximately 0.7071067811865476.

## SQRT2

Constant number value for the square root of 2, which is approximately 1.4142135623730951.

## 37.44.1 Abs (abs)

Returns the absolute value of x.

### Parameters

Name	Type	Description
x	Float	Number

### Returned value

Float. It contains the absolute value of x.

## 37.44.2 Acos (acos)

Returns the arc cosine of x.

### Parameters

Name	Type	Description
x	Float	Number

### Returned value

Float. It contains the arc cosine of x. The result is expressed in radians and ranges from +0 to +pi.

## 37.44.3 Asin (asin)

Returns the arc sine of x.

### Parameters

Name	Type	Description
x	Float	Number

### Returned value

Float. It contains the arc sine of x. The result is expressed in radians and ranges from -pi/2 to +pi/2.

## 37.44.4 Atan (atan)

Returns the arc tangent of x.



## Parameters

Name	Type	Description
x	Float	Number

## Returned value

Float. It contains the arc tangent of x. The result is expressed in radians and ranges from  $-\pi/2$  to  $+\pi/2$ .

## 37.44.5 Atan2 (atan2)

Returns the arc tangent of the quotient  $y/x$ .

## Parameters

Name	Type	Description
y	Float	Number
x	Float	Number

## Returned value

Float. It contains the arc tangent of the quotient  $y/x$ , where the signs of y and x are used to determine the quadrant of the result. The result is expressed in radians and ranges from  $-\pi$  to  $+\pi$ .

## 37.44.6 Ceil (ceil)

Returns the smallest integer number that is not less than x.

## Parameters

Name	Type	Description
x	Float	Number

## Returned value

Integer. It contains the smallest integer number that is not less than x.

## 37.44.7 Cos (cos)

Returns the cosine of x.

## Parameters

Name	Type	Description
x	Float	Number

## Returned value

Float. It contains the cosine of x. The argument is expressed in radians.

## 37.44.8 Exp (exp)

Returns the result of e raised to the power of x.

## Parameters

Name	Type	Description
x	Float	Number

## Returned value

Float. It contains e raised to the power of x.

## 37.44.9 Floor (floor)

Returns the greatest integer number that is not greater than x.

## Parameters

Name	Type	Description
x	Float	Number

## Returned value

Integer. It contains the greatest integer number that is not greater than x.

## 37.44.10 Log (log)

Returns the natural logarithm of x.

## Parameters

Name	Type	Description
x	Float	Number

## Returned value

Float. It contains the natural logarithm of x.

## 37.44.11 Max (max)

Returns the largest of the specified values.

## Parameters

Name	Type	Description
a	Float	Value1
(optional) b	Float	Value2
(optional) c	Float	Value3
(optional) d	Float	Value4
(optional) e	Float	Value5
(optional) f	Float	Value6
(optional) g	Float	Value7

## Returned value

Float. It contains the largest of the specified values.

## 37.44.12 Min (min)

Returns the smallest of the specified values.

### Parameters

Name	Type	Description
a	Float	Value1
(optional) b	Float	Value2
(optional) c	Float	Value3
(optional) d	Float	Value4
(optional) e	Float	Value5
(optional) f	Float	Value6
(optional) g	Float	Value7

### Returned value

Float. It contains the smallest of the specified values.

## 37.44.13 Pow (pow)

Returns the result of raising x to the power of y.

### Parameters

Name	Type	Description
x	Float	Number
y	Float	Number

---

## Returned value

Float. It contains x raised to the power of y.

### 37.44.14 Random (random)

Returns a random number.

## Parameters

None

## Returned value

Float. It contains a random number between 0 (included) and 1 (excluded).

### 37.44.15 Round (round)

Returns a number rounded to the nearest integer.

## Parameters

Name	Type	Description
x	Float	Number

## Returned value

Integer. It contains a number rounded to the nearest integer. If the number is equally close to two integer values, then the larger integer value is returned.

## 37.44.16 Sin (sin)

Returns the sine of x.

### Parameters

Name	Type	Description
x	Float	Number

### Returned value

Float. It contains the sine of x. The argument is expressed in radians.

## 37.44.17 Sqrt (sqrt)

Returns the square root of x.

### Parameters

Name	Type	Description
x	Float	Number

### Returned value

Float. It contains the square root of x.

## 37.44.18 Tan (tan)

Returns the tangent of x.

## Parameters

Name	Type	Description
x	Float	Number

## Returned value

Float. It contains the tangent of x. The argument is expressed in radians.

## 37.45 Member

An object representing a single dimension member. If you use this object as string, specify the internal key format. The object also provides access to the representations of the member.

### externalKey

The member's representation as external key.

### externalNoncompoundedKey

The member's representation as external non-compounded key.

### getAttributeMember

Returns an attribute member that belongs to this specific dimension member

Table 127: Parameters

Name	Type	Description
attribute	DimensionAttribute	Returns an attribute member that belongs to this specific dimension member

Returned Value: AttributeMember. It contains an attribute member of the dimension member.



---

## **internalKey**

The member's representation as internal key.

## **internalNoncompoundedKey**

The member's representation as internal non-compounded key.

## **text**

The member's representation as text.

# **37.46 Navigation Panel**

## **37.46.1 Set Dimensions (setDimensions)**

Sets the dimensions.

### **Parameters**

Table 128:

Name	Type	Description
dimensionArray	Array of Dimension	Array of dimensions

### **Returned value**

None

---

## 37.47 Pagebook

### 37.47.1 Get Page Count (getPageCount)

Returns the number of pages.

#### Parameters

None.

#### Returned value

Integer. Returns the name of the selected page.

### 37.47.2 Get Selected Page (getSelectedPage)

Returns the name of the selected page.

#### Parameters

None

#### Returned value

String. It contains the name of the selected page.

### 37.47.3 Get Selected Page Index (getSelectedPageIndex)

Returns index of the selected page. The first page has the index 0.

## Parameters

None

## Returned value

Integer. It contains the index of the selected page. The first page has index 0.

### 37.47.4 Get Selected Page by Name (getSelectedPageByName)

Selects the page by its name.

The selected page is the visible page of the pagebook.

## Parameters

Table 129:

Name	Type	Description
pageName	PageName	Name of page to select

## Returned value

None

### 37.47.5 On Select (onSelect)

Calls the script that is executed when the user selects a page.

## Parameters

None

## Returned value

None

### 37.47.6 Set Selected Page by Name (setSelectedPageByName)

Selects the page by its name.

The selected page is the visible page of the pagebook.

## Parameters

Table 130:

Name	Type	Description
pageName	PageName	Name of page to select

## Returned value

None

### 37.47.7 Set Selected Page Index (setSelectedPageIndex)

Selects page with the specified index.

The selected page is the visible page of the pagebook.

## Parameters

Name	Type	Description
index	Integer	Index of the page to select. The first page has index 0.

---

## Returned value

None

## 37.48 Panel

### 37.48.1 On Click (onClick)

Calls the script that is executed when the user clicks the component.

## Parameters

None

## Returned value

None

## 37.49 PDF

A global object providing export to PDF functions.

### 37.49.1 Export Application (exportApplication)

Exports an application to PDF, report style.

## Parameters

None

## Returned Value

None

## 37.49.2 Export Panel Screen (exportPanelScreen(panelArray))

Exports a panel or multiple panels to PDF. The exported content looks like a screenshot.

## Parameters

Name	Type	Description
panelArray	Array of Panel	Panel component or array of Panel components.

## Returned Value

None

## 37.49.3 Export Application Screen (exportApplicationScreen)

Exports an application to PDF, WYSIWYG.

## Parameters

None

## Returned Value

None

---

## 37.50 Planning

A global object providing planning functionality.

### 37.50.1 Client Reset (`clientReset`)

Rolls back to the last successfully recalculated state.

#### Parameters

None

#### Returned value

None

### 37.50.2 Has Client Changes (`hasClientChanges`)

Returns whether there is unsaved planning data on the client.

#### Parameters

None

#### Returned value

Boolean. True if there is unsaved planning data on the client and false if there is not.

---

## 37.50.3 Has Unsaved Changes (hasUnsavedChanges)

Returns whether there is unsaved planning data.

### Parameters

None

### Returned value

Boolean. True if there is unsaved planning data and false if there is not.

## 37.50.4 Recalculate (recalculate)

Recalculates changed planning data. Returns a status indicating whether the recalculation executed correctly.

### Parameters

None

### Returned value

Boolean. True if the recalculation executed correctly and false if an error occurred during recalculation.

## 37.50.5 Reset (reset)

Rolls back to the last saved server state.

### Parameters

None



---

## Returned value

None

## 37.50.6 Save (save)

Saves changed planning data to the server. Returns a status indicating whether the save operation executed correctly.

## Parameters

None

## Returned value

Boolean. True if the save operation executed correctly and false if an error occurred during this operation.

## 37.51 PlanningFunction

An object representing a planning function.

### 37.51.1 Clear All Filters (clearAllFilters)

Removes filters for all dimensions.

## Parameters

None

## Returned value

None

### 37.51.2 Clear Filter (clearFilter)

Removes the filter for a dimension.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the planning function to remove the filter from

## Returned value

None

### 37.51.3 Copy Filters (copyFilters)

Copies filter values of common dimensions from a data source.

#### Note

Values of static filters will not be copied, even if the dimension is available as a filter in the target data source. Therefore, values of static filters need to be retrieved from the data source and applied to the target data source separately. Values of static filters can be retrieved with `getStaticFilterExt` and `getStaticFilterText` from dimensions of a data source.

## Parameters

Name	Type	Description
dataSourceAlias	DataSource Alias	Data source alias to copy filters from

Name	Type	Description
isCopyMeasuresFilter (optional)	Boolean	Specifies whether the filter values for measures are copied as well (default: false).

## Returned value

None

## 37.51.4 Execute (execute)

Performs the planning function. Returns a status indicating whether the planning function executed correctly.

## Parameters

None

## Returned value

Boolean. True if the planning function executed correctly and false if an error occurred while executing the planning function.

## 37.51.5 Get Dimension Text (getDimensionText)

Returns the localized text of a dimension.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the planning function

## Returned value

String. It contains the localized text of the dimension.

## 37.51.6 Get Dimensions (getDimensions)

Returns an array of dimensions of an axis.

## Parameters

None

## Returned value

DimensionArray. It contains all dimensions.

## 37.51.7 Get Filter Ext (getFilterExt)

Returns the filter value of a data source dimension in external key format.

### Note

Use this method to pass the filter value to other methods.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the planning function

## Returned value

String. It contains the filter value in external key format.

## 37.51.8 Get Filter Text (getFilterText)

Returns the filter value of the specified dimension in external key format.

### Note

Use this method to display the filter value.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the planning function

### Returned value

String. It contains the filter value.

## 37.51.9 Get Member List (getMemberList)

Retrieves a list of dimension members.

### Parameters

Name	Type	Description
dimension	Dimension	Dimension of the planning function
memberPresentation	MemberPresentation	Presentation of member keys
memberDisplay	MemberDisplay	Textual display of members
maxNumber	Integer	Maximum number of members to be returned
(optional) allMemberText	String	Text of the item that represents all members. If no text is specified, the item is not added to the list

## Returned value

ValueTextList. It contains a list of dimension members.

### Example

In the following example, a list box is populated with members from a planning function:

```
LISTBOX_1.setItems(DS_1.getMemberList("0D_CUSTOMER",  
MemberPresentation.INTERNAL_KEY, MemberDisplay.TEXT_KEY, 100));
```

## 37.51.10 Set Filter (setFilter)

Sets a filter for a dimension in the internal key format.

The existing filter is replaced.

### Note

If you want to specify filter values in the external key format use the method `setFilterExt` instead.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the planning function to be filtered
value	Array of Filter	Filter to be set

## Returned value

None.

## 37.51.11 Set Filter Ext (setFilterExt)

The existing filter is replaced.

Sets a filter for a dimension in the external key format.

You can specify the input string as a filter value. The input string syntax allows you to specify complex selections, such as intervals and excluding selections. In contrast to method `setFilter`, the members are specified in the external fully-compounded key member presentation format, for example **K4/001.2012**.

#### **i** Note

External key values might be dependent on the user's regional settings. If the application is intended to be used by a multi-lingual group of users, or if you want to specify filter values with the internal key (for example, to avoid localization problems), use the method `setFilter` instead.

## Parameters

Name	Type	Description
dimension	Dimension	Dimension of the planning function to be filtered
value	InputString	Filter to be set

## Returned value

None

## 37.52 PlanningObjectWithVariables

An object representing a planning object with variables.

### 37.52.1 Copy Variable Value (copyVariableValue)

Copies the value of a variable from a data source.

## Parameters

Name	Type	Description
dataSourceAlias	DataSourceAlias	Data source alias to copy the variable from
variableFrom	Variable	Data source variable to copy from
(optional) variableTo	Variable	Planning object variable to copy to. If omitted, the variable from the DataSource is copied to a variable with the same name in the planning object (if available)

## Returned value

None

## 37.52.2 Get Variable Value Ext (getVariableValueExt)

Returns the variable value of a planning object variable in external key format.

### Note

Use this method to pass the variable value to other methods.

## Parameters

Name	Type	Description
variable	Variable	Data source variable

## Returned value

String. It contains the variable value in external key format.



## 37.52.3 Set Variable Value Text (setVariableValueText)

Returns the variable value of a planning function variable.

### Note

Use this method to display the variable.

## Parameters

Name	Type	Description
variable	Variable	Data source variable

## Returned value

String. It contains the variable value.

## 37.52.4 Set Variable Value (setVariableValue)

Sets planning object values in internal key format.

Variable values on planning objects are independent of values set on the application.

With this method you can set only single members or hierarchy nodes in the internal key format as variable values. If you want to set variable values for the following cases, use `setVariableValueExt` instead:

- formula variables
- text variables
- hierarchy variables
- variables representing precalculated values sets (buckets)
- multiple single values
- intervals
- variables representing a selection option
- using external format

For hierarchy nodes, prefix the internal non-compounded presentation of the hierarchy node with **HIERARCHY\_NODE/NodeType/**, for example **HIERARCHY\_NODE/0HIER\_NODE/ROOT**.

## Parameters

Name	Type	Description
variable	Variable	Query variable to set
value	VariableValue	Variable value in internal key format

### Restriction

It is not possible to use the crosstab and chart method `getSelectedMember` for setting variables for compounded dimensions.

## Returned value

None

### Example

In the following example a fixed string value is set to a variable:

```
PF_1.setVariableValue("OVAR", "4711");
```

In the following example the value is set to a variable returned from a component selection:

```
PF_1.setVariableValue("OVAR", DROPDOWN_1.getSelectedValue());
```

## 37.52.5 Set Variable Value Ext (setVariableValueExt)

Sets query variable values in external key format.

Variable values on planning objects are independent of values set on the application. Use this method if other selections than single members or hierarchy nodes need to be set as variable value.

### Note

External key values might be dependent on the user's regional settings. If the application is intended to be used by a multi-lingual group of users, it is therefore recommended to use the `setVariableValue` method when possible.

## Parameters

Name	Type	Description
variable	Variable	Query variable to set
value	InputString	Variable value to set in external key format

## Returned value

None

### Example

In the following example a fixed string value is set to a variable:

```
PF_1.setVariableValue("0VAR", "4711");
```

In the following example the value returned from a component selection is set to a variable:

```
PF_1.setVariableValue("0VAR", DROPDOWN_1.getSelectedValue());
```

## 37.52.6 Set Variable Value Range (setVariableValueRange)

Sets the value range of a variable.

## Parameters

Name	Type	Description
variable	Variable	Data source variable
variableFrom	VariableValue	Variable value to start range, in internal key format
(optional) variableTo	VariableValue	Variable value to end range, in internal key format

---

## Returned value

None

## 37.53 PlanningSequence

An object representing a planning sequence.

### 37.53.1 Execute (execute)

Performs the planning sequence. Returns a status indicating whether the planning sequence executed correctly.

## Parameters

None

## Returned value

Boolean. True if the planning sequence executed correctly and false if an error occurred while executing the planning sequence.

## 37.54 Popup

### 37.54.1 Hide (hide)

Hides popup.

## Parameters

None

---

## Returned value

None

### 37.54.2 Is Showing (isShowing)

Returns wheter popup is shown.

## Parameters

None

## Returned value

Boolean. True if popup is shown or false if popup is hidden.

### 37.54.3 Show (show)

Shows popup.

## Parameters

None

## Returned value

None

### 37.55 SdkDataSource

An object representing an Sdk data source.

---

This object has the following methods:

- `clearAllFilters` [Clear All Filters \(clearAllFilters\)](#) [page 572]
- `clearFilter` [Clear Filter \(clearFilter\)](#) [page 573]
- `getData` [Get Data \(getData\)](#) [page 580]
- `getDataAsString` [Get Data as String \(getDataAsString\)](#) [page 581]
- `getDimensionText` [Get Dimension Text \(getDimensionText\)](#) [page 584]
- `getDimensions` [Get Dimensions \(getDimensions\)](#) [page 584]
- `getFilterText` [Get Filter Text \(getFilterText\)](#) [page 585]
- `getMeasuresDimension` [Get Measures Dimension \(getMeasuresDimension\)](#) [page 586]
- `getMembers` [Get Members \(getMembers\)](#) [page 588]
- `setFilter` [Set Filter \(setFilter\)](#) [page 605]

## 37.56 SearchExpression

A SearchExpression is used for search operations.

## 37.57 SingleMemberFilter

A SingleMemberFilter is a filter composed of a single dimension member. The dimension member is specified in internal key format.

## 37.58 Splitcell Container

### 37.58.1 Get Data Sources (getDataSources)

Returns a list of data sources.

#### Parameters

None

---

## Returned Value

Array of DataSourceAlias. It contains data source aliases.

## 37.58.2 Is Display Mode (isDisplayMode)

Returns whether the split cell container is in display mode.

### Parameters

None

### Returned value

Boolean. True if the split cell container is in display mode, false if the split cell container is not in display mode.

## 37.58.3 On Delete (onDelete)

Calls the script that is executed when the user deletes an object from the split cell container.

### Parameters

None

### Returned value

None

## 37.58.4 On Drop (onDrop)

Calls the script that is executed when the user drops an object on the split cell container.

## Parameters

None

## Returned value

None

### 37.58.5 Set Display Mode (setDisplayMode)

Enables or disables the display mode of the split cell container.

An enabled display mode of the split cell container does not allow user interaction.

## Parameters

Name	Type	Description
isDisplayMode	Boolean	Specifies whether to enable or disable the display mode of the split cell container

## Returned value

None

## 37.59 State

### 37.59.1 Back One Step (backOneStep)

Goes back one step. Reverts the last change in the analysis application.



---

## Parameters

None

## Returned Value

None

### Note

Supported platforms - BI platform, SAP NetWeaver, Local. SAP HANA is not supported.

## 37.59.2 Back To Start (backToStart)

Goes back to start. Reverts all changes in current analysis application.

## Parameters

None

## Returned Value

None

### Note

Supported platforms - BI platform, SAP NetWeaver, Local. SAP HANA is not supported.

## 37.59.3 Is Back To Start Available (isBackToStartAvailable)

Returns whether the `backToStart` method can be used to revert all changes in the analysis application.

---

## Parameters

None

## Returned value

Boolean. True if this method can be used at this time, false if this method cannot be used at this time.

### Note

Supported platforms - BI platform, SAP NetWeaver, Local. SAP HANA is not supported.

## 37.59.4 Is Back One Step Available (`isBackOneStepAvailable`)

Returns whether the `backOneStep` method can be used to revert the last change in the analysis application.

## Parameters

None

## Returned Value

Boolean. True if this method can be used at this time, false if this method cannot be used at this time.

### Note

Supported platforms - BI platform, SAP NetWeaver, Local. SAP HANA is not supported.

## 37.59.5 Set Personalization (`setPersonalization`)

Saves the current state of this analysis application as the default state. The saved state will be applied to the application each time it is opened. Valid for the current version of the analysis application only.

---

## Parameters

None

## Returned value

None

### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.59.6 Delete Personalization (deletePersonalization)

Deletes the default state for this analysis application, if one exists.

## Parameters

None

## Returned value

None

### Note

Supported platforms - BI platform, SAP NetWeaver, Local

## 37.60 String

### 37.60.1 Index Of (indexOf)

Returns the index of the first occurrence of the specified substring within a string. Optionally, you can provide an index from where to start the search.

## Parameters

Name	Type	Description
searchFor	SearchExpression	Substring to search for
(optional) startIndex	Integer	Index from where to start searching (default: 0)

## Returned value

Integer. It contains the index of the first occurrence of the specified substring within a string. The first character of that string has index 0. The returned value is -1 if the substring was not found.

### Example

In the following example, the returned index is 6:

```
"Hello world!".indexOf("world");
```

In the following example, the returned index is -1:

```
"Hello world!".indexOf("sailor");
```

## 37.60.2 Length (length)

The string's number of characters.

## 37.60.3 Split (split)

Splits the string at a separator string into string fragments.

## Parameters

Name	Type	Description
separator	SearchExpression	Separator string
(optional) limit	Integer	The maximum number of fragments to return

## Returned value

StringArray. It contains the string fragments.

### Example

In the following example, the array contains the string fragments `one`, `two`, and `three`:

```
var array = "one|two|three".split("|");
```

## 37.60.4 Substring (substring)

Returns a new string that is a substring of the given string.

The substring is composed of the characters of the given string from the given `startIndex` up to the given `endIndex` - 1. If `endIndex` is omitted, then the substring includes the characters up to the last character of the given string.

### Parameters

Name	Type	Description
<code>startIndex</code>	Integer	Start index of the substring, inclusive
(optional) <code>endIndex</code>	Integer	End index of the substring, exclusive

## Returned value

String. It contains the substring of the given string.

## 37.61 Tab

### 37.61.1 Get Text (getText)

Returns text.

---

## Parameters

None

## Returned value

String. It contains the text of this component.

## 37.61.2 Is Enabled (isEnabled)

Returns whether component is enabled.

## Parameters

None

## Returned value type

Boolean. True if component is enabled or false if component is disabled.

## 37.61.3 Set Enabled (setEnabled)

Enables or disables component.

Disabled components do not allow user interaction.

## Parameters

Name	Type	Description
isEnabled	Boolean	Specifies whether to enable or disable the component

## Returned value

None

## 37.61.4 Set Text (setText)

Sets text.

## Parameters

Name	Type	Description
text	String	Text to be set

## Returned value

None

## 37.62 Tabstrip

### 37.62.1 Get Selected Tab (getSelectedTabIndex)

Returns the name of the selected tab.

## Parameters

None

## Returned value

String. It contains the name of the selected tab.

## 37.62.2 Get Selected Tab Index (getSelectedTabIndex)

Returns index of the selected text.

### Parameters

None.

### Returned value type

Integer. It contains the index of the selected tab. The first tab has index 0.

## 37.62.3 Get Tab (getTab)

Returns a tab.

### Parameters

Table 131:

Name	Type	Description
tabIndex	Integer	Index of tab to return. The first tab has index 0.

### Returned value type

Tab. It is the tab indicated by the given index.

## 37.62.4 On Select (onSelect)

Calls the script that is executed when the user selects a tab.



## Parameters

None

## Returned value

None

### 37.62.5 Set Selected Tab (setSelectedTab)

Selects the tab by its name.

The selected tab is the visible tab of the tabstrip.

## Parameters

Table 132:

Name	Type	Description
tabName	TabName	Name of the tab to select

### 37.62.6 Set Selected Tab Index

Selects tab with the specified index.

## Parameters

Name	Type	Description
tabIndex	Integer	Index of tab to select. The first tab has index 0.

## Returned value

None

---

## 37.63 Text

### 37.63.1 Get Text (getText)

Returns text.

#### Parameters

None

#### Returned value

String. It contains the text of this component.

### 37.63.2 Get Tooltip (getTooltip)

Returns the tooltip of the component.

#### Parameters

None.

#### Returned value

String. It contains the tooltip text.

### 37.63.3 Set Text (setText)

Sets text.

## Parameters

Name	Type	Description
text	String	Text

## Returned value

None

## 37.63.4 Set Tooltip (setTooltip)

Sets the tooltip of the component.

## Parameters

Name	Type	Description
tooltip	String	Tooltip text

## Returned value

None.

## 37.64 Timer

### 37.64.1 Start (start)

Starts timer.

## Parameters

None

---

## Returned Value

None

### 37.64.2 Stop (stop)

Stops timer.

## Parameters

None

## Returned Value

None

### 37.64.3 Is Running (isRunning)

Returns true if timer is running, false otherwise.

## Parameters

None

## Returned Value

Boolean. True if timer is running, false if the timer is not running.

---

## 37.65 Tree

### 37.65.1 Get Bottom Margin (getBottomMargin)

Returns the bottom margin of the component.

#### Parameters

None

#### Returned value

Integer. It contains the bottom margin of component in pixels. It is -1 if the current value is *auto*.

### 37.65.2 Get CSS Class (getCSSClass)

Returns the Cascading Style Sheet (CSS) class name of the component.

#### Parameters

None

#### Returned value

String. It contains the CSS class name of the component.

### 37.65.3 Get Height (getHeight)

Returns the height of the component.

---

## Parameters

None

## Returned value

Integer. It contains the height of component in pixels. It is -1 if the current value is *auto*.

### 37.65.4 Get Left Margin (getLeftMargin)

Returns the left margin of the component.

## Parameters

None

## Returned value

Integer. It contains the left margin of component in pixels. It is -1 if the current value is *auto*.

### 37.65.5 Get Right Margin (getRightMargin)

Returns the right margin of the component.

## Parameters

None

## Returned value

Integer. It contains the right margin of component in pixels. It is -1 if the current value is *auto*.

---

## 37.65.6 Get Selected Member (getSelectedMember)

Returns the selected member. If the tree items are property-bound to a dimension, then the selected member (or the first of multiple selected members) is returned. Otherwise an empty Member object is returned. If nothing has been selected an empty Member object is returned.

### Parameters

None

### Returned Value

Member. It contains the selected member.

## 37.65.7 Get Selected Members (getSelectedMembers)

Returns the selected members. If the tree items are property-bound to a dimension, then the selected members are returned. Otherwise an empty array is returned.

### Parameters

None

### Returned Value

Array of Member. It contains the selected members.

## 37.65.8 Get Selected Text (getSelectedText)

Returns the selected text. If the tree items are property-bound to a dimension, then the returned value depends on the display setting of the dimension: If the display setting is "Text", "Key & Text", or "Text & Key" then the selected text (or the first of multiple selected texts) is returned. Otherwise an empty string ("") is returned.

---

## Parameters

None

## Returned Value

String. It contains the selected text.

### 37.65.9 Get Selected Texts (getSelectedTexts)

Returns the selected texts. If the tree items are property-bound to a dimension, then the returned value depends on the display setting of the dimension: If the display setting is "Text", "Key & Text", or "Text & Key" then the selected texts are returned. Otherwise an empty array is returned.

## Parameters

None

## Returned Value

Array of String. It contains the selected texts.

### 37.65.10 Get Selected Value (getSelectedValue)

Returns the selected value. If the tree items are property-bound to a dimension, then the returned value depends on the display setting of the dimension: If the display setting is "Key", "Key & Text", or "Text & Key" then the selected value (or the first of multiple selected values) is returned. Otherwise an empty string ("") is returned.

## Parameters

None



---

## Returned Value

String. It contains the selected value.

### 37.65.11 Get Selected Values (getSelectedValues)

Returns the selected values. If the tree items are property-bound to a dimension, then the returned value depends on the display setting of the dimension: If the display setting is "Key", "Key & Text", or "Text & Key" then the selected values are returned. Otherwise an empty array is returned.

## Parameters

None

## Returned Value

Array of String. It contains the selected values.

### 37.65.12 Get Top Margin (getTopMargin)

Returns the top margin of the component.

## Parameters

None

## Returned value

Integer. It contains the top margin of the component in pixels. It is -1 if the current value is *auto*.

---

## 37.65.13 Get Width (getWidth)

Returns the width of the component.

### Parameters

None

### Returned value

Integer. It contains the width of component in pixels. It is -1 if the current value is *auto*.

## 37.65.14 Hide Loading State (hideLoadingState)

Hides the loading indicator on the component.

### Parameters

None

### Returned value

None

## 37.65.15 Is Visible (isVisible)

Returns whether component is visible.

### Parameters

None

**Returned value**

Boolean. True if component is shown or false if component is hidden.

**37.65.16 Remove Model (removeModel)**

Removes tree model.

**Parameters**


None

**Returned Value**

None

**37.65.17 Set Bottom Margin (setBottomMargin)**

Sets the bottom margin of the component.

 **Note**  
This function has no effect, if the current value is *auto*.

**Parameters**

Table 133:

Name	Type	Description
bottomMargin	Integer	Bottom margin of component in pixels

**Returned value**

None

## 37.65.18 Set CSS Class (setCSSClass)

Sets the Cascading Style Sheet (CSS) class name of the component.

### Parameters

Table 134:

Name	Type	Description
className	String	Name of the CSS class

### Returned value

None

## 37.65.19 Set Height (setHeight)

Sets the height of the component.

### Note

This function has no effect if the current value is *auto*.

### Parameters

Table 135:

Name	Type	Description
height	Integer	Height of the component in pixels

### Returned value

None

## 37.65.20 Set Left Margin (setLeftMargin)

Sets left margin of the component.

### Note

This function has no effect if the current value is *auto*.

### Parameters

Table 136:

Name	Type	Description
leftMargin	Integer	Left margin of component in pixels

### Returned value

None

## 37.65.21 Set Model (setModel)

Sets the tree model.

### Parameters

Name	Type	Description
model	TreeModel	Tree model

### Returned Value

None

## 37.65.22 Set Right Margin (setRightMargin)

Sets right margin of the component.

### Note

This function has no effect if the current value is *auto*.

## Parameters

Table 137:

Name	Type	Description
rightMargin	Integer	Right margin of component in pixels

## Returned value

None

## 37.65.23 Set Top Margin (setTopMargin)

Sets top margin of the component.

### Note

This function has no effect if the current value is *auto*.

## Parameters

Table 138:

Name	Type	Description
topMargin	Integer	Top margin of component in pixels

## Returned value

None

## 37.65.24 Set Visible (setVisible)

Shows or hides component.

### Parameters

Name	Type	Description
isVisible	Boolean	Specifies whether to show or hide component

### Returned value

None

## 37.65.25 Set Width (setWidth)

Sets width of the component.

### Note

This function has no effect if the current value is *auto*.

### Parameters

Table 139:

Name	Type	Description
width	Integer	Width of component in pixels

### Returned value

None

## 37.65.26 Show Loading State (showLoadingState)

Shows a loading indicator on the component.

### Parameters

Name	Type	Description
(optional) text	String	Text to show in the loading indicator

### Returned value

None

## 37.66 Variable

Variables are parameters of a query. They act as placeholders for, for example, members of dimensions, hierarchies, or hierarchy nodes.

### dimensionName

Returns the name of the associated dimension.

### inputEnabled

Returns true if the variable is input-enabled and false if it is not.

### name

The variable's name.



---

## **text**

The variable's text.

## 38 Working with the Local Mode of the Design Tool

### From local mode to full mode of SAP BusinessObjects Design Studio

When starting the design tool of SAP BusinessObjects Design Studio after the installation, it is being launched in local mode by default. The local mode can be used to create first analysis applications for presenting and evaluating SAP BusinessObjects Design Studio with users at the customer site. You can save the analysis applications on your local system. Without the integration to the BI platform, the technical prerequisites are on a minimal level and an easy and fast evaluation can take place. The key users of a company - possibly together with consultants - can investigate if SAP BusinessObjects Design Studio is able to cover their requirements regarding data analysis and reporting in their company. After a successful evaluation phase and with the decision to implement SAP BusinessObjects Design Studio, the necessary requirements regarding the BI platform need to be provided.

As soon as the BI platform infrastructure has been set up by the administrator, you as an application designer who has been working in local mode can easily switch the default startup mode in the [Preferences](#) dialog box and connect to the BI platform. Next, you can upload your locally saved analysis applications to the BI platform.

#### Restriction

The local mode of the design tool is not designed for an offline usage of the design tool. You should not use it for downloading analysis applications from the BI platform, changing and saving them locally, and then uploading the applications again to the BI platform, repeating these steps several times.

### 38.1 Launching SAP BusinessObjects Design Studio

#### Context

To launch SAP BusinessObjects Design Studio, choose **Start > All Programs > SAP BusinessObjects > Design Studio**. The design tool is launched with the welcome page.

#### Related Information

[Starting with the Welcome Page \[page 33\]](#)



---

## 38.2 Creating New Analysis Applications

### Context

When creating new analysis applications, you can select different templates. You can choose between blank templates or predefined templates that correspond to various design and business needs. When you choose a template, the system automatically creates a copy of this template. You can change this copy according to your needs.

### Procedure

1. Click  [Application](#) > [New...](#)  in the menu of the design tool or click [Create Analysis Application](#) on the Welcome page. The [New Application](#) dialog box is displayed.
2. On the [New Application](#) dialog box, perform the following steps:
  - a. Enter a unique name for your application in the field under [Application Name](#).
  - b. Choose the rendering type for the components in your application: [SAPUI5](#) or [SAPUI5 m](#). The SAP UI5m rendering type creates applications using the SAPUI5 main control set. For more information about the use of the rendering types, see [Using the SAPUI5 m Library \[page 45\]](#). Depending on your choice, the available templates for the rendering type are displayed automatically.
  - c. Choose the template you want to use for your application. A short description of the template is displayed at the right side of the dialog box.
  - d. Choose [Create](#). The template is opened in the editor where you can change and edit it.

### Results

You have created a new application. Now you can define the content of your application by adding components and data sources to the blank template or adjusting the predefined ready-to-run templates according to your needs. For more information, see the links in the *Related Topics* section.

### Related Information

[Adding Components to an Application \[page 47\]](#)

[Adding a Data Source \[page 68\]](#)

[Assigning a Data Source to a Component \[page 97\]](#)

[Using the Script Editor \[page 277\]](#)

[Saving an Application \[page 102\]](#)

[Executing an Application \[page 724\]](#)

[Properties of the Application \[page 325\]](#)

## 38.3 Maintaining Settings in the Design Tool in Local Mode

You can define settings for the design tool in the *Preferences* dialog box. To start the *Preferences* dialog box, choose **Tools > Preferences > Application Design**.

Change the default values as required and choose *Apply* to make these values effective. If you want to reset the default values, choose *Restore Defaults*. To close the *Preferences* dialog box, press *OK*.

### General

You can switch the startup mode. In the default setting, the design tool starts in local mode after installation. If you want to use SAP BusinessObjects Design Studio integrated into one of the supported platforms, select the required platform:

- *SAP BusinessObjects BI Platform*
- *SAP HANA*
- *SAP NetWeaver*

After restarting the design tool, you need to log on to the selected platform.

#### Note

In the *Logon to <selected platform>* dialog box, you can still decide to log on locally by clicking *Skip*, even if the startup mode is set to one of the supported platforms. The design tool starts in local mode.

In the *Undo History Size* field, you can change the default value 50 and enter a number between 20 and 999. This value defines how often application designers can undo their changes when working in the design tool.

### Embedded Web Server

You can define the network port for the embedded Web server of the design tool. Enter a number between 1024 and 65535. When an application is executed, the port number is shown in the URL. If the port is set to 5555 for example, the URL is `http://localhost:5555/aad/web.do?APPLICATION=MYAPP`. If you leave the default value 0, the system automatically assigns a network port.

#### Tip

If users create bookmarks for their applications, the network port has to be set to a fixed number. Auto-assigning the network port does not guarantee that the bookmarks will work properly.

If you want to test your application on a mobile device, you need to allow external access to the embedded Web server. Select the *Allow external access to embedded Web server* checkbox. The design tool must be running on your PC.

### Caution

Note that allowing external access is a security risk. Everyone connected to the same network can potentially access all of your applications and can see the same data that you can (as the applied data authorizations are the same).

### Restriction

Before other users can open the analysis application on a mobile device, the [Logon to <BW or SAP HANA system>](#) dialog box always appears on the PC with the design tool that is running.

- SAP HANA data sources used in the analysis application: the application designer of the PC that the design tool is running on has to enter his/her credentials and click [Log on](#) in the [Logon to <SAP HANA system>](#) dialog box.
- BW data sources used in the analysis application: activate Single-Sign On. The application designer of the PC that the design tool is running on has to confirm the client and language (or has to change these entries if required) and then click [Log on](#) in the [Logon to <BW system>](#) dialog box.

## Application Recovery

In the default setting, the system automatically saves unsaved applications every minute. You can configure the auto-save time interval as required. Select the [Save application recovery information every <1> minute](#) checkbox and enter the required number for the auto-save time interval.

### Tip

The design tool needs to be restarted for the new interval to become active.

There is a background job that searches for unsaved applications in the given interval. If unsaved applications are found, the system extracts their XML code and stores it under `<user home directory>\Analysis-workspace\.metadata\.plugins\com.sap.ip.bi.zen`. The content of this file is encrypted using Eclipse secure store technology.

### Note

The auto-save function does not replace saving an application. Saving an application or closing it and answering the *Save changes?* question with either *Yes* or *No* will delete the XML from the `autosaves` file.

In the event of a system crash, the auto-saved XML persists. When restarting the design tool and opening the affected application again, the designer is informed that an auto-saved version of the application exists.

- If the designer decides to restore the auto-saved version, the system uses the stored XML, saves the application and opens the application in the layout editor. The auto-saved XML is removed.
- If the designer decides to discard the auto-saved version, the auto-saved XML is removed as well.

## Member Selection

Application designers need to pick single members of a dimension when using statements like `setFilter` for a dropdown box, for example. They can pick the members in the content assistance of the [Script Editor](#) dialog box or in the [Select Member](#) dialog box. The [Select Member](#) dialog box can list a small or large number of members, depending on the maximum threshold number of members. You can set the default threshold in the [Preferences](#) dialog box.

- You can define the maximum number of members that are displayed in the [Select Member](#) dialog box. Enter the required number in the [Maximum number of members to fetch from backend in dialog](#) checkbox. The default threshold is 1000.
- You can define the maximum number of members that are displayed in the content assistance of the [Script Editor](#) dialog box. Enter the required number in the [Maximum number of members to fetch from backend in content assistance](#) checkbox. The default threshold is 20. If the number of available members exceeds this threshold number, the content assistance does not list single members. Instead it offers the [Select Member...](#) entry, which opens the [Select Member](#) dialog box.

### ➔ Tip

Designers can access the content assistance in the script editor by pressing `CTRL` + `SPACE`.

- You can also decide whether the system should display warnings in the script editor whenever designers manually enter non-existent values. To activate the warnings, select the [Display warnings for manually entered invalid values](#) checkbox.

## Prompt Handling

In the [Prompts](#) dialog box, application designers and application users set values for prompts. For SAP BW data sources, prompts are defined as variables. When working with data sources with defined variables, application designers might be prompted to set the required values before continuing their work in the design tool (if there are mandatory variables without default values, or if variables have invalid default values, for example). SAP BusinessObjects Design Studio stores all valid prompt values of each analysis application in the user's cache file in `<user home directory>\Analysis-workspace\.metadata\.plugins\com.sap.ip.bi.zen\cache`. This provides application designers with a smooth workflow in the design tool. If this was not the case, the [Prompts](#) dialog would appear an analysis application is reloaded, or when the initial state of the data source is modified in the [Initial View](#) dialog box, for example.

In the [Preferences](#) dialog box, you can specify whether the [Prompts](#) dialog box appears when an analysis application is executed locally:

- If you want to simulate how an application user opens the analysis application, leave the checkbox deselected (This is the default setting). When you execute an analysis application locally, the [Prompts](#) dialog box appears and you can set the prompt values like an end user would do.
- If you want to use the prompt values from the cache file, select the [Use cached prompt values for local execution](#) checkbox. The [Prompts](#) dialog box does not appear and application designers can test their analysis applications quickly.

### Note

If the *Prompts* dialog box appears despite this setting, check if the *Force Prompts On Startup* property of the analysis application is set to true. This property forces the *Prompts* dialog box to appear, regardless of your choice for the *Use cached prompt values for local execution* checkbox.

If you want to clear the prompt values of an analysis application in the cache file, press *Clear Prompt Value Cache...* You can select the required analysis application(s) for this cache deletion. When the application is reloaded, the *Prompts* dialog box will appear and you can set new values, for example.







## Report-Report Interface















With the report-report interface application designers can jump to predefined jump targets in analysis applications.

If the jump target configured for the data source query is itself a query, the target query is launched as a BEx Web application by default. To avoid this and to ensure that the jump is handled by SAP BusinessObjects Design Studio and that the target query is displayed as an analysis application, you need to specify a generic analysis template.






Under *Application to use for query jumps*, enter the name of the locally saved generic analysis template.

## Application Templates

SAP BusinessObjects Design Studio includes a set of templates that offer designers an easy way to get started. When creating new applications, designers can choose between different templates that are optimized for desktop Web browser applications or mobile applications in the *New Application* dialog box ( *Application*  *New...* ). For more information, see “Creating a new analysis application” in the *Application Designer Guide: Designing Analysis Applications* under  *Help*  *Help Contents*  in the design tool.

Application designers can also create analysis applications and provide them as templates for other application designers. Under  *Tools*  *Preferences*  *Application Design*  *Application Templates*  , you can add the path to the folder where these templates are stored. Here you can also define your own template categories that indicate the target device types recommended for a specific template. The template categories are then listed in the *New Application* dialog box ( *Application*  *New...* ), and also in the *Export Application As Template* dialog box ( *Application*  *Export as Template...* ). For more information, see “Exporting Applications As Templates” in the *Application Designer Guide: Designing Analysis Applications* under  *Help*  *Help Contents*  in the design tool.

## Network Connections

Under  *Tools*  *Preferences*  *Application Design*  *Network Connections*  , you can specify the proxy settings to be used when opening connections. For more information, see “Network Connection Preferences” in the online help. You can access this chapter by pressing F1 or the *Help* button in the *Preferences* dialog box.

## Support Settings

Under **Tools** > **Preferences** > **Application Design** > **Support Settings**, you can specify the amount of information stored in a log file and activate functions to record traces or collect statistics data. For more information, see the links in the [Related Information](#) section below.

Under **SAP HANA Data Sources (HTTP)**, you can specify that you use an HTTP proxy for communication with the SAP HANA system. If you have configured that your SAP HANA system can only be accessed using a proxy, select the **Use HTTP Proxy** checkbox.

- Under **Proxy Host**, enter the name of the system hosting the proxy service used by the HTTP destination.
- Under **Proxy Port**, enter the port to connect on the system hosting the proxy service.

You can also use this setting for support purposes: Using Fiddler as a proxy, you can record Fiddler traces, which help SAP diagnosing your issues with SAP HANA HTTP data sources. To set Fiddler as a proxy, select the **Use HTTP Proxy** checkbox.

- Under **Proxy Host**, enter **localhost**.
- Under **Proxy Port**, enter **8888**.

For more information, see SAP Note [2166049](#).

In the status line of the design tool, the indicator **HTTP Proxy: Off/On** shows you if this setting is switched on or not. By double-clicking on the indicator you can quickly access the **Preferences** page to change your current setting.

### **i** Note

This setting applies to SAP HANA data sources that are connected using HTTP as the protocol for communication only (not ODBC).

## Related Information

[Activating Runtime Traces \[page 320\]](#)

[Activating SAP JCo Traces \[page 321\]](#)

[Configuring the Report-Report Interface for Analysis Applications in Local Mode \[page 727\]](#)

## 38.4 Storage of Applications and Images

To work efficiently with the design tool, you need to know where your applications are stored and where to store the images and icons that you want to insert in your applications.



---

## Analysis applications

The applications you create are automatically stored in the folder <userhome>\Analysis-workspace\com.sap.ip.bi.zen\repository.

## Analysis applications history

The analysis applications history in the Application menu is persisted in your cache file <home directory>\.sap\com.sap.ip.bi\cache.

## Images and Icons

There are several possible locations where you can store the images and icons that you want to use in your application:

- in the applications directory <userhome>\Analysis-workspace\com.sap.ip.bi.zen\repository\MyApplication\MyImage.jpg
- in a subfolder of the relevant application directory, for example, <userhome>\Analysis-workspace\com.sap.ip.bi.zen\repository\MyApplication\MySubFolder\MyImage.jpg
- Internet or intranet

## Data source history

The data source history in the [Add Data Source](#) dialog box is persisted in <home directory>\.sap\com.sap.ip.bi\cache.

# 38.5 Selecting a Connection

## Prerequisites

Before you can choose a connection, you have to create connections to the BI backend systems containing the business data. SAP BusinessObjects Design Studio can access SAP HANA systems or SAP BW systems as BI backend systems.

## Context

Connections represent BW or SAP HANA systems. They are defined and configured by your administrator. Connections have to be active if you want to select a data source and use it immediately. However, you can also work with inactive connections when designing and validate the data sources for this connection later when the connection is active. The connection is automatically active when the backend system (SAP HANA or SAP BW) is up and running.

## Procedure

In the *Connection* box, click *Browse...* The *Select Connection* dialog opens where you can choose one of the created connections.

### Note

You can see all created connections in the design tool under ► *Tools* ► *Preferences* ► *Backend Connections*. ►  
To display newly defined connections in the corresponding table, click *Reload All Connections*.

## Results

You have selected a connection and can now select a data source based on this connection.

### Note

You can see all created connections in the design tool under ► *Tools* ► *Preferences* ► *Backend Connections*. ►  
To display newly defined connections in the corresponding table, click *Reload All Connections*.

## Related Information



[Selecting a Data Source \[page 71\]](#)

## 38.5.1 Defining Connections to BI Backend Systems

### Context

Before you can add data sources to your analysis applications, you have to create connections to the BI backend systems containing the business data. SAP BusinessObjects Design Studio can access SAP HANA systems or SAP BW systems as BI backend systems.

## Procedure

1. Open the design tool and choose **Tools > Preferences > Application Design > Backend Connections**.
2. Decide whether you want to create a SAP BW connection or a SAP HANA connection.
  - For SAP BW connections, click  **Launch SAP Logon** in the upper right corner of the **Connections maintained in SAP Logon** area.
    1. In the **SAP Logon** dialog box, click **New**.
    2. Add the required system parameters.
  - For SAP HANA connections, click  **ODBC Data Source Administrator** in the upper right corner of the **SAP HANA Connections maintained in ODBC Data Source Administrator** area.
    1. Click **Add...**
    2. Select the driver **HDBODBC32**.

### Note

This driver is only available if the SAP HANA client tools are installed on your local machine.
    3. Click **Finish**.
    4. Enter the SAP HANA system in the **Data Source Name** field.
    5. Enter the corresponding server and port in the **Server:Port** field.
    6. Click **OK** and again **OK**.
3. To display newly defined connections in the corresponding table, click **Reload All Connections**.
4. To close the dialog box, click **OK**.

## Results

The defined connections will be listed in the **Select Connection** dialog box, which appears after browsing for a different connection than the preselected one in the **Add Data Source** dialog box.

## 38.6 Saving an Application Using a Different Name



### Prerequisites

You have made changes to an existing application.

### Context

You want to save the changed application using a different name. Perform one of the following steps:

## Procedure

Click  **Application** . The **Save as** dialog box opens. The system suggests the current name for the application.

- In the **Name** box, type in a new name for the application and click **Save**. The new application name is displayed in the outline view of the editor.
- In the list of existing applications, select the one that you want to use for your application and click **Save**. The system asks you to confirm that you want to overwrite an existing application. Click **Overwrite**. The chosen application name is displayed in the relevant tab in the editor.



### Caution

Once you have confirmed that you want to overwrite an existing application, you cannot undo the changes.

You cannot select the name of an application that is open in another editor. Select another application name.

## Results

You have now saved the application using a new or different name.

## 38.7 Executing an Application

### Context

You can always execute an application locally in your Web browser while you are working on it, and you don not need to save it first. This enables you to easily check your design steps in the application.

## Procedure

Click  **Application**  in the menu or toolbar. The application is displayed in your Web browser.

---

## 38.8 Executing an Application on a Mobile Device

### Prerequisites

- To execute a mobile application, make sure that you have enabled external access to the embedded Web server. Activate this setting under ► [Tools](#) ► [Preferences](#) ►.
- Execute the application on your desktop Web browser.

### Context

To execute the application on a mobile device:

### Procedure

1. Click [Send to Mobile Device \(using QR code ®\)](#). The dialog box [QR Code](#) is displayed with the URL of the current application encoded.
2. If there is more than one suitable network adapter/interface with at least one IP address assigned, choose the required option in the dropdown box next to [IP address to use](#). The system generates a new QR code.
3. Scan the QR code with a mobile device, for example, an iPad (iPad 2 or higher), using one of the various QR code scanner apps.

### Results

The application opens in the mobile Safari browser on the iPad or iPhone and navigates to the application URL.

## 38.9 Collecting Support Information in Local Mode

### Context

If you encounter problems in the design tool, you can collect the relevant information to send to SAP in a zip file.

## Procedure

1. In the design tool, choose ► [Help](#) ► [Support](#) ► [Collect Support Information...](#) .
2. Select the target folder for the zip file.
3. Click [OK](#).
4. Choose the analysis applications you want to add.
5. Click [Package](#).  
The support information is saved in the file `DS_Support.zip`.
6. To view the content of the zip file, click [View...](#)
7. Click [OK](#).

## Results

You can attach the zip file to a customer message and send it to SAP.

# 38.10 Coordinating the Translation of Translatable Texts in Local Mode

## Context

The texts in analysis applications that are created by the application designers (for example, button texts) are translatable. To collect these texts for translation, the texts are stored in the `localization.properties` file that is located in the directory of your analysis application (`content.biapp` file).

The `localization.properties` file contains all translation-relevant property values and all manual entries in the [Text Pool](#) component.

### Note

Every time the analysis application is saved, the system updates the `localization.properties` file. Do not modify the file manually.

## Procedure

1. To prepare the translation of the `localization.properties` file, copy the file and append the required ISO language code as required.  
For example: `localization_de.properties` for the translation into German

2. Send the files to the people responsible for your translations.
3. After you receive the translated files, save the files in the same directory on your machine.

## Results

When executing the analysis application locally, you see the translated texts displayed as specified in your language settings in the Web browser.

### ➔ Tip

You can override the browser setting using the URL parameters `LANGUAGE` and `COUNTRY`. For example, you can specify American English by appending `&LANGUAGE=en&COUNTRY=US` to the URL of your analysis application.

**In the design tool, the original texts entered by the application designer are always displayed.**

### Example

The `localization.properties` file can contain the following texts:

```
# XMSG
BUTTON_1.TEXT=Filter
# XMSG
CHECKBOX_1.TEXT=Show Totals
# XMSG
TEXT_1.TEXT=Please enter your filter criteria
# XMSG
TEXT_POOL.Key1=This is {0}.
```

## 38.11 Configuring the Report-Report Interface for Analysis Applications in Local Mode

### Context

Before application users can use the report-report interface (RRI) in analysis applications, you need to configure the jump targets for the query that is used as data source in the analysis application. For more information, see

- “Using the Report-Report Interface” on SAP Help Portal at <http://help.sap.com>
- “BEx Query as Recipient” on SAP Help Portal at <http://help.sap.com>

If the jump target, which is configured for the data source query, is also a query, the target query is launched as a BEx Web application by default. In order to avoid this and ensure that the jump is handled by Design Studio and the target query is displayed as an analysis application, you need to specify a generic analysis template.

## Procedure

1. In the design tool, create an analysis application that can be used as the generic analysis template.  
As this analysis application will be used for all query jumps, it should be a very generic application, which can handle basic analysis of an arbitrary query. In general, it must conform to the following constraints:
  - Contains only one data source, which is loaded in script.
  - Accepts query ID and system ID as URL parameters `XQUERY` and `XSYSTEM` respectively.
  - Loads the single data source (usually in the `On Startup` script) by calling `assignDataSource()` and passing the `XQUERY` and `XSYSTEM` parameters to the corresponding parameters of `assignDataSource()`.

### ➔ Tip

A good starting point, and an example of a valid generic analysis template, would be to select [Generic Analysis Template](#) when creating an application in the design tool. Note that you do not need to assign a data source to this [Generic Analysis Template](#). When application users jump to the target query, the necessary query information is automatically added to the [Generic Analysis Template](#) as URL parameters (`XQUERY`, `XSYSTEM`).

2. Save your generic analysis template on your local machine.
3. To configure the runtime in order to use this analysis application as the generic analysis template for query jumps, proceed as follows:
  - a. Choose **Tools** > **Preferences** > **Application Design**.
  - b. In the **Report-Report Interface** section at the bottom of the page, under **Application to use for query jumps**, enter the name of the locally saved generic analysis template.

## Results

In addition to the `XQUERY` and `XSYSTEM` URL parameters, which are passed to the receiving application, the filters and selection context are also passed, which are needed for dimensions relevant to the target query from the sending application. The mappings from source to target query are processed on the BW system, and the application of the appropriate filter values are processed by the Design Studio runtime.

### ⚠ Restriction

Only query targets that are on the same system as the source query can be launched as analysis applications. All other targets are processed as BEx Web applications.



## 39 Terminology Essentials

Before you read any further, it might be useful to understand some basic OLAP and analysis design tool terminology.

<b>(analysis) application</b>	An entity created in the design tool of SAP BusinessObjects Design Studio. An application usually consists of user interface components like charts, crosstabs and buttons and data source aliases.
<b>connection</b>	A representation of BW or SAP HANA systems (in the design tool) that is used to add data sources.
<b>(user interface) component</b>	A design element and entity in the design tool. Components are used to visualize data (such as chart, crosstab) or to enable application users to interact with the data (for example, change filters or select drill-down dimensions). The appearance and behavior of components can be changed by editing their properties.
<b>data source</b>	A BW query or query view, a SAP HANA analytic or calculation view, used in the design tool.
<b>data source alias</b>	An instance of a data source at runtime of an application and an entity in the design tool.
<b>data binding</b>	A reference to a data source alias that provides the data for a component. Data binding is defined in the design tool and describes the relationship between components and data source aliases. These relationships are displayed in the outline view of the design tool.
<b>dimension</b>	A collection of related data members, which represents one aspect of a business; for example, products or sales.
<b>event</b>	A system notification about a specific user interaction, for example, a click on a button.
<b>measure</b>	A number or quantity that records a directly observable value or performance. Examples of measures include: sales, revenue, fixed costs, sales quantity, or number of employees.
<b>script</b>	Series of statements which are created by the user of the design tool (using the script editor or the statement wizard). By adding a script to a component, you can influence the behavior of this component and thus enable user interaction, also referred to as events, at runtime. A script typically consists of several statements.
<b>script editor</b>	A tool within the design tool to specify the actions that should take place when an event is triggered by an application user.
<b>statement</b>	A programmatic instruction within a script. The execution of a statement is typically triggered by user interaction with the component.
<b>statement wizard</b>	Dialog within the script editor that guides the application designer through the necessary steps and finally creates a script statement.

---

# Important Disclaimers and Legal Information

## Coding Samples

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

## Accessibility

The information contained in the SAP documentation represents SAP's current view of accessibility criteria as of the date of publication; it is in no way intended to be a binding guideline on how to ensure accessibility of software products. SAP in particular disclaims any liability in relation to this document. This disclaimer, however, does not apply in cases of wilful misconduct or gross negligence of SAP. Furthermore, this document does not result in any direct or indirect contractual obligations of SAP.

## Gender-Neutral Language

As far as possible, SAP documentation is gender neutral. Depending on the context, the reader is addressed directly with "you", or a gender-neutral noun (such as "sales person" or "working days") is used. If when referring to members of both sexes, however, the third-person singular cannot be avoided or a gender-neutral noun does not exist, SAP reserves the right to use the masculine form of the noun and pronoun. This is to ensure that the documentation remains comprehensible.

## Internet Hyperlinks

The SAP documentation may contain hyperlinks to the Internet. These hyperlinks are intended to serve as a hint about where to find related information. SAP does not warrant the availability and correctness of this related information or the ability of this information to serve a particular purpose. SAP shall not be liable for any damages caused by the use of related information unless damages have been caused by SAP's gross negligence or wilful misconduct. All links are categorized for transparency (see: <http://help.sap.com/disclaimer>).



**go.sap.com/registration/  
contact.html**

© 2016 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 <http://www.sap.com/corporate-en/legal/copyright/index.epx> for additional trademark information and notices.