



# BusinessObjects XI Integration for SAP Solutions User's Guide

## Copyright

© 2009 SAP® BusinessObjects™. All rights reserved. SAP BusinessObjects and its logos, BusinessObjects, Crystal Reports®, SAP BusinessObjects Rapid Mart™, SAP BusinessObjects Data Insight™, SAP BusinessObjects Desktop Intelligence™, SAP BusinessObjects Rapid Marts®, SAP BusinessObjects Watchlist Security™, SAP BusinessObjects Web Intelligence®, and Xcelsius® are trademarks or registered trademarks of Business Objects, an SAP company and/or affiliated companies in the United States and/or other countries. SAP® is a registered trademark of SAP AG in Germany and/or other countries. All other names mentioned herein may be trademarks of their respective owners.

2009-07-17

# Contents

<b>Chapter 1</b>	<b>Introduction</b>	<b>7</b>
	About this guide.....	8
	Who should read this guide.....	8
<b>Chapter 2</b>	<b>What's new in BusinessObjects XI integration for SAP Solutions</b>	<b>9</b>
	Extending platform and language support.....	10
	Mass distribution of personalized SAP Crystal and Web Intelligence reports.....	11
<b>Chapter 3</b>	<b>Reporting with SAP BW</b>	<b>13</b>
	Reporting off BW Queries.....	14
	Formatted reporting overview.....	14
	Creating new queries for reports.....	19
	Selecting existing queries for reports.....	23
	Saving reports to BW.....	25
	Using your own reports as templates.....	27
	Building your first report off the SAP DemoCube.....	28
	Building a report off a BW hierarchy.....	38
	Reporting off BW Queries and Cubes: MDX Query Driver.....	53
	MDX Query driver overview.....	53
	Formatted reporting overview.....	54
	Building a report off a BW query.....	66
	Building a report off a BW hierarchy.....	71
	Building a report off a BW query with a hierarchy node variable.....	76
	Building a report off a BW query with a hierarchy variable and a hierarchy node variable.....	79

## Contents

Crystal Reports and variable pick lists.....	82
Reporting off Operational Data Stores.....	85
Operational Data Store driver overview.....	85
Selecting an ODS object.....	85
Designing and Translating Reports.....	89
Designing reports for translation.....	90
Publishing and Viewing Reports.....	94
Publishing predefined reports to BusinessObjects Enterprise.....	94
Publishing reports from SAP BW to BusinessObjects Enterprise.....	96
Viewing reports.....	96
Reporting off SAP Data.....	98
Overview of the Data Access Components drivers.....	98
Connecting to SAP with Crystal Reports.....	99
Crystal Reports configuration options.....	102
BusinessObjects InfoView.....	104
 <b>Chapter 4    Reporting with SAP Relational Data Sources</b>	 <b>107</b>
Reporting off Tables, Views, Clusters, and Functions.....	108
Open SQL driver overview.....	108
Selecting tables, views, functions, and clusters.....	109
Linking tables and views.....	113
Reporting off ABAP functions.....	114
Reporting off ABAP data clusters.....	121
Mapping ABAP data clusters.....	122
Reporting off InfoSets and SAP Queries.....	139
InfoSet Query driver overview.....	139
Accessing an InfoSet or SAP query.....	140
Selecting an InfoSet or SAP query in Crystal Reports.....	140

Contents

<b>Chapter 5</b>	<b>SAP specific features in InfoView</b>	<b>145</b>
	Working with My Groups.....	146
	Publishing and BusinessObjects XI Integration for SAP.....	146
	Personalization in BusinessObjects Enterprise.....	147
	What is personalization?.....	147
	Parameters.....	147
	Scheduling reports that contain parameters.....	151
<b>Appendix A</b>	<b>More Information</b>	<b>153</b>
<b>Index</b>		<b>157</b>

## Contents



# Introduction



# 1

## chapter



## About this guide

BusinessObjects Integration for SAP Solutions provides you, the SAP user, with the ability to extract the business intelligence contained within your SAP systems. BusinessObjects InfoView allows you to share your reports over the Web, and SAP Authentication enables Single Sign On between your SAP system and BusinessObjects Enterprise.

BusinessObjects Integration for SAP Solutions consists of several main components: Crystal Reports is the report-design tool that allows you to report off your SAP data; BusinessObjects Enterprise provides the framework for managing, scheduling, and distributing reports over the Web. This solution allows tools such as Voyager, Web Intelligence, LiveOffice as well as Xcelsius to report of SAP.

Please refer to SAP-specific chapters within the user's guides of these tools for further details.

## Who should read this guide

This guide is intended for SAP and BusinessObjects users who want to design and report of SAP data sources using BusinessObjects tools.





# What's new in BusinessObjects XI integration for SAP Solutions



# 2

chapter



This section gives a high-level overview of the key new features provided in BusinessObjects XI Integration for SAP Solutions.

BusinessObjects XI Integration for SAP Solutions delivers the strongest business intelligence solution for SAP customers, with new capabilities aimed at adding value to SAP products while lowering the total cost of ownership and providing an SAP-familiar user experience, especially for end-user self-service in ad-hoc query, reporting, and analysis.

## Extending platform and language support

BusinessObjects XI Integration for SAP Solutions provides additional platform support for SAP integration.

### **New platform**

Linux SUSE10 SP2

### **New web application servers**

- Tomcat 6
- Weblogic 10.3
- Webshpere7
- SAP Web Application Server 7.10

### **New web browsers**

- Internet Explorer 8
- Safari 3.2.1

### **New language packs**

- Finnish
- Czech

### **Other upgrades**

The SAP Unicode RFC library will now be supported on Windows platforms.

## Mass distribution of personalized SAP Crystal and Web Intelligence reports

Report publishing capabilities in BusinessObjects XI provide the ability to broadcast personalized SAP Crystal and Web Intelligence reports to a large user population based on SAP user role definitions. With these new capabilities there is no need to replicate SAP data access authorization settings in BusinessObjects Enterprise. For more information about Publications, see the *BusinessObjects Enterprise XI Publishing Guide*.

## 2 | What's new in BusinessObjects XI integration for SAP Solutions

*Mass distribution of personalized SAP Crystal and Web Intelligence reports*



# Reporting with SAP BW



## 3 chapter



## Reporting off BW Queries

This section introduces you to Crystal Reports and the integrated features of the SAP Tools toolbar. It shows how to create and select queries for reports and how to save reports to BW. It concludes with a tutorial that guides you through the creation of a report that is based on the SAP DemoCube.

### Formatted reporting overview

With the BW Query driver, you can design formatted Crystal reports that are based on the data that is stored in SAP Business Information Warehouse (BW). You base these reports on queries that you create with SAP's Business Explorer (BEx) Query Designer. Once you have designed your Crystal reports, you can save them to BW and, if required, target them for translation in order to make them available to users in multiple languages.

This section briefly shows how to start SAP's BEx Query Designer from within Crystal Reports, but focuses primarily on Crystal Reports and your task of selecting BW queries as data sources for Crystal reports. This section concludes with a step-by-step tutorial that guides you through the creation of a sales report that displays data from the SAP DemoCube.

BusinessObjects Enterprise allows you to share Crystal reports over the Web so that all users can view your formatted content. For details about publishing reports to BusinessObjects Enterprise, see [Publishing and Viewing Reports](#) on page 94.

Your BW queries may contain SAP variables that appear as parameters in BusinessObjects Enterprise and Crystal Reports. To accommodate these variables, BusinessObjects XI Integration for SAP Solutions employs dynamic pick lists. A dynamic pick list provides you with a list of possible values to choose for a parameter (variable). Additionally, the values that are displayed in a dynamic pick list correspond to your user rights—you see only pick list values that you have the rights to report off.




### Commands and connection settings in Crystal Reports




Crystal Reports includes several integrated tools that allow you to report off your BW data. You can access many of these tools from two locations: click

the appropriate button on the SAP Tools toolbar, or click the corresponding command on the SAP menu. This section describes the available commands and shows how to change the related settings in Crystal Reports.

**Tip:**

To show/hide the SAP Tools toolbar in Crystal Reports, on the **View** menu, click **Toolbars**. Select the **SAP Tools** check box, and click **OK**. Commands without icons are available only on the SAP menu.

Button	Command	Function
	Create New Report from a Query	Create a new report based on a query that is already defined in BW.
	Open Report	Open a Crystal report that has already been saved in BW.
	Save Report	Save the report that is currently open to BW. You have the option to prepare the report for translation and publish it to BusinessObjects Enterprise.
	Save Report As	Save the report to BW with a different description and/or in a different role.

Button	Command	Function
	Start BW Query Designer	Launch the BW Query Designer to create a new query, or to modify a query that is already defined in BW.
	Disconnect	Disconnect from SAP BW.
	Settings	Change your connection settings.
	SAP mode	Change a predefined SAP report into a regular Crystal report.

For instructions on how to change your connection settings, refer to [Changing your connection settings](#) on page 16. For instructions on how to show field descriptions and technical field names, refer to [Showing field descriptions and technical field names](#) on page 18.

## Changing your connection settings

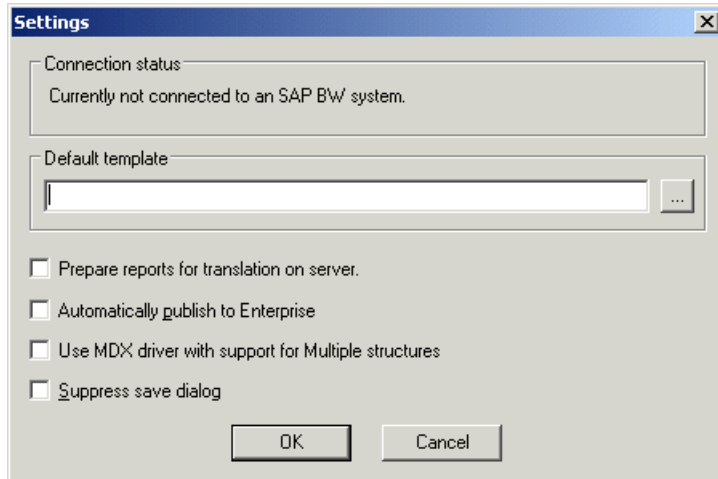
1. On the **SAP** menu in Crystal Reports, click **Settings**.

### Tip:

 You can also click the corresponding button on the toolbar.

The Settings dialog box appears.





2. Select from the available options to customize the behavior of Crystal Reports:

- **Default template**

In this field, you can specify a report (.rpt file) that you want to use as a template for any new reports that you create from BW queries. For details, see [Using your own reports as templates](#) on page 27.

- **Prepare reports for translation on server**

If you select this option, the reports that you save in BW are prepared for translation on the SAP system. For details, see [Preparing reports for translation](#) on page 90.

- **Automatically publish to Enterprise**

If you select this option, the report is automatically published to BusinessObjects Enterprise when you save it in BW. You can view published reports through several ways: from the SAP Easy Access Menu, the SAP Enterprise Portal, or BusinessObjects InfoView. For details, see [Publishing and Viewing Reports](#) on page 94.

- **Use MDX driver with support for Multiple structures**

If you select this option, Crystal Reports uses the MDX Query driver to connect to the data source rather than using the BW Query driver. For details about the MDX Query driver, see [MDX Query driver overview](#) on page 53.

- **Suppress save dialog**

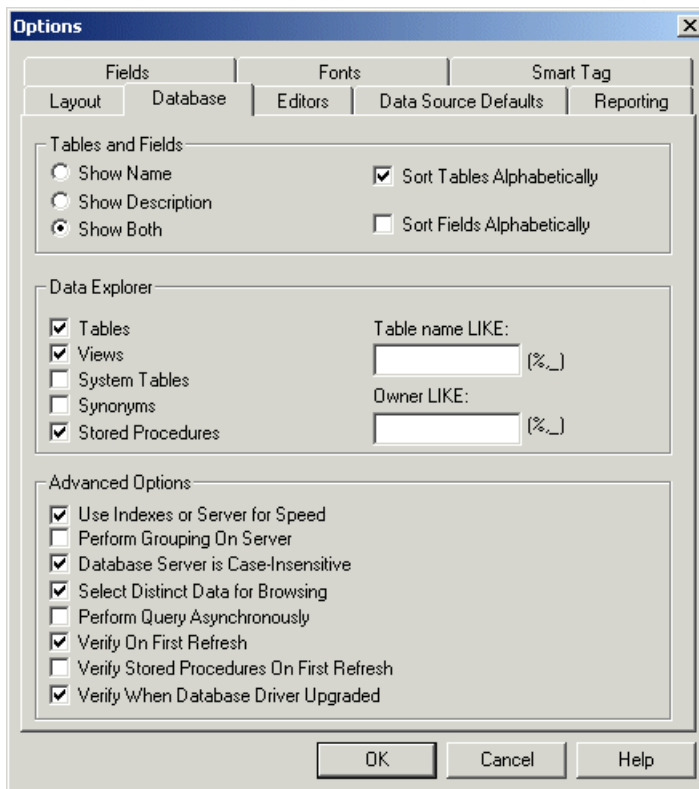
If you select this option, you are not prompted every time you save reports to BW, and the settings that you specify here are used for all reports.

## Showing field descriptions and technical field names

1. On the **File** menu, click **Options**.

The "Options" dialog box appears, with the Layout tab active.

2. Click the **Database** tab.



3. In the "Tables and Fields" area, select **Show Both**.

When you select this option, the program identifies tables and fields using both the names and the descriptions that were assigned to them in the SAP system.

4. Click **OK**.

## Creating new queries for reports

If you have not already defined one or more queries within BW, you can start SAP's BEx Query Designer from within Crystal Reports. The query that you define is automatically selected as the data source for a new Crystal report.

### Tip:

When you define a query, you specify the subset of data that you want to retrieve from a particular InfoProvider that is stored in BW. Therefore, you must be able to access at least one InfoProvider on your BW system.

## To create a new query for a report

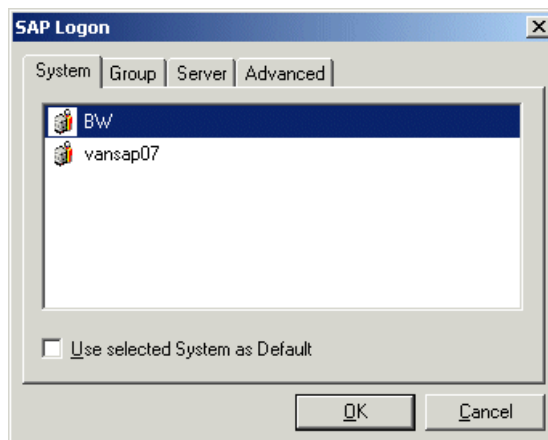
1. On the **SAP** menu in Crystal Reports, click **Start BW Query Designer**.

### Tip:



You can also click the corresponding button on the toolbar.

The "SAP Logon" dialog box appears.

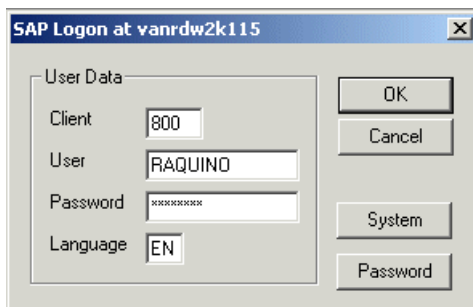


2. On the **System** tab, select your BW system, then click **OK**.

**Note:**


If your system is not displayed, click the Group or Server tab and provide the valid BW system information. Contact your administrator if you are unsure of the entries that need to be made.

The next "SAP Logon" dialog box prompts you for User Data.

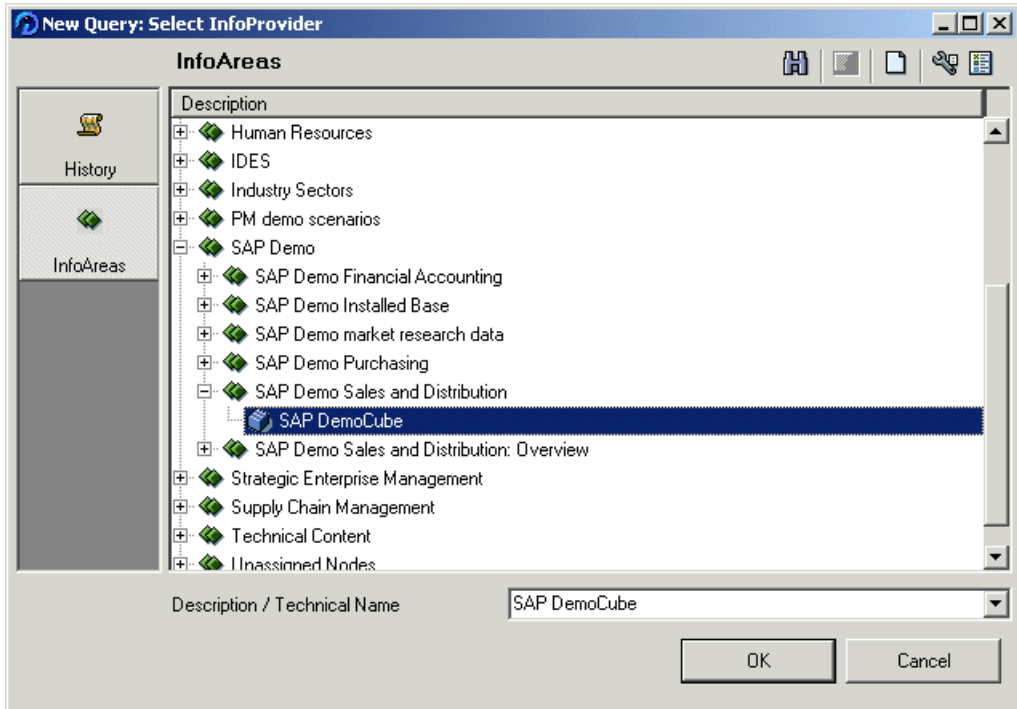
The image shows a screenshot of the "SAP Logon at vanrdw2k115" dialog box. It has a title bar with the text "SAP Logon at vanrdw2k115" and a close button. The main area is titled "User Data" and contains four input fields: "Client" with the value "800", "User" with the value "RAQUINO", "Password" with masked characters "xxxxxxxx", and "Language" with the value "EN". To the right of these fields are four buttons: "OK", "Cancel", "System", and "Password".

3. Type your usual SAP user credentials in the **Client**, **User**, **Password**, and **Language** fields. Then, click **OK**.

The Query Designer appears.

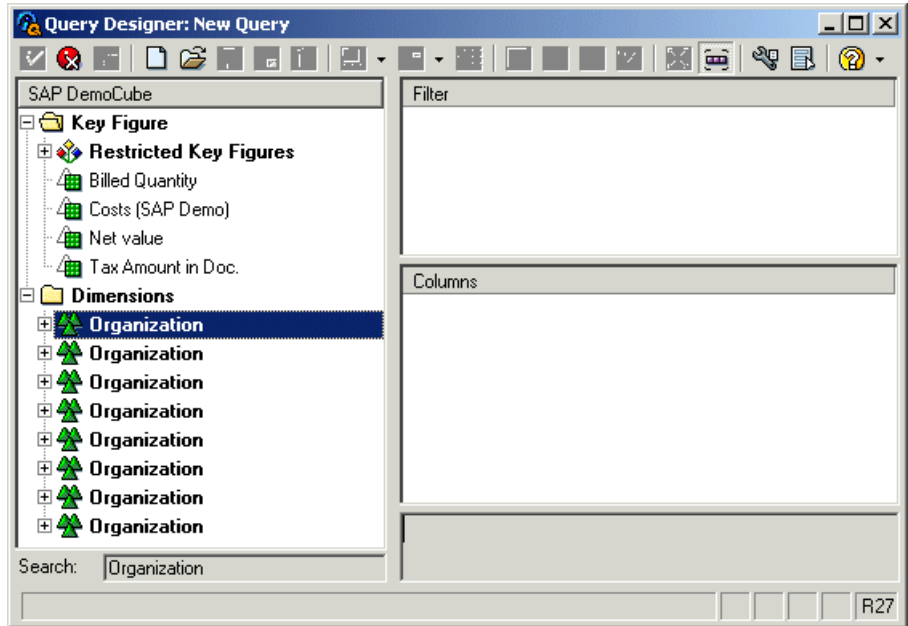
4.  On the Query Designer's toolbar, click **New Query**.

The "New Query: Select InfoProvider" dialog box displays the InfoAreas and InfoProviders that are available on your system.



5. Select the InfoProvider on which you want to base your query, then click **OK**.



The Query Designer automatically displays the InfoProvider in Flat mode, a display mode, which is recommended when designing Crystal reports using the BW Query driver. For details about Flat mode, see the SAP Library > Business Information Warehouse > Business Explorer.



6. Define your query by selecting measures and characteristics from the Key Figure and Dimensions lists and dragging them to the Columns area.

**Note:**

For complete information about using the Query Designer, see the SAP Library > Business Information Warehouse > Business Explorer > Query Design: BEx Query Designer.

7.  Click **Query Properties**.
8. Click the Extended tab and ensure that **Allow External Access to this Query** is selected.  
  
This option allows other programs, like Crystal Reports, to access this query.
9.  Click **Save Query**.
10. In the Save Query dialog box, click **Roles** and select the role where you want to save the query.
11. Type a **Description** and a **Technical Name** for the query, then click **Save**.

The Technical Name must uniquely identify the query and must adhere to SAP naming conventions.

12.  Click **Quit and Use Query**.

Crystal Reports generates a report that uses your query as its data source. The report opens in the Design tab of Crystal Reports. You can now add fields, titles, charts, and other objects to the report.

## Selecting existing queries for reports

If you or someone else in your organization has already defined one or more queries for reporting purposes within BW, you can select those queries as data sources for new Crystal reports.

### Note:

The Data Explorer also provides an SAP BW Query node, which can be used to connect to and select a BW query for a report. Use Data Explorer when attempting to report on an InfoCube directly using the MDX driver.

## To select an existing query for a report

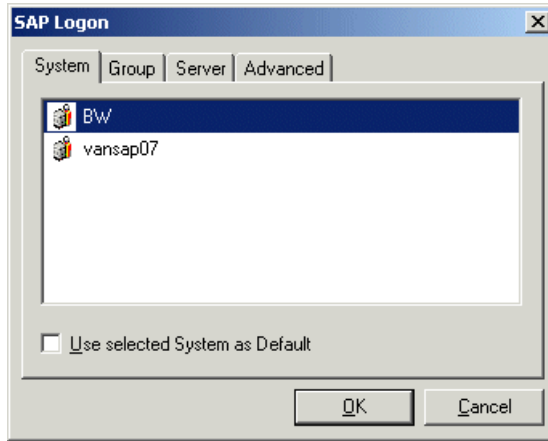
1. On the **SAP** menu in Crystal Reports, click **Create New Report from a Query**.

### Tip:



You can also click the corresponding button on the toolbar.

The "SAP Logon" dialog box appears.

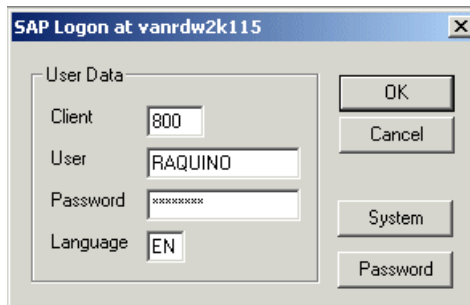


2. On the **System** tab, select your BW system, then click **OK**.

**Note:**

If your system is not displayed, click the **Group** or **Server** tab and provide the valid BW system information. Contact your administrator if you are unsure of the entries that need to be made.

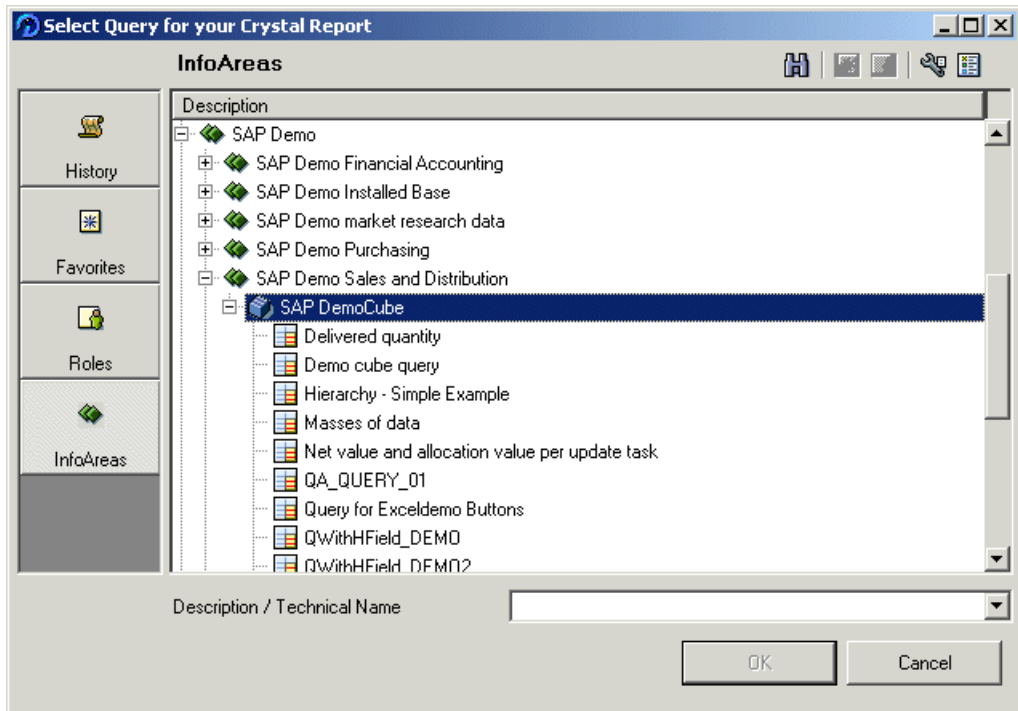
The next "SAP Login" dialog box prompts you for User Data.



3. Type your usual SAP user credentials in the **Client**, **User**, **Password**, and **Language** fields, then click **OK**.

The "Select Query for your Crystal Report" dialog box displays the queries that are available through your Favorites, Roles, and InfoAreas in BW.





4. Select the query with the data that you want to report off, then click **OK**.

Crystal Reports generates a report that uses your query as its data source. The report opens in the Design tab of Crystal Reports. You can now add fields, titles, charts, and other objects to the report.

## Saving reports to BW

After designing a report based on a query, you can save the report to a role in BW. At the same time, you can prepare the report for translation or automatically publish it to BusinessObjects Enterprise.

Before saving your report, check its title by clicking Summary Info on the File menu. The title that appears here is displayed to users in BusinessObjects Enterprise. The technical name of the query is used as the default title, but you can replace it with a descriptive title that is more meaningful to SAP users.

**Tip:**

Use the Save As command on the File menu to save the report to disk.

## To save a report to BW

1. On the **SAP** menu, click **Save Report**.

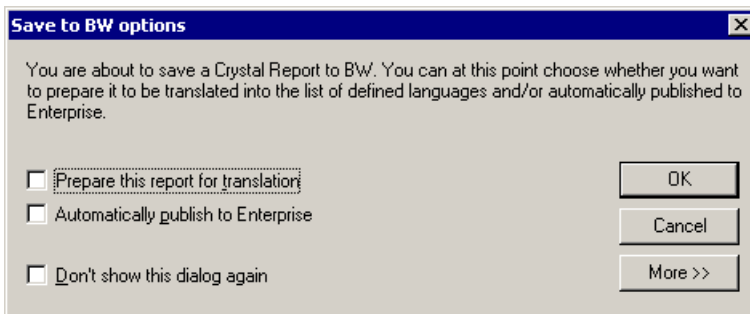
**Tip:**



You can also click the corresponding button on the toolbar.

2. In the "Save a Crystal Report to BW" dialog box, select the role where you want to save the new Crystal report.
3. If you are using NetWeaver 7.0.2 or higher, type a **Technical Name** for the report (maximum 25 characters long).
4. Type a **Description** for the report (optional if you entered a **Technical Name**).
5. Click **Save**.

Depending on your connection settings, the "Save to BW options" dialog box may appear.



6. Select from the available options:

- **Prepare this report for translation**

When you select this option, the strings in your report are extracted and made available for translation within SAP. For details, see [Designing reports for translation](#) on page 90.

- **Automatically publish to Enterprise**

For more information on publishing reports, see [Publishing and Viewing Reports](#) on page 94.

If you are unable to publish your report to BusinessObjects Enterprise, you may need to consult with your administrator about licensing.

- **Don't show this dialog again**

If you select this option now, you can modify your default settings later by clicking Settings on the SAP menu. For details, see [Commands and connection settings in Crystal Reports](#) on page 14.

7. Click [OK](#).

## Using your own reports as templates

By creating customized reports that serve as templates, you can speed up the process of designing and formatting the reports that you generate from BW queries. You can, for instance, create reports that include customized titles, corporate logos, or contact details in the Report Header. As well, you can place copyright information along with page numbers in the Page Footer. You can then select one of these reports as the default template, which "Crystal Reports" will use whenever you generate reports from BW queries.

### To create a report to use as a template

1. Using Crystal Reports, create a blank report without specifying a data source.
2. Add the elements that you want included in the template (text objects, graphics, hyperlinks, Special Fields, and so on).

**Note:**

- You may add objects to only the Report Header, Page Header, Report Footer, and Page Footer sections; these are the sections that are merged into the final report.
  - When adding graphics to a template, you can insert them as pictures or as OLE objects (static, embedded, or linked).
  - You can add Special Fields to your report template, as long as the fields are in no way dependent upon your SAP data. Predefined Special Fields are available from the Field Explorer in Crystal Reports.
3. When you have finished designing the template, save your work.

## To specify a default template

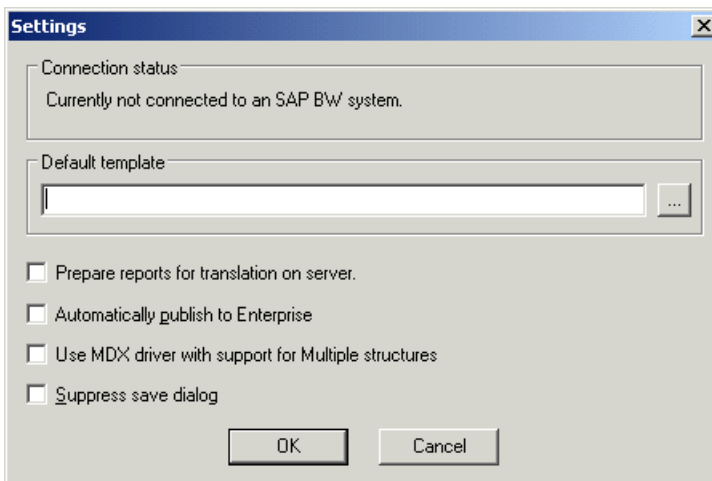
1. On the **SAP** menu in Crystal Reports, click **Settings**.

**Tip:**



You can also click the corresponding button on the toolbar.

The "Settings" dialog box appears.



2. In the **Default template** field, specify the report (.rpt file) that you want to use as a template for all new reports that you create from BW queries.
3. Click **OK**.

**Note:**

If you later rename, move, or delete your report template, Crystal Reports will prompt you with an error message when you attempt to create a new report from a BW query. You must specify a new report template (or none at all) before you can create the new report.

## Building your first report off the SAP DemoCube

This tutorial provides an introduction to Crystal Reports as you design a sales report. You begin by creating a query, which is selected as the report's data source. You then:

- Insert database fields.
- Group and sort data.
- Summarize data.
- Add a template.
- Specify a title.
- Preview the report.

## How to create a new query

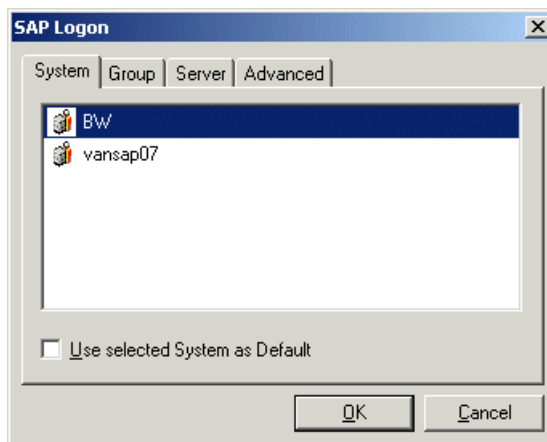
1. On the **SAP** menu in Crystal Reports, click **Start BW Query Designer**.

### Tip:



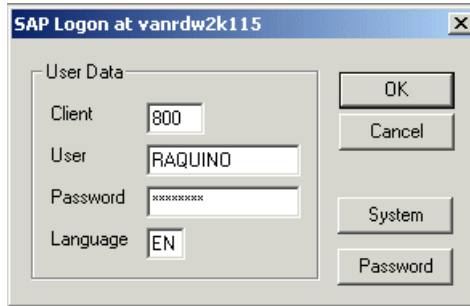
You can also click the corresponding button on the toolbar.

The "SAP Logon" dialog box appears.



2. On the **System** tab, select your BW system, then click **OK**.

The next "SAP Logon" dialog box prompts you for User Data.

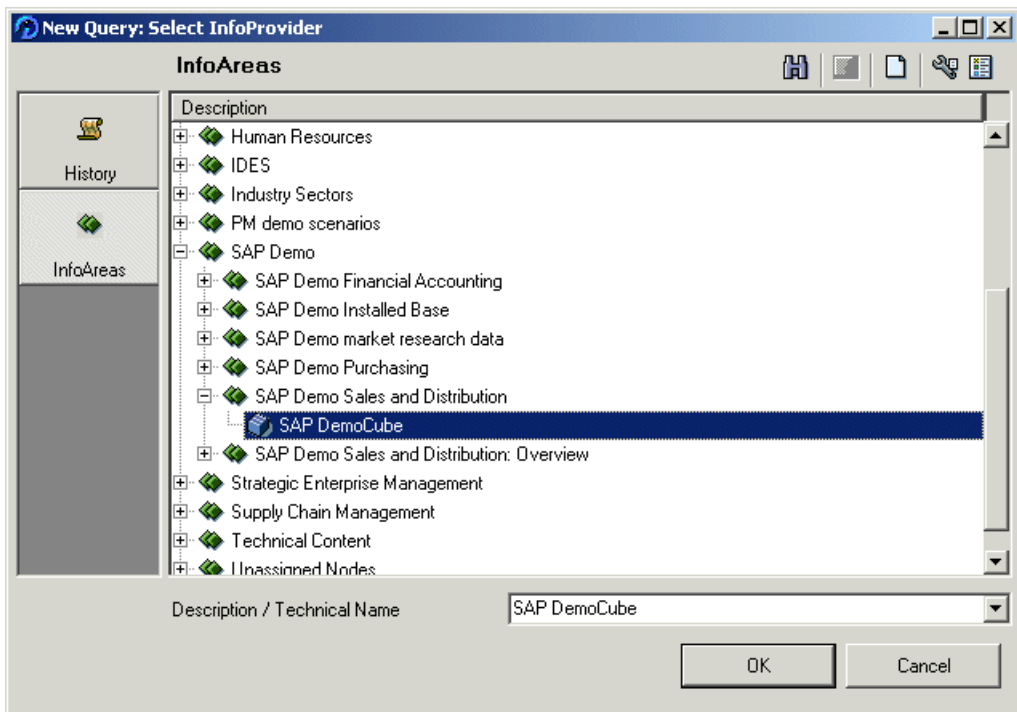


3. Type your usual SAP user credentials in the **Client**, **User**, **Password**, and **Language** fields. Then, click **OK**.

The Query Designer appears.

4.  On the Query Designer's toolbar, click **New Query**.

The "New Query: Select InfoProvider" dialog box displays the InfoAreas and InfoProviders that are available on your system.

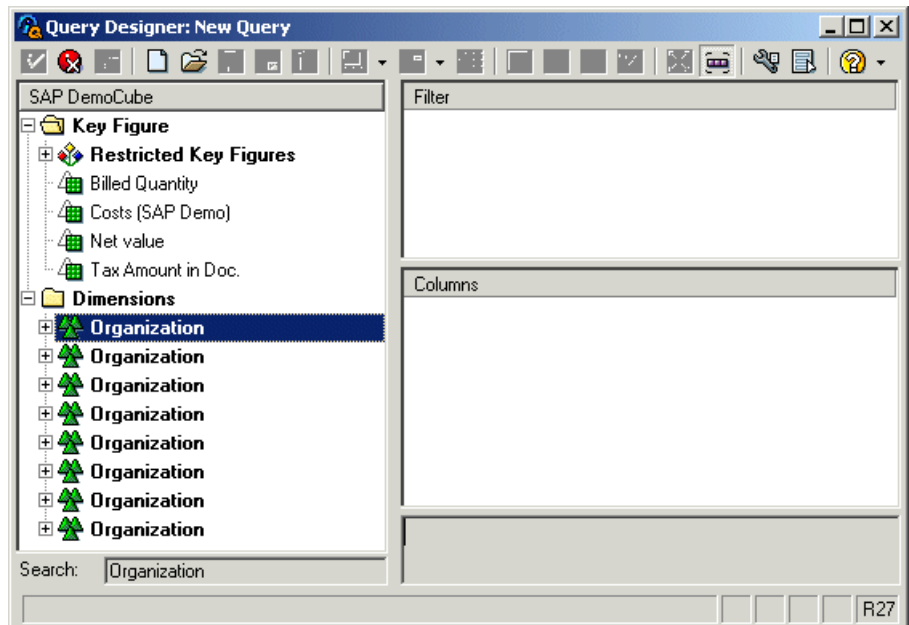


5. Select the **SAP DemoCube** and click **OK**.

**Note:**

The location of the SAP DemoCube varies, so you may need to browse to find it.

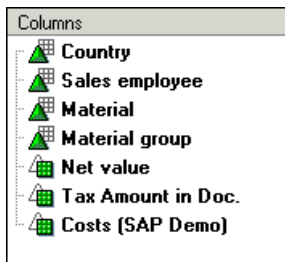
The Query Designer automatically displays the InfoProvider in Tabular Display mode, which is required when designing Crystal reports. For details about Tabular Display mode, see the SAP Library > Business Information Warehouse > Business Explorer.




6. Select the following characteristics from the Dimensions list, and drag them to the Columns area:
  - Country
  - Sales employee
  - Material
  - Material group
7. Select the following measures from the Key Figure list, and drag them to the Columns area:
  - Net value

- Tax Amount in Doc.
- Costs (SAP Demo)

The Columns area should now appear as follows:




8.  Click **Query Properties**.
9. Click the Extended tab and ensure that **Allow External Access to this Query** is selected.

This option allows other programs, like Crystal Reports, to access this query.

### Note:

For complete information about using the Query Designer, see the SAP Library > Business Information Warehouse > Business Explorer > Query Design: BEx Query Designer.

## To save the query in BW

1.  Click **Save Query**.
2. In the "Save Query" dialog box, click **Roles** and select the role where you want to save the query.
3. Type a **Description**, such as `My sample sales query`
4. Type a **Technical Name**, such as `SALES_DEMO_QUERY`

The Technical Name must uniquely identify the query—that is, the name can appear only once across all InfoProviders in BW. The Technical Name can be up to 30 characters long and must begin with a letter.

5. Click **Save**.



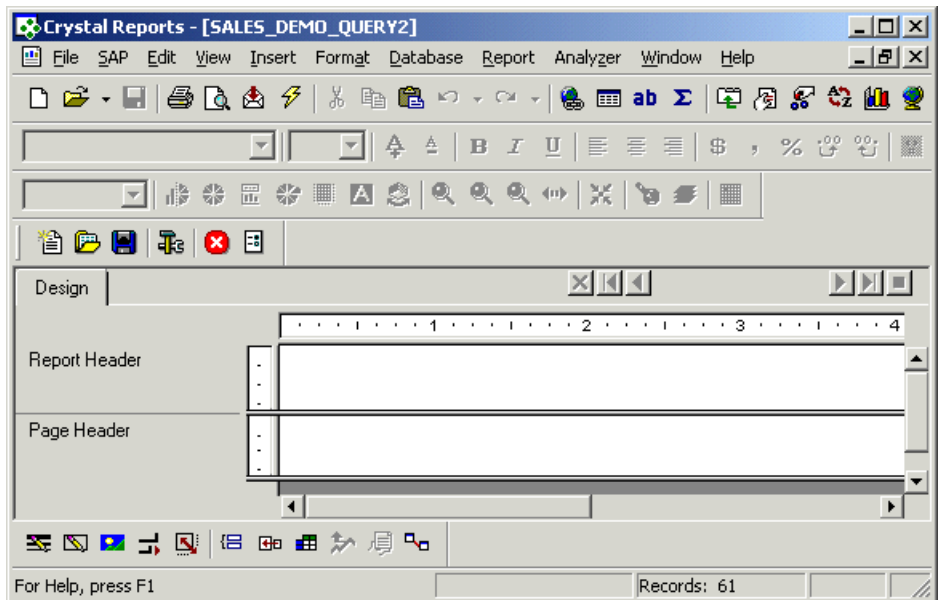
## How to create a new report based on the query

This section guides you through the process of creating a new report based on the query you created in the last section, [How to create a new query](#) on page 29.

### To create a new report based on the query

1.  While still in the BW Query Designer, click **Quit and Use Query**.

Crystal Reports generates a report that uses your query as its data source. The report opens in the **Design** tab of Crystal Reports. You can now add fields, titles, charts, and other objects to the report.

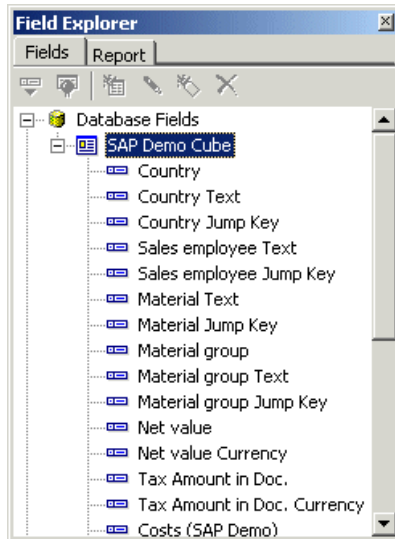


#### Tip:

To ensure that Crystal Reports displays technical names and/or descriptions for SAP tables and fields, click **Options** on the **File** menu. Click the **Database** tab, and make your selection in the "Tables and Fields" area.

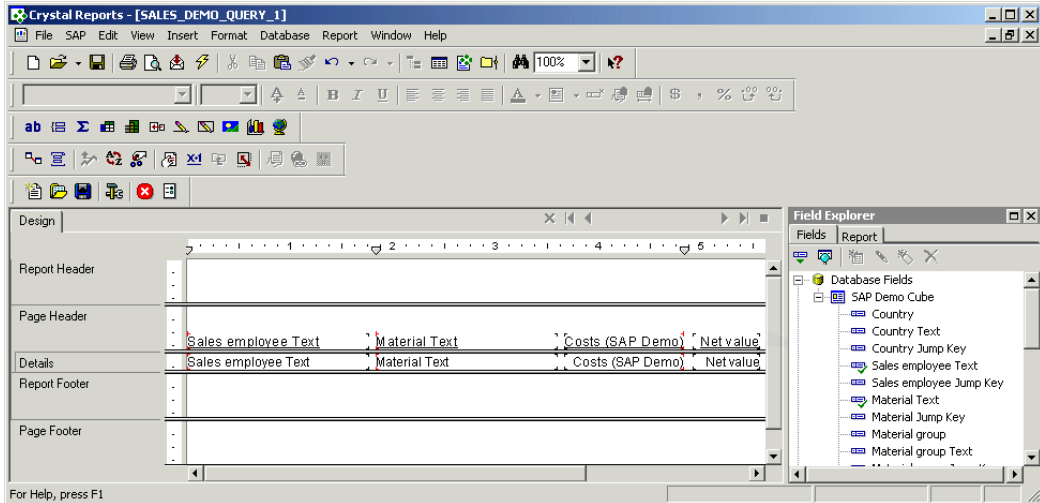
2. On the **View** menu, click **Field Explorer**.

The "Field Explorer" appears. You can expand **Database Fields** and "My sample sales query" to see a list of **Description** fields available for your report.



#### To select fields for the report

1. Drag and drop the following fields from the Field Explorer to the Details section of the report:
  - Sales employee Text
  - Material Text
  - Costs (SAP Demo)
  - Net value

**Tip:**

You can drag and drop fields on the **Design** tab to reposition them.

2. In the Page Header area of the report, double-click **Costs (SAP Demo)**, highlight the text, and type *Costs* in its place.

This changes the default column heading that is displayed in the report.

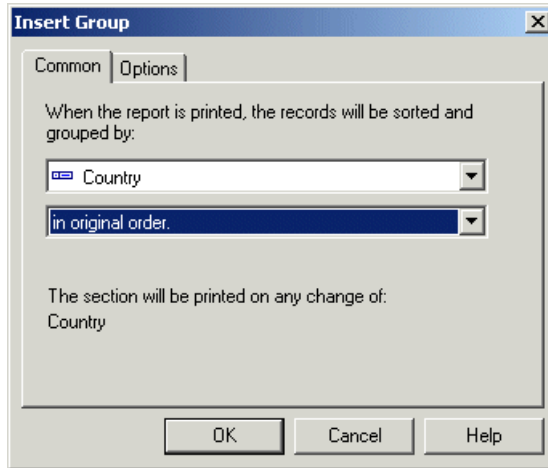
## How to group and sort data

This section guides you through the process of grouping and sorting data based on the report you created in the last section, [How to create a new report based on the query](#) on page 33.

### To group and sort the report's data

1. On the **Insert** menu, click **Group**.

The "Insert Group" dialog box appears.



2. Click the first list and select **Country** as the field by which you want the data grouped.
3. Click the second list and select **in ascending order** as the sort direction.
4. Click **OK**.
5. On the **Insert** menu, click **Group**.
6. Click the first list and select **Material Text** as the field you want the data grouped by.
7. Click the second list and select **in original order** as the sort direction.
8. Click **OK**.

Your report should now appear like this:

Report Header				
Page Header	Sales employee Text	Material Text	Costs	Net Value
Group Header #1:	Group #1 Name			
Group Header #2:	Group #2 Name			
Details	Sales employee Text	Material Text	Costs (SAP Demo)	Net value
Group Footer #2:				
Group Footer #1:				
Report Footer				
Page Footer				

When you view this report, the records are first grouped according to the values in the Country field. The resultant groups are then displayed in the report in ascending alphabetical order. The records in each Country group are further divided into secondary groups based on the values in the Material Text field. The resultant groups are displayed in their original order.

## How to summarize and preview data

This section guides you through the process of summarizing and previewing data based on the report you created in the last section, [How to create a new report based on the query](#) on page 33.

### To select a template for your report

1. On the **Report** menu, click **Template Expert**.
2. In the **Available Templates** list, click **High Contrast**.
3. Click **OK**.

### To add a title to your report

1. On the **File** menu, click **Summary Info**.

The "Document Properties" dialog box appears with the Summary tab displayed.

2. In the **Title** field, type the text you want to display as the title of the report, such as `Sales grouped by country and material`.
3. Click **OK**.

The title is inserted into the template's Page Header when you preview the report.

### To preview and save your report

1. On the **Report** menu, click **Refresh Report Data**.

Crystal Reports refreshes the report against the query in BW and displays your newly designed report in the **Preview** tab.

2. On the **SAP** menu, click **Save Report**, and save the report to a role in BW.

For details, see [Saving reports to BW](#) on page 25.

You may want to resize or reposition report elements, or further modify the report's design. You can, for instance, select and drag the borders of columns and column headings in order to resize them in the Preview tab; or, you can quickly display only a subset of the data by filtering the records with the Select Expert.

The *Crystal Reports Online Help* provides extensive conceptual and procedural information, tutorials, and examples to help you use Crystal Reports to its fullest potential. Access the help by pressing **F1** anywhere within Crystal Reports.

## Building a report off a BW hierarchy



This tutorial demonstrates how to report off a BW query with a hierarchy using Crystal Reports. You begin by creating a simple BW query with a hierarchy. You then complete these steps:

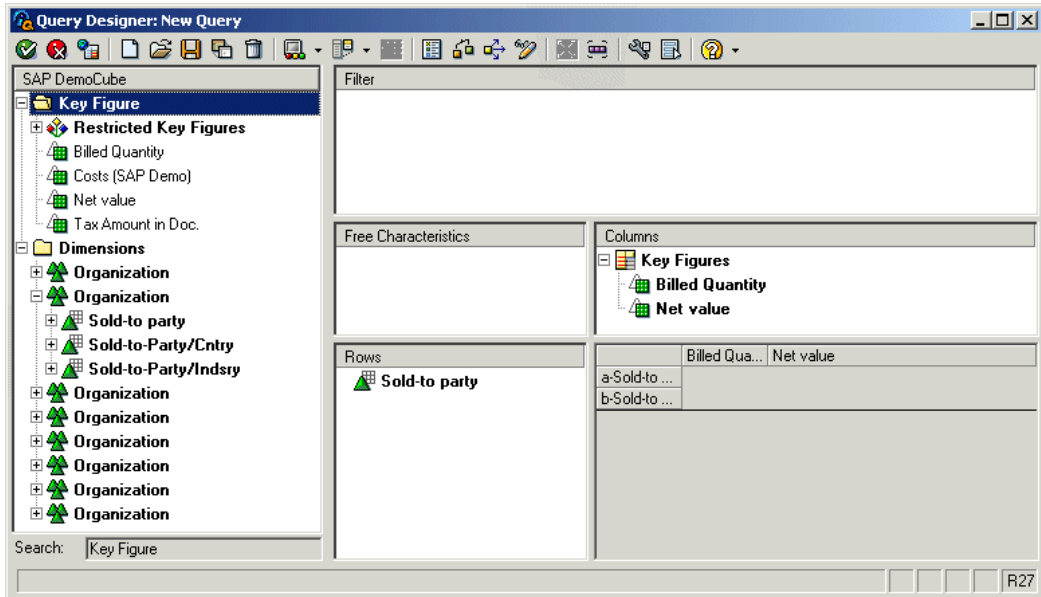
- Create a new report and group the data hierarchically.
- Insert a summary.
- Adjust object positioning dynamically.

### Note:

This tutorial builds on the topics covered in [Building your first report off the SAP DemoCube](#) on page 28, which provides more details on steps such as logging on to your SAP system, selecting the SAP DemoCube, and saving queries.

## To design a simple BW query with a hierarchy

1. On the **SAP** menu in Crystal Reports, click **Start BW Query Designer**.  
Log on to your SAP system if prompted.
2.  On the Query Designer's toolbar, click **New Query**.
3. Select the **SAP DemoCube** and click **OK**.
4.  On the Query Designer's toolbar, click **Table Display** to toggle to the view of the Query Designer that shows the Free Characteristics and Rows areas as well as the Filter and Columns areas.



5. Select the following measures from the **Key Figure** list, and drag them to the Columns area:
  - Billed Quantity
  - Net Value
6. Select the **Sold-to party** characteristic from the **Dimensions** list, and drag it to the Rows area.
7. In the Rows area, right-click **Sold-to party**, and click **Properties**.  
The Characteristic Properties dialog box appears.

**Characteristic Properties for Characteristic Sold-to party**

**Description** Sold-to party **Technical Name** OD\_SOLD\_TO

**Display As**  
Key and Name

**Display of Results**  
 Suppress Results Rows: Never  
 Normalize to: No Normalization  
☐ Cumulated

**Display Hierarchy**  
☐ Active  
 Hierarchy Name:   
☐ Sort Within the Hierarchy According to: (As in Hierarchy)  

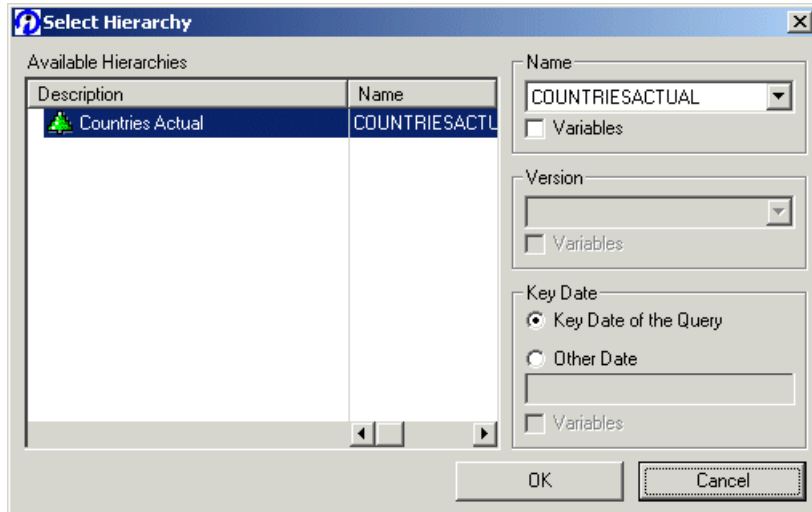
Hierarchy Properties	U	Value
Expand to Level	<input type="checkbox"/>	0
Position of Lower-Level Nodes	<input type="checkbox"/> Down	
Values of Posted Nodes	<input type="checkbox"/> Display	
Nodes with Only One Lower-Level Node	<input type="checkbox"/> Display	

**Sort Order**  
☐ Sorting within the characteristic according to: (As in the Query)

OK Cancel

8. Next to the **Hierarchy Name** field, click the **Values** button.  
The "Select Hierarchy" dialog box appears.






9. In the Available Hierarchies list, select **Countries Actual** and click **OK**.

You are returned to the "Characteristic Properties" dialog box.


10. In the Hierarchy Properties list, click the **Expand to Level** check box and, in the **Value** field, type 5

The Expand to Level property allows you to set a maximum number of levels for a hierarchy in Crystal Reports. In this example, you set the number of levels of hierarchy to 5.

Hierarchy Properties		U	Value
Expand to Level	<input checked="" type="checkbox"/>	5	
Position of Lower-Level Nodes	<input type="checkbox"/>	Down	
Values of Posted Nodes	<input type="checkbox"/>	Display	
Nodes with Only One Lower-Level Node	<input type="checkbox"/>	Display	

11.  Click **Query Properties**.
12. Click the **Extended** tab and ensure that **Allow External Access to this Query** is selected.

This option allows other programs, like Crystal Reports, to access this query.

13.  Click **OK** and then save your query to the appropriate location.

For this example, in the **Description** field, type Hierarchy - Simple Example, and in the Technical Name field, type QRY\_HIERARCHY\_SIMPLE

14.  Click **Quit and Use Query**.

You now have a simple query with five possible levels of hierarchy. You can display the query in either the Business Explorer Analyzer or as a dataset in Crystal Reports. To format the data, create a formatted Crystal report

using groups and summaries, as described in [Creating a report based on a BW hierarchy](#) on page 43.

## Creating a report based on a BW hierarchy

Crystal Reports 2008 has enhanced hierarchical grouping capabilities. These enhancements allow you to easily create reports based on BW queries that contain hierarchies with variable levels. Crystal Reports is able to determine each group's parent and children, and at the same time update hierarchical summaries.

Crystal Reports gives you increased layout control so that groups are now nested in a hierarchy, and you have more fine-grained control over hierarchy indenting. You can indent specific objects in a section according to their hierarchy level, instead of automatically indenting all objects in the hierarchy.

### Some general information about hierarchical reporting

When you group data hierarchically, Crystal Reports sorts information based on the relationship between two fields. A hierarchical relationship must be inherent in the data that you use for the report:

- Parent and child fields must be of the same data type for the program to recognize a relationship between them.
- The data in the parent field must be a subset of the data in the child field.
- For the top level of a hierarchy to appear in a report, the value must appear in the child data, and the corresponding row in the parent data must be empty.
- There cannot be any circular logic in the data (that is, A cannot be related to B, while B is related to C, and C is related back to A).

For example, if you want to show the hierarchical relationship of the people who work in a department, you could group data by employee name (the child field) and specify the hierarchy by using the field that lists whom the employee reports to (the parent field).

In addition to a visual representation of the hierarchy inherent in your data, a Crystal report that you have grouped hierarchically has several other characteristics:

- When you drill down on a group in the hierarchy, the drill-down view also shows the records that are lower in the hierarchy.
- The report contains hierarchical group footers that include the records that are lower in the hierarchy of each group. You can summarize data across the hierarchies.
- Use the conditional-X-position feature to ensure that the indenting you set to show hierarchical relationships does not affect other fields in same section of your report.
- Hierarchical levels are supported in the formula language through the use of the `GroupingLevel` and `HierarchyLevel` functions.

**Note:**

You cannot use hierarchical summaries in formulas.

## Creating the report

This section uses the query that you created in [To design a simple BW query with a hierarchy](#) on page 38.

This tutorial will use the following fields:

- *Hierarchy Name* Node ID

This field reflects the child values.

- *Hierarchy Name* Parent ID

This field reflects the parent values. Combined with the Actual Node ID, Crystal Reports uses these fields to determine the hierarchy based on the information received from the SAP BW query.

- *Hierarchy Name* Canonical ID

This field is the ID of the Node ID.

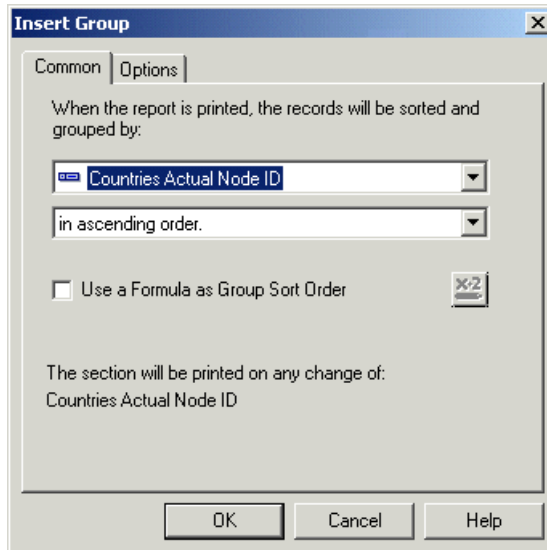
- *Hierarchy Name* Node Description

This field contains the description of the Actual Node ID, or child values.

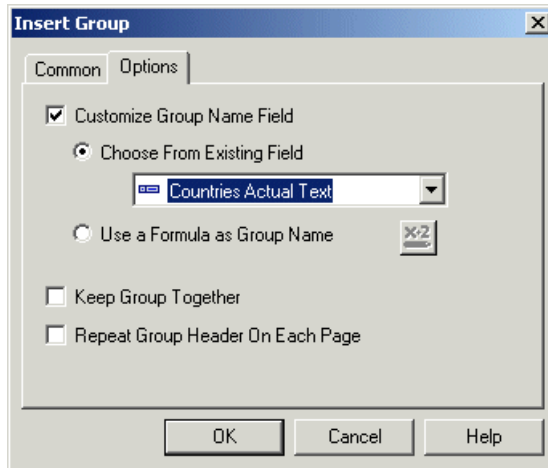
### To design a report based on a BW query with a hierarchy

1. On the **Insert** menu click **Group**.

2. In the "Insert Group" dialog box, select the Actual Node ID to base the group on. This is the child field, that is, the field that you want to see the hierarchy of.



3. Select **in ascending order**.  
By default, the group header of the report displays the value of the field you are grouping on.
4. Click the **Options** tab and select the **Customize Group Name Field** check box.
5. Click **Choose From Existing Field** and select *Hierarchy Name Actual Node Description*.

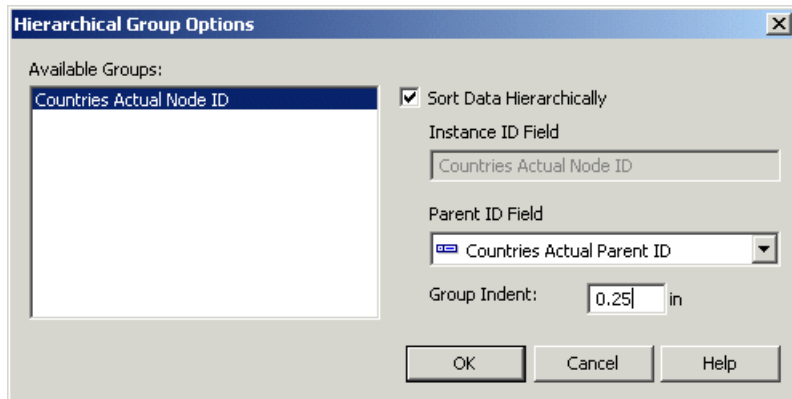


6. Click **OK**.
7. On the **Report** menu click **Hierarchical Grouping Options**.
8. Select **Sort Data Hierarchically** and on the **Parent ID Field** list click *Hierarchy Name* Actual Parent ID.

### Note:

The Instance ID Field and Parent ID Field must be of the same data type. For example, if the Instance ID Field holds string data, then the Parent ID Field must also hold string data.

9. In the **Group Indent** field, enter the amount that you want each hierarchy level indented.



### Note:

The value that you enter in the Group Indent field affects all other objects that are in the same area as your hierarchical group. For example, if your report contains a salary field on the same line as the name of the employee, the salary field is also indented when you use the employee field to create a hierarchical group. To indent only the hierarchy records and not the other objects, leave this value as 0 (zero) and use the conditional-X-position feature.

### 10. Click **OK**.

The report data is now displayed hierarchically.

In this example the Details section has been hidden to more clearly display the hierarchical relationship of the data.

RH		
PH		
GH1	1	<u>Countries Actual Node ID</u>
GF1		Not Assigned Sold-to party
GH1		Countries Actual
GH1		Americas
GH1		US
GH1		Booktree Inc.
GF1	2	
GF1		
GF1		
GH1		Europe
GH1		England
GH1		Adecom SA
GF1	3	
GH1		COMPU Tech. AG
GF1		
GF1		
GH1		France
GH1		DelBont Industries
GF1	4	
GH1		Minerva Industries
GF1		
GF1		
GH1		Germany
GH1		Becker AG
GF1	5	
GH1		Becker Berlin
GF1		
GF1		
GF1		
GF1		

## Inserting a summary

At this point you may want to insert a summary onto your report. You can now do this in the regular Crystal Reports workflow.

### To insert a summary

1. On the **Insert** menu click **Summary**.
2. On the **Choose the field to summarize** list, click the field that you want to insert the summary on. In this example, it is the Billed Quantity field.
3. On the **Calculate this summary** list, click the summary operation you want.
4. On the **Summary Location** list, click a location for the summary. In this example, it is the hierarchical group.

#### Note:

You do not need to select Summarize across hierarchy in order to have the data summarized correctly in this example because Crystal Reports can recognize the Hierarchy node summaries that the SAP BW data delivers.

The report now displays the summary data across the hierarchy.

#### Note:

When you drill down on a hierarchical group, the drill down page always includes groups lower in the hierarchy.



RH	-			
PH	-			
GH1	-	<u>Countries Actual Node ID</u>		
GF1	-	Not Assigned Sold-to party		
GH1	-		14,970.00	
GH1	-	Countries Actual		
GH1	-	Americas		
GH1	-	US		
GH1	-	Booktree Inc.		
GF1	-		3,058.00	
GF1	-		3,058.00	
GF1	-		3,058.00	
GH1	-	Europe		
GH1	-	England		
GH1	-	Adecom SA		
GF1	-		1,328.00	
GH1	-	COMPU Tech. AG		
GF1	-		770.00	
GF1	-		2,098.00	
GH1	-	France		
GH1	-	DelBont Industries		
GF1	-		278.00	
GH1	-	Minerva Industries		
GF1	-		192.00	
GF1	-		470.00	
GH1	-	Germany		
GH1	-	Becker AG		
GF1	-		790.00	
GH1	-	Becker Berlin		
GF1	-		1,764.00	
GF1	-		2,554.00	
GF1	-		5,122.00	
GF1	-		8,180.00	
RF	-			

The summary can be placed in the Group Header or Group Footer.

The summary values are indented. To fix this see, [Indenting a hierarchy without affecting other fields](#) on page 49.

## Indenting a hierarchy without affecting other fields

This procedure demonstrates how to indent a hierarchy while keeping all other fields on a report in their original positions. It takes advantage of two new functions included in Crystal Reports XI, `HierarchyLevel` and `GroupingLevel`.

This procedure continues directly from the example outlined in [Creating the report](#) on page 44.

### To indent a hierarchy without affecting other fields

1. On the **Report** menu, click **Hierarchical Grouping Options**.
2. Ensure that the **Group Indent** field is set to zero.
3. Click **OK**.

The report now displays the data without any indent.

4. Right-click the group header field and click **Size and Position**.
5. Click the Conditional Formula button beside the X position value field.
6. In the Formula Workshop, enter the formula:

```
HierarchyLevel (GroupingLevel ({ [Hierarchy Name] Actual Node  
ID})) * 250
```

#### Note:

- The field {[Hierarchy Name] Actual Node ID} appears with its technical name into the formula.
- Positions are measured in twips; there are 1440 twips in an inch.

The report now displays the summary data in its original position and the group headers hierarchically.

PH			
		<u>Countries Actual Node ID</u>	
GH1		Not Assigned Sold-to party	
GF1			14,970.00
GH1		Countries Actual	
GH1		Americas	
GH1		US	
GH1		Booktree Inc.	
GF1			3,058.00
GF1			3,058.00
GF1			3,058.00
GH1		Europe	
GH1		England	
GH1		Adecom SA	
GF1			1,328.00
GH1		COMPU Tech. AG	
GF1			770.00
GF1			2,098.00
GH1		France	
GH1		DelBont Industries	
GF1			278.00
GH1		Minerva Industries	
GF1			192.00
GF1			470.00
GH1		Germany	
GH1		Becker AG	
GF1			790.00
GH1		Becker Berlin	
GF1			1,764.00
GF1			2,554.00
GF1			5,122.00
GF1			8,180.00
RF			

## New Functions

Two functions new in Crystal Reports XI make this functionality possible.

## Report-Report Interface overview

Report-Report Interface (RRI) is a feature that allows you to create a relationship between a query in BW (referred to as the sender query) and a Crystal report that is published to BusinessObjects Enterprise with a data source that is a different BW query (the receiver query).

This relationship allows users who view the sender query in BW to drill down on specific rows and go to an associated Crystal report that displays related data from the receiver query. The sender and receiver queries can be any pair of BW queries, provided that the receiver query contains at least one

variable that is based on a characteristic from the result set of the sender query.

In this scenario, the associated Crystal report is refreshed on demand against the receiver query. Information about the user's request is passed from the sender query, through the Crystal report (as a report parameter), and into the receiver query (as a variable). This allows the Crystal report to display data that is filtered based on the context of the user's original request.

### To set up Report-Report Interface in BW

1. Create a new query in BW.

This will be the sender query that users will view from within BW.

2. Create another new query in BW, and add at least one variable based on a characteristic that is included in the result set of the sender query.

#### Note:

For more information on how to use variables, see [Personalization in BusinessObjects Enterprise](#) on page 147.

This will be the receiver query. Users will view its data in a Crystal report.

#### Tip:

You can use the SAP BW Variables Wizard to add the required variable to your query.

3. Design a Crystal report that is based on the receiver query.

For details about creating a report based on a BW query, see [Selecting existing queries for reports](#) on page 23.

4. Save the report to BW and publish it to BusinessObjects Enterprise.


### To associate the sender query with the Crystal report

1. In BW, execute transaction `rsbbs`
2. Click the **Query** tab.
3. In the **Sender** field, type the technical name of the sender query, and then press **Enter**.

#### Tip:

Press F4 to browse to the query.

4. Click **Create**.
5. In the **Report Type** list, select **BW Crystal Report**.


6.  Click the **Look Up** button in the Report area.
7. Select the Crystal report that you want to associate with the sender query.

**Tip:**

Click the down arrow on the long bar under the Restrictions tab to filter the list of reports.

8. Click **Transfer** (or **F8** on your keyboard).

9.  Click **Save**.

10.  Click the **Back** button (or **F3** on your keyboard) repeatedly until you have backed out of the Maintain Sender/Receiver Assignment transaction (rsbbs).

**Note:**

Backing out of this transaction is recommended to avoid complications with BW placing a lock on the query.

Now, when users view the sender query in BW, they can click a row, select Goto or the JUMP key, and select the related Crystal report to view. (If RRI is not set up for a characteristic in that row, no report will appear as a Goto option.)

## Reporting off BW Queries and Cubes: MDX Query Driver

This section introduces you to Crystal Reports and the integrated features of the MDX Query Driver. It shows how to format and build reports off a BW query and hierarchy.

### MDX Query driver overview

The SAP BW MDX Query driver introduces new flexibility when reporting off BW cubes and queries. By writing reports with the MDX Query driver, you gain direct access to BW cubes. You can also now use display attributes, Multiple Structures, and Free Characteristics in your Crystal reports.

Additionally, the MDX Query driver provides the ability to create reports from queries that contain hierarchy variables and hierarchy node variables. The driver creates specific fields that allow you to specify pick lists for your variables in Crystal Reports.

## Formatted reporting overview

With the BW MDX Query driver, you can design formatted Crystal reports that are based on the data that is stored in SAP Business Information Warehouse (BW). You base these reports either on the queries that you create with SAP's Business Explorer (BEx) Query Designer or directly on cubes. Once you have designed your Crystal reports, you can save them to BusinessObjects Enterprise and, if required, target them for translation in order to make them available to users in multiple languages.

This section focuses on Crystal Reports and the task of using the MDX Query driver to select BW queries and cubes as data sources for Crystal reports. This section ends with a series of tutorials that guide you through the creation of reports off a sample query.

### Note:

It is recommended that you complete the tutorials in sequential order because each tutorial builds on the query and report you built in the previous tutorial.

BusinessObjects Enterprise allows you to share Crystal reports over the Web so that all users can view your formatted content. For details about publishing reports to BusinessObjects Enterprise, see [Publishing and Viewing Reports](#) on page 94.

Your BW queries may contain SAP variables that appear as parameters in BusinessObjects Enterprise and Crystal Reports. To accommodate these variables in BusinessObjects Enterprise, BusinessObjects Integration for SAP Solutions employs dynamic pick lists. A dynamic pick list provides you with a list of possible values to choose for a parameter (variable). Additionally, the values that are displayed in a dynamic pick list correspond to your user rights—you see only the pick list values that you have the rights to as defined by SAP BW security settings.

### Note:

In Crystal Reports, the pick lists are static but in BusinessObjects Enterprise they are dynamic.

## Accessing BW queries for reporting

This section outlines the ways in which you can create BW queries and select existing BW queries to use as data sources for your Crystal reports. To access the BEx Query Designer to create new queries, you can use the SAP Tools toolbar in Crystal Reports or you can use the Programs menu to navigate to it. To select existing queries, you can use the SAP Tools toolbar in Crystal Reports or you can use the Database Explorer in Crystal Reports.

### Creating new queries for Crystal Reports

There are two ways you can access the BEx Query Designer to create a new BW query for Crystal Reports. You can use the SAP Tools toolbar within Crystal Reports or you can use the Programs menu.

For details about the SAP Tools toolbar, see [Commands and connection settings in Crystal Reports](#) on page 14.

#### Note:

The tutorials in this section use the SAP Tools toolbar to create new queries for reporting. However, when creating new queries you can use whichever method you prefer.


#### To create a BW query from the Crystal Reports SAP Tools toolbar

1. On the **SAP** menu in Crystal Reports, click **Settings**.

#### Tip:



You can also click the corresponding button on the toolbar.

2. Select the **Use MDX driver with support for Multiple structures** check box and click **OK**.
3. On the **SAP** menu, click **Start BW Query Designer**.
4. When prompted, log on to your SAP BW system.
5.  On the Query Designer's toolbar, click **New Query**.


The "New Query: Select InfoProvider" dialog box displays the InfoAreas and InfoProviders that are available on your system.

6. Select the InfoProvider on which you want to base your query, then click **OK**.


7. Define your query by selecting measures and characteristics from the Key Figure and Dimensions lists and dragging them to the Columns area.

**Note:**

When you create a new query for reporting purposes, it is recommended that you place Key Figures in the Columns area of the Query Designer and Characteristics in the Rows area.

8.  Click **Query Properties**.
9. Click the **Extended** tab and ensure that **Allow External Access to this Query** is selected.

This option allows other programs, like Crystal Reports, to access this query.

10.  Click **Save Query**.
11. In the "Save Query" dialog box, click **Roles** and select the role where you want to save the query.
12. Type a **Description** and a **Technical Name** for the query, then click **Save**.

The Technical Name must uniquely identify the query—that is, the name can appear only once across all InfoProviders in BW. The Technical Name can be up to 30 characters long and must begin with a letter.

13.  Click **Quit and Use Query**.

Crystal Reports generates a report that uses your query as its data source. The report opens in the **Design** tab of Crystal Reports. You can now add fields, titles, charts, and other objects to the report.

### To create a BW query directly from the BEx Query Designer

1. Go to **Start > Programs > Business Explorer > Query Designer**.
2. When prompted, log on to your SAP BW system.
3. On the Query Designer's toolbar, click **New Query**.

The "New Query: Select InfoProvider" dialog box displays the InfoAreas and InfoProviders that are available on your system.


4. Select the InfoProvider on which you want to base your query, then click **OK**.




5. Define your query by selecting measures and characteristics from the Key Figure and Dimensions lists and dragging them to the Columns area.

**Note:**

When you create a new query for reporting purposes, it is recommended that you place Key Figures in the Columns area of the Query Designer and Characteristics in the Rows area.

6.  Click **Query Properties**.
7. Click the **Extended** tab and ensure that **Allow External Access to this Query** is selected.

This option allows other programs, like Crystal Reports, to access this query.

8.  Click **Save Query**.
9. In the "Save Query" dialog box, click **Roles** and select the role where you want to save the query.
10. Type a **Description** and a **Technical Name** for the query, then click **Save**.

The Technical Name must uniquely identify the query—that is, the name can appear only once across all InfoProviders in BW. The Technical Name can be up to 30 characters long and must begin with a letter.

## Selecting BW Queries for a report

There are two ways you can access BW queries to use as Crystal Reports data sources. You can use the SAP Tools toolbar in "Crystal Reports" or you can use the Database Explorer in Crystal Reports.

### Using the SAP Tools toolbar to select an MDX query

**Note:**

- The tutorials in this section use the SAP Tools toolbar to create new queries for reporting. However, when creating new queries you can use whichever method you prefer.
  - Before you can see a query in the Database Explorer, you will need to set the "Allow External access to the Query" option in its properties.
1. On the **SAP** menu click **Create New Report from a Query**.

**Tip:**



You can also click the corresponding button on the toolbar.

2. When prompted, log on to your SAP BW system.

The "Select Query for your Crystal Report" dialog box displays the queries that are available through your Favorites, Roles, and InfoAreas in BW.

3. Select the query with the data that you want to report off, then click **OK**.

Crystal Reports generates a report that uses your query as its data source. The report opens in the **Design** tab of "Crystal Reports". You can now add fields, titles, charts, and other objects to the report.

### Using the Database Expert to select an MDX query

1. On the **File** menu in Crystal Reports, click **New** and then click **Blank Report**.

The Database Expert appears.

2. Under Available Data Sources, expand **Create New Connection** and then expand **SAP BW MDX Query**.

The "SAP System Logon" dialog box appears.

3. Select the appropriate BW system and click **Next**.

The next "SAP logon" dialog box prompts you for user credentials

4. Type your usual SAP user credentials in the **Client**, **Username**, and **Password** fields, then click **Next**.

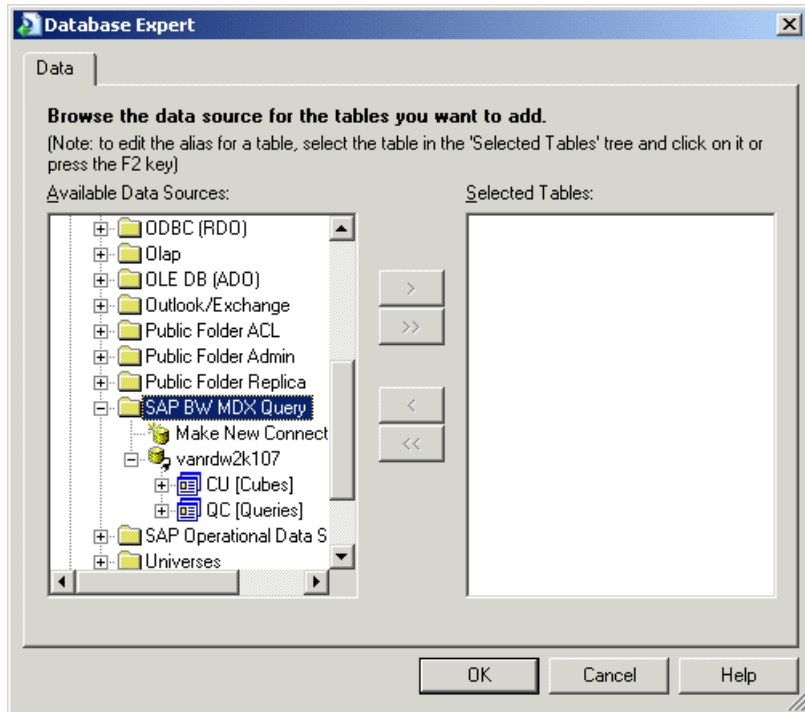
**Tip:**

To set the logon language, you must specify your Preferred Viewing Local in Crystal Reports under **View > Preferred Viewing Local**.

The next "SAP logon" dialog box asks you whether you would like to generate RFC trace files.

5. If you want the SAP Communication Library to create trace files for all client/server communication (for troubleshooting purposes), select the **Generate RFC trace files** check box.
6. Click **Finish**.

You are returned to the "Database Expert". There are now two entries under your SAP BW MDX Query connection. One is labeled CU (Cubes) and the other is labeled (QC) Queries.



7. Expand **QC (Queries)** to reveal the defined MDX BW Queries that are available as report data sources.

**Tip:**

If no items are found, check the Options setting of the Data Explorer.

- a. Right-click **QC (Queries)** and click **Options**.

This opens the "Options" dialog box with only the **Database** tab visible.

- b. In the Data Explorer area, ensure that **Stored Procedures** is selected and then click **OK**.

You are returned to the Database Expert.

- c. Select **SAP BW MDX Query** and press the F5 key to refresh the Database Expert.

- d. Expand **Current Connections** and then expand **QC (Queries)**.

Now there are lists of available queries.

8. Select the query that you want to report off and click the right-arrow button.
9. Click **OK**.

"Crystal Reports" generates a blank report that uses your query as its data source. You can now add objects to the report.

### Showing field descriptions and technical field names

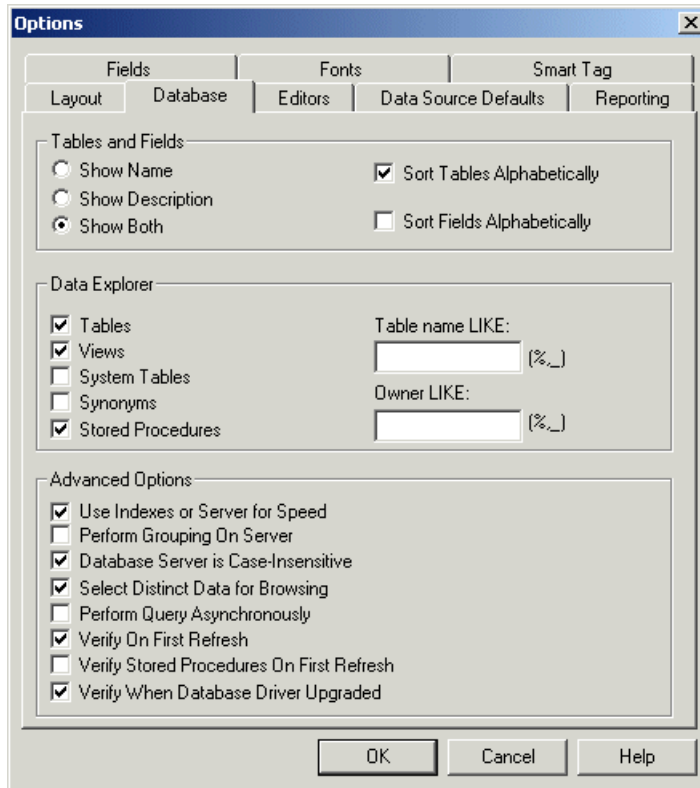
In Crystal Reports you can choose to identify fields by their description, name, or both. In some cases you may prefer to show both and in other cases you may prefer to use just one. For example, when working with Multiple Structures, it is recommended that you show only the descriptions. The Global Unique Identifiers (GUID) that SAP assigns to Multiple Structures are reflected in Crystal Reports and can be difficult to work with.

#### To adjust the field name and description settings

1. On the **File** menu, click **Options**.

The "Options" dialog box appears, with the Layout tab active.

2. Click the **Database** tab.



3. In the “Tables and Fields” area, select **Show Name**, **Show Description**, or **Show Both**.
4. Click **OK**.

**Tip:**

For more information about formatted reporting, see the SAP Library > Business Information Warehouse > Business Explorer > Formatted Reporting: Crystal Reports Integration.

### Tips regarding field names in Crystal Reports

When you are viewing attribute names in Crystal Reports, the number that the attribute name is preceded by gives you information about what kind of information the field contains.

- 2 means that the attribute is the member key

- 1, 4, or 5 means that the attribute is a member description

When you are viewing characteristic names in Crystal Reports, if the characteristic name is followed by many spaces and then the name of a hierarchy, the characteristic contains that hierarchy.

For example, if the Material characteristic contains the HIGHTECH hierarchy, it appears as `[0Material HIGHTECH]`.

The following fields are used for hierarchical grouping and parameter binding when there is a hierarchy node variable on a characteristic. These fields should not be used for reporting.

- Nodeld
- Parent Nodeld
- MemberUniqueName

### Note:

Putting the MemberUniqueName on a report will cause performance issues.

When a query contains a hierarchy variable, the MDX Query driver creates an extra field called Hierarchies in Crystal Reports that allows you to create a list of values for the variable.

The Hierarchies field has the following attributes:

- Hierarchy names
- Hierarchy descriptions

This field has no purpose other than to create lists of values for hierarchy variables. This field should not be used for reporting.

## Selecting cubes for reports

If you or someone else in your organization has defined one or more cubes for reporting purposes within BW, you can select them as data sources for new Crystal Reports.

### To select an existing query for a report

1. On the **File** menu in Crystal Reports, click **New** and then click **Blank Report**.

The "Database Expert" appears.

2. Under "Available Data Sources", expand **Create New Connection** and then expand **SAP BW MDX Query**.

The "SAP System Logon" dialog box appears.

3. Select the appropriate BW system and click **Next**.

The next "SAP logon" dialog box prompts you for user credentials

4. Type your usual SAP user credentials in the **Client**, **Username**, and **Password**, fields, then click **Next**.

**Tip:**

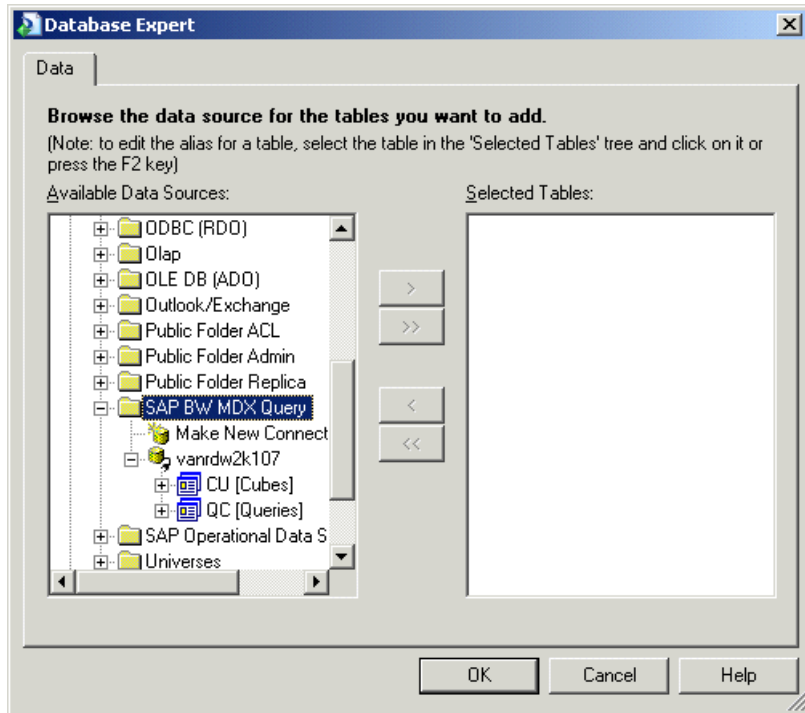
To set the logon language, you must specify your Preferred Viewing Local in Crystal Reports under **View > Preferred Viewing Local**.

The next "SAP logon" dialog box asks you whether you would like to generate RFC trace files.

5. If you want the SAP Communication Library to create trace files for all client/server communication (for troubleshooting purposes), select the **Generate RFC trace files** check box.

6. Click **Finish**.

You are returned to the "Database Expert". There are now two entries under your SAP BW MDX Query connection. One is labeled CU (Cubes) and the other is labeled (QC) Queries.



7. Expand **CU (Cubes)** to see the defined BW cubes that are available.

**Tip:**

If no items are found, check the Options setting of the Data Explorer.

- a. Right-click **CU (Cubes)** and click **Options**.
  - b. This opens the "Options" dialog box with only the Database tab visible.
  - c. In the "Data Explorer" area, ensure that **Stored Procedures** is selected and then click **OK**.
  - d. You are returned to the "Database Expert".
  - e. Select **SAP BW MDX Query** and press the F5 key to refresh the Database Expert.
  - f. Expand **Current Connections** and then expand **CU (Cubes)**.
  - g. Now there are lists of available cubes.
8. Select the cube that you want to report off and click the right-arrow button.
  9. Click **OK**.



"Crystal Reports" generates a blank report that uses your cube as its data source. You can now add objects to the report.

## Multiple Structures and Crystal Reports

The MDX Query driver recognizes Multiple Structures and allows you to use them in your Crystal reports. Multiple Structures show in Crystal Reports as a single dimension. That is, they show with no attributes. The structure consists only of the members that are in the query.

## Saving reports to BW

After designing a report based on a query, you can save the report to a role in BW. At the same time, you can prepare the report for translation or automatically publish it to BusinessObjects Enterprise.

Before saving your report, check its title by clicking Summary Info on the File menu. The title that appears here is displayed to users in BusinessObjects Enterprise. The technical name of the query is used as the default title, but you can replace it with a descriptive title that is more meaningful to SAP users.

### Tip:

Use the Save As command on the File menu to save the report to disk.

### To save a report to BW

1. On the **SAP** menu, click **Save Report**.

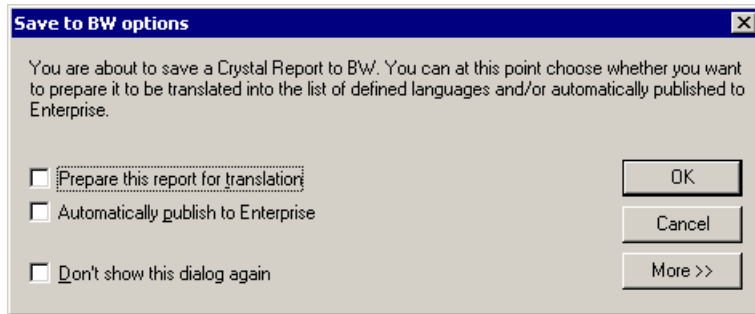
### Tip:



You can also click the corresponding button on the toolbar.

2. In the "Save a Crystal Report to BW" dialog box, select the role where you want to save the new Crystal report.
3. If you are using NetWeaver 7.0.2 or higher, type a **Technical Name** for the report (maximum 25 characters long).
4. Type a **Description** for the report (optional if you entered a **Technical Name**).
5. Click **Save**.

Depending on your connection settings, the "Save to BW options" dialog box may appear.



6. Select from the available options:

- **Prepare this report for translation**

When you select this option, the strings in your report are extracted and made available for translation within SAP. For details, see [Designing reports for translation](#) on page 90.

- **Automatically publish to Enterprise**

For more information on publishing reports, see [Publishing and Viewing Reports](#) on page 94.

If you are unable to publish your report to BusinessObjects Enterprise, you may need to consult with your administrator about licensing.

- **Don't show this dialog again**

If you select this option now, you can modify your default settings later by clicking Settings on the SAP menu. For details, see [Commands and connection settings in Crystal Reports](#) on page 14.

7. Click [OK](#).

## Building a report off a BW query

This tutorial provides an introduction to reporting off a query using the BW MDX Query driver. You will perform the following tasks in this tutorial:

- Create a query
- Select the query as a data source for a Crystal report

- Insert database fields
- Group and sort data
- Summarize data
- Preview the report

## How to create a simple BW query

This section guides you through the process of creating a query, saving it to BW, and releasing it for use by Crystal Reports.

### Note:

This tutorial uses the SAP Tools toolbar method to create a new BW Query.

### To create a simple BW query

1. On the **SAP** menu in Crystal Reports, click **Start BW Query Designer**.

#### Tip:



You can also click the corresponding button on the toolbar.

2. When prompted, log on to your SAP BW system.

The "Query Designer" opens.

3.  On the Query Designer's toolbar, click **New Query**.

The "New Query: Select InfoProvider" dialog box displays the InfoAreas and InfoProviders that are available on your system.

4. Select the **Customer-Cube** and click **OK**.


#### Note:

The location of the Customer-Cube varies, so you may need to browse to find it.

5. Select the following characteristics from the **Dimensions** list, and drag them to the "Rows" area:
  - Division
  - Material


#### Tip:

You can find Division under the Sales area data dimension.

6. Select the following measures from the Key Figure list, and drag them to the Columns area:
  - Invoiced quantity
  - Sales Volume
7.  Click **Query Properties**.
8. Click the **Extended** tab and ensure that **Allow External Access to this Query** is selected.

This option allows other programs, like Crystal Reports, to access this query.

### To save the query in BW

1.  Click **Save Query**.
2. In the "Save Query" dialog box, click **Roles** and select the role where you want to save the query.
3. Type a **Description**, such as `Sample Material query`
4. Type a **Technical Name**, such as `MATERIALSAMPLE01`.

The Technical Name must uniquely identify the query—that is, the name can appear only once across all InfoProviders in BW. The Technical Name can be up to 30 characters long and must begin with a letter.

5. Click **Save**.

### To use the query in Crystal Reports

-  Click **Quit and Use Query**.

Crystal Reports generates a report that uses your query as its data source. The report opens in the Design tab of Crystal Reports. You can now add fields, titles, charts, and other objects to the report.

### How to create a new report based on the query

This section guides you through the process of creating a new report based on the query you created in the last section, [How to create a simple BW query](#) on page 67.

## To create a new report


- On the **View** menu, click **Field Explorer**.

The "Field Explorer" appears. Expand Database Fields and Sample Material query to see the list of fields available for your report.

### Tip:

To change how the fields are identified in Crystal Reports, see [Showing field descriptions and technical field names](#) on page 60.

## To select fields for the report

1. Expand **Key Figures**.
2. Drag Invoiced Quantity from the "Field Explorer" and place it in the Details section of the report.
3.  Click the **Refresh** button to view the result set.

Only one value is returned on the report. This value represents the aggregated value of this Key Figure across all Characteristics in the cube. To display more detailed results, you must group the data on one or more dimensions.

## To group and sort data

### Note:

If you group on an attribute value, report performance is degraded. This occurs because the MDX query is slow to return data. To retain processing speed, it is recommended that you group on the Member Caption field whenever possible, as outlined in this tutorial.

1. On the **Insert** menu click **Group**.  
 The Insert Group dialog box appears.
2. On the first list click **Division**.  
 This field is the Member Caption for Division.
3. On the second list click **in ascending order**.
4. Click **OK**.
5. On the **Insert** menu click **Group**.
6. On the first list click **Material**.

This field is the Member Caption for Material.

7. On the second list click **in ascending order**.
8. Click the **Refresh** button to view the result set.

When you view this report, the records are first grouped according to the value in the Division characteristic. The resultant groups are then displayed in the report in ascending alphabetical order. The records in each Division group are further divided into secondary groups based on the values in the Material dimension. The resultant groups are also sorted in ascending alphabetical order.

**Tip:**

If you want to see the MDX query statement that the MDX Query driver is passing to the server, you can use the Show SQL Query option in CR. To see the query, on the **Database** menu, click **Show SQL Query**.

This MDX statement can be tested directly against SAP BW using the transaction mdxtest.

## How to summarize data and save report

This section guides you through the process of summarizing data and saving report based on the report you created in the last section, [How to create a new report based on the query](#) on page 68.

### To summarize the data

1. On the **Insert** menu click **Summary**.
2. The **Insert Summary** dialog box appears.
3. In the **Choose the field to summarize** list, select **Invoiced Quantity**.
4. In the **Calculate this summary** list, select **Sum**.
5. In the **Summary Location** list, select **Group #1:[Field Name]**.

In this case, [Field Name] indicates the name of the field that the group is based on.

6. Click the **Refresh** button to view the result set.

When you view the report, there is one summary in Group Footer 1. This summary represents a sum of the Invoiced Quantity for each Division.

## To save your report

- On the **SAP** menu, click **Save Report**, and save the report to a role in BW.

For details, see [Saving reports to BW](#) on page 65.

You may want to resize or reposition report elements, or further modify the report's design. You can, for instance, select and drag the borders of columns and column headings in order to resize them in the Preview tab; or, you can quickly display only a subset of the data by filtering the records with the Select Expert.

The *Crystal Reports Online Help* provides extensive conceptual and procedural information, tutorials, and examples to help you use Crystal Reports to its fullest potential. Access the help by pressing **F1** anywhere within Crystal Reports.

## Building a report off a BW hierarchy

This tutorial demonstrates how to report off a BW query with a hierarchy using the MDX Query driver in Crystal Reports.

In this tutorial you will:

- Create a simple BW query with a hierarchy.
- Create a new report and group the data hierarchically.
- Insert a summary.
- Adjust object positioning dynamically.

### Note:

This tutorial builds on the topics covered in [Building a report off a BW query](#) on page 66, which provides more details on steps such as logging on to your SAP system, selecting the Customer-Cube and saving queries.

## How to create a simple BW query with a hierarchy

This section uses the query that you created in the last tutorial, [Building a report off a BW query](#) on page 66. You will be adding a hierarchy to the query on the Material dimension.

For general information about hierarchical reporting, go to [Some general information about hierarchical reporting](#) on page 43.

### To create a BW query with a hierarchy

1. On the **SAP** menu in Crystal Reports, click **Start BW Query Designer**.
2. Click **Open Query**.
3. Select the query you created in the last tutorial.

If you have not created a query, see [Building a report off a BW query](#) on page 66.

4. In the Rows area, right-click **Material** and click **Properties**.

The "Characteristic Properties" dialog box appears.

5. Next to the Hierarchy Name field, click the **Values** button.

The "Select Hierarchy" dialog box appears.

6. In the **Available Hierarchies** list, select **Material class** and click **OK**.

You are returned to the Characteristic Properties dialog box.

7. Click **OK** and then click **Save Query As...** to save your query to the appropriate location.

For this example, in the Description field, type Simple Hierarchy for MDX, and in the Technical Name field, type HIERARCHY\_SIMPLE\_MDX

8. Click **Quit and Use Query**

You now have a simple query with three possible levels of hierarchy. You can display the query in either the Business Explorer Analyzer or as a dataset in Crystal Reports.



## How to build a report off the BW hierarchy

The report that you build in this section starts using the same field as the report you created in [Building a report off a BW query](#) on page 66. The difference comes with the grouping. Instead of grouping the data like you did in the previous tutorial, you will group the data hierarchically. This way, the hierarchy that exists in the data is reflected in the report.

This tutorial will use the following fields to create a hierarchical display:

- *Hierarchy Name* Node ID

This field reflects the child values.

- *Hierarchy Name* Parent ID

This field reflects the parent values. Combined with the Actual Node ID, Crystal Reports uses these fields to determine the hierarchy based on the information received from the SAP BW query.

- *Hierarchy Name*

This field contains the captions of the hierarchy members.

### To build a hierarchy report off the BW hierarchy

- On the **View** menu, click **Field Explorer**.

The "Field Explorer" appears. Expand Database Fields and Simple Hierarchy for MDX to see the list of fields available for your report.

#### Tip:

You may want to set the field name and description display options to make it easier to select fields. To change how the fields are identified in Crystal Reports, see [Showing field descriptions and technical field names](#) on page 60

### To select fields for the report

1. Expand **Key Figures**.
2. Drag Invoiced Quantity from the Field Explorer and place it in the Details section of the report.

### To group and sort data

1. On the **Insert** menu click **Group**.

The "Insert Group" dialog box appears.

2. On the first list click **Material Node ID**.
3. On the second list click **in ascending order**.
4. Click the **Options** tab and select the **Customize Group Name** field.
5. From the **Choose from Existing** field list, select **Material**.

Material is the hierarchy name, or Member Caption.

6. Click **OK**.
7. On the **Report** menu click **Hierarchical Grouping Options**.

The "Hierarchical Group Options" dialog box appears.

8. In the Available Groups area, ensure that Material Node ID is selected.

Material Node ID is probably the only group listed.

9. Select the **Sort Data Hierarchically** check box.
10. In the Parent ID field, select **Material Parent Node ID**.
11. In the Group Indent: field, type the amount that you would like each hierarchy level to be indented.

This entry depends on how you want your report to look and on what unit of measurement you are using in CR.

12. Click the **Refresh** button to view the result set.

The result set for the report is now displayed hierarchically.

## How to summarize data

This section guides you through the process of summarizing data based on the query you created in the last section, [How to build a report off the BW hierarchy](#) on page 73.

### To summarize the data

1. On the **Insert** menu click **Summary**.
2. The **Insert Summary** dialog box appears.
3. In the **Choose the field to summarize** list, select **Invoiced Quantity**.
4. In the **Calculate this summary** list, select **Sum**.
5. In the **Summary Location** list, select **Group #1:[Field Name]**.

In this case, [Field Name] indicates the name of the field that the group is based on.

**Note:**

You do not need to select Summarize across hierarchy in order to have the data summarized correctly in this example because Crystal Reports can recognize the Hierarchy node summaries that the SAP BW data delivers.

The report now displays the summary data across the hierarchy. However, the summary values are indented. To fix this see, [How to indent a hierarchy without affecting other fields](#) on page 75.

### An important note about group summaries

If you mix hierarchical grouping with regular grouping and insert a summary at the regular group level, the summary may be incorrect. That is, the existing summaries are summarized again along with the detail records.

In this case, you must use server-side calculated summaries at the regular group level.

## How to indent a hierarchy without affecting other fields

This procedure demonstrates how to indent a hierarchy while keeping all other fields on a report in their original positions.

### To indent a hierarchy without affecting other fields

1. On the **Report** menu, click **Hierarchical Grouping Options**.
2. Ensure that the **Group Indent** field is set to zero.
3. Click **OK**.

The report now displays the data without any indent.

4. Right-click the group header field and click **Size and Position**.
5. Click the **Conditional Formula** button beside the X position value field.
6. In the Formula Workshop, enter the formula:

```
HierarchyLevel (GroupingLevel ({ [Hierarchy Name] Node ID} ) )
* 250
```

**Note:**

- The field {[Hierarchy Name] Node ID} appears with its technical name in the formula.
- Positions are measured in twips; there are 1440 twips in an inch.
- The report now displays the summary data in its original position and the group headers hierarchically.

## Building a report off a BW query with a hierarchy node variable

When you design your reports, you might want to give users the opportunity to restrict the amount of data that is returned by Crystal Reports. To do this, you can create a hierarchy node variable in your query. This variable will be treated as a parameter in Crystal Reports.

You will perform the following tasks in this tutorial:

- Create a query with a hierarchy node variable
- Create a new report based on this query
- Bind the hierarchy node variable to the MemberUniqueName field in Crystal Reports.
- Preview the report


**Note:**

This tutorial builds on the query that you built in the previous tutorial. If you have not built a query yet, follow the steps outlined in [How to create a simple BW query](#) on page 67 and [How to create a simple BW query with a hierarchy](#) on page 72.

## How to create a query with a hierarchy node variable

This section guides you through the process of creating a hierarchy node variable on the Material hierarchy. When you report off this query using Crystal Reports, you will receive a prompt to select a hierarchy node to use for the report data.

## To create a hierarchy node variable

1. On the **SAP** menu in Crystal Reports, click **Start BW Query Designer**.
2.  Click **Open Query**.
3. Select the query you created in the last tutorial.

### Note:

If you have not created a query, complete the steps outlined in [How to create a simple BW query](#) on page 67 and [How to create a simple BW query with a hierarchy](#) on page 72.

4. In the Rows area, right-click **Material** and click **Restrict**.  
 The Selection for [Dimension] dialog box appears.
5. Ensure that the **Selection** field is set to Single Values, and that the **Hierarchy** field is set to the hierarchy you created for this dimension.

In this example, the hierarchy is Material class.

6. Click the **Variable** tab.
7. Right-click in the white space and click **New Variable**.

The SAP BW Variables Wizard opens.

8. Click **Next** on the Introduction window.
9. Ensure that the **Type of Variable** field says Hierarchy Node.
10. Fill in the rest of the fields as follows:
  - In the **Variable Name** field, type a name such as HNV01.
  - In the **Variable Description** field, type a name such as Test Hierarchy Node Variable.
  - In the **Processing by** field, ensure it says User Entry/Default Value.
  - In the **Characteristic** field, ensure it says Material.
11. Click **Next**.  
 The Details window appears.
12. On the **Variable Entry is** list click **Optional** or **Mandatory**.
13. Click **Next** to accept the other default values.

The "Default Values" window appears.

14. Click **Next** to accept the default values.


15. Click **Finish** to create the variable.

You are returned to the "Selection for Dimension" dialog box. Your new hierarchy node variable is visible in the variable list.

### To add the hierarchy node variable to the query

1. Click the hierarchy node variable named Test Hierarchy Node Variable.
2. Click the right arrow to move the variable to the Selection area of the dialog box.
3. Click **OK**.

You are returned to the BEx Query Designer. Your new hierarchy node variable is visible in the Rows area under the Material characteristic. You are now ready to save the query.

4.  Click **Save Query As....** to save the query with a new name and description.

For this example, in the Description field, type Simple Hierarchy w Node Variable, and in the Technical Name field, type HIERARCHY\_SIMPLE\_HNV.

5.  Click **Quit and Use Query**.

You now have a query with a simple hierarchy on Material class and a hierarchy node variable based on the Material class hierarchy.

### How to build a report off the query with a hierarchy node variable

This section guides you through the process of creating a list of default values for the parameter that appears in "Crystal Reports".

1. Create the same report that you created in [How to build a report off the BW hierarchy](#) on page 73 except you are now using the query called Simple Hierarchy w Node Variable as the data source.
2. Click the **Refresh** button when you have finished designing the report.

#### Note:

If your variable is optional, ensure that the **Set to Null** check box is clear when you select a parameter value. The **Set to Null** check box overrides

any value you select from the list of default parameter values. This check box is not always cleared when you select a parameter value.

## Building a report off a BW query with a hierarchy variable and a hierarchy node variable

This tutorial guides you through adding a hierarchy variable to the query you created in the last tutorial. The hierarchy variable allows users to choose what hierarchy they would like Crystal Reports to use when displaying the data. The hierarchy node variable allows users to then choose what node of the hierarchy they would like to see data from.

You will perform the following tasks in this tutorial:

- Add a hierarchy variable to your existing query
- Create a new report based on this query
- Bind the hierarchy node variable to the MemberUniqueName field.
- Bind the hierarchy variable to the Hierarchies field.
- Preview the report

### Note:

This tutorial builds on the query that you built in the previous tutorial. If you have not built a query yet, follow the steps outlined in [How to create a simple BW query](#) on page 67, [How to create a simple BW query with a hierarchy](#) on page 72, and [How to create a query with a hierarchy node variable](#) on page 76.

## How to create a hierarchy variable

This section guides you through the process of creating a hierarchy variable on your existing query and resetting the hierarchy node variable to correspond to this new hierarchy variable.

In the last tutorial, you set the hierarchy node variable to the Material class hierarchy because that is the hierarchy that existed in the query. In this tutorial you will change the hierarchy of the query from the Material class hierarchy to a variable. To avoid unexpected behavior, the hierarchy node variable

should be based on the same hierarchy as the query. Therefore, you will change the hierarchy node variable to reflect the new hierarchy variable in this query.

### To create a hierarchy variable

1. On the **SAP** menu in Crystal Reports, click **Start BW Query Designer**.

2.  Click **Open Query**.


3. Select the query you created in the last tutorial.

This query already contains a hierarchy and a hierarchy node variable.

If you have not created a query, complete the steps outlined in [How to create a simple BW query](#) on page 67, [How to create a simple BW query with a hierarchy](#) on page 72 and [How to create a query with a hierarchy node variable](#) on page 76.

4. In the Rows area, right-click **Material** and click **Properties**.

The "Properties for Characteristic Material" dialog box appears.

5.  Click the **Values** icon.

The "Select Hierarchy" dialog box appears.

6. Select the **Variables** check box and then click the **New Variable** icon.

The SAP BW Variables Wizard appears.

7. Ensure that the **Type of Variable** field says Hierarchy.

8. Fill in the rest of the fields as follows:

- In the **Variable Name** field, type a name such as HV01.
- In the **Variable Description** field, type a name such as Test Hierarchy Variable.
- In the **Processing by** field, ensure it says User Entry/Default Value.
- In the **Characteristic** field, ensure it says Material.

9. Click **Next**.

The "Details" window appears.

10. On the **Variable Entry is** list click **Optional** or **Mandatory**.

The "Default Values" window appears.



11. Click **Next** to accept the default values.
12. Click **Finish** to create the variable.

You are returned to the "Select Hierarchy" dialog box. Your new hierarchy variable is visible in the Name list.

13. Select the new hierarchy variable from the Name list and click **OK**.

In this example, the hierarchy variable name appears as HV01(Test Hierarchy Variable).

Now that the display hierarchy is set to a variable rather than the Material class hierarchy, you must reset the hierarchy node variable to the new hierarchy variable.

### To set the hierarchy node variable

1. In the Rows area, click the Test Hierarchy Node Variable, and click **Restrict**.

The "Selection for Material" dialog box appears.

2. In the Selection area, right-click the `Test Hierarchy Node Variable`.
3. Click **Select Variable Hierarchy**.
4. Select the **Variables** check box to expose the variable hierarchies in the Name list.
5. From the Name list, click the hierarchy variable you created earlier in this tutorial.

In this example, the hierarchy name is HV01 (Test Hierarchy Variable).

6. Click **Save Query As...** and name the query `Simple Hierarchy Variable w HN Variable`.
7. Click **Quit and Use Query**.

You now have a simple query that will prompt the user to choose a particular hierarchy to use when displaying results and to restrict that result set to a particular hierarchy node.

## How to build a report off the query with a hierarchy variable and hierarchy node variable

This section guides you through the process of creating a basic report off your new query and creating a list of default values for the hierarchy variable in Crystal Reports.

### To build a report off the query with a hierarchy variable and hierarchy node variable

1. Create the same report that you created in [Building a report off a BW query](#) on page 66 except you are now using the query called `Simple Hierarchy Variable w HN Variable` as the data source.

2. Click the **Refresh** button.

You are presented with two parameter prompts. One prompts you to enter a value for the hierarchy node variable and one prompts you to enter a value for the hierarchy variable.

3. Select a value for the Hierarchy parameter.

**Note:**

Be sure to select this value first because the Hierarchy Node values are set based on the selection you make here.

4. Select a value for the Hierarchy Node parameter.

## Crystal Reports and variable pick lists

If you do not use the SAP Tools toolbar to access the BEx Query Designer, Crystal Reports does not create default pick lists for any BW variables that are in your query.

**Note:**

You do not need to do this when viewing the report in BusinessObjects Enterprise. In BusinessObjects Enterprise the pick list is dynamic.

## Creating default values list for hierarchy node variable

### To create a list of default values for the hierarchy node variable

**Note:**

This procedure is based on the query you created in [Building a report off a BW query with a hierarchy variable and a hierarchy node variable](#) on page 79.

1. In the Field Explorer, expand **Parameter Fields**, and right-click **[HNV01]**.

The parameter, [HNV01], in CR represents the hierarchy node variable you created for this query.

2. Click **Edit**.

The Edit Parameter dialog box opens.

3. Ensure that the **List of Values** field is set to **Static**.

4. From the **Value Field** list, select **Material MemberUniqueName**.

The attributes appear with their technical names in this list.

5. From the **Description Field** list, select **Material**, which is the Member Caption field.

The attributes appear with their technical names in this list.

6. Click **Actions** and then click **Append all database values**.

The Value and Description lists are populated with values from the BW system.

7. In the Values Options area, set the Prompt Text to the message you would like users to receive in the parameter prompt.

8. Ensure that Allow custom values and Allow multiple values are set to False.

9. Click **OK**.

Now when you refresh the report and are presented with the parameter prompt, you are able to pick the parameter value from a list of values.

**Note:**

If your variable is optional, ensure that the Set to Null check box is clear when you select a parameter value. The Set to Null check box overrides any value you select from the list of default parameter values. This check box is not always cleared when you select a parameter value.

## Creating list of default values for hierarchy variable

### Note:

This procedure is based on the query you created in [Building a report off a BW query with a hierarchy variable and a hierarchy node variable](#) on page 79.

When a query contains a hierarchy variable, the MDX Query driver creates an extra field called Hierarchies in Crystal Reports that allows you to create a list of values for the variable.

1. In the Field Explorer, expand **Parameter Fields**, and right-click **[HV01]**.

The parameter, [HV01], in CR represents the hierarchy variable you created for this query.

2. Click **Edit**.

The Edit Parameter dialog box opens.

3. Ensure that the **List of Values** field is set to **Static**.
4. From the Value list, select **Hierarchies-[0MATERIAL]**.
5. From the Description list, select **Hierarchies-[0MATERIAL]-Description**.
6. Click **Actions** and then click **Append all database values**.

The Value and Description lists are populated with values from the BW system.

7. In the Values Options area, set the Prompt Text to the message you would like users to receive in the parameter prompt.
8. Ensure that Allow customer values and Allow multiple values set to False.
9. Click **OK**.

### To create a list of default values for the hierarchy node variable

- Repeat the steps outlined in the section [To create a list of default values for the hierarchy node variable](#) on page 83.

Now when you refresh the report, you are presented with two prompts, each with a list of default values.

# Reporting off Operational Data Stores

This section introduces the Operational Data Store Driver and how to report off it.

## Operational Data Store driver overview

The Operational Data Store (ODS) driver allows you to use existing ODS objects from your BW Data Warehouse as a data source in Crystal Reports. ODS objects are collections of info objects that act as smaller organizational units for the data in your BW Data Warehouse. Through this driver, the user can design multiple reports by quickly referencing the same ODS objects.

Reports that are based on ODS objects behave similarly to reports that are based on stored procedures in mainstream databases. Because the data stored in the ODS objects is not multidimensional and contains no parameters, it can be accessed and reported from quickly. This driver also pushes record selection to the server, which further speeds up report processing.

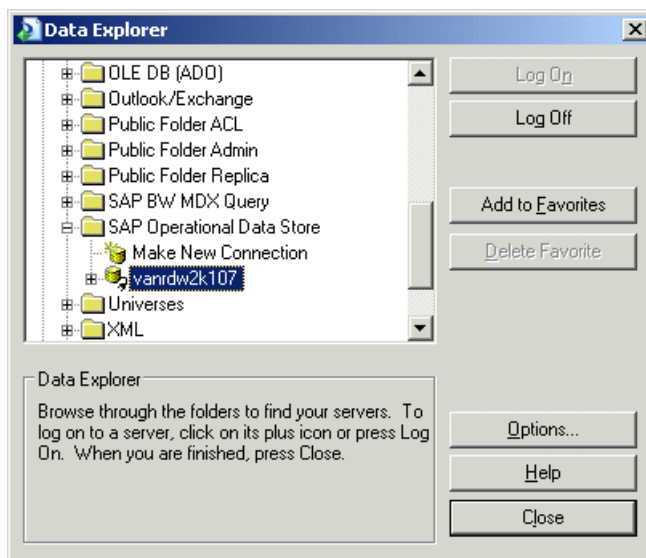
## Selecting an ODS object

1. Start Crystal Reports.
2. Connect to SAP as outlined in [Using the Log On Server command](#) on page 100.

**Tip:**

You can also use the method outlined in [Using the Report Wizards](#) on page 99.

Whichever method you choose to use, select SAP Operational Data Store in the Data Explorer.



3. Click **Options**.

The "Options" dialog box appears with only the Database tab displayed.

**Options**

Database

**Tables and Fields**

☒ Show Name ☒ Sort Tables Alphabetically

☐ Show Description ☐ Sort Fields Alphabetically

☐ Show Both

**Data Explorer**

☒ Tables  Table name LIKE: (%)

☒ Views  Owner LIKE: (%)

☐ System Tables

☐ Synonyms

☒ Stored Procedures

**Advanced Options**

☒ Use Indexes or Server for Speed

☐ Perform Grouping On Server

☒ Database Server is Case-Insensitive

☒ Select Distinct Data for Browsing

☐ Perform Query Asynchronously

☒ Verify On First Refresh

☐ Verify Stored Procedures On First Refresh

☒ Verify When Database Driver Upgraded

☒ Automatic Smart Linking

OK Cancel Help

**Tip:**

If you chose to use the report wizard, right-click SAP Operational Data Stores and click Options.

4. In the Data Explorer area, select the options for the types of data that you want to make available for the report.

In this case, ensure that Stored Procedures is selected.

5. Use the **Table name LIKE** and **Owner LIKE** fields to select only a subset of the available data types.

Use any of the following techniques when filtering:

- Type full or partial names of tables.

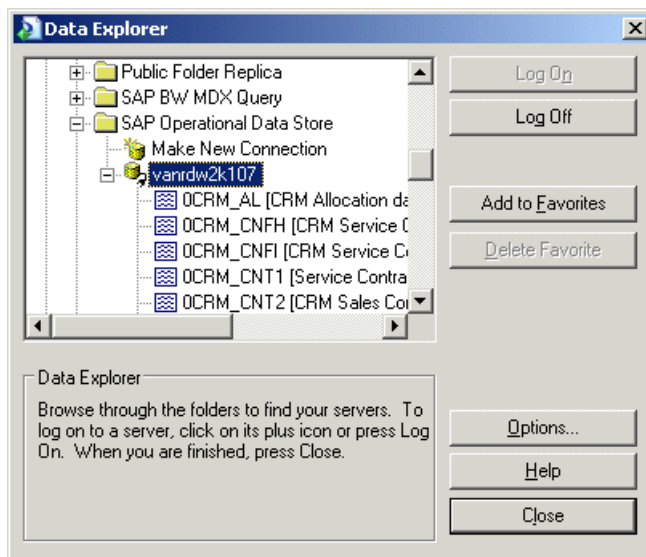
- Add wildcards to select multiple ODS objects.
  - Use a percentage sign (%) to denote any number of characters.
  - Use an underscore (\_) to denote a single character.

**Note:**

The % and \_ wildcards correspond respectively to the asterisk (\*) and question mark (?) wildcards used in Windows. The % and \_ wildcards correspond respectively to the asterisk (\*) and plus sign (+) wildcards in SAP.

- Clear both fields and click OK to proceed without filtering.
6. In the Tables and Fields area, click **Show Both**.
  7. When you have specified the ODS that you want to see, click **OK**.

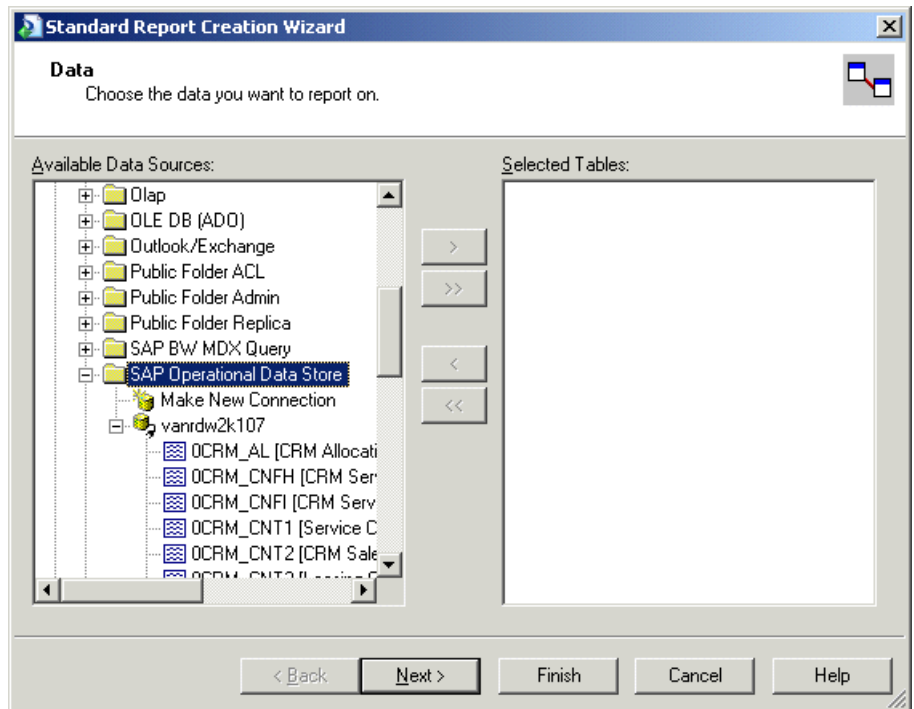
The Data Explorer displays the selected tables with descriptions defined by the SAP administrator.



8. Click **Close**.
9. Open a new, blank report.

The Database Expert opens.





10. In the Available Data Sources area, double-click the ODS that you want to use as a data source for the report.

The ODS moves to the Selected Tables area.

11. Click **OK**.

You can now add fields and continue to design your report.

## Designing and Translating Reports

This section introduces you to the Report Wizards in Crystal Reports and provides reporting details that are relevant within SAP environments. The section also shows how to prepare reports for translation and how to conditionally modify the report's design based on the user's logon language.

## Designing reports for translation

If your BW system supports users in multiple languages, you may want to prepare your reports for translation within BW. Once you publish the translated reports to BusinessObjects Enterprise, SAP users can view the reports that correspond to their preferred viewing locale. This section describes how to prepare your report for translation, and how to use a report's `LanguageCode%` formula to make language-specific modifications to reports.

### Preparing reports for translation

When you save a report to BW, you are given the option to prepare the report for translation. When you select this option, Crystal Reports extracts the string contents from all the text objects that you have inserted into the report (and replaces the original strings with placeholders in the .rpt file). Use the standard translation transactions within BW to access the language table that contains the original strings and translate them into all the languages that you require. Once you finish this process, the translated strings will be inserted back into the report when you publish to BusinessObjects Enterprise.

When you publish a translated report from BW to BusinessObjects Enterprise, the BW Report Publisher receives a single .rpt file, the original report strings, a list of destination languages, and all the translated report strings from BW. For each language, the BW Report Publisher copies the original .rpt file and replaces the placeholders with the corresponding report strings. The BW Report Publisher also marks each report with a language code (for details, see [Conditional formatting with the LanguageCode% formula](#) on page 91). The BW Report Publisher then generates multiple .rpt files (one for each language), and publishes each of these objects to BusinessObjects Enterprise.

#### Note:

When users refresh or schedule translated reports, new data is retrieved from BW, but the report strings are not refreshed. To update the translations of your report strings, publish the reports again from BW to BusinessObjects Enterprise.

### To prepare all reports for translation

1. Start Crystal Reports and open a report that you want to prepare for translation.

2. On the **SAP** menu, click **Settings**.

**Tip:**



You can also click the corresponding button on the toolbar.

The Settings dialog box appears.

3. Ensure that **Prepare reports for translation on server** is selected.

This option changes your default settings, so reports are always prepared for translation on the server.

If you prefer to select this option for individual reports, clear the **Suppress save dialog** option. You are then prompted with a **Prepare this report for translation** option every time you save a report to BW.

4. Click **OK**.
5. On the **SAP** menu, click **Save Report** and proceed as usual.

For details, see [Saving reports to BW](#) on page 25.

## Conditional formatting with the LanguageCode% formula

When you first create a report that is based on a BW query, Crystal Reports automatically generates a formula named LanguageCode%. You can find this formula in the Field Explorer. Rather than performing a calculation, this formula consists of a string value—a single letter—that serves to associate the report with a particular logon language.

When you first create and design your report, the formula contains the code for your current language ("E" for English, "D" for German, and so on). This is the report's master language. When you prepare reports for translation and subsequently publish them to BusinessObjects Enterprise, the BW Report Publisher modifies the LanguageCode% formula in each translated report. When an SAP user views a Crystal report or logs on to BusinessObjects InfoView, the application filters the available reports: if a translated report is available in the user's logon language, it is displayed for the user; if a translated report is unavailable, the report appears in its master language.

The LanguageCode% formula is intended to provide you, the report designer, with a tool for conditionally formatting reports in ways that accommodate users with different logon languages. For example, suppose that your report displays values from a Product Category field, the string values of which are

translated within SAP. As a report designer, the difficulty lies in that the German strings are longer than the equivalent English strings.

To solve this problem, you could use the `LanguageCode%` to conditionally reduce the font size for the translated version of the report (as demonstrated in the following steps). Alternatively, you could insert two Details sections—each containing the same database fields—and increase the width of the fields in the second section to accommodate for the longer German strings. Your conditional formatting formula would then show/hide the Details sections according to the report's language.

**Note:**

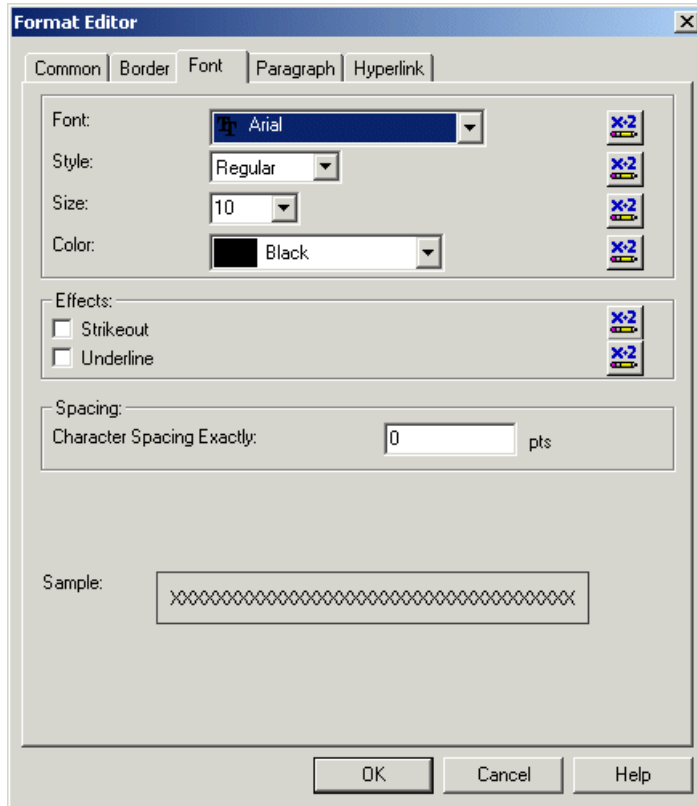
For complete instructions on conditionally formatting fields and using the Format Editor, see the *Crystal Reports Online Help*.


#### To reduce a field's font size conditionally

1. Open the report in Crystal Reports, and click the **Design** tab.
2. In the Details section of the report, right-click the field that contains the font that you want to conditionally format.
3. On the shortcut menu, click **Format Field**.

The "Format Editor" dialog box appears.

4. Click the **Font** tab.



5.  Click the **Formula** button that corresponds to the Size field.

The Formula Workshop opens a new formula and names it Font Size.

6. In the text area, type this formula (which uses Crystal Syntax):

```
if {@LanguageCode%} = "D"
then 10
else 12
```

This formula ensures that the font size of the field that is currently selected is reduced from 12 pts to 10 pts when the user's logon language is German (or when the user schedules a German instance of the report).

7.  Click **Save and close**.

## Publishing and Viewing Reports

This section describes how to publish Crystal reports to BusinessObjects Enterprise. It also describes how to view those reports from BusinessObjects InfoView and through other applications.

### Publishing predefined reports to BusinessObjects Enterprise

There are several ways to publish valid Crystal report files to BusinessObjects Enterprise:

- When you open a report in Crystal Reports, you can simultaneously save it to BW and publish it to BusinessObjects Enterprise. For details, see [Saving reports to BW](#) on page 25.
- If you have already added a number of reports to BW, you can publish them in batches to BusinessObjects Enterprise. For details, see [Batch publishing reports from BW](#) on page 94.
- You can add Crystal reports to the system through the Publishing Wizard, the Central Management Console, or the Import Wizard. For details, see the *BusinessObjects Enterprise Administrator's Guide* (`admin.pdf` in the `docs` directory of your BusinessObjects Enterprise product distribution).

### Batch publishing reports from BW

You can publish large numbers of Crystal reports using the Content Administration Workbench. For more information on the Content Administration Workbench, see the “Configuring publishing in the Content Administration Workbench” section of the *BusinessObjects XI Integration for SAP Solution Guide*.

### Migrating development content to a production BW system

If you have deployed BusinessObjects Integration for SAP Solutions in your development BW environment, you can import your reporting content to the

BusinessObjects Enterprise system that is configured for use with your production BW environment.

Before importing your content, consider the following:

- BW treats Crystal reports (.rpt files) as native objects. If the Crystal reports are stored in the repository of your development BW system, you can transport the BW content and then batch publish the reports to BusinessObjects Enterprise. (For details about transporting content between BW systems, see the SAP Library.) This procedure ensures that the database information for each report is updated by the BW Report Publisher.
- If you have deleted some or all of the Crystal reports from the repository of your development BW system, you can use the Import Wizard to import the report objects from one BusinessObjects Enterprise installation to another. When using the Import Wizard, be sure to set the correct database information on each of the report files that you import.
- If the number of report files that you want to migrate is small, you may find it easier to change each report's database information in the Central Management Console (CMC). (Go to the Objects management area, locate the report, and click the Database link on its Process tab.)

**Tip:**

For details about the Import Wizard and the CMC, see the *BusinessObjects XI Integration for SAP Installation Guide*.

After you migrate your content, use the Content Administration Workbench to perform maintenance on your reports. Report maintenance tasks include synchronizing information about reports between BusinessObjects Enterprise and BW (Update status), deleting unwanted reports (Delete reports), and updating reports migrated from previous versions of BusinessObjects Enterprise (Post-migration).

For more information about maintaining you reports, see “Maintaining reports” in the *BusinessObjects Integration for SAP Solutions Installation and Administration Guide*.

## Publishing reports from SAP BW to BusinessObjects Enterprise

There are two standard ways to publish reports to BusinessObjects Enterprise. This section discusses the integrated BW publishing options that allow you to:

- Immediately publish the reports that you create off BW queries to BBusinessObjects Enterprise.
- Publish the reports in batches from BW to BusinessObjects Enterprise.

If you have Crystal Reports installed on your machine, you can design your own report based on a BW query. You can then simultaneously save the report to BW and publish it to BusinessObjects Enterprise from within Crystal Reports. To enable this feature, in Crystal Reports, click **Settings** on the **SAP** menu, and ensure that **Automatically publish to BusinessObjects Enterprise** is selected.

(The SAP menu appears only after you install the BusinessObjects Integration for SAP Solutions. For details, see the *BusinessObjects XI Integration for SAP Installation Guide*.)

### Note:

You can publish large numbers of Crystal reports using the Content Administration Workbench. For more information on the Content Administration Workbench, see the “Configuring publishing in the Content Administration Workbench” section of the *BusinessObjects Integration for SAP Solutions Installation and Administration Guide*.

## Viewing reports

You can view Crystal reports through a number of applications, depending upon how BusinessObjects Enterprise has been integrated with your BW system. For example, you can log on to BusinessObjects InfoView with your usual SAP credentials, or you can launch reports in a web browser from within the "SAP Easy Access" interface.



## To view a published report through BusinessObjects InfoView

1. Open your web browser.
2. Enter the appropriate URL for InfoView:
  - For Java InfoView go to `http://webserver:portnumber/InfoViewApp/`
  - For .NET InfoView go to `http://webserver/InfoViewApp/`

Replace *webserver* with the name of the web server and *portnumber* with the port number that is set up for BusinessObjects Enterprise. You may need to ask your administrator for the name of the web server, the port number, or the exact URL to enter.

### Tip:

If you have the BusinessObjects XI Integration for SAP installed, you can also go to **Start > Programs > BusinessObjects XI 3.1 > BusinessObjects Enterprise** and select one of the following:

- BusinessObjects Enterprise .NET InfoView
- BusinessObjects Enterprise Java InfoView

The "Log On to InfoView" page is displayed.

3. In the **Authentication** list, ensure that **SAP** is selected.

Log On to InfoView

Welcome to BusinessObjects

Enter your user information and click Log On.  
(if you are unsure of your account information, contact your system administrator)

SAP System:

SAP Client:

User name:

Password:

Authentication: SAP

4. In the **SAP SystemID** field, type the three-character System ID (SID) for your SAP system. (Contact your administrator if you are unsure of the appropriate SID.)
5. In the **SAP Client** field, type your usual three-digit SAP client number.
6. In the **User Name** and **Password** fields, type your usual SAP logon credentials.
7. Click **Log On**.

You are now logged on to BusinessObjects InfoView.

8. Click the **My Groups** folder for quick access to all of the objects that have been saved in your various SAP roles and published to BusinessObjects Enterprise.

**Tip:**

For more information about using BusinessObjectsInfoView click **Help** to access the *BusinessObjects InfoView User's Guide*.

## To view a published report through SAP Easy Access

1. Log on to SAP Easy Access.
2. Browse your roles to locate the reports that have been saved to BW.

**Tip:**



Look for the Crystal report icon that is used in BW.

3. Double-click the report.

The report is displayed in your web browser. You may be prompted to log on to the SAP Web Application Server and/or BusinessObjects Enterprise. If so, provide your usual credentials.

## Reporting off SAP Data

This section shows how to connect to SAP from Crystal Reports, how to create reports off your SAP data with the Data Access Components, and how to access your reports through BusinessObjectsInfoView. The section also discusses the sample reports and the configuration options in Crystal Reports.

## Overview of the Data Access Components drivers

The Data Access Components consist of the BW Query driver, InfoSet Query driver, OLAP ABAPI driver, Open SQL driver, ODS driver, and the BW MDX Query driver. These drivers enable you to design reports off your SAP tables, ABAP functions, ABAP data clusters, ABAP queries, InfoSets, Operational Data Stores, and cubes. With the BW Query driver, you can design formatted

Crystal reports that are based on the data that is stored in SAP Business Information Warehouse (BW).

The Crystal reports that you create can be refreshed against the current SAP data by any user who has installed the drivers along with Crystal Reports. You can also schedule these reports within BusinessObjects Enterprise, so your users can access customized reports with current data on a regular basis.

This section includes the following topics:

- [Connecting to SAP with Crystal Reports](#) on page 99
- [Crystal Reports configuration options](#) on page 102
- [BusinessObjects InfoView](#) on page 104

**Note:**

The procedures in this section are based on Crystal Reports XI.

## Connecting to SAP with Crystal Reports

Once you have installed the Data Access Components (as described in the *BusinessObjects XI Integration for SAP Installation Guide*), you can start Crystal Reports and select a driver for your report. There are two ways to select a driver in Crystal Reports: use one of the Report Wizards, or select **Log On Server** from the **File** or **Database** menu. Both methods are described below.

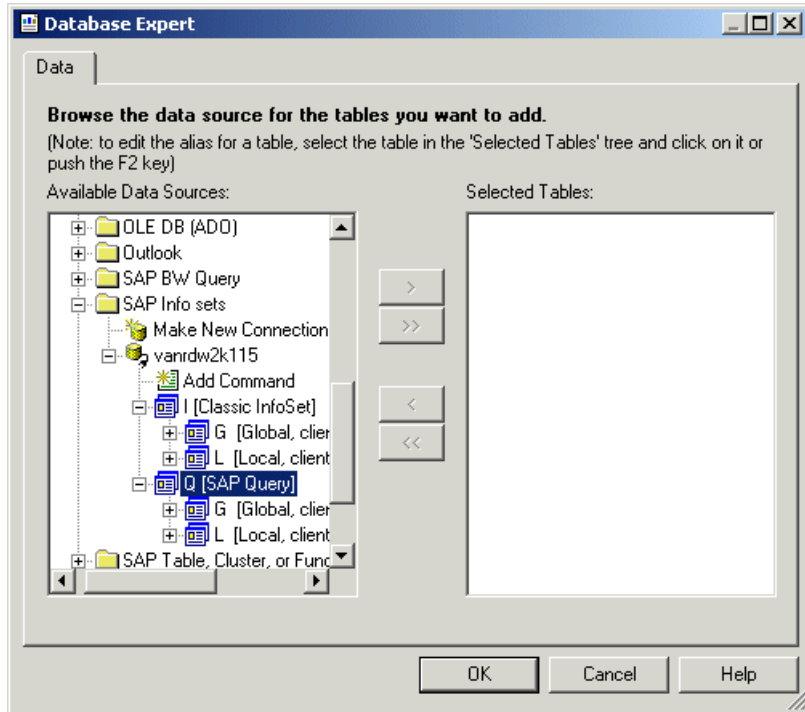
**Note:**

Crystal Reports includes several integrated tools that allow you to report off your data. For more information on the SAP Tools toolbar, see [Commands and connection settings in Crystal Reports](#) on page 14.

## Using the Report Wizards

1. Start Crystal Reports.
2. Click the **Start Page** tab if the Start Page is not already visible in Crystal Reports.
3. In the New Reports area, click **Blank Report**.

The "Database Expert" dialog box appears.



4. Expand **Create New Connection**, then expand the folder that corresponds to the driver that you want to use:
  - To use the BW Query driver, expand "SAP BW Query."
  - To use the Info Set driver, expand "SAP Info sets."
  - To use the Open SQL driver, expand "SAP Table, Cluster, or Function."
  - To use the ODS driver, expand "SAP Operational Data Store."
  - To use the BW MDX Query driver, expand "SAP BW MDX Query."

Once you've selected a driver, the "SAP System Logon" dialog box appears. For more details, see [Logging on to the SAP server](#) on page 101.

## Using the Log On Server command

1. Start Crystal Reports.

2. On the **File** menu (or **Database** menu, if a report is already open), click **Log On or Off Server**.
3. In the Data Explorer, expand **Create New Connection**. Then, expand the folder that corresponds to the driver that you want to use:
  - To use the BW Query driver, expand "SAP BW Query."
  - To use the Info Set driver, expand "SAP Info sets."
  - To use the Open SQL driver, expand "SAP Table, Cluster, or Function."
  - To use the ODS driver, expand "SAP Operational Data Store."
  - To use the BW MDX Query driver, expand "SAP BW MDX Query."

Once you've selected a driver, the "SAP System Logon" dialog box appears. For more details, see [Logging on to the SAP server](#) on page 101.

## Logging on to the SAP server

Regardless of which SAP driver you select, you must specify the SAP system that you want to report off and provide valid R/3 or BW user credentials for that system. Each driver prompts you for this logon information before it allows you to select the elements that you want to include in the report. Your user credentials are also requested when you run any existing reports that are built off SAP data sources.

### To log on to the SAP server

1. Once you have selected a driver, the SAP System Logon dialog box appears.
2. Select the SAP system that you want to report off and click **Next**.

#### Note:

If your SAP administrator has not configured SAP Logon, your SAP system will not appear in the Available SAP systems list. For more information about this, see the *BusinessObjects XI Integration for SAP Installation Guide*.

The next "SAP System Logon" dialog box prompts you for user logon credentials.

3. In the **Client** field, type your usual three-digit SAP client number.
4. In the **Username** and **Password** fields, type your usual SAP logon credentials, then click **Next**.

**Note:**

When selecting Secure Network Communication (SNC) enabled entries, the password field will be disabled.

The next "SAP System Logon" dialog box prompts you for extended logon parameters.

5. If you want the SAP Communication Library to create trace files for all client/server communication (for troubleshooting purposes), select the **Generate RFC trace files** check box.

6. Click **Finish**.

Crystal Reports logs you onto the SAP system and displays the connection in the Data Explorer (or in the Available Data Sources list of a Report Wizard).

**Note:**

It is strongly recommended that you filter the tables that are displayed under the new connection by right-clicking the connection, selecting Options, and modifying the appropriate settings in the "Options" dialog box. (For details, see [Crystal Reports configuration options](#) on page 102.) You can choose not to specify filter options for the tables that are returned to the Data Explorer; however, if the SAP system contains numerous tables, a significant wait may occur while Crystal Reports retrieves the list of data sources from your SAP system.

## Crystal Reports configuration options

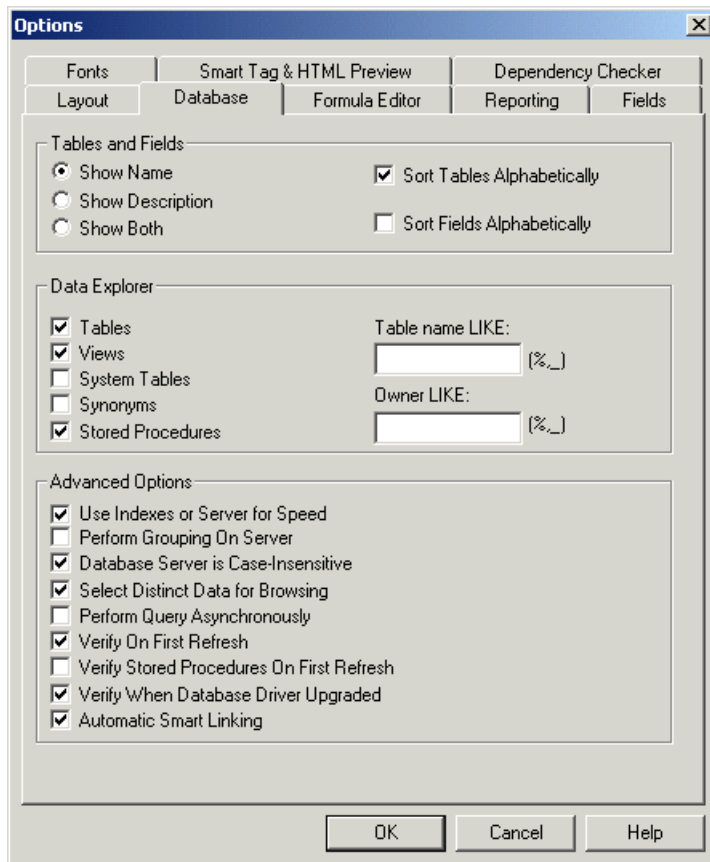
When you install Crystal Reports, you may need to change your global reporting options to improve the experience of reporting off SAP. The steps in this section show how to configure Crystal Reports to list tables and fields by both name and description. The section also shows how to filter the tables that are displayed in the Data Explorer.

Your SAP system may contain thousands of tables, each with a predefined technical name. The options that are described in this section ensure that you can select specific tables for a report without having to search through a lengthy list.

## To change configuration options

1. On the **File** menu, click **Options**.

The "Options" dialog box appears.



2. Click the **Database** tab.
3. In the Data Explorer area, select the check boxes for the types of data that you want to make available for the report.
4. Use the **Table name LIKE** and/or the **Owner LIKE** fields to select only a subset of the available data types. Use any of the following techniques when filtering:
  - Type full or partial names of tables.

- Add wildcards to select multiple queries or InfoSets: use a percentage sign (%) to denote any number of characters; use an underscore (\_) to denote a single character. (The % and \_ wildcards correspond respectively to the \* and ? wildcards that are used within Windows.)
  - Clear both fields and click OK to proceed without filtering. Note, however, that there is a limit to the number of objects that can be displayed in the Data Explorer. If the number of queries or InfoSets in SAP exceeds this limit (which is by default 8000), you may receive an Out of Memory error or a blank Data Explorer.
5. In the Tables and Fields area, select **Show Both**.
  6. Click **OK**.

## BusinessObjects InfoView

SAP functionality is integrated into the standard BusinessObjectsInfoView. Through BusinessObjectsInfoView, you can access all of the Crystal reports that have been saved in your SAP BW roles and published to BusinessObjects Enterprise. In addition, you can carry out all of the reporting tasks that are supported by the standard InfoView.

BusinessObjectsInfoView provides a number of features that are specific to SAP reporting environments. These features include the ability to log on with your usual SAP user name and password, a My Groups folder that provides quick access to all of the Crystal reporting content that has been saved in your various SAP BW roles, and the ability to view and schedule reports in multiple languages.

You can configure your BusinessObjectsInfoView settings and save your favorite reports. Once you find the reports that you want to view, BusinessObjectsInfoView offers report viewers with printing and exporting capabilities.

### To access BusinessObjects InfoView

1. Open your web browser.
2. Enter the appropriate URL for InfoView:
  - For Java InfoView go to <http://webserver:portnumber/InfoViewApp/>
  - For .NET InfoView go to <http://webserver/InfoViewApp/>



Replace *webserver* with the name of the web server and *portnumber* with the port number that is set up for BusinessObjects Enterprise. You may need to ask your administrator for the name of the web server, the port number, or the exact URL to enter.

**Tip:**

If you have the BusinessObjects XI Integration for SAP installed, you can also go to **Start > Programs > BusinessObjects XI 3.1 >**

**BusinessObjects Enterprise** and select one of the following:

- BusinessObjects Enterprise .NET InfoView
- BusinessObjects Enterprise Java InfoView

The "Log On to InfoView" page is displayed. Click **Help** to see the *InfoView Online Help*.





# Reporting with SAP Relational Data Sources



# 4

chapter



# Reporting off Tables, Views, Clusters, and Functions

This section describes introductory and advanced tasks, which are related to reporting off tables, views, clusters, and functions with the Open SQL driver. Advanced tasks include reporting off ABAP data clusters and functions and using the Cluster Definition tool.

## Open SQL driver overview

The Open SQL driver is the most powerful of the BusinessObjects XI Integration for SAP Solutions drivers because of the flexibility that it provides for accessing data in SAP. By writing reports with the Open SQL driver, you gain easy access to SAP's transparent tables, pool tables, cluster tables, and views. If you are currently involved with the production of ABAP reports, you will find it easy to quickly build reports with this driver.

The advanced functionality of the driver allows you to report against ABAP functions and ABAP data clusters. ABAP developers can therefore perform advanced tasks such as reporting against data clusters in HR. The Cluster Definition tool is provided as part of the server installation for users who need to add, edit, and delete data cluster definitions. Developers can also utilize ABAP to augment report processing.

Additionally, the Open SQL driver provides the ability to secure the data to which users have access. You can place limitations either on individual tables, or, at a more granular level, on the individual rows within a table. For more information about using the Security Definition Editor to customize your own set of data access authorizations, see the "Defining Security for the Data Access Kit" section in the *BusinessObjects XI Integration for SAP Installation Guide*.

## Objects supported by the Open SQL driver

The Open SQL driver provides access to several low-level objects within SAP: transparent tables, pool and cluster tables, views, and ABAP data clusters and functions.

## Transparent tables

This type of database object is similar to the traditional database table. Transparent tables contain the majority of the SAP application data from which you can derive Business Intelligence. Most SAP applications use one or more transparent tables to store and retrieve business transactions. You can link these tables to other transparent tables in order to pull information from other application areas of the SAP system.

## Pool and cluster tables

Pool and cluster tables are made up of a logical grouping of other database tables. They are primarily used to store application configuration and control information rather than the actual application data. Because these tables are only a logical view of several tables combined, they are accessible only through the Open SQL driver. Native database access does not provide the same view of this data.

## Views

A view appears as one table with a list of fields, but actually consists of several tables linked together. Views provide a more abstract view of the data and allow you to build reports without having to worry about linking multiple tables in a report. Views are maintained within SAP and need to be configured before you use them for reporting. For more information about views, see your SAP documentation.

## ABAP data clusters and functions

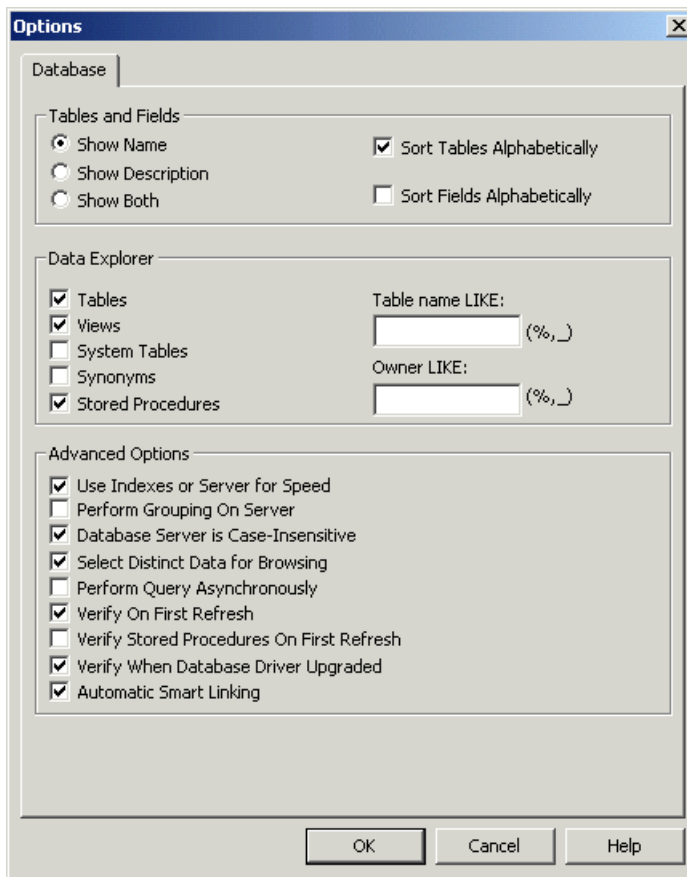
Reporting off ABAP data clusters and functions are advanced topics that usually require some understanding of the ABAP programming language. For details, see [Reporting off ABAP functions](#) on page 114 and [Reporting off ABAP data clusters](#) on page 121.

## Selecting tables, views, functions, and clusters

Once you have selected the Open SQL driver and provided valid SAP user credentials, use the Options dialog box to choose the tables, views, functions, or clusters that you want to add to the report. You can filter by the type of table and by the actual table name. Note that, in this situation, table also refers to views, ABAP data clusters, and ABAP functions.

## To select tables, views, functions, and clusters

1. Start Crystal Reports.
2. Click the **Start Page** tab if the Start Page is not already visible in Crystal Reports.
3. Click **Blank Report** in the New Reports area.  
The "Database Expert" dialog box appears.
4. In the **Create New Connection** folder, expand **SAP Table, Cluster, or Function**.  
The "SAP System Logon" dialog box appears.
5. Log on to an SAP system, provide your user logon credentials, and specify extended logon parameters as prompted. Then, click **Finish**.  
You return to the "Database Expert" dialog box.
6. Right-click the connection you created and select **Options**.  
The "Options" dialog box appears, with only the **Database** tab displayed.



7. In the Data Explorer area, select the check boxes for the types of data that you want to make available to the report.
8. Use the **Table name LIKE** field to select only a subset of the available data types. Use any of the following techniques when filtering:
  - Type full or partial names of tables.
  - Add wildcards to select multiple queries or InfoSets: use a percentage sign (%) to denote any number of characters; use an underscore (\_) to denote a single character. (The % and \_ wildcards correspond respectively to the \* and ? wildcards that are used within Windows. The % and \_ wildcards also correspond respectively to the \* and + wildcards that are used within SAP.)

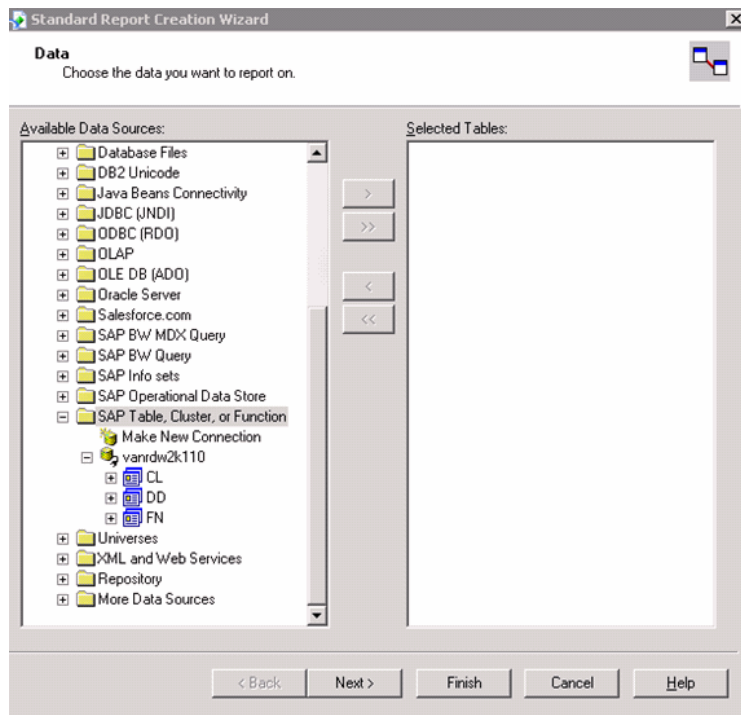
- Clear the field and click **OK** to proceed without filtering. Note, however, that there is a limit to the number of objects that can be displayed in the Data Explorer. If the number of queries or InfoSets in SAP exceeds this limit (which is by default 8000), you may receive an Out of Memory error or a blank Data Explorer.

9. In the Tables and Fields area, select **Show Description**.

10. Click **OK**.

The Database Expert displays the selected clusters, tables, or functions, along with the descriptions that were defined by the SAP administrator. Three branches appear under the data connection that you created:

- ABAP Function modules
- Data Clusters
- Data Dictionary



11. In the Available Data Sources area, double-click the tables, clusters, or functions that you want to add to the report.



**Tip:**

You can also CTRL-click multiple tables and then click >.

**12. Click OK.**

If you selected multiple tables for the report, you must provide table-linking information. For more details, see [Linking tables and views](#) on page 113. If you selected a single table, you can now add fields to the report.

## Linking tables and views

Reports that are based on SAP tables (transparent, pool, and cluster tables, as well as views) behave similarly to reports that are based on mainstream databases. If you include more than one table or view in a report, you must describe the relationship between the tables by using the Links tab in the Database Expert in Crystal Reports.

Due to the standardized field-naming conventions that are used by SAP, you may find that the Auto-Link feature (on the Links tab of the Database Expert) generates unnecessary links between tables.

### To clear Auto-Links

1. On the **Database** menu, click **Database Expert**.
2. On the **Links** tab, click **Clear Links**.
3. Click **OK**.

**Note:**

The **Links** tab appears in the Database Expert only when two or more tables are selected for the report.

The following client field and language field features are specific to SAP tables.

### Client field

Most tables in SAP contain a client field (native field name is MANDT). Because individual SAP systems can be configured to hold data for multiple clients, the client field specifies which rows of data in a table are for which clients. The Open SQL driver automatically handles this configuration. When

you connect to SAP, you are prompted to provide your client number: this information eliminates the need to create a selection formula that filters the clients. As well, if a report contains multiple tables, you do not need to include a link on the client field—this, too, is automatically handled by the Open SQL driver.

## Language field

Some tables in SAP, specifically those that contain text descriptions, have multiple entries—one for each language that is installed on the SAP system. Consequently, with these tables you must create a selection formula on the Language Key (native field name is SPRAS) and specify the language that you want to display. For example:

```
{table.spras} = "E"
```

For additional help on table-linking and on general report design topics, see the *Crystal Reports Online Help*.

## Reporting off ABAP functions

Reporting off ABAP Functions is an advanced topic that requires an understanding of the ABAP programming language.

While technically not a table, an ABAP function is often used in SAP reporting to provide calculations, and to retrieve data that is based on parameters that are passed into the function (for example, an employee number passed to a function that returns the employee's address). For this reason, the Open SQL driver provides the ability to call ABAP functions that have defined return types and to display their results. ABAP functions are also useful in situations where complex pre-processing of the data must occur before viewing, or in circumstances where you may want to further optimize the performance of a query.

## How it works

Using the steps listed in [Selecting tables, views, functions, and clusters](#) on page 109, select a function. In the Database Expert's Available Data Sources area, right-click ABAP Function modules and select **Options**. Then, type the

full or partial function name in the Table name LIKE field. When you add it to the report, the function appears on the Links tab of the Database Expert just like a table (assuming there is more than one table or function already added to the report). All the input and output parameters are displayed as fields in the table.

Specify the input parameters either by linking them to another table's fields or by adding them to a record selection formula. If the output of the function is one or more tables, the "virtual table" consists of rows that are the cross-product of the values in all the tables. If the output of the function is one or more values or structure instances, the "virtual table" consists of a single row that contains all the values/structure values.

## Limitations

For functions to be used in a report, they must meet two conditions:

- Functions must have defined return types for each of the output parameters.
- Functions must not have whole tables as input parameters.

### Note:

If a function does not meet the first condition, you can write a "wrapper" function with defined return types, and then call the original function from within the wrapper function.

Finally, the ability to call entire programs is not supported—you may call individual functions only.

## Examples

The following examples illustrate the different types of ABAP functions that can be called by the Open SQL driver and show how they appear in Crystal Reports.

### Functions with values as inputs and outputs

```
Function F
Importing
  VALUE(NAME) TYPE C
  VALUE(DEPARTMENT) TYPE C
Exporting
```

```
VALUE(Years_of_service) TYPE I
VALUE(Age) TYPE I
```

This function would appear as a table called "F" with the following schema:

Field	Type
Name	String
Department	String
Years_of_service	Integer
Age	Integer

The first two fields, Name and Department, are input parameters. You must either link to them from another table (or function), or specify them with a record selection formula (for example, {F.Name} = "Richard").

For example, if table "T" contains departments that are over budget, you can link {T.Department} to {F.Department} and have a record selection formula that specifies {F.Name} = "Richard". This formula will get the years of service and age of all people named Richard in departments over budget.

### Functions with values as inputs and outputs and in/out (CHANGING) parameters

```
Function F
IMPORTING
  VALUE(NAME) TYPE C
  VALUE(DEPARTMENT) TYPE C
EXPORTING
  VALUE(Years_of_service) TYPE I
  VALUE(Age) TYPE I
CHANGING
  DATEFIELD TYPE D
```

This function would appear as a table called "F" with the following schema:

Field	Type
Name	String
Department	String
Years_of_service	Integer
Age	Integer
DateField_In	Date
DateField_Out	Date

When working with the Name, Department, and DateField\_In fields, you must either link them to another table, or specify them in a record selection formula.

For example, if table "T" contains departments that are over budget, you can link {T.Department} to {F.Department} and have a record selection formula that specifies {F.Name} = "Richard" and DateField = Date(1999,04,14). This formula will get the years of service, age, and start date of all people named Richard in departments over budget on April 14, 1999.

## Functions with structures as inputs and outputs

```
Function F
IMPORTING
  VALUE(ADDRESS) LIKE S_ADDR STRUCTURE S_ADDR
EXPORTING
  VALUE(CONTACTINFO) LIKE S_CINFO STRUCTURE S_CINFO
S_CINFO is defined as:
PHONE(20) TYPE C
EMAIL(100) TYPE C
S_ADDR is defined as:
```

```
STREET(100) TYPE C  
ZIPCODE(6) TYPE N
```

This function would appear as a table called "F" with the following schema:

Field	Type
Address.street	String
Address.zipcode	Integer
ContactInfo.Phone	String
ContactInfo.email	String

As before, you must either link the street and zip code fields to or from another table, or define them in a record selection formula.

### Functions with a table as their output

```
Function F  
IMPORTING  
  VALUE(NAME) TYPE C  
EXPORTING  
  VALUE(GENDER) TYPE C  
TABLES  
POSITIONS STRUCTURE S_POSN  
S_POSN is defined as:  
TITLE(20) TYPE C  
STARTDATE TYPE D
```

In this case, the function takes a name as an argument and returns the age and position of that person. Suppose that Tom's gender is male, and that the Positions table returned by F for Tom contains:

Title	StartDate
Mail Clerk	1/1/93
Manager, Mail	3/4/95
VP, Mail	7/12/98

When the function is called, the result will be a table called "F" that contains:

Name	Gender	Positions.Title	Positions.Start Date
Tom	M	Mail Clerk	1/1/93
Tom	M	Manager, Mail	3/4/95
Tom	M	VP, Mail	7/12/98

## Functions with multiple tables as their output

```
Function F
IMPORTING
  VALUE(NAME) TYPE C
EXPORTING
  VALUE(GENDER) TYPE C
TABLES
POSITIONS STRUCTURE S_POSN
OPTIONS STRUCTURE S_OPTN
S_POSN is defined as before and S_OPTN is defined as:
GRANTDATE TYPE D
NUMBER TYPE I
```

Suppose that the data from Positions is as before, and the Options table looks like:

GrantDate	Number
4/4/93	1000
5/6/97	15300

When the function is called, the result will be a table called "F" that contains:

Name	Gender	Posi tions.Title	Positions. StartDate	Options. GrantDate	Options. Number
Tom	M	Mail Clerk	1/1/93	4/4/93	1000
Tom	M	Manager, Mail	3/4/95	4/4/93	1000
Tom	M	VP, Mail	7/12/98	4/4/93	1000
Tom	M	Mail Clerk	1/1/93	5/6/97	15300
Tom	M	Manager, Mail	3/4/95	5/6/97	15300
Tom	M	VP, Mail	7/12/98	5/6/97	15300



## Reporting off ABAP data clusters

Reporting off ABAP data clusters is an advanced topic that requires an understanding of the ABAP programming language.

### What is a data cluster?

Unlike most of the SAP system, the Human Resources module stores much of its data in ABAP data clusters, rather than in data dictionary tables such as transparent, pool, or cluster tables. Data clusters are actually stored in some transparent tables. Any such transparent table can contain many different types of clusters. Each cluster of data is stored in encoded binary fields. A cluster can contain any number of individual components or pieces of data, which may range from single values to structures to tables.

You can access a data cluster in an ABAP program by running the ABAP command `IMPORT`. This command takes parameters—such as cluster database name, cluster ID, and so on—as well as a list of pieces of data to import. The `IMPORT` command also copies the contents of the data cluster into local variables within the calling code. A typical transparent table that contains cluster data has a structure similar to the following:

client	area	key	line counter	user data	data length	data cluster
--------	------	-----	--------------	-----------	-------------	--------------

The combination of transparent table name and area identifies which type of cluster to import, while the key identifies the specific cluster. For HR, the cluster key is typically some combination of personnel number and something else. The key is stored in a single text field, but may actually represent more than one logical key field (for example, personnel number + a sequence number for HR Payroll data).

### How it works

The difficulty in working with clusters from a generic, relational data access point of view is that the definition of the clusters is not stored in a repository

such as the ABAP dictionary (unlike the definitions of transparent, pool, and cluster tables, for instance). Furthermore, the format of these clusters may change from site to site due to customizations. Finally it is fundamentally difficult to map these as tables in the relational sense because, although the data represented is relational, it is stored in independent clusters in an almost hierarchical form.

In order to overcome these issues, the individual components of data clusters must be mapped as tables. You can do this by creating a dictionary of the clusters on a particular SAP system. The dictionary can then be accessed by the Open SQL driver. To build this dictionary, use the Cluster Definition tool, which was installed by your administrator with the BusinessObjects XI Integration for SAP Solutions transport files.

## Mapping ABAP data clusters

The Cluster Definition tool (transaction ZCDD) enables you to create a dictionary of the data clusters on your SAP system. Once you have added a data cluster to the dictionary, you can select the cluster in Crystal Reports and report off its data with the Open SQL driver. You will usually require prior experience with ABAP programming because you need to find the cluster definitions within R/3 before you enter them with the Cluster Definition tool.

Combined, the following procedures show how to map the SALDO table (Time balances per period) to your dictionary of data clusters with the Cluster Definition tool. The SALDO table is located in the B2 (PDC Data (Month)) cluster area of the ABAP data cluster that is stored in the PCL2 (HR cluster 2) transparent table. In this case, PCL2 is the cluster database, B2 is the cluster area or type, and SALDO is the table. The procedures listed below correspond to SAP R/3 4.6b:

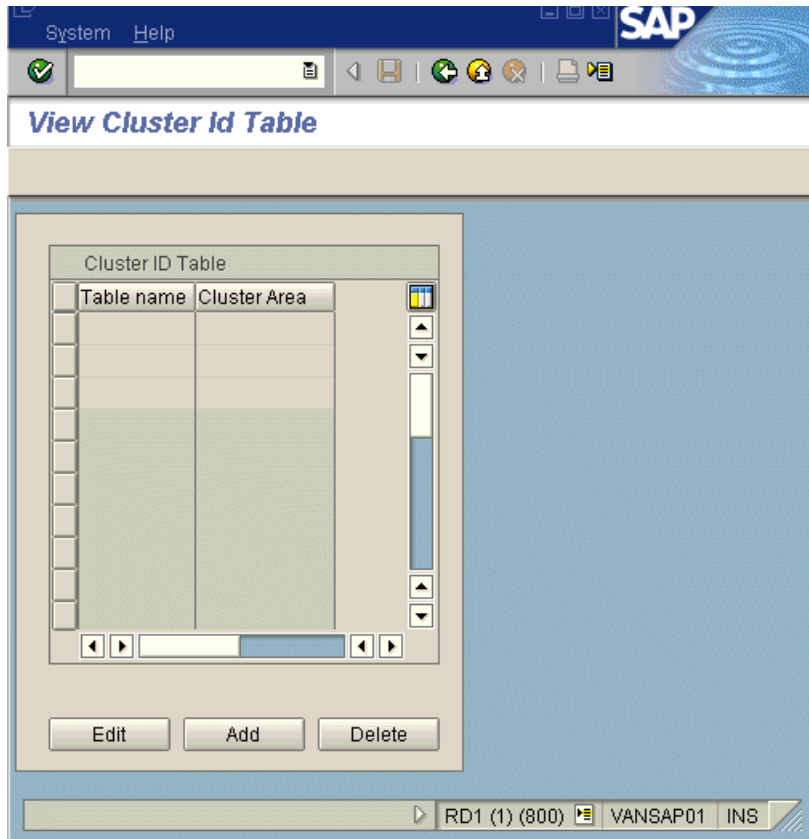
### Creating a new cluster entry in the dictionary

This section guides you through the process of defining a new cluster entry in the dictionary.

#### To create a new cluster entry in the dictionary

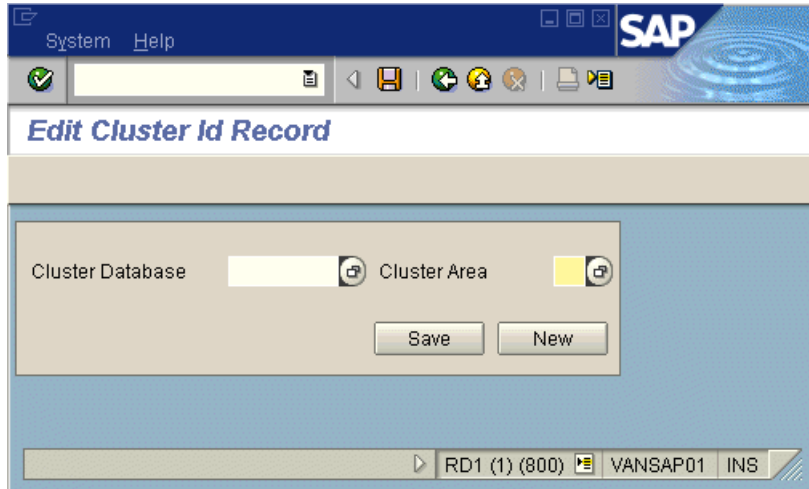
1. From the SAPGUI, execute transaction `ZCDD`.

The Cluster Definition tool displays its "View Cluster Id Table" screen.



2. Click **Add** to create a new entry.

The "Edit Cluster ID Record" screen appears.



3. In the **Cluster Database** field, type the name of the table in which the ABAP data cluster is stored.

For this example, type PCL2.

**Note:**

To find out which transparent table contains a required ABAP data cluster, ask a subject matter expert about the module on which you are working. Or, if you are trying to locate the data from a particular program (such as an SAP report like RPC11XE0), look for the table in the ABAP code. When you find the table, look at the data in the RELID field for a list of cluster areas.

4. In the **Cluster Area** field, type the name of the cluster area or RELID.

For this example, type B2.

5. Click **Save** and exit the Cluster Definition tool.

You now need to find the key to the cluster area. Specifically, you need the name of the dictionary structure or table in which the key fields are stored, and you need the names of those key fields. To locate the cluster area's key fields, go to the next section, [Locating the cluster area's key fields](#) on page 125.

## Locating the cluster area's key fields

This section guides you through the process of locating the cluster area's key fields based on the new cluster entry you created in the last section, [Creating a new cluster entry in the dictionary](#) on page 122.

There are several ways to find this information within SAP. This example uses the Object Navigator to locate the key information for the B2 cluster area.

### To locate the cluster area's key fields

1. Execute transaction `SE80` to access the Object Navigator.
2. Select **Program** from the list and type the name of the program that holds the key information related to the cluster area.

For this example, type `RPCLSTB2`, which is the program that holds the key information for the B2 cluster area.

3. Press **Enter**.

The Object Navigator displays the information for the specified program.

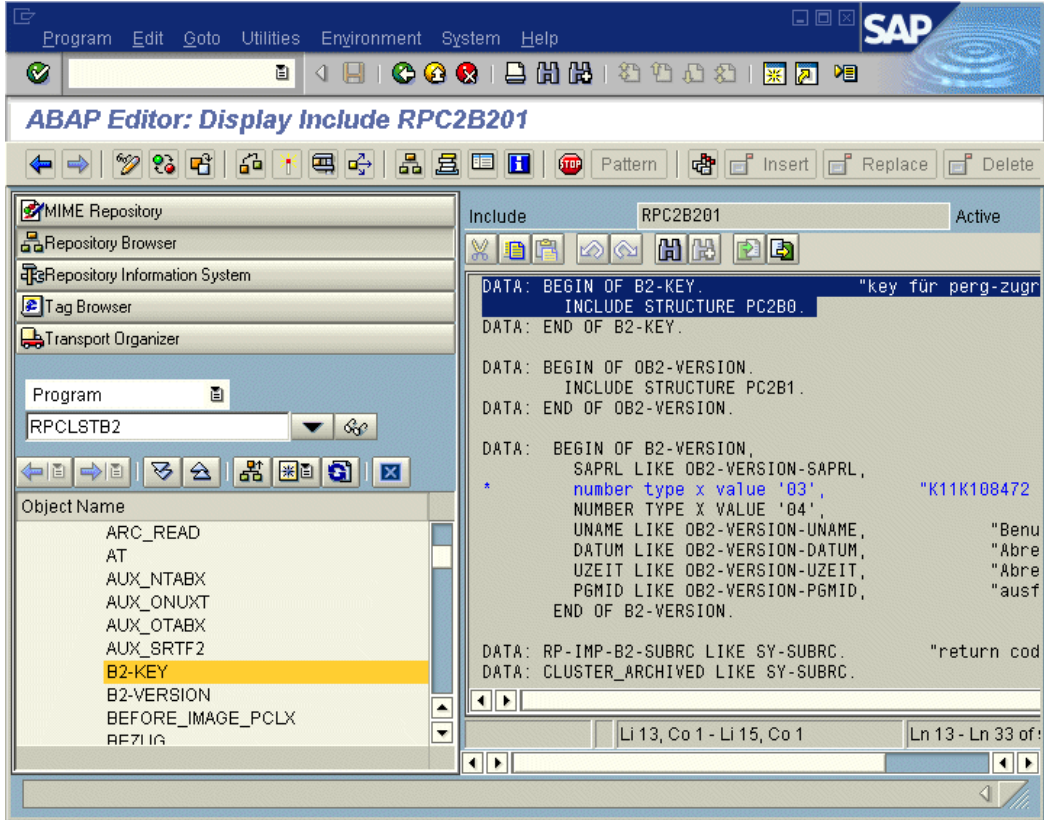
4. In the **Object name** list, expand the **Fields** folder.
5. Double-click the entry that corresponds to your cluster area's key.

For this example, double-click the `B2-KEY` entry.

The relevant ABAP code is displayed.

## 4 Reporting with SAP Relational Data Sources

### Reporting off Tables, Views, Clusters, and Functions



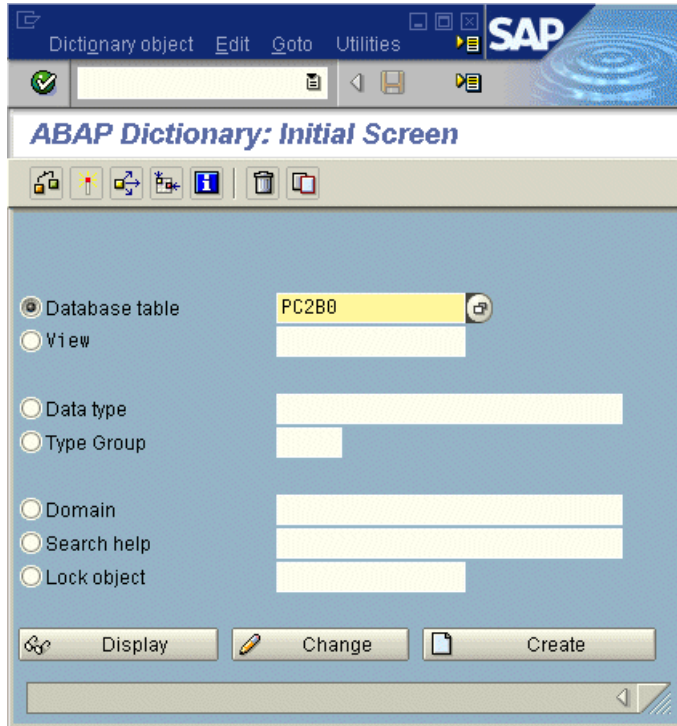
6. In the ABAP code, look for the heading that denotes the beginning of the cluster area's key, and note the structure that is included in the following line.

For this example, look under the heading DATA: BEGIN OF B2-KEY for the line INCLUDE STRUCTURE PC2B0 (highlighted in the image above). This shows that the PC2B0 structure contains the fields of the key.

Now that you've located the name of the structure that holds the key fields, you need to determine the names of those key fields.

7. Create a new session in R/3 and execute transaction SE11 to access the ABAP/4 Dictionary.

The "ABAP Dictionary: Initial" Screen appears.

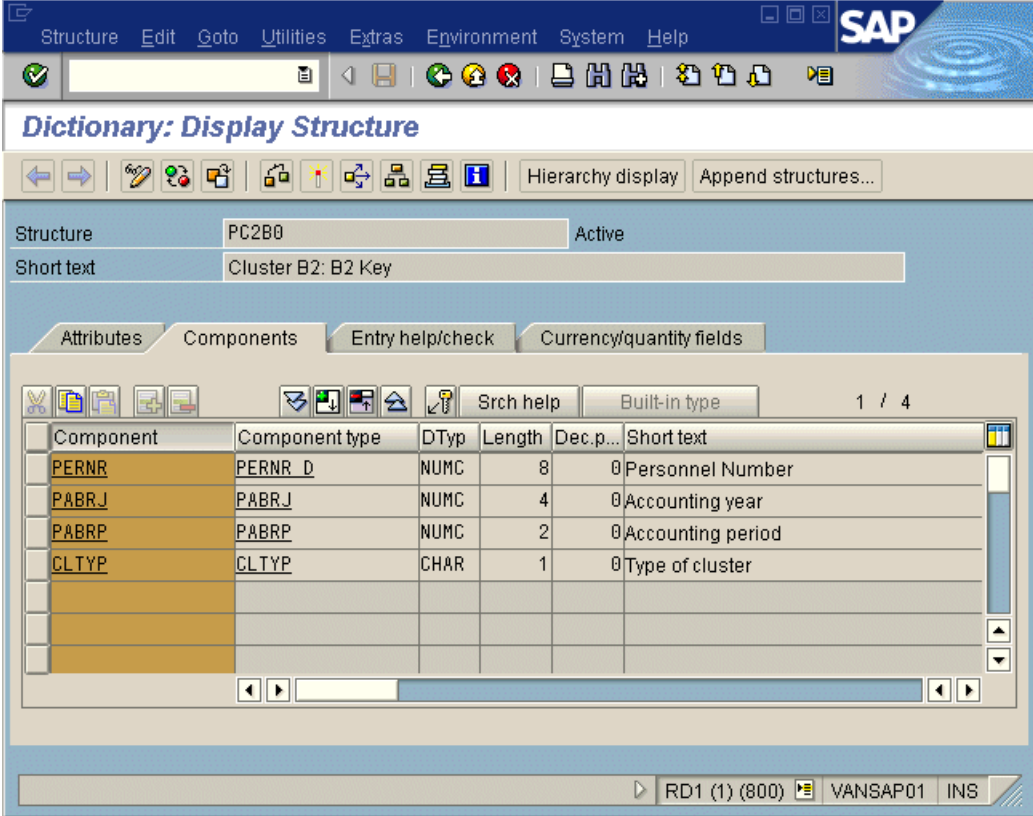


8. Select **Database table** and type the name of the structure that contains the key fields.

For this example, type PC2B0.

9. Click **Display**.

The "Dictionary: Display Structure" screen appears.



10. Make a note of the names of the key fields displayed in the **Component** column.

In this case, the four key fields are PERNR, PABRJ, PABRP, CLTYP.

You now need to add the key fields to your new dictionary entry in the Cluster Definition tool. To add these key fields to the dictionary entry, go to next section, [Adding the key fields to the dictionary entry](#) on page 128.

## Adding the key fields to the dictionary entry

This section guides you through the process of adding the key fields to the dictionary entry based on the key fields that you noted in the last section, [Locating the cluster area's key fields](#) on page 125.



Following are the steps involved in adding the four key fields to the dictionary entry that you created for the B2 cluster area.

**To add the key fields to the dictionary entry**

1. In the SAP Easy Access screen, execute transaction `ZCDD` to access the Cluster Definition tool.
2. In the "View Cluster Id Table" screen, select the new entry that you created.

For this example, select the PCL2 B2 entry.

3. Click **Edit**.

The "View Cluster Record" screen appears.

## 4 Reporting with SAP Relational Data Sources

### Reporting off Tables, Views, Clusters, and Functions

System Help

Cluster Database: PCL2 Cluster Area: B2 Edit New Delete

Cluster Key Definition

Key Num	Key Name	Reference Table	Reference Field	Data Type	Data Length
---------	----------	-----------------	-----------------	-----------	-------------

Edit Cluster Key New Cluster Key Delete Cluster Key

Cluster Data Definition

Display Name	Piece Name	Class	Ref Table	Ref Field	Data Type	Data Length
--------------	------------	-------	-----------	-----------	-----------	-------------

Edit Cluster Data New Cluster Data Delete Cluster Data

#### 4. Click **New Cluster Key**.

The "Edit Cluster Key Record" screen appears.

5. In the **Key Number** field, type the sequence number of the key field that you want to specify.

For this example, type 001 (this is the first entry for this cluster area's key).

6. In the **Key Name** field, type the name that you want to appear when you use this key field in Crystal Reports.

For this example, type K\_PERNR.

**Tip:**

In general, follow a naming convention. For instance, start a key field name with K\_ to specify it as a key field, as has been done here.

7. Use the **Reference Table** field to define the key field's data type by referencing an existing field of the same type in another table.

For this example, type PC2B0, because it is the name of the structure that holds the Reference Field.

8. In the **Reference Field** field, type the name of the field to which you want to refer. This field must exist in the Reference Table.

For this example, type PERNR.

**Note:**

If the key field is not already defined in the data dictionary, and if you do not define it by referencing an existing field, you can manually specify the ABAP type and Length in order to define the key field's data type. In this example, the key field is defined by reference, so you can ignore the ABAP type and Length fields.

9. Click **Save**; then click **Back** to return to the View Cluster Record screen.
10. Repeat steps 4 to 9 for each field in the key. In the **Key Number** field, be sure to increment the sequence number by one every time you define a new key field for this cluster area.

The values for each field in this example are:

Key Number	Key Name	Reference Table	Reference Field
002	PABRJ	PC2B0	PABRJ
003	PABRP	PC2B0	PABRP
004	CLTYP	PC2B0	CLTYP

Once you have entered all of the definitions, the "View Cluster Record" screen looks like this.

**View Cluster Record**

Cluster Database: PCL2 Cluster Area: B2 Edit New Delete

**Cluster Key Definition**

Key Num	Key Name	Reference Table	Reference Field	Data Type	Data Length
1	PERNR	PC2B0	PERNR		0
2	PABRJ	PC2B0	PABRJ		0
3	PABRP	PC2B0	PABRP		0
4	CLTYP	PC2B0	CLTYP		0

Edit Cluster Key New Cluster Key Delete Cluster Key

**Cluster Data Definition**

Display Name	Piece Name	Class	Ref Table	Ref Field	Data Type	Data Length

Edit Cluster Data New Cluster Data Delete Cluster Data

RD1 (1) (800) VANSAP01 INS

Now you need to determine the structure of the table that you want to access. In this example, the table in question is SALDO. There are several ways to determine the SALDO table's structure within SAP. The next section [Determining the structure of the cluster table](#) on page 134 explains one way.

## Determining the structure of the cluster table

This section guides you through the process of determining the structure of the cluster table based on the cluster entry you created in the previous section. This example below again uses the Object Navigator to locate the required information about the SALDO table.

### To determine the structure of the cluster table

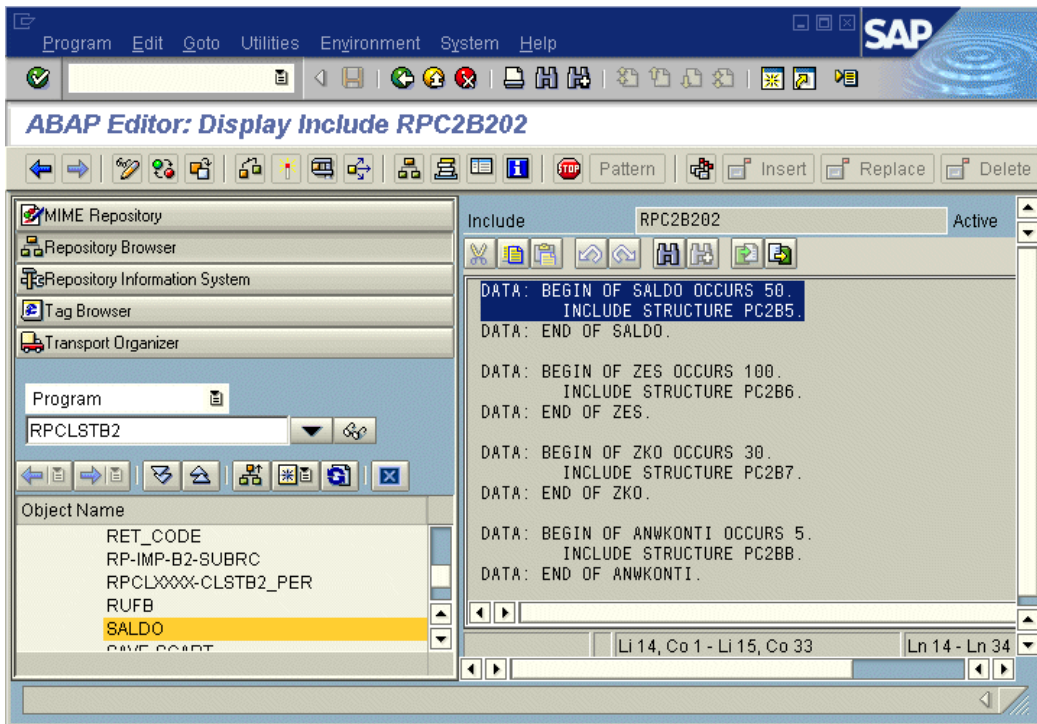
1. In the SAP Easy Access screen, execute transaction `SE80` to access the Object Navigator.
2. Select **Program** from the list and type the name of the program that holds the key information related to the cluster area.

For this example, type `RPCLSTB2`, which is the program that holds the key information for the B2 cluster area and its SALDO table.

3. In the **Object name** list, expand the **Fields** folder.
4. Double-click the entry that corresponds to the table that you want to access.

For this example, double-click the SALDO entry.

The relevant ABAP code is displayed.



5. In the ABAP code, look for the heading that denotes the beginning of the table, and note the structure that is included in the following line.

For this example, look under the heading DATA: BEGIN OF SALDO OCCURS 50 for the line INCLUDE STRUCTURE PC2B5 (highlighted in the image above). This shows that the PC2B5 structure defines the fields of the SALDO table.

Now that you've determined which structure defines the fields of the table you want to report off, you must add this information to your new dictionary entry by following the steps in section, [Adding the table to the dictionary entry](#) on page 136.

## Adding the table to the dictionary entry

This section guides you through the process of adding the table to the dictionary entry based on the cluster table's structure you created in the last section, [Determining the structure of the cluster table](#) on page 134.

### To add the table to the dictionary entry

1. In the SAP Easy Access screen, execute transaction `ZCDD` to access the Cluster Definition tool.
2. In the "View Cluster Id Table" screen, select your new entry.

For this example, select the PCL2 B2 entry.

3. Click **Edit**.
4. In the "View Cluster Record" screen, click **New Cluster Data**.

The "Edit Cluster Data Record" screen appears.

System Help

**Edit Cluster Data Record**

Cluster Database: PCL2 Cluster Area: B2

Piece Name	
Display Name	
Piece Class	(T,B, or S)
Reference Table	
Reference Field	
ABAP Type	
Length	

Save New



5. In the **Piece Name** field, type the name of the table as it appears in the cluster area.

For this example, type SALDO.

6. In the **Display Name** field, type the name of the table as you want it to appear in Crystal Reports.

For this example, type B2SALDO.

**Tip:**

In general, use a naming convention. For instance, combine the table name with the name of the cluster area, as has been done here (in case there is another SALDO table in a different cluster area).

7. In the **Piece Class** field, type the abbreviation appropriate to the data piece: (T)able, (S)tructure, or (B)asic type.

For this example, type T.

8. Use the **Reference Table** field to define the structure of the new data piece by referencing an existing structure of the same type in another table.

For this example, type PC2B5.

**Note:**

In this example, you need not complete the remaining fields (Reference Field, ABAP Type, and Length). Use these fields to add specific information if the structure of the table is not defined in the data dictionary.

9. Click **Save**; then click **Back** to return to the "View Cluster Record" screen.

Now that you have entered all of the definitions, you have finished creating your new dictionary entry in the Cluster Definition tool. The View Cluster Record screen looks like this.

## 4 Reporting with SAP Relational Data Sources

### Reporting off Tables, Views, Clusters, and Functions

System Help

View Cluster Record

Cluster Database: PCL2 Cluster Area: B2 Edit New Delete

Cluster Key Definition

Key Num	Key Name	Reference Table	Reference Field	Data Type	Data Length
1	PERNR	PC2B0	PERNR		0
2	PABRJ	PC2B0	PABRJ		0
3	PABRP	PC2B0	PABRP		0
4	CLTYP	PC2B0	CLTYP		0

Edit Cluster Key New Cluster Key Delete Cluster Key

Cluster Data Definition

Display Name	Piece Name	Class	RefTable	RefField	Data Type	Data Length
B2SALD0	SALD0	T	PC2B5			0

Edit Cluster Data New Cluster Data Delete Cluster Data

RD1 (1) (800) VANSAP01 INS

When you create a new report with the Open SQL driver, you can report off your customized ABAP data cluster by searching for the table named B2SALDO.

Although it takes some time to define each data cluster, keep in mind that once the definition is complete all users can access the cluster—regardless of whether or not they have ABAP experience.

# Reporting off InfoSets and SAP Queries

This section describes the combined InfoSet/ABAP Query driver and shows how to select an InfoSet or an SAP query as a data source for a Crystal report.

## InfoSet Query driver overview

The combined InfoSet/ABAP Query driver allows you to use an existing SAP query or InfoSet (previously known as a Functional Area) as a data source for a Crystal report. The SAP query or InfoSet is built within SAP by a content expert, who then makes the data set available to the business user. Through this driver, the user can design multiple reports by quickly referencing the same query or InfoSet.

Because an InfoSet can be built on top of a logical database in SAP, it can inherit all of the business logic and security that is programmed into the logical database. However, the report that is designed through the driver can contain only as much data as the SAP query or InfoSet has been designed to retrieve.

Reports that are based on SAP queries and InfoSets behave similarly to reports that are based on stored procedures in mainstream databases. You can insert subreports that are based on different queries or InfoSets into the main report.

### Note:

- If you want to join two or more InfoSets or SAP queries, you should consider creating a brand new InfoSet or SAP query instead. The new join will be processed on the SAP server allowing the report to respond faster.
- Your SAP queries and InfoSets may contain parameters that you can use to limit the data that is returned from the SAP system. For example, you might specify a date range to get data for only a certain period. If parameters are included in the SAP query or InfoSet, Crystal Reports prompts you to specify the values that you want to include in the query.

**Tip:**

You can use transaction `sq02` to create an InfoSet in SAP, and transaction `sq01` to create an SAP query. For more information about designing SAP queries and InfoSets, refer to your SAP documentation.

## Accessing an InfoSet or SAP query

User groups are groups that are managed in SAP. They allow you to control user access to InfoSets and SAP queries

An SAP query can belong only to a single user group (which must be shared with the InfoSet that the SAP query is based on). In contrast, InfoSets can belong to multiple user groups. For a user to report off an InfoSet (or SAP query based on an InfoSet), the user and the InfoSet/SAP query must belong to a common user group.

**Tip:**

To associate existing InfoSets and users with additional user groups in SAP, use transaction `sq03`.

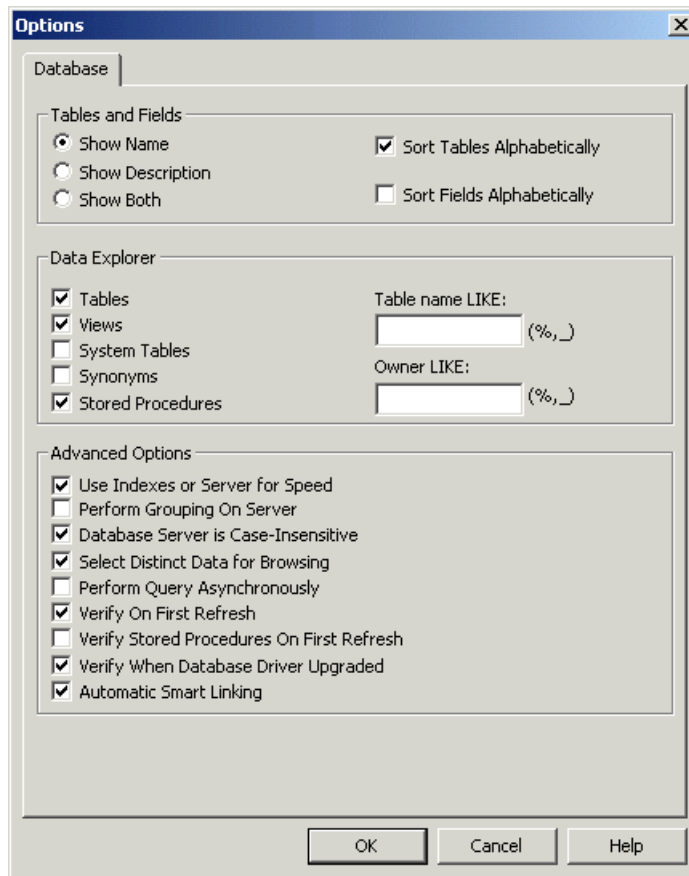
## Selecting an InfoSet or SAP query in Crystal Reports

1. Start Crystal Reports.
2. Connect to SAP as instructed in [Using the Log On Server command](#) on page 100. Select the SAP InfoSet driver, and provide your user credentials.

The "Data Explorer" dialog box appears.

3. Click **Options**.

The "Options" dialog box appears with only the **Database** tab displayed



4. In the Data Explorer area, select the check boxes for the types of data that you want to make available for the report.
5. Use the **Table name LIKE** field to select only a subset of the available data types. Use any of the following techniques when filtering:
  - Type full or partial names of tables.
  - Add wildcards to select multiple queries or InfoSets: use a percentage sign (%) to denote any number of characters; use an underscore (\_) to denote a single character. (The % and \_ wildcards correspond respectively to the \* and ? wildcards that are used within Windows. The % and \_ wildcards also correspond respectively to the \* and + wildcards that are used within SAP.)

- Clear both fields and click OK to proceed without filtering. Note, however, that there is a limit to the number of objects that can be displayed in the Data Explorer. If the number of queries or InfoSets in SAP exceeds this limit (which is by default 8000), you may receive an Out of Memory error or a blank Data Explorer.

**Note:**

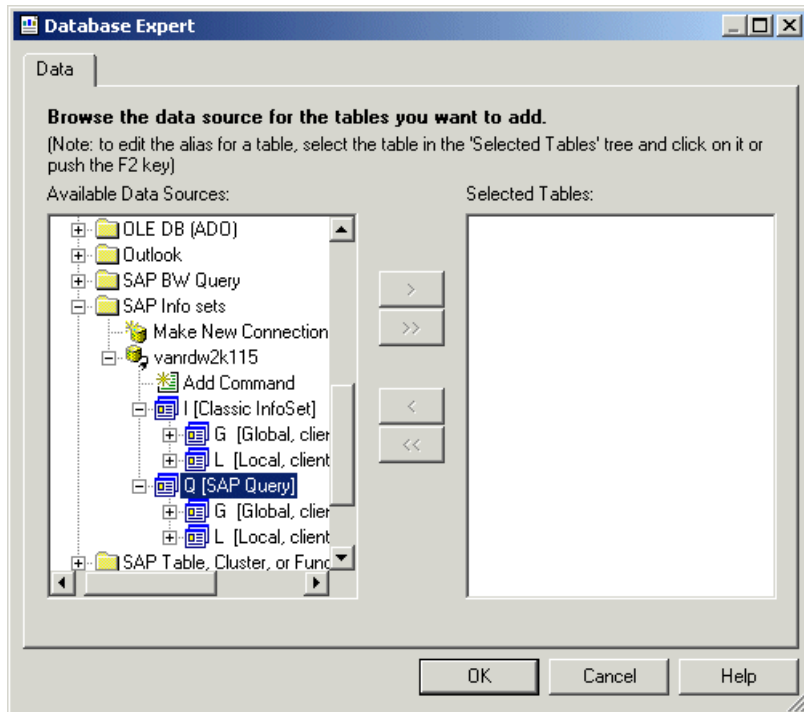
The InfoSet driver does not make use of the Owner LIKE field.

6. In the Tables and Fields area, select **Show Both**.
7. When you have specified the data sources that you want to see, click **OK**.

The Data Explorer displays the selected queries or InfoSets, along with the descriptions defined by the SAP administrator.

8. Click **Close**.
9. Open a new, blank report.

The Database Expert appears.



**Tip:**

You can also add InfoSets or SAP queries to existing reports. To do so, open the desired report and, on the **Database** menu, select **Database Expert**.

10. In the Available Data Sources area, locate the query or InfoSet that you want to use as a data source for the report:
  - Expand either I (for Info Set) or Q (for SAP query).
  - Expand either G (to search in the Global area) or L (to search in the Local area).

**Note:**

Queries and InfoSets in the Local area are client-dependent.

11. Double-click the query or InfoSet that you want to use as a data source for the report.

**Note:**

If the query or InfoSet contains parameters, the Enter Parameter Values dialog box will prompt you for information. Use the dialog box to specify the data that you want to include, and then click OK.

12. In the "Data Explorer" dialog box, click **OK**.

**Tip:**

For additional information about parameters and general report design topics, see the *Crystal Reports Online Help*.







## SAP specific features in InfoView

5

chapter



Text in development

## Working with My Groups

When you log on to BusinessObjectsInfoView with your SAP user credentials, the My Groups folder provides you with quick access to all of the Crystal reporting content that has been saved in your various SAP BW roles. Click the My Groups folder to access this view.

**Note:**

The My Groups folder is available only after your administrator has imported one or more of your roles that contain Crystal reports.

Each of your SAP BW roles is represented in BusinessObjectsInfoView as a folder that contains reports and/or additional subfolders. Search through the various folders to locate your reports.

Depending on your rights, you may be able to schedule different reports and/or view their latest instances. In addition, you can copy or move reports to your Favorites view so they are easier to locate the next time you log on.

## Publishing and BusinessObjects XI Integration for SAP

In general, Publishing in the BusinessObjects Integration for SAP functions in the same way as Publishing in a regular installation of BusinessObjects Enterprise. However, two important differences exist in the Publishing workflow for BusinessObjects Integration for SAP:

- You do not use the settings in the "Personalization" section to personalize source documents for SAP recipients. SAP recipients already have profile values mapped to their user accounts outside of the BusinessObjects Enterprise system. This serves as a built-in personalization method for SAP recipients. It is unnecessary to create profiles and profile values for SAP recipients in BusinessObjects Enterprise, or to map profiles to source document fields.
- The only report bursting method that works for a publication intended for SAP recipients is **One database fetch per recipient**. This method maximizes security and ensures that the database logon credentials of each publication recipient are processed individually.

For more information about Publishing, see the *BusinessObjects Enterprise Publisher's Guide*.

# Personalization in BusinessObjects Enterprise

BusinessObjects Integration for SAP Solutions supports the personalization of variables in reports that are generated from BW queries.

## What is personalization?

Reports that are based off BW queries potentially make use of variables. These variables contain values that the BW queries use to limit or specify returned data. Typically, you run reports with variables in two ways:

- Enter the value that will be used to run the report.
- Select the value from a list of predefined values.

In SAP Business Explorer (BEx), personalization enables users to enter a value for the variable and save it as their personal default value. As a result, when the report is run, the data that is generated is based on the variable value of their choice. If, in the future, the users want to run the same report, their saved personalized values are available for them to use.

### Note:

Personalized values are user-specific—that is, the personalized values that users set for themselves are not implemented onto other users. Other users who work with the same report can set their own personalized values, or use any of the preset values that are given to the variable.

For more information about personalization, see the documentation for your SAP BW system or contact your system administrator.

## Parameters

In BusinessObjects Integration for SAP Solutions, variables within reports are represented as parameters on BusinessObjects InfoView . Before users can view or schedule the report, they are prompted to choose a value from

a dynamic pick list for each parameter. The values that are in the list are the values that were assigned to the variable in the SAP environment and filtered based on the rights the user has in the SAP system.

From the "Enter Prompt Values" dialog box, you have the following options:

- Run the report with its default parameter values.
- Select values from the dynamic pick lists and run the report with your chosen parameter values.
- Enter a value for each parameter and run the report with those values.
- Run the report with null values for the parameters.
- Personalize a value for each parameter and run the report with those personalized values.

**Note:**

Some options are available only if they are enabled in the report's referenced BW query or in BusinessObjects Enterprise.

## Working with default parameter values

The default value for a report parameter is set in the SAP environment (while the query is being designed). Because the report in BusinessObjects Enterprise is based off a BW query, the default value for the query variable automatically becomes the default value for the report parameter.

### To run a report with default parameter values

1. Log on to BusinessObjectsInfoView.
2. In the **My Groups** folder of BusinessObjectsInfoView, navigate to the object that you want to view.
3. Click the object's title.

The "Enter Prompt Values" screen appears.

4. Click **OK**.

The report appears in your Crystal report viewer. It contains only data that is based on the default values that were assigned to the parameter.

For information about setting your own default values see [Personalizing parameter values](#) on page 150.

## Choosing parameter values from dynamic pick lists

The items that populate a dynamic picklist for a parameter are based on the values that were assigned to the variable in the SAP environment. The values that are displayed on BusinessObjectsInfoView correspond to your user rights; you see only the values for which you have the rights to report.

**Note:**

This option is only available for BW Query based parameters.

### To run a report with parameter values from a dynamic pick list

1. Log on to BusinessObjectsInfoView.
2. In the **My Groups** folder of BusinessObjectsInfoView, navigate to the object that contains variables.
3. Click the object's title or click **View Latest Instance**.  
The "Enter Prompt Values" dialog box appears.
4. Beside a parameter, click the "..." button.  
The Picklist screen appears.
5. Search the list for the value that you want for the parameter. When you find it, click its hyperlink.  
You are returned to the "Enter Prompt Values" dialog box. The value that you selected appears in the edit field for the parameter.
6. Repeat Steps 3 and 4 for other parameters, and then click **Execute**.  
The report appears in your Crystal report viewer. It contains only data that is based on the parameter values that you selected.

## Running a report with null parameter values

When you run a report with null parameter values, upon execution, no values are passed to the query; the values that are used to run the report are either the preset default values or the preset personalized values of the variables. (Personalized values override default values.) If a variable has neither a default value nor a personalized value, then the report will attempt to run without a value for the variable. Depending on your query, you may receive an error if a value for the variable is necessary for the report to run.

**Note:**

- This feature is primarily used when scheduling reports. For more information, see [Null parameter values in scheduled reports](#) on page 152.
- This option is only available for BW Query based parameters.

**To run a report with null parameter values**

1. Log on to BusinessObjectsInfoView.
2. In the **My Groups** folder of BusinessObjectsInfoView, navigate to the object that contains variables.
3. Click the object's title or click **View Latest Instance**.  
The "Enter Prompt Values" dialog box appears.
4. Ensure that the **Set to null** checkbox under each parameter is selected.
5. Click **OK**.

The report appears in your Crystal report viewer. It contains only data that is based on the default/personalized value that was originally assigned to the variables in SAP.

**Personalizing parameter values**

Personalization is a feature that enables you to set your own default value for a parameter and save it for future use. When you set a personalized value for a parameter, the personalized value effectively becomes your new default value for the parameter.


**Note:**

- This option is only available for BW Query based parameters and only applies to reports that are based on BW Query or MDX driver.
- Personalized values are user-specific—that is, the personalized values that users set for themselves are not implemented for other users. Other users who work with the same report can set their own personalized values or use any of the preset values that are given for the parameter.

**To run a report with personalized parameter values**

1. Log on to BusinessObjectsInfoView.
2. In the **My Groups** folder of BusinessObjectsInfoView, navigate to the object that contains variables.
3. Click the object's title.

The "Enter Prompt Values" dialog box appears. The default values for the report parameters are each listed beside "Current value."

4. You can personalize a value for a parameter in two ways:
  -  Select a value from the list and click the **Personalize** icon to set it as your personalized value.
  - Click the parameter's edit field, type a value, and then click the **Personalize** icon to set it as your personalized value.

You can then view the report with data that is based on the personalized value of your choice. If, in the future, you decide to view the same report, the personalized value is used to run the report, unless otherwise specified.

For more information about personalization, see the documentation for your SAP BW system or contact your system administrator.

## Scheduling reports that contain parameters

Aside from enabling users to view reports that prompt for parameter (variable) values, BusinessObjects Integration for SAP Solutions gives users the ability to schedule and distribute these reports over the web. Depending on your needs, you can schedule the report to run once or on a recurring basis.

### To schedule reports that contain parameters

1. Navigate to the object you want to schedule.
2. Click the **Schedule** link for the object.

The Schedule page appears.

3. Select **Parameters** from the list of options.  
 Provide the parameter values/settings for your report.
4. Select and specify other scheduling options you want to use.

#### Tip:

For details about the options available to you, see "Scheduling objects" in the *InfoView Online Help*.

**5. Click **Schedule**.**

The report will now run according to the schedule that you defined.

## **Personalized parameter values in scheduled reports**

When you set a recurring schedule for a report that contains a personalized parameter value, BusinessObjects Enterprise will use the personalized value to generate the report for all the times that it is scheduled to run. If you change the personalized value of the parameter, the scheduled report will still continue to display data that is based on the original personalized value even though it has been changed. This occurs because, upon personalizing a value, BusinessObjects Enterprise saves the value and sets it as the permanent user-specific parameter value for the report. To change the personalized value of the parameter, you need to do one of the following:

- Reschedule your report with the new parameter value of your choice.
- Schedule the report to run with null values for the parameter. This ensures that personalization values are evaluated when the report runs.

## **Null parameter values in scheduled reports**

When you schedule a report to run with null as a parameter value, upon execution, no value is passed to the query; the value that is used to run the report is either the preset default value or the preset personalized value of the variable (Personalized values override default values.) If you decide to change the parameter value that is being used to run the report, in BusinessObjects Enterprise, you can click the report and reschedule it with your new parameter. Because the report was initially scheduled to run with a null parameter value, no value is saved with the report. As a result, when the report is once again run, the new parameter value is used to generate the data.





More Information



A

appendix



Information Resource	Location
SAP BusinessObjects product information	<a href="http://www.sap.com">http://www.sap.com</a>
SAP Help Portal	<p>Select <a href="http://help.sap.com">http://help.sap.com</a> &gt; SAP BusinessObjects.</p> <p>You can access the most up-to-date documentation covering all SAP BusinessObjects products and their deployment at the SAP Help Portal. You can download PDF versions or installable HTML libraries.</p> <p>Certain guides are stored on the SAP Service Marketplace and are not available from the SAP Help Portal. These guides are listed on the Help Portal accompanied by a link to the SAP Service Marketplace. Customers with a maintenance agreement have an authorized user ID to access this site. To obtain an ID, contact your customer support representative.</p>
SAP Service Marketplace	<p><a href="http://service.sap.com/bosap-support">http://service.sap.com/bosap-support</a> &gt; Documentation</p> <ul style="list-style-type: none"> <li>• Installation guides: <a href="https://service.sap.com/bosap-instguides">https://service.sap.com/bosap-instguides</a></li> <li>• Release notes: <a href="http://service.sap.com/releasenotes">http://service.sap.com/releasenotes</a></li> </ul> <p>The SAP Service Marketplace stores certain installation guides, upgrade and migration guides, deployment guides, release notes and Supported Platforms documents. Customers with a maintenance agreement have an authorized user ID to access this site. Contact your customer support representative to obtain an ID. If you are redirected to the SAP Service Marketplace from the SAP Help Portal, use the menu in the navigation pane on the left to locate the category containing the documentation you want to access.</p>
Developer resources	<p><a href="https://boc.sdn.sap.com/">https://boc.sdn.sap.com/</a></p> <p><a href="https://www.sdn.sap.com/irj/sdn/businessobjects-sdklibrary">https://www.sdn.sap.com/irj/sdn/businessobjects-sdklibrary</a></p>

Information Resource	Location
SAP BusinessObjects articles on the SAP Community Network	<a href="https://www.sdn.sap.com/irj/boc/businessobjects-articles">https://www.sdn.sap.com/irj/boc/businessobjects-articles</a> These articles were formerly known as technical papers.
Notes	<a href="https://service.sap.com/notes">https://service.sap.com/notes</a> These notes were formerly known as Knowledge Base articles.
Forums on the SAP Community Network	<a href="https://www.sdn.sap.com/irj/scn/forums">https://www.sdn.sap.com/irj/scn/forums</a>
Training	<a href="http://www.sap.com/services/education">http://www.sap.com/services/education</a> From traditional classroom learning to targeted e-learning seminars, we can offer a training package to suit your learning needs and preferred learning style.
Online customer support	<a href="http://service.sap.com/bosap-support">http://service.sap.com/bosap-support</a> The SAP Support Portal contains information about Customer Support programs and services. It also has links to a wide range of technical information and downloads. Customers with a maintenance agreement have an authorized user ID to access this site. To obtain an ID, contact your customer support representative.
Consulting	<a href="http://www.sap.com/services/bysubject/businessobjectsc consulting">http://www.sap.com/services/bysubject/businessobjectsc consulting</a> Consultants can accompany you from the initial analysis stage to the delivery of your deployment project. Expertise is available in topics such as relational and multidimensional databases, connectivity, database design tools, and customized embedding technology.



# Index

## A

- ABAP data clusters 108
  - advanced topics 108
  - and the Cluster Definition tool 121
  - and the Open SQL driver 108
  - described 109, 121
  - mapping as tables 121, 122
  - reporting off 121
  - selecting for reports 109
- ABAP functions 108
  - advanced topics 108
  - and the Open SQL driver 108
  - calculations and functions 114
  - described 109, 114
  - examples 115
  - selecting for reports 114
- ABAP queries 139
  - and parameters 139
  - and subreports 139
- advanced reporting topics 108
- Automatically publish to BusinessObjects Enterprise 16

## B

- batch publishing 94, 96
- Business Explorer (BEx) Query Designer 14
- BusinessObjects Enterprise 147
  - personalization in 147
  - publishing reports to 94
  - scheduling reports 98
  - viewing reports through 96
- BW queries 14
  - and formatted reporting 14, 54

- BW queries (*continued*)
  - connection settings 14, 55
  - creating for reports 19, 67
  - default report title 25, 65
  - dynamic pick lists 14, 54
  - field descriptions and technical names 60
  - selecting for reports 23, 62
  - with hierarchy 43, 72
  - with hierarchy node variable 78
  - with hierarchy variable 79
- BW Query driver, connection settings 14, 55
- BW query, design with hierarchy 38, 72
- BW, publishing reports from 96

## C

- calculations, and ABAP functions 114
- client field 113
- Cluster Definition tool 121
  - tutorial 122
  - using 122
- cluster tables 109
  - selecting for reports 109
- clusters. See ABAP data clusters 121
- commands, in Crystal Reports 14, 55
- conditional formatting 91
- configuration options 14, 55, 102
- connecting 99
  - to the SAP system 99
  - with the BW Query driver 23
- connection settings 14, 55
- creating 19
  - data clusters 122
  - queries 19, 67, 72, 76
  - reports 98

## Index

creating (*continued*)  
reports to use as templates 27

## D

data access 98  
connecting with the Data Access  
Components 99  
Data Access Components 98  
connecting to SAP 99  
data clusters. See ABAP data clusters 121  
Data Explorer 23  
data sources 98, 108  
ABAP data clusters 108  
ABAP functions 108  
BW queries 14  
creating queries 19  
cubes 62  
ODS 85  
Operational Data Store 85  
selecting BW queries 23  
transparent, pool, and cluster tables 108  
views 108  
databases, logical 139  
default templates 27  
default values 148  
designing reports 98  
dictionaries of data clusters 121  
Display database field by, configuration options  
102  
drivers 98  
BW Query 14  
connecting to SAP 99  
Data Access Components 98  
MDX Query 53  
Open SQL driver 108  
Operational Data Store 85  
selecting for reports 99  
dynamic pick lists 14, 54

## E

examples, ABAP functions 115

## F

field descriptions 60  
formatted reporting 14, 54  
formatting  
translated reports 91  
with the LanguageCode% formula 91  
Formula Functions 50  
GroupingLevel 50  
HierarchyLevel 50  
formulas, LanguageCode% 91  
functions. See also ABAP functions 109

## G

GroupingLevel function 50  
Groups, My 146

## H

hierarchies 72  
creating in BW query 72  
in BW queries 71  
in BW query 43  
reporting off 73  
reporting on 44  
with hierarchy node variable 76, 78  
Hierarchy Node Variable 76  
building report from 76, 78  
creating query with 76  
Hierarchy Variable 79  
building report from 79, 82  
creating 79  
HierarchyLevel function 50  
Human Resources modules, and data clusters  
121

**I**

Import Wizard 94  
 importing 94  
     BW content 94  
 InfoSet driver, selecting data sources 140  
 InfoSets 139  
     and parameters 139  
     and subreports 139  
     selecting for reports 140  
 InfoView 104  
 input parameters, and ABAP functions 114  
 installation 8  
 iViews  
     saving reports as 25

**L**

language field 114  
 LanguageCode% formula 91  
 languages 90  
     translated reports 90  
     viewing reports in 96  
 limitations, and ABAP functions 115  
 linking 113  
     client field 113  
     language field 114  
     tables and views 113  
 List tables and fields by, configuration options 103  
 Log On Server command 100  
 logging on to SAP 101  
 logical databases 139  
 logon dialog box 101

**M**

MANDT field 113  
 mapping ABAP data clusters 122  
 MDX Query driver 53  
     connection settings 55  
 migrating reporting content in BW 94

multidimensional data, reporting off 53  
 Multiple Structures 65  
 My Groups 146  
 My Roles. See My Groups 146

**N**

naming, report titles 25, 65  
 null values 149

**O**

ODS driver 85  
     overview 85  
     selecting data source 85  
 Open SQL driver 108  
     and ABAP functions 114  
     and the Cluster Definition tool 121  
     objects supported by 108  
     overview 108  
     selecting data sources 109  
 Operational Data Store driver 85  
     selecting data source 85  
 options, configuring Crystal Reports 14, 55, 60, 102

**P**

parameters 82, 139  
     and ABAP functions 114  
     and ABAP queries 139  
     and InfoSets 139  
     and personalization 150  
     and variables in BW queries 147  
     choosing from dynamic picklists 149  
     default values 148  
     manually creating default values 82  
     null values 149  
     scheduling reports with 151  
 personalization 147  
 pick lists 149

## Index

- pool tables 109
  - selecting for reports 109
- pre-processing, ABAP functions 114
- predefined reports
  - publishing 94
- Prepare reports for translation on server 16
- Prepare this report for translation 26, 65
- publishing 94
  - from BW 96
  - in batches 94
  - predefined reports 94
- Publishing
  - and SAP integration 146
- Publishing Wizard 94

## Q

- queries 19
  - creating 19
  - in SAP BW 14
  - selecting for reports 23
  - with hierarchy 43
- query, design with hierarchy 38, 72

## R

- record selection
  - on client field (MANDT) 113
  - on language field (SPRAS) 114
- report designer 99
  - connecting to SAP 99
  - logging on to SAP 101
- report templates 27
- Report Wizards
  - selecting drivers 99
- reporting 14, 108
  - advanced topics 108
  - off ABAP data clusters 121
  - off ABAP functions 114
  - off BW queries 14, 66
  - off tables, views, clusters, and functions 108

- reports 98
  - creating
    - off BW hierarchies 38, 71
    - off hierarchy node variables 76
    - off hierarchy variables 79
    - off Operational Data Stores 85
    - off SAP data 98
  - migrating with BW content 94
  - off the SAP DemoCube 28
  - preparing for translation 90
  - publishing 94
  - reporting off BW's multidimensional data 53
  - saving to BW 25, 65
  - scheduling 98, 151
  - selecting
    - a driver 99
    - an SAP system 101
    - queries for 23
  - titles 25, 65
  - translating 90
  - tutorial on creating 28, 66, 71, 76, 79
  - viewing 96, 104
- return types, using wrapper functions 115
- Roles, My. See Groups, My 146

## S

- sample reports 98
- SAP BW 14
  - connection settings 14, 55
  - migrating systems 94
  - publishing reports from 94
  - saving reports to 25
- SAP DemoCube, reporting off 28
- SAP Easy Access 96
- SAP InfoView 96, 104
- SAP parameters 76, 82
  - Hierarchy Node variable 76
  - Hierarchy variable 82
- SAP system 99
  - connecting with the drivers 99
  - logging on with the drivers 101



- SAP System Logon dialog box 101
- SAP Tools toolbar 14, 55
- saving reports to BW 25
- scheduling reports 98, 151
- selecting 23, 62
  - ABAP functions 114
  - ABAP queries 140
  - an SAP system 101
  - BW queries 23, 62
  - cubes 62
  - drivers for reports 99
  - InfoSets 140
  - reports as templates 27
  - tables, views, functions, and clusters 109
- server, log on command 100
- settings, for connecting to BW 14, 55
- smart linking 113
- SPRAS field 114
- Suppress save dialog 16

## T

- tables 113
  - linking 113
  - selecting for reports 108
- technical field names 60
- templates 27
  - to create a report using 27
- titles, of reports 25, 65
- toolbar 14, 55
- transaction ZCDD, Cluster Definition tool 122
- translated reports 90
  - formatting 91
  - preparing 90
  - viewing 96

- transparent tables 109
  - described 109
  - selecting for reports 109
- troubleshooting 115
  - ABAP functions 115
  - configuration options 14, 55, 102
- tutorial 38, 66, 76, 79
  - BW Query driver
    - building hierarchical report 38
    - building simple report 28
  - MDX Query driver
    - building hierarchical report 71
    - building report with hierarchy node variable 76
    - building report with hierarchy variable 79
    - building simple report 66

## V

- variable
  - picklist, creating in Crystal Reports 82
- viewing reports 96, 104
- views 109
  - in SAP, described 109
  - linking 113
  - reporting off 108

## W

- wizards, report 99
- wrapper functions, defining return types 115

## Z

- ZCDD, Cluster Definition tool 122

## Index