BW306

Enterprise Reporting, Query and Analysis (Part II)

SAP NetWeaver

Date

Training Center

Instructors

Education Website

Participant Handbook

Course Version: 2006 Q2
Course Duration: 5 Day(s)
Material Number: 50077636

SAP®

An SAP course - use it to learn, reference it for work

Copyright

Copyright © 2007 SAP AG. All rights reserved.

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

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

Trademarks

- Microsoft®, WINDOWS®, NT®, EXCEL®, Word®, PowerPoint® and SQL Server® are registered trademarks of Microsoft Corporation.
- IBM®, DB2®, OS/2®, DB2/6000®, Parallel Sysplex®, MVS/ESA®, RS/6000®, AIX®, S/390®, AS/400®, OS/390®, and OS/400® are registered trademarks of IBM Corporation.
- ORACLE® is a registered trademark of ORACLE Corporation.
- INFORMIX®-OnLine for SAP and INFORMIX® Dynamic ServerTM are registered trademarks of Informix Software Incorporated.
- UNIX®, X/Open®, OSF/1®, and Motif® are registered trademarks of the Open Group.
- Citrix®, the Citrix logo, ICA®, Program Neighborhood®, MetaFrame®, WinFrame®, VideoFrame®, MultiWin® and other Citrix product names referenced herein are trademarks of Citrix Systems, Inc.
- HTML, DHTML, XML, XHTML are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.
- JAVA® is a registered trademark of Sun Microsystems, Inc.
- JAVASCRIPT® is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.
- SAP, SAP Logo, R/2, RIVA, R/3, SAP ArchiveLink, SAP Business Workflow, WebFlow, SAP EarlyWatch, BAPI, SAPPHIRE, Management Cockpit, mySAP.com Logo and mySAP.com are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other products mentioned are trademarks or registered trademarks of their respective companies.

Disclaimer

THESE MATERIALS ARE PROVIDED BY SAP ON AN "AS IS" BASIS, AND SAP EXPRESSLY DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR APPLIED, INCLUDING WITHOUT LIMITATION WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WITH RESPECT TO THESE MATERIALS AND THE SERVICE, INFORMATION, TEXT, GRAPHICS, LINKS, OR ANY OTHER MATERIALS AND PRODUCTS CONTAINED HEREIN. IN NO EVENT SHALL SAP BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR PUNITIVE DAMAGES OF ANY KIND WHATSOEVER, INCLUDING WITHOUT LIMITATION LOST REVENUES OR LOST PROFITS, WHICH MAY RESULT FROM THE USE OF THESE MATERIALS OR INCLUDED SOFTWARE COMPONENTS.

g2007313103842

About This Handbook

This handbook is intended to complement the instructor-led presentation of this course, and serve as a source of reference. It is not suitable for self-study.

Typographic Conventions

American English is the standard used in this handbook. The following typographic conventions are also used.

Type Style	Description
Example text	Words or characters that appear on the screen. These include field names, screen titles, pushbuttons as well as menu names, paths, and options.
	Also used for cross-references to other documentation both internal (in this documentation) and external (in other locations, such as SAPNet).
Example text	Emphasized words or phrases in body text, titles of graphics, and tables
EXAMPLE TEXT	Names of elements in the system. These include report names, program names, transaction codes, table names, and individual key words of a programming language, when surrounded by body text, for example SELECT and INCLUDE.
`Example text`	Screen output. This includes file and directory names and their paths, messages, names of variables and parameters, and passages of the source text of a program.
`Example text`	Exact user entry. These are words and characters that you enter in the system exactly as they appear in the documentation.
`<Example text>`	Variable user entry. Pointed brackets indicate that you replace these words and characters with appropriate entries.

 SAP

Icons in Body Text

The following icons are used in this handbook.

Icon	Meaning
	For more information, tips, or background
	Note or further explanation of previous point
	Exception or caution
	Procedures
	Indicates that the item is displayed in the instructor's presentation.

Contents

Course Overview

This courses focuses on using the design components of the NetWeaver 2004s BEx Suite and the xApps Visual Composer.

Target Audience

This course is intended for the following audiences:

- SAP BI Project Team Members
- SAP BI Consultants

Course Prerequisites

Required Knowledge

- Basic reporting skills using NetWeaver 2004s BI Business Explorer
- Ideally, participant will have attended BW305 SAP BI – Enterprise Reporting, Query & Analysis (Part I)

Recommended Knowledge

- Knowledge of backend topics of NetWeaver 2004s BI
- Web development knowledge is helpful but not essential.

Course Goals

This course will prepare you to:

- Develop sophisticated reports in the Business Explorer Suite by using advanced features and functions of the reporting components.
- Publish BI reports and analyses to the NetWeaver Portal.

Course Objectives

After completing this course, you will be able to:

- Describe the purpose and outline the key features of each component of the BEx Suite and also xApps Visual Composer.
- Design and build customized workbooks using the Design Mode of the BEx Analyzer and also use native Excel formulas to integrate Excel functionality with BI functions.
- Develop highly formatted reports for displaying on the Web and also for printing using the new Report Designer.

- Develop web applications that use the full range of features of the Web Application Designer.
- Expand your usage of the BEx Broadcaster to distribute NetWeaver BI reports.
- Publish a variety of BI reports to the NetWeaver Portal and organize them efficiently using iViews, pages and folders with portal roles.
- Put into practice the skills developed in this class to develop a comprehensive reporting solution based on detailed business requirements.
- Explain the possibilities presented with the tight integration of the xApps Visual Composer with NetWeaver BI.
- Discuss the issues with migrating reporting objects between BW 3.x and NetWeaver 2004s BI.

SAP Software Component Information

The information in this course pertains to the following SAP Software Components and releases:

Unit 1

NetWeaver 2004s BI Reporting Tools Overview

Unit Overview

The BEx Suite in NetWeaver 2004s BI provides a broad collection of tools and services for creating and using the data stored in the data warehouse. In this unit we will review the components of the BEx Suite.

Unit Objectives

After completing this unit, you will be able to:

- Describe the purpose and outline the key features of each component of the BEx Suite
- Describe the purpose and outline the key features of the xApps Visual Composer

Unit Contents

Lesson: Business Explorer Tools and Visual Composer

Lesson Overview

This lesson provides an overview of the BEx tools and the xApps Visual Composer.

Lesson Objectives

After completing this lesson, you will be able to:

- Describe the purpose and outline the key features of each component of the BEx Suite
- Describe the purpose and outline the key features of the xApps Visual Composer

Business Example

Your organization has asked you provide a high level overview of the BI tools and also they want to know how Visual Composer fits into the picture.

Business Explorer Suite
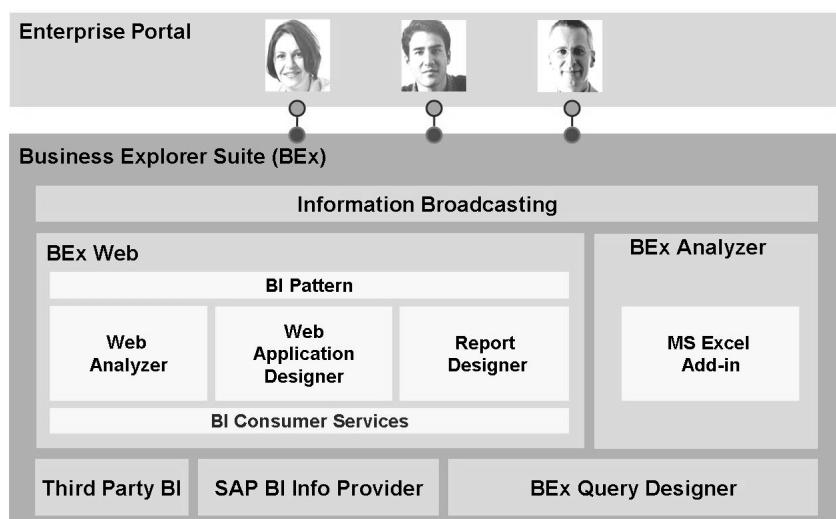
Figure 1: Business Explorer Suite

The SAP NetWeaver Business Intelligence Suite, the Business Explorer (BEx), provides flexible reporting and analysis tools for strategic analyses and decision-making support within a business. These tools include query, reporting, and analysis functions. As an employee with access authorization, you can evaluate historical or current data at various levels of detail and from different perspectives, not only on the Web but also on the portal and in Microsoft Excel.

You can also use the Business Explorer tools to create planning applications and for planning and data entry within BI Integrated Planning. You can use BEx Information Broadcasting to distribute business intelligence content by e-mail, either as precalculated documents with historical data, or as links **to** live data. You can also publish this content on the portal (in Knowledge Management folders or collaboration rooms).

SAP NetWeaver 2004s BI provides two versions of the following tools:

- BEx Query Designer
- BEx Web Application Designer
- BEx Broadcaster
- BEx Analyzer

One version of the above tools is based on BW 3.x,technology, the other version is based on Netweaver 2004s BI technology. SAP supplies both versions of the tools in NetWeaver 2004s BI so that customers can continue to maintain their existing reporting objects using the older tools. Customers are not forced to migrate their reporting objects just because they have upgraded their BW 3.x system to Netweaver 2004s BI. It is possible to migrate all BW 3.x objects to Netweaver 2004s BI objects.as required, this migration is done automatically when the old 3.x object is opened and saved in the new 2004s tools.

The Report Designer is brand new for NetWeaver 2004s., so there is no 3.x version.

The diagram above describes the architecture of the BEx tools. Notice how the Query Designer plays a fundamental role by the way it sits beneath all the other tools. This is because the Query Designer is used to build the queries that the other tools will then use in the development of BI applications. You will also notice that the Web Application Designer, the Report Designer and the Web Analyzer are all part of a family called BEx Web. This, of course, is because all these tools are used to create output for the web.

The Analyzer is based on Microsoft Excel and utilizes an add in component developed by SAP to provide BI connectivity and OLAP functionality. The BEx Analyzer is not part of the BEx Web.

Sitting at the top of the Business Explorer area of the diagram is the Information Broadcaster. This has been positioned in the diagram deliberately to sit above all the tools because the Broadcaster can take the output from any of the BEx tools and distribute it according to the rules you supply.

Finally you see the Portal included at the very top of the diagram. The portal provides the desktop for the display and interaction with any BI report.

> **Note:** The Portal is a **mandatory** component for displaying any web output developed using NetWeaver 2004s BI tools.

Query Designer

Figure 2: BEx Query Designer

The Query Designer is used to develop and maintain an extremely important reporting object within BI, namely the **query**. Within a query you define:

- The source of the BI data (InfoProvider)
- Required characteristics and key figures (InfoObjects)
- Additional run-time calculations
- Filters for the selection of required data
- Rules for exceptions and conditions
- Default presentation settings

Once the query is built it can be used to pass BI data to a number of BEx reporting tools such as the Analyzer, Web Application Designer, Report Designer, Information Broadcaster and Web Analyzer. Although the Query Designer can be accessed stand alone from it's own menu option, it can also be accessed using a short-cut link from most of the BEx tools such as the Web Application Designer, Analyzer and Report Designer.

It is important to remember that a single query can be re-used in many different front end applications. As an example, Query A produces a list of the top 10 customers for the month based on profitability. The result of this query need to be seen in both a Sales Management Dashboard (which you may have built using the Web Application Designer) and also the same results need to be shown in a formatted report (built using the Report Designer). So queries are re-useable objects; in fact you should be planning your query development carefully so that you enjoy maximum re-use of the queries in your NetWeaver BI environment.

This approach will certainly reduce the maintenance effort when changes need to be implemented to these reports.

> **Note:** BEx Query Designer skills are taught in detail when you attend the class BW305 and also TBW20.

BEx Analyzer
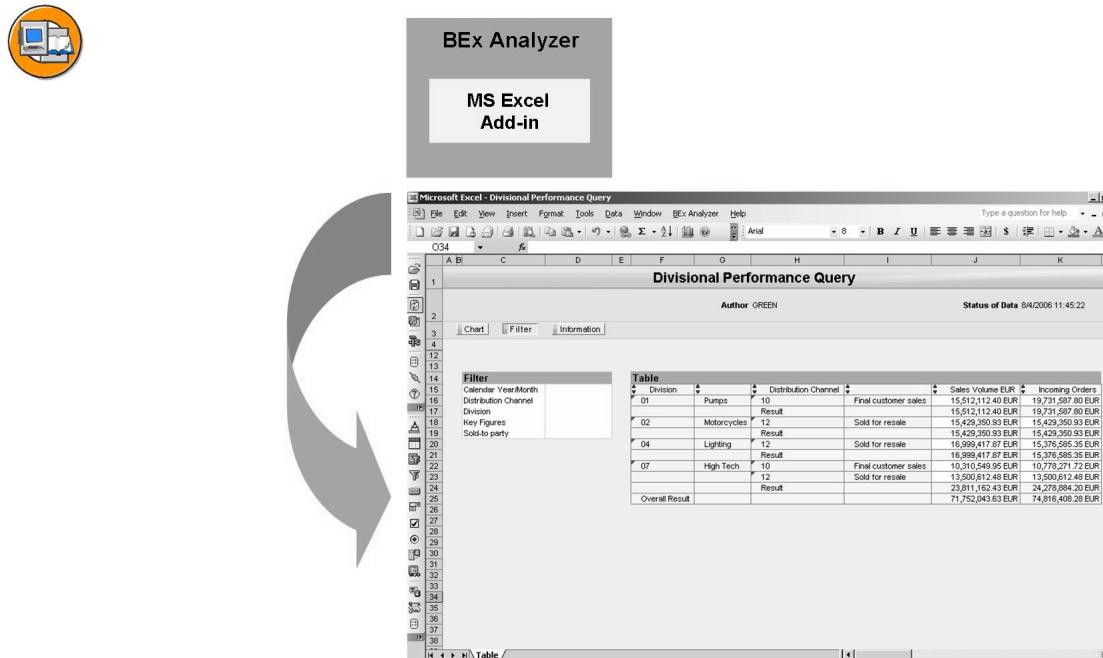
Figure 3: BEx Analyzer

The BEx Analyzer is used to present query data to the users in a Microsoft Excel workbook. As well as a wide range of analysis tools provided by SAP NetWeaver BI (drill down, swap characteristics, currency translation etc) the Analyzer also offers the user the full functionality of Microsoft Excel. So for example, the user may decide to add some conditional formatting and cell formatting (standard Excel functions) to the report results to highlight weaknesses in sales performance. Using the Analyzer the user can navigate through the report result using either intuitive drag and drop actions or a comprehensive context menu or a combination of the two methods.

The BEx Analyzer has two modes.

- Analysis Mode
- Design Mode

As the name suggests **analysis mode** allows the user to work with the report results analyzing the business data by making use of any navigation features supplied by the developer.

> **Note:** The analysis mode of the Analyzer is covered in BW305 and TBW20.

The **design mode** puts the workbook into a mode where the developer creates customized Excel based reports which integrate results from different BEx queries. The design mode also provide the developer with a wide choice of features aimed at improving the interaction experience of the user, such as drop down lists, radio buttons and push buttons.

> **Caution:** Be careful not to confuse the BEx Analyzer with the BEx Web Analyzer, they are not the same thing.

Web Application Designer
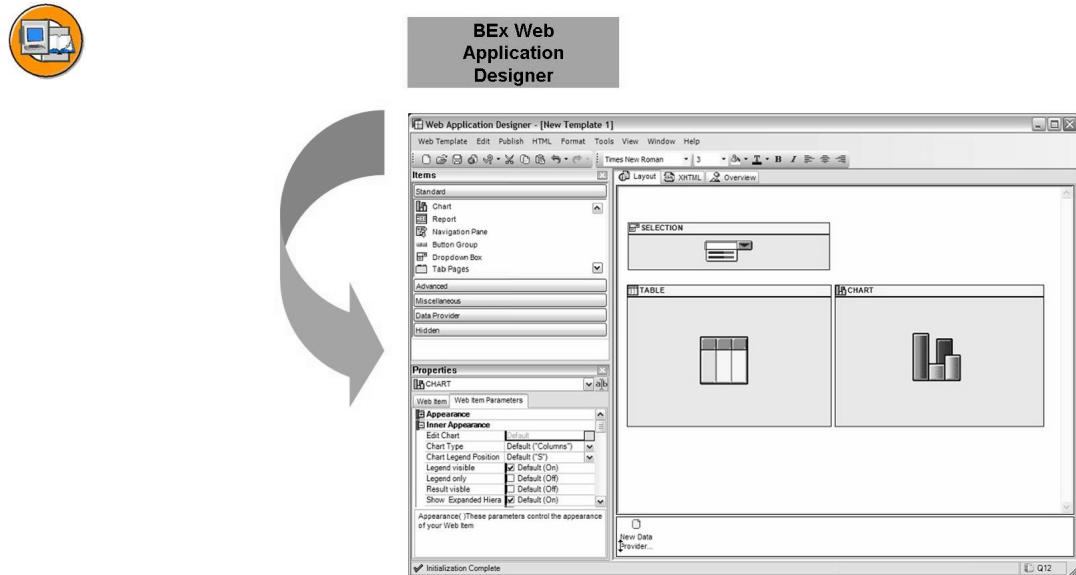
Figure 4: BEx Web Application Designer

With the BEx Web Application Designer you are able to produce customized web based business reports, called 'web applications' A typical web application is made up from a number of web items such as result tables perhaps presented across tabs, charts, buttons, drop down lists etc.

> **Hint:** SAP provides a ready made default web template for the output of any simple web based reports. This template is called the Web Analyzer. See the section later for more information on this.

The Web Application Designer allows the developer to create his own customized web templates, using a flexible, easy to use tool kit. The Web Application Designer uses standard web based technologies such as XHTML, XML, XPATH and Javascript, which means you can integrate NetWeaver BI web applications with any external (non SAP) web applications. You can also decide to enhance your NetWeaver BI web applications with your own custom code although it must be emphasized that the BI Web Application Designer does not require any previous knowledge of web technologies in order to create powerful web applications.

The Web Application Designer offers access to a large number of Web Design API's. This means that developers can create custom commands for scenario specific BI web applications. For example, you could build a menu with a set of predefined navigation filters to the query results or you could perhaps create a push button which sends a command to modify settings of the chart so that you could switch the bar chart to a pie chart. For less experienced developers the Web Application Designer provides a wizard to guide them through the key steps in building a web template.

BI Patterns

The Netweaver BI Pattern Wizard is embedded within the Web Application Designer and makes creating web applications easy and quick by providing ready made pattern templates as a starting point for generating your own web applications. The templates are predefined by SAP but can be adjusted very easily using simple wizard based prompts to create scenario specific web applications.

Using the pattern wizard ensures a uniform design of web reports. This means that the users always find the same functionality in the same place in all web reports. It also means you can produce web applications without having to learn all the features of the Web Application Designer as the templates and the web items they contain have already been put in place by SAP. To utilize templates simply supply the missing information to complete them and decide which features of the template you would like to use. There are different patterns available to suit the needs of the reporting community. There are patterns which are used to develop simple web applications aimed at casual users, and there are patterns which offer more features for the expert user.

Note: A web template created with the pattern wizard cannot be edited in the Web Application Designer. To modify a pattern based template you can simply run the pattern wizard again for the template and make adjustments.

Web Analyzer
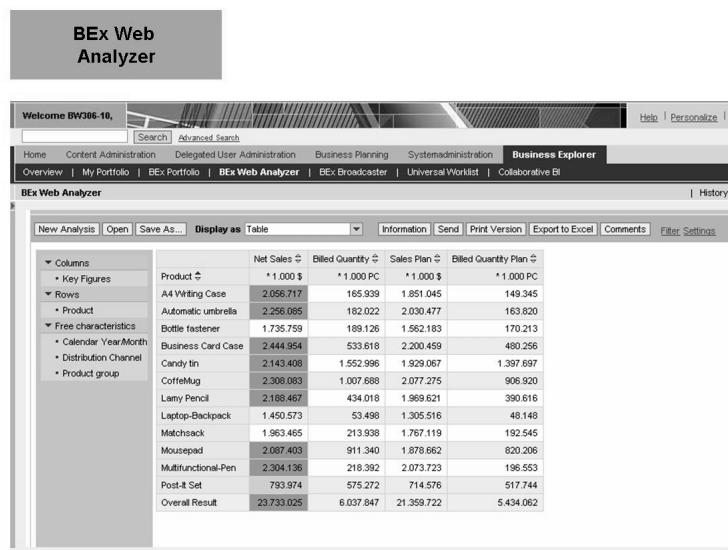
Figure 5: BEx Web Analyzer

The BEx Web Analyzer is a predefined web template developed and supplied by SAP and is used to present your query results on the web. This template contains a large number of useful web items such as drop down lists for each characteristic and buttons to send the report results printer or to generate a PDF. Although the Web Analyzer is not complex, it is typically used by power users as it contains more features and functions than many casual users would require. The BEx Analyzer allows a user to select a query, query view or even an entire InfoProvider as the source of the analysis.

> **Hint:** The Web Analyzer is a standard web template offering standard navigation controls. It is designed with power users in mind who want to do ad-hoc analysis of business data. Implementing the Web Analyzer is a great way to introduce users to web applications without the need for time consuming web template development using the Web Application Designer.

It is possible to modify the standard web template used by the Web Analyzer and add your own features or remove any items you don't want to make available to users. Power users can develop their own analysis using the Web Analyzer, they simply need to specify the dataprovider which will source the data, this could be a query, query view or even an infoprovider. The user then drags and drops various characteristics and key figures into the analysis to create a custom report. The

analysis results can be saved for later reuse. The Web Analyzer is typically used for ad-hoc analysis of a specific, single result set. It is not a cockpit or dashboard type web application (for that you would use the Web Application Designer).

Hint: An example of how the Web Analyzer could be used:

A developer creates a generic query for the procurement team. This is also known as a 'fat' query; there are very few filters on the data, perhaps just a filter on the current year and purchasing organization. The procurement team then use the Web Analyzer to execute this query, they add some extra filters and modify the output settings, perhaps adding a chart. They can then save their analysis and re-call this later or share the result with colleagues by publishing the results to the portal. Then another procurement team member open the same query, drills down to see their own data, then save the results to their personal favorites area in the portal. This means that you have allowed the procurement team to build their own reports but you have restricted the amount of data first by defining a query which they must all use to select the data set.

Caution: Be sure to always use the full term **Web Analyzer** to avoid confusion with the **Analyzer** .

Report Designer
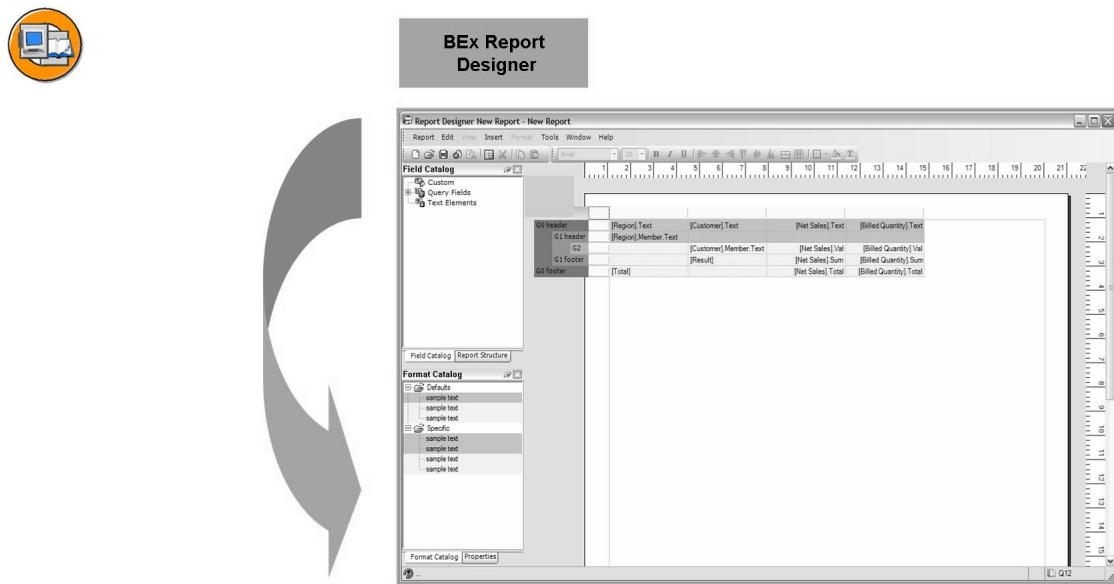
Figure 6: Report Designer

When presentation and formatting of the report results is the top priority then the BEx Report Designer should be used. The Report Designer is part of the BEx Web family of tools and allows you to take the results of a query and apply a variety of formatting techniques to dramatically improve the presentation. These techniques include using color to emphasize sections of the report or important characteristic values (ie, my team), adding images (logos, banners), changing the position of the result cells, or adding headers and footers. The formatted results can be displayed on the web, exported to a PDF or printed. Navigation within a formatted report is possible, although it is limited.

> **Note:** See later in Unit 3 for a discussion on the subject of limited navigation with formatted reports.

Formatted results can be broadcast using the BEx Broadcaster so you can be sure the top managers always receive the well formatted quarterly results report exactly on time and posted directly to each manager. The formatted reports can be integrated into web applications using a web item 'Report'. This means that you could use the layout and organization features of the Web Application Designer (such as tabs, drop down lists) to present a series of well formatted reports that are also printer friendly.

Broadcaster
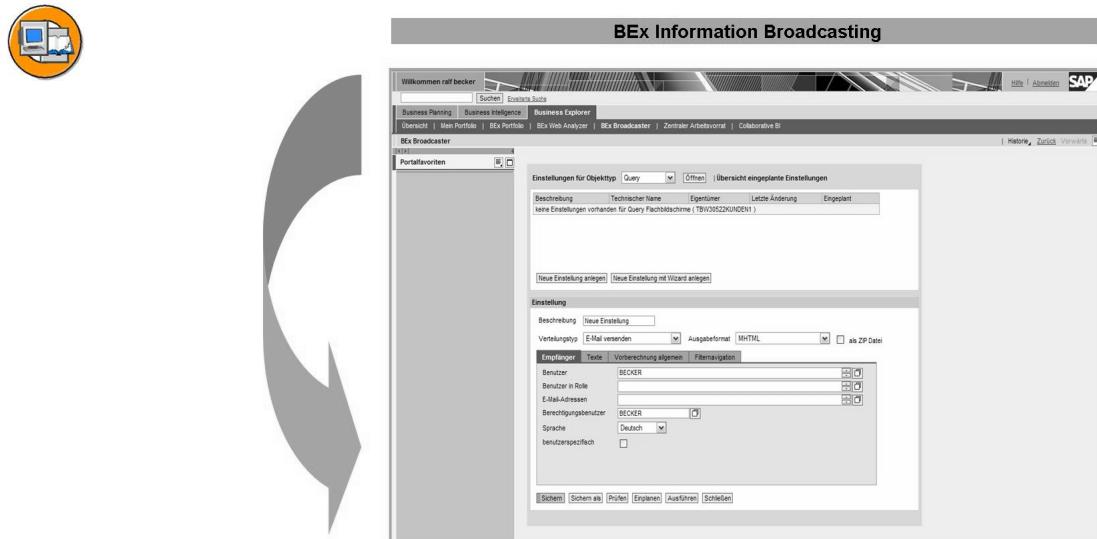
Figure 7: Broadcaster

With the BEx Broadcaster you define rules which determine how the results of any BI analysis can be distributed to a target audience. With the BEx Broadcaster you define broadcast settings to capture all the parameters of the broadcast. These parameters include the identity of the source of the data which includes queries, query views, web applications and workbooks. Also the parameters specify the

output type of the broadcast, this could include external email clients, NetWeaver Portal folders, printer, PDF. The target audience can be defined based on user ids, email addresses, user roles and also via the alert category assignments for exception analysis broadcasts. Broadcasts can be triggered at various time points. Broadcast can be sent immediately (perhaps as a one time event), sent once at a specified time, repeatedly sent at a regular time, only when there is new data to send, or only when there is an exception raised. You can decide if you want to send the precalculated results (like a snapshot) or whether you should send a link to the online data (so the user always sees the current data). Broadcasts can also play a part in improving the reporting performance of your NetWeaver BI. For example you could use the broadcaster to pre-fill the OLAP cache with commonly used business data, create precalculated stores of commonly used data, or to create precalculated reports and to store them in the KM folder accessible through the portal.

Visual Composer
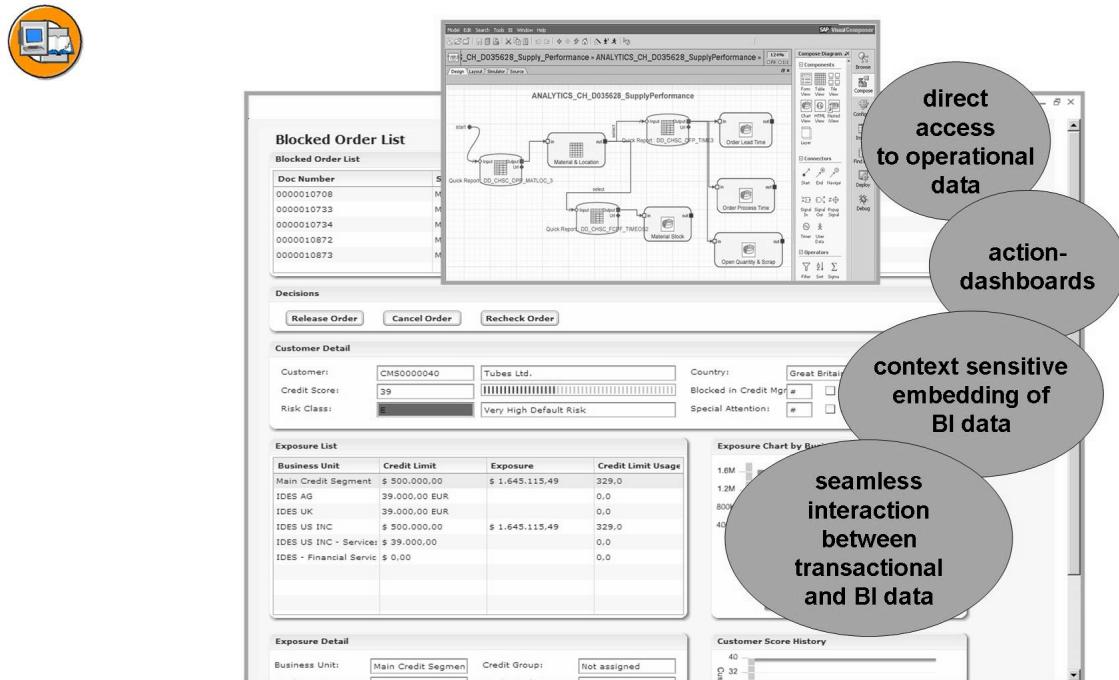
Figure 8: Visual Composer

With Visual Composer, SAP offers a powerful tool that enables business process experts to rapidly and easily model analytical applications and prototypes without writing a single line of code. Users of Visual Composer can quickly implement new and adapt existing analytical applications in order to support their companies'

ever-changing business processes. Visual Composer helps bridge the gap between the business process expert and IT, thus reducing the need for expensive consulting services, and lowering the TCO for the company.

The Visual Composer uses simple drag and drop techniques to build analytical applications. Analytical applications not only provide data for analysis but also provide the application interface used to **act on the analysis**. When deployed the Visual Composer turns your model into real code and publishes your application automatically to the NetWeaver Portal. Visual Composer focuses on the creation and adaptation of analytical and freestyle applications which are not restricted by system boundaries. It is a major step towards becoming the central modeling tool for creating and adapting composite applications as part of SAP's Enterprise SOA strategy.

> **Note:** Visual Composer is not part of Netweaver BI, in fact it is a component of xApps Analytics. However we include some basic coverage of this powerful component in this class simply because there is very tight integration between BI content and Visual Composer.

BI content (SAP supplied and your own) can be exposed to the Visual Composer via the BI Integration Wizard. This means that your analytical applications can easily use BI queries to present business data to the user as well as reports from transactional and non SAP systems.

Lesson Summary

You should now be able to:

- • Describe the purpose and outline the key features of each component of the BEx Suite
- • Describe the purpose and outline the key features of the xApps Visual Composer

Unit Summary

You should now be able to:

*   Describe the purpose and outline the key features of each component of the BEx Suite
*   Describe the purpose and outline the key features of the xApps Visual Composer

Related Information

For all the units in this course, additional information can be found at the following web sites:

*   SAP Help Portal – **http://help.sap.com**
*   SAP Service Marketplace – **http://service.sap.com/bi**
*   SAP Developer's Network – **http://sdn.sap.com**

Unit 2

BEx Analyzer for Business Experts

Unit Overview

With NetWeaver 2004s, the BEx Analyzer gained many new features, including the *Design Mode* toolbar and much tighter integration with native Excel formatting and formula functions.

Unit Objectives

After completing this unit, you will be able to:

- Design and build customized workbooks using the Design Mode of the BEx Analyzer.
- Use native Excel formula to integrate Excel functionality with BI functions

Unit Contents

Lesson: Bex Analyzer Design Mode

Lesson Overview

In this lesson we will introduce you to the Design Mode of the BEx Analyzer. Here you will learn how to create custom workbooks using configurable design item.

Lesson Objectives

After completing this lesson, you will be able to:

- Design and build customized workbooks using the Design Mode of the BEx Analyzer.

Business Example

Users are asking for more interaction and advanced features with the Analyzer workbooks. You need to learn more about using design items in the Analyzer Design Mode in order to create sophisticated workbooks.

Design Mode

Before we discuss the BEx Analyzer design mode lets take a look at what can be achieved by using this mode. You can see that the workbook below contains a number of useful interaction features such as check boxes, drop down lists, radio button and push buttons. The design mode of the BEx Analyzer lets the developer take control of the user interface to produce highly customized workbooks which meet the needs of the growing demands of the business user in terms of ease of use and advance capabilities.
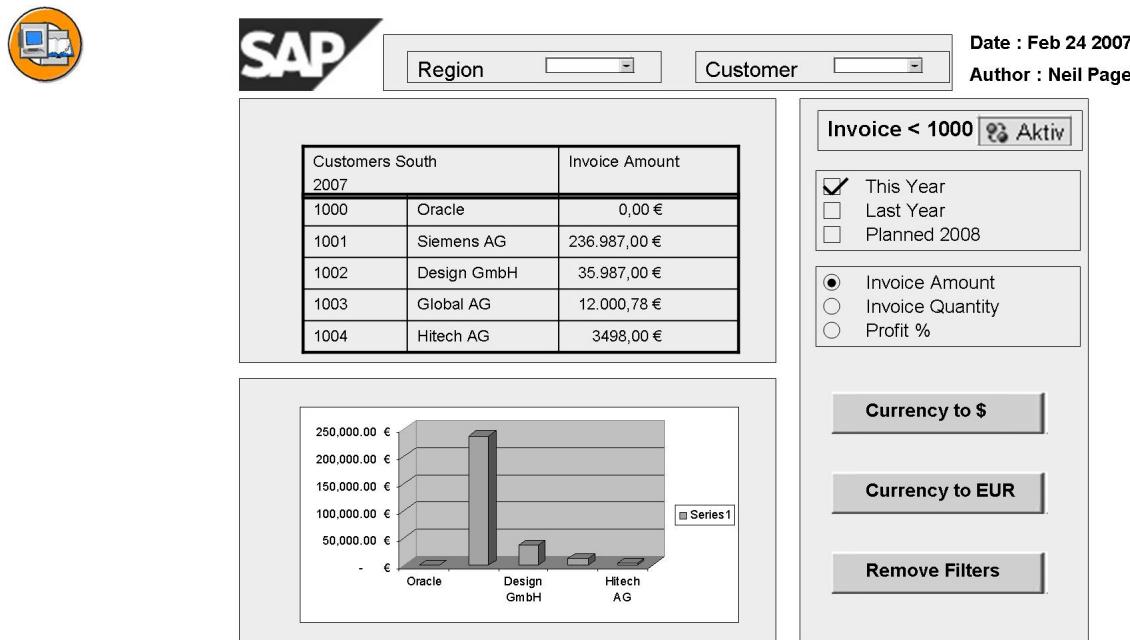
Figure 9: Example of a customized workbook

Let's start at the beginning. The BEx Analyzer has two modes, **analysis mode** and **design mode**.

Analysis mode is used for performing OLAP analysis on query results (the mode the user always uses). Design mode is used by the developer for designing the interface for query applications, or to put in a very simple way, analysis mode is the execution environment and the design mode is the development environment of the BEx Analyzer. In the BEx Analyzer you will find two toolbars each representing the functions of the two modes. Notice also the name of the toolbar is provided above the icons.
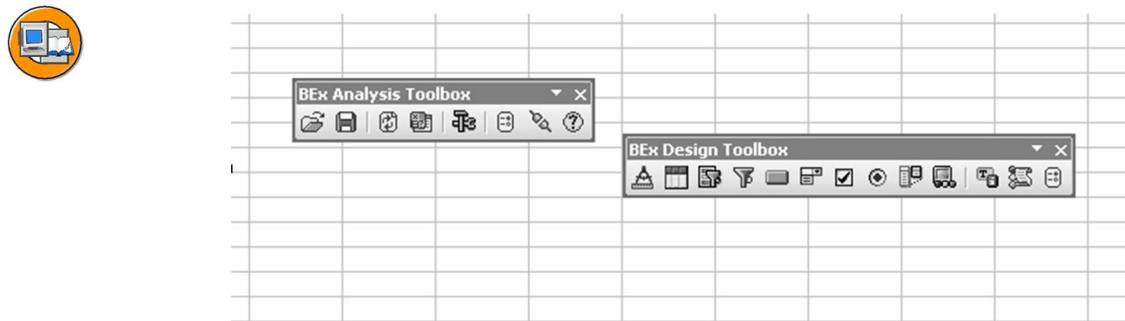
Figure 10: BEx Analyzer Toolbars - Analysis Mode vs. Design Mode

In this lesson we will focus on the design mode.

In BEx Analyzer's design mode, you design the interface for your query applications. A query itself, essentially a complex select statement, is an abstract object without visualization. Once you embed it into an Excel workbook and enter design mode, you add its visualization by designing its interface. In design mode, your workbook appears as a collection of design items represented by their respective icons.
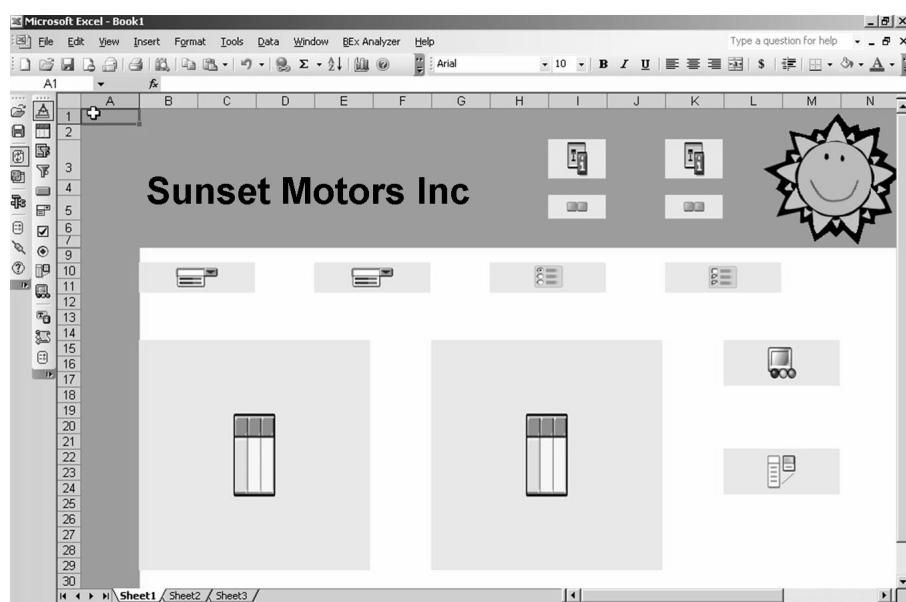
Figure 11: Design Mode showing layout of Design Items in a workbook

Once you switch to analysis mode, the results of your query appear according to how you have configured these design items. Thus, with design items you create an interface that dictates how you will analyze and navigate the results in analysis mode. Results of the query are displayed in the analysis grid design item, in which you also navigate and analyze the query's results, with the assistance of the navigation pane design item. Further customize the interface of your query by adding and arranging design items such as dropdown boxes, radio button groups, and checkboxes with which you filter your results. In addition, you can use the editing functions in Microsoft Excel in order to format your workbook, print results areas, insert additional worksheets, and create graphics and charts.

Switching between Analysis mode and Design mode

The functions of analysis mode and design mode are typically both active, and BEx Analyzer switches automatically to the right mode when you choose a function. For example if you are in analysis mode, and you select a tool on the design toolbar in order to insert a design item, Analyzer switches automatically to design mode, turning design items into their icon representations and allowing you to work with them and their properties. There can be times, however, when you

want to manually switch in and out of design mode. For example, after you have inserted a design item and you want to view the query results, you want to switch into analysis mode. In this case, you use Exit Design Mode.

To manually switch between analysis mode and design mode use the design toolbar icon

Design Items

You use design items to design the interface to your query and thus create query applications. All the design items except for the Messages item are based on data providers. A dataprovider identifies the source of the data. For example, if you wanted a set of radio buttons to offer possible key figures to the user, you would need to identify the query from where the key figures could be found.

There are eleven design items you can use when building the interface to your queries. They are provided here in a summary form. We will describe them each in detail and provide information about each setting in the next section.

Analysis Grid		Displays the results of a query. The analysis grid is the main design item in which you can navigate and perform OLAP functions
Navigation Pane		Provides access to all characteristics and structures in the query for use in navigation and analysis.
List of Filters		Lists all currently active filters
Button		Lets you execute a customized command on your results.
Dropdown Box		Allows you to set a filter via selection from the dropdown box.
Checkbox Group		Allows you to set a filter via checkbox selection.
Radio Button Group		Allows you to set a filter via radio button selection.

List of Conditions		Lists existing conditions and their status, and lets you activate and deactivate them
List of Exceptions		Lists existing exceptions and their status, and lets you activate and deactivate them
Text		Displays text elements for the query.
Messages		Displays messages associated with the application (the workbook

Analysis Grid

You use the analysis grid as the central design item in your worksheets. The analysis grid displays query results in data cells, and characteristics and structures in either rows or columns of a table. When you design a workbook, you can use the analysis grid together with the navigation pane and other design items to create a query application with full access to all OLAP functions. In analysis mode, you can navigate in the grid using either the context menu, using drag and drop, or using icons.

Figure 12: Analysis Grid

You configure properties for the analysis grid on the following tab pages in the Analysis Grid Properties dialog box.

- General – you can configure data provider, cell range, and behavior options
- Clipping – you configure whether to clip or scroll the grid display
- Associated Charts – allows you to associate Microsoft Excel charts with the analysis grid

Analysis Grid Properties

Property	Description
Analysis Grid Name	Displays the name of the grid, which is generated automatically and is unique. This name is used in the BEx Analyzer Design Toolbar menu to refer to a particular instance of the analysis grid.
General Tab Page	
Data Provider	You assign an existing data provider to the analysis grid, create and assign a new one, or change or delete a data provider. The initial view of a data provider corresponds to the query view.
Area	You manipulate coordinates for a cell or cells in this field to move or resize the button
Use Formulas	You select this checkbox to replace each cell in the analysis grid with a Microsoft Excel formula. This property of the Analysis Grid design item is set by the context menu function Convert to Formula.
Adjust Print Area	Adjusts the Microsoft Excel print area automatically to the size of the analysis grid, and also repeats the header rows and header on every page.

Enable Cell Protection	Locks (protects) all cells that are not defined in the query as input ready, and unlocks all input-ready cells. This prevents you from changing data in any cells other than input-ready cells. This is mainly used with planning functions and prevents you from entering data where inappropriate.
Apply Formatting	Deselect this checkbox to switch off the display of formatting, including icons and background color. Displaying formatting during navigation can be time consuming. You may therefore want to switch if off for performance reasons. An example is if you are navigating to achieve a result that you want to export to a text-based file. Scroll bars and scrolling icons are not affected by this setting.
Allow Navigation	Deselect this checkbox to deactivate the context menu and drag and drop functions in analysis mode. A similar option is available for the navigation pane. This prevents you from analyzing and navigating in the query
AutoFit	When you select this option, the width of the cells in the analysis grid expands as required to fit the contents. If this option is selected, after you navigate and the analysis grid is rendered, the columns are resized horizontally to the minimum size needed to render the whole content of the column.
Display Sort Icons	The icons for sorting characteristics and key figures in ascending and descending order are displayed in a report only when this option is selected
Display Hierarchy Icons	The icons for expanding or collapsing a hierarchy are displayed in a report only when this option is selected

Suppress New Lines	In BI applications that use data providers that are assigned an input-ready query, the system allows you to manually enter data in new input-ready rows. Select this checkbox to suppress the option of creating new input-ready rows.
Do Not Delete Number Formats	When Converting to Formula Number formats are deleted by default when you convert data in cells for the result set into formulas. Select this checkbox to retain the unit when you work in formula mode.
Clipping Tab Page	

 23 SAP

You use the clipping options to specifically define the size of the analysis grid using clipping or scroll bars. Clipping settings work together with cell coordinate settings in the Range field. You can independently configure horizontal or vertical clipping using the following options: Horizontal options are as follows:

- Clip – the analysis grid can only extend as far to the right as the right-most column defined in the Range field. The horizontal display beyond that is clipped, or not displayed
- Full Size – the analysis grid is displayed starting from the left-most column defined in the Range field, but expands to the right to be as wide as required to display the results
- Scroll – the analysis grid displays within the right and left coordinates configured in the Range field, and a scroll bar allows you to scroll to the right or to the left within the results. You can scroll using the following icons in the scroll bar: scroll full left, scroll left, scroll full right, or scroll Right

Vertical clipping options are as follows:

- Clip – the analysis grid can only extend as far down as the lower-most row defined in the Range field. The vertical display beyond that is clipped, or not displayed
- Full Size – the analysis grid is displayed starting from the top-most row defined in the Range field, but expands down as many rows as required to display the results.
- Scroll – the analysis grid displays within the upper and lower coordinates configured in the Range field, and a scroll bar allows you to scroll up or down within the results. You can scroll using the following icons in the scroll bar: scroll to bottom, scroll down, scroll up, or scroll to top.

Associated Charts Tab Page

You use this function to associate Microsoft Excel with the analysis grid, so that the charts always represent the results of the query. The name of any chart you have inserted on any sheet in the workbook appears in the Associated Charts list. Select the checkbox next to the chart or charts you want to associate with this analysis grid. When you associate a chart with a grid, the chart is automatically updated with the cell ranges and values in the grid, even if the grid resizes and the values within it change when you navigate.

Navigation Pane

The navigation pane design item provides access to all dimensions (characteristics and structures) in the query for use in navigation and analysis. Use the navigation pane as the staging area for dimensions in the query results. Together with the analysis grid, the navigation pane forms the backbone of a query application. In the navigation pane, dimensions and key figures in the query are listed individually on separate rows. The navigation pane also provides an overview of current filter and drill states.
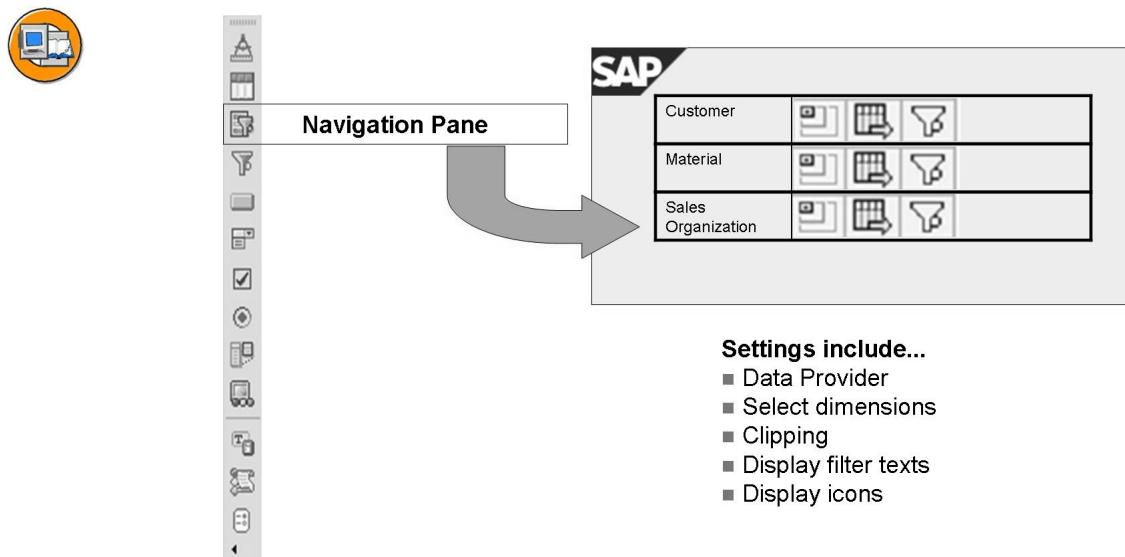
Figure 13: Navigation Pane

You configure the properties for the navigation pane on the following tab pages in the Navigation Pane Properties dialog box:.

• General – you can configure the data provider, cell range, and AutoFit options

• Dimensions – you can select the dimensions to display in the navigation pane

• Display Settings – allows you to configure various settings that affect how the pane is displayed

• Clipping – you can configure whether to clip or scroll the navigation pane display

Property	Description
Navigation Pane Name	Displays the name of the pane, which is generated automatically and is unique. This name is used in the BEx Analyzer Design Toolbar menu to refer to a particular instance of the navigation pane.
Display Technical Names/Do Not Display Technical Names	Use this button on any tab page to toggle the display of technical names
General Tab Page	
Data Provider	You assign an existing data provider to the navigation pane, create and assign a new one, or change or delete a data provider. The initial view of a data provider corresponds to the query view.
Area	You manipulate coordinates for a cell or cells in this field to move or resize the navigation pane
AutoFit	When you select this option, the width of the cells in the navigation pane (except the third column) expands as required to fit the contents. When the pane is auto fit, the columns are resized horizontally to the minimum size needed to render the whole content of column. This setting does not apply to the third column (the filer area) because filter values can be very long.
Dimensions Tab Page	
Available Dimensions	Lists all the available dimensions in the query. Select a dimension in this list and choose Add the Selected Dimensions to the Displayed Dimensions to transfer it to the Selected Dimensions list. This allows you to specify which dimensions are to be displayed in the navigation pane. By default, all dimensions are displayed.

Selected Dimensions	Lists the dimensions that are to be displayed in the navigation pane. You can use this setting to reorder the list of selected dimensions. By default, all dimensions are displayed although this list is empty. Select a dimension in this list and choose Remove Selected Dimensions from the Displayed Dimensions to remove it, or Move the Selected Displayed Dimension Up or Move the Selected Displayed Dimension Down to reorder it.
Display Settings Tab Page	
Display Filter Texts	Deselect this checkbox to turn off the display of the filter area of the navigation pane. In this case, the whole third column of the pane is hidden.
Allow Navigation	Deselect this checkbox to deactivate the context menu, drag and drop, and icon functions in analysis mode. A similar option is provided for the analysis grid. This prevents you from analyzing and navigating in the query.
Display Icons	Deselect this checkbox to turn off the display of icons in the navigation pane. Scroll bars and scrolling icons are not affected by this setting.
Show Dimensions on Rows	If selected, the dimensions that appear in the rows in the current navigational state of the analysis grid are displayed in the navigation pane
Show Dimensions on Columns	If selected, the dimensions that appear in the columns in the current navigational state of the analysis grid are displayed in the navigation pane.
Show Dimensions on Free Characteristics	If selected, the dimensions that appear in the free characteristics in the current navigational state of the analysis grid are displayed in the navigation pane.

 SAP

Number of Dimensions Per Row	If you have selected the Display Dimensions Horizontally property, you use this field to select the number of dimensions to display in each row of the navigation pane, horizontally across the sheet. This effectively vertically shortens your navigation pane
Display Dimensions Horizontally	You select this option to display multiple dimensions for each row in the navigation pane. You then configure the number of dimensions using the Number of Dimensions Per Row property.
Clipping Tab Page	
You use the clipping options to specifically define the vertical size of the analysis grid using clipping or scroll bars. Clipping settings work together with cell coordinate settings in the Range field. You configure vertical clipping using the following options: • Clip – the navigation pane can only extend as far down as the lower-most row defined in the Range field. The vertical display beyond that is clipped, or not displayed. • Full Size – the navigation pane is displayed starting from the top-most row defined in the Range field, but expands down as many rows as required to display the results. • Scroll – the navigation pane displays within the upper and lower coordinates configured in the Range field, and a scroll bar allows you to scroll up or down within the results. You can scroll using the following icons in the scroll bar: scroll to bottom, scroll down, scroll up, scroll to top.	

List of Filters

This item simply displays a list of active filters for one or more dimensions. Since the navigation pane also displays currently active filters, this item might be useful if you have suppressed the display of filters in the navigation pane (by deselecting Display Filter Texts in the navigation pane properties), or if you don't display a navigation pane at all in your workbook and yet you want to display the current status of filters.
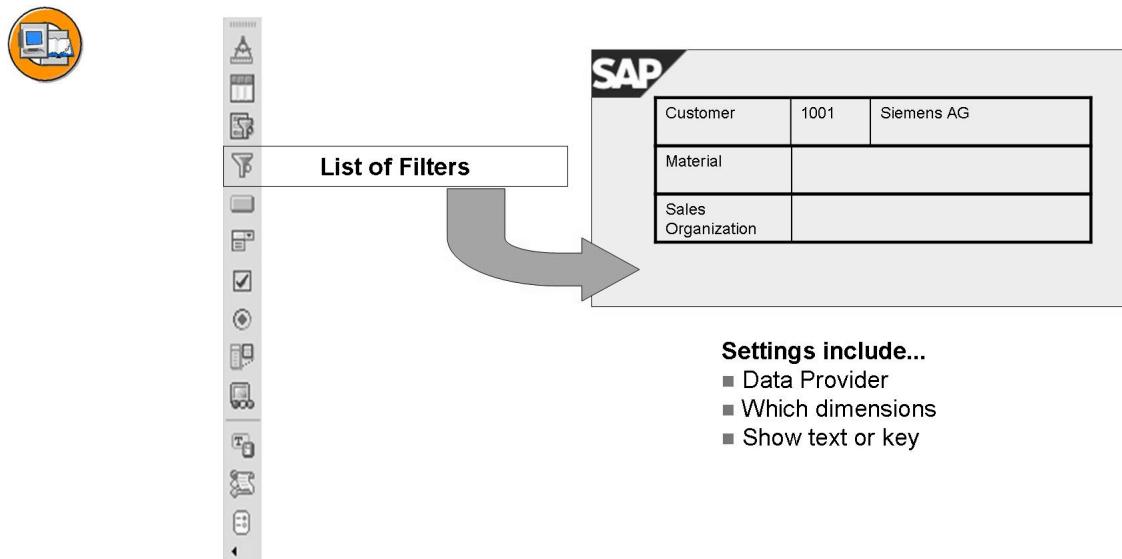
Figure 14: List of FIlters

Configure properties for the list of filters on two different tabs in the Filter List Properties dialog box

- General – lets you configure data provider, cell range, and characteristic text display
- Dimensions – lets you select the dimensions for which to display filters

Filter List Properties

Property	Description
Filter List Name	Displays the name of the list of filters, which is generated automatically and is unique. This name is used on the BEx Analyzer Design Toolbar menu to refer to a particular instance of the list of filters
Display Technical Names / Do Not Display Technical Names	Use this button on any tab to toggle the display of technical names.
General Tab	
Data Provider	Assign an existing data provider to the list of filters, create and assign a new one, or change or delete a data provider. The initial view of a data provider corresponds to a query or query view.

2007/Q2   **29** SAP

Range	Manipulate coordinates for a cell or cells in this field to move or resize the list of filters
Display Characteristic Text	Select this checkbox to display the name of the dimension in a field next to the list of its selected filters. This acts as a sort of label for each selected dimension in the filter list.
Dimension Tab	
Available Dimensions	Lists all the available dimensions in the query. Select a dimension in this list and choose the Add the Selected Dimensions to the Displayed Dimensions button to transfer it to the Selected Dimensions list. You determine in this way the dimensions for which to display a list of filters. By default, all dimensions are displayed.
Selected Dimensions	Lists the dimensions for which you have chosen to display filters, and allows you to reorder or remove them. By default, all dimensions are displayed although this list is empty. Select a dimension in this list and choose the Remove Selected Dimensions from the Displayed Dimensions button to remove it, or the Move the Selected Displayed Dimension Up or Move the Selected Displayed Dimension Down buttons to reorder it.
Presentation Style	Determines how the filter values are displayed in the list. Select from the following values: Key, Key and Text, Text and Key, Text.

Button

Use the Button design item to customize a specific command or sequence of commands. In the syntax, based on the Web API Reference, all data provider-specific commands and parameters are accepted, except functions to export to Microsoft Excel. Using a button, you can customize any steps you take

when you manually navigate a query using the context menu. When you choose the button, you execute the command or commands and the query results are updated accordingly in the analysis grid.
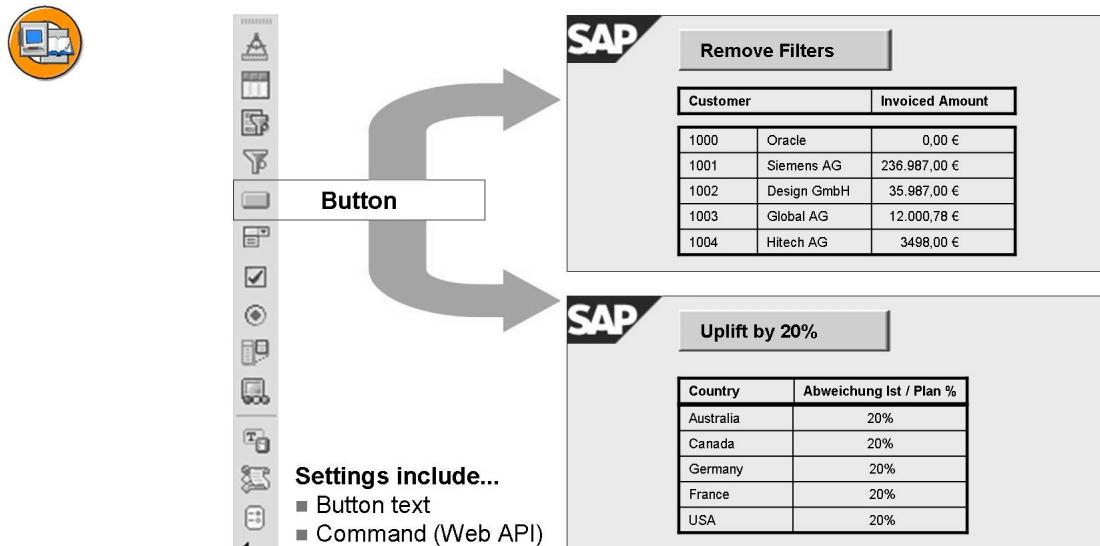
Figure 15: Button

Button Properties

Property	Description
Button Name	Displays the name of the button, which is generated automatically and is unique. This name is used in the BEx Analyze Design Toolbar menu to refer to a particular instance of the button
Area	Manipulate coordinates for a cell or cells in this field to move or resize the button.

2007/Q2   **31**

Button Text	The text (caption) you want to display on the button. You can use this to describe the command the button executes.
Command Range	Optional) Specify a three-column range of cells in the worksheet that contains the command parameters. In these cells, you provide the same parameters (Name, Index, and Value) as you do for Static Parameters (see below). But in the worksheet these can be variable; they can change during navigation based on query results or on a value you manually specify. When the button is chosen to initiate the command, all values in Static Parameters are evaluated first. Then all the parameters in the Command Range are appended.
Static Parameters	Configure the command or commands to execute by supplying name-value pairs to send to the server. A command can consist of multiple name-value pairs; configure one for each row in this table. Any data provider-specific command in the Web API Reference (except functions to export to Microsoft Excel) is accepted • Name: The name of the command. • Index: The sequence in which the command is to be executed. The start command begins with 0. You can configure as many commands as you want. • Value: The value for the command .

Dropdown Box

Use the dropdown box to easily filter a selected dimension. Since filtering is a function also provided in the navigation pane, you might not want to use a navigation pane in the same worksheet as a dropdown box. You might use the dropdown box if, for example, you don't want to offer the whole range of navigation that the navigation pane provides, but you want to be able to conveniently set filters. The checkbox group and radio button group design items provide similar functionality, except the checkbox group allows you to select multiple filter values at once. In the dropdown box properties, you configure the values for the dimension to appear in the dropdown list. When you then select a value from the dropdown box in analysis mode, the dimension is filtered by your selection, and the query results in the analysis grid update accordingly.

Configure properties for the dropdown box on two different tabs in the Dropdown Box Properties dialog box

- General – lets you configure data provider, cell range, and label display
- Dimensions – lets you select and configure the dimension you wish to filter
- Target Data Provider – lets you select multiple data providers for which you wish to apply a filter

Dropdown Box Properties

Property	Description
Dropdown Box Name	Displays the name of the dropdown box, which is generated automatically and is unique. This name is used on the BEx Analyzer Design Toolbar menu to refer to a particular instance of the dropdown box.
Display Technical Names / Do Not Display Technical Names	Use this button on any tab to toggle the display of technical names.
General Tab	
Data Provider	Assign an existing data provider to the dropdown box, create and assign a new one, or change or delete a data provider. The initial view of a data provider corresponds to a query or query view
Range	Manipulate coordinates for a cell or cells in this field to move or resize the dropdown box.
Display Label	Select this checkbox to display the name of the dimension to be filtered next to the dropdown box.

 SAP

Dimensions Tab	
Dimension	Select the dimension you wish to filter.
Text Type	Determines which text type to display for the values of the dimension: • Default Text: • Short Text: • Middle Text: • Long Text:
Read Mode	Determines the method of retrieving the list of filter values: • Posted Values • Dimension Table • Master Data Table See help text for detailed explanation of these options.

Display	Determines how the filter values are displayed in the list. Select from the following values: • Text • Key • Key and Text • Text and Key
Target Data Provider Tab	
Data Target Provider Name	The query results of a data provider can be filtered by the value of a dimension for which a dropdown box is designed. This filtering is not applied in other data providers within a workbook even though the same dimension is reused. That means you need to create individual filters for the same dimension, which is used in different data providers. However, the target data provider feature allows you simplify the above scenario. In this tab page, you can view all data providers being used in the workbook. You can select the required or all data providers to apply the same filtering for specific or all data providers used in a workbook.

Checkbox Group

Use the checkbox group to easily filter a selected dimension by multiple values at once. Since filtering is a function also provided in the navigation pane, you might not want to use a navigation pane in the same worksheet as the checkbox group. You might use the checkbox group if, for example, you don't want to offer the whole range of navigation that the navigation pane provides, but you want to be able to conveniently set a group of filter values at once. The dropdown box and radio button group design items provide similar functionality, except they restrict filer selection to one value at a time. In the checkbox group properties, you configure the values for the dimension you wish to filter. In analysis mode, each value appears next to one of a group of checkboxes, and any currently filtered values are checked. Two buttons at the bottom of the list, Submit and

Clear, let you either submit or clear checked filter selections, upon which the dimension filters are updated according to your selection, and the query results in the analysis grid are updated

Configure properties for the checkbox group on two different tabs in the Checkbox Group Properties dialog box:

*   General – lets you configure data provider, cell range, AutoFit, and horizontal display
*   Dimensions – lets you select and configure the dimension you wish to filter

Checkbox Group Properties

Property	Description
Checkbox Group Name	Displays the name of the checkbox group, which is generated automatically and is unique. This name is used on the BEx Analyzer Design Toolbar menu to refer to a particular instance of the checkbox group.
Display Technical Names / Do Not Display Technical Names	Use this button on any tab to toggle the display of technical names
General Tab	
Data Provider	Assign an existing data provider to the checkbox group, create and assign a new one, or change or delete a data provider. The initial view of a data provider corresponds to a query or query view.
Range	Manipulate coordinates for a cell or cells in this field to move or resize the checkbox group. See Moving and Resizing Design Items
AutoFit	When selected, the width of the cells in the checkbox group expands as needed to accommodate the longest value listed.
Display Dimensions Horizontally	Select this option to display the values of the checkbox group horizontally, in one row across the sheet.
Dimensions Tab	
Dimension	Select the dimension you wish to filter.

Text Type	Determines which text type to display for the values of the dimension: • Default Text • Short Text • Middle Text: • Long Text:
Read Mode	Determines the method of retrieving the list of filter values: • Posted Values • Dimension Tables • Master Data Tables See help text for more details on these options.
Display	Determines how the filter values are displayed in the list. Select from the following values: • Text • Key • Key and Text • Text and Key
Maximum Number of Displayed Values	Use this selector to set the maximum number of values to display in the checkbox group.

Radio Button Group

⦿

Use the radio button group to easily filter a selected dimension. Since filtering is a function also provided in the navigation pane, you might not want to use a navigation pane in the same worksheet as the radio button group. You might use the radio button group if, for example, you don't want to offer the whole range of navigation that the navigation pane provides, but you want to be able to conveniently set filters. The dropdown box and checkbox group provide similar functionality, except the checkbox group allows you to select multiple filter values at once. In the radio button group properties, you configure the values for the dimension you wish to filter. In analysis mode, each value appears next to one of a group of radio buttons. When you then select one of the radio buttons, the dimension is filtered by your selection, and the query results in the analysis grid update accordingly

2007/Q2   **37** SAP

Configure properties for the radio button group on two different tabs in the Radio Button Group Properties dialog box:

- General – lets you configure data provider, cell range, AutoFit, and horizontal display
- Dimensions – lets you select and configure the dimension you wish to filter
- Target Data Provider – lets you select multiple data providers for which you wish to apply a filter

Radio Button Group Properties

Property	Description
Radio Button Group Name	Displays the name of the radio button group, which is generated automatically and is unique. This name is used on the BEx Analyzer Design Toolbar menu to refer to a particular instance of the radio button group.
Display Technical Names / Do Not Display Technical Names	Use this button on any tab to toggle the display of technical names.
General Tab	
Data Provider	Assign an existing data provider to the radio button group, create and assign a new one, or change or delete a data provider. The initial view of a data provider corresponds to a query or query view. See Configuring Data Providers
Range	Manipulate coordinates for a cell or cells in this field to move or resize the radio button group. See Moving and Resizing Design Items
AutoFit	When selected, the width of the cells in the radio button group expands as needed to accommodate the longest value listed.
Display Dimensions Horizontally	Select this option to display the values of the radio button group horizontally, in one row across the sheet.
Dimensions Tab	
Dimension	Select the Dimensions you wish to filter

Text Type	Determines which text type to display for the values of the dimension: • Default Text • Short Text • Middle Text: • Long Text:
Read Mode	Determines the method of retrieving the list of filter values: • Posted Values • Dimension Tables • Master Data Tables See help text for more details on these options.
Display	Determines how the filter values are displayed in the list. Select from the following values: • Text • Key • Key and Text • Text and Key

 SAP

Maximum Number of Displayed Values	Use this selector to set the maximum number of values to display in the radio button group.
Target Data Provider Tab	
Target Data Provider Name	The query results of a data provider can be filtered by the value of a dimension for which a radio button is designed. This filtering is not applied in other data providers within a workbook even though the same dimension is reused. That means you need to create individual filters for the same dimension, which is used in different data providers. However, the target data provider feature allows you simplify the above scenario. In this tab page, you can view all data providers being used in the workbook. You can select the required or all data providers to apply the same filtering for specific or all data providers used in a workbook.
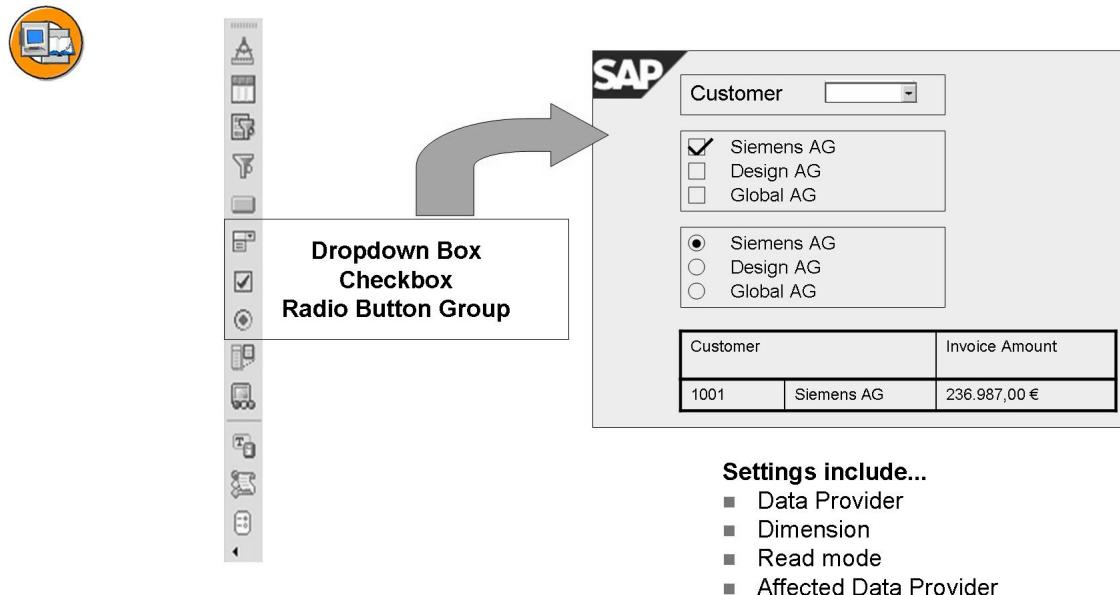
Figure 16: Dropdown list, Checkbox Group and Radio Button Group

List of Conditions

If conditions have been defined in the query, you can use the list of conditions design item to display them in rows in the worksheet. The description of each condition appears in a cell, and its status (Active or Inactive) and an icon (Activate / Deactivate) displays in the cell next to it. In analysis mode, click the icon to toggle the active status or evoke the context menu and choose**Activate** to activate a selected inactive condition. Choose **Deactivate** to deactivate a selected active condition Conditions filter the results so that only the part of the results you are interested in is displayed, so when you activate or deactivate a condition, the navigational state of the analysis grid updates accordingly.
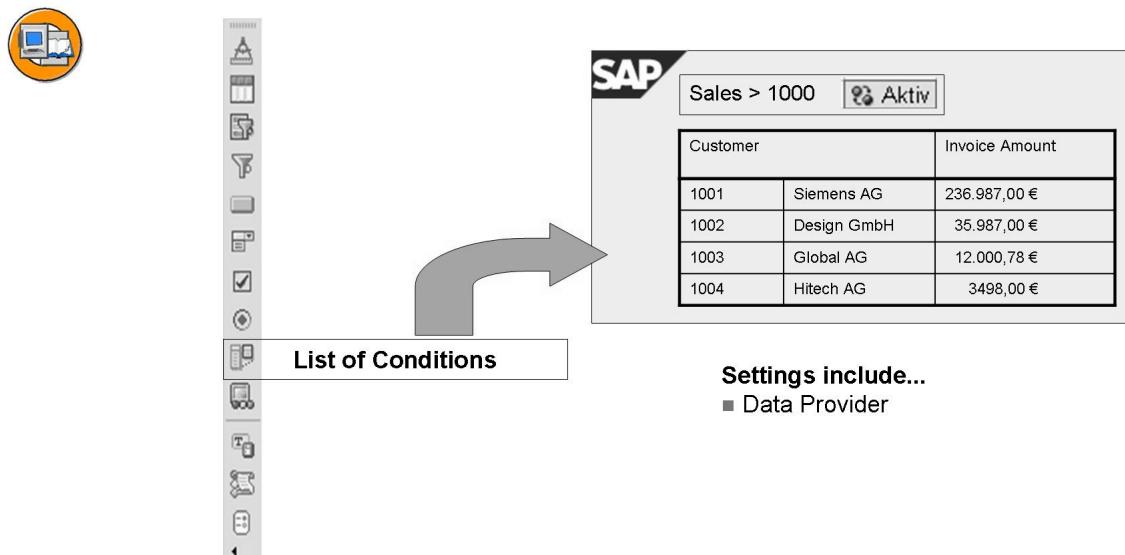
Figure 17: List of Conditions

List of Conditions Properties

Property	Description
Condition List Name	Displays the name of the list of conditions, which is generated automatically and is unique. This name is used on the BEx Analyzer Design Toolbar menu to refer to a particular instance of the list of conditions.
Data Provider	Assign an existing data provider to the list of conditions, create and assign a new one, or change or delete a data provider. The initial view of a data provider corresponds to a query or query view.
Range	Manipulate coordinates for a cell or cells in this field to move or resize the list of conditions

List of Exceptions

If exceptions have been defined in the query, you can use the list of exceptions design item to display them in rows in the worksheet. The description of each exception appears in a cell, and its status (Active or Inactive) and an icon (Activate / Deactivate) displays in the cell next to it. In analysis mode, click the icon to toggle the active status or evoke the context menu and choose **Activate** to activate a selected inactive exception. Choose **Deactivate** to deactivate a selected active exception Exceptions set the background color of data cells in results alerting you to exceptional values, so when you activate or deactivate an exception, the background color of relevant data cells in the analysis grid updates accordingly
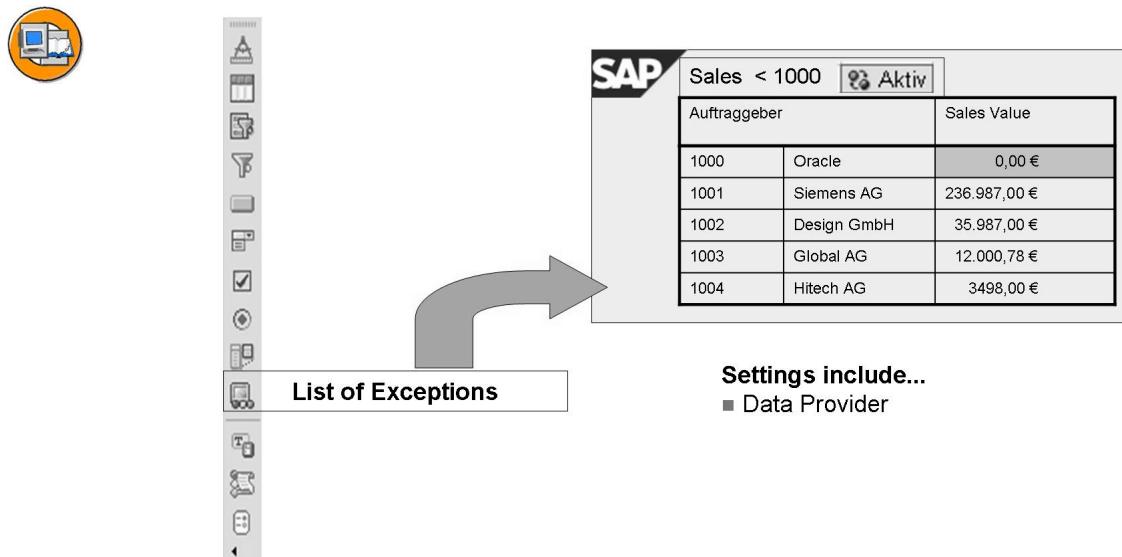
Figure 18: List of Exceptions

List of Exceptions Properties

Property	Description
Exception List Name	Displays the name of the list of exceptions, which is generated automatically and is unique. This name is used on the BEx Analyzer Design Toolbar menu to refer to a particular instance of the list of exceptions
Data Provider	Assign an existing data provider to the list of exceptions, create and assign a new one, or change or delete a data provider. The initial view of a data provider corresponds to a query or query view.
Range	Manipulate coordinates for a cell or cells in this field to move or resize the list of exceptions

Text

Use the text item to display text-based information saved with the query such as author, query description, InfoProvider, and any global filters configured in the query. You configure which text elements you want to see, and they then display in a list in your worksheet.

Configure properties for the text item on three different tabs in the Text Properties dialog box (see Inserting Design Items to learn about how to insert the text item and access its properties):

- General – lets you configure data provider, cell range, AutoFit, and caption display
- Constants – lets you select from a list of text constants to display
- Filters – lets you select from a list of global filters to display
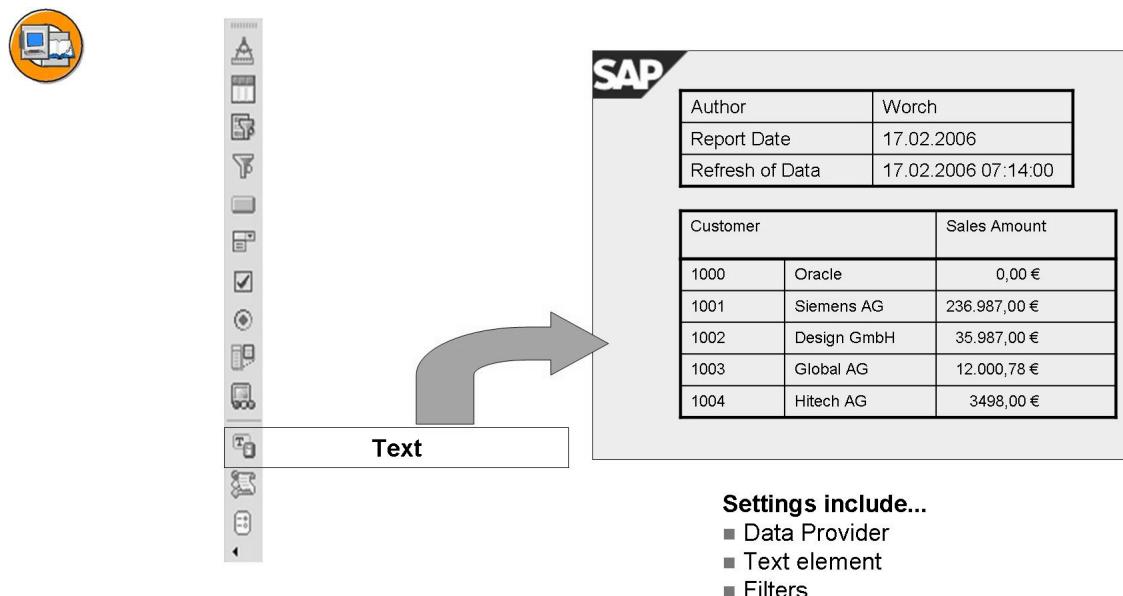
Figure 19: Text

Text Properties

Property	Description
Text Name	Displays the name of the text item, which is generated automatically and is unique. This name is used on the BEx Analyzer Design Toolbar menu to refer to a particular instance of the text item.
General Tab	

Data Provider	Assign an existing data provider to the text item, create and assign a new one, or change or delete a data provider. The initial view of a data provider corresponds to a query or query view. .
Range	Manipulate coordinates for a cell or cells in this field to move or resize the text item
AutoFit	When selected, the width of the cells in the text item expands as needed to accommodate the contents.
Display Caption	Select this checkbox to display the name of the text element in a field next to the text itself. This acts as a sort of label for each selected text element.

Constants Tab

After you have selected a data provider on the General tab, the list of available constant text elements appears on this tab, including:

- Author: The user who defined the query.
- Last Changed by: The user who last changed the query definition.
- InfoProvider: The InfoProvider with the data reported on in the query
- Query Technical Name: Technical name for the query that you enter when you save it
- Query Description: Description of the query
- Key Date: The key date is the date for which time-dependent master data is selected
- Changed At: This time tells you when the query definition was last changed.
- Current User: The user who has this query open or has inserted it into a workbook
- Last Refreshed: When the query data was last refreshed.
- Status of Data: The point in time when data for the latest request that can be used for reporting was posted to the InfoProvider
- Relevance of the Data (date): The date of the Status of Data text element.
- Relevance of the Data (hour): The time of the Status of Data text element
- Status of Data From: (MultiProviders only)
- Status of Data To: (MultiProviders only)

Select the checkbox next to the text elements you wish to display. You may also choose **Select All** to select all checkboxes and hence all constants. Choose **Deselect All** to deselect all checkboxes and hence all constants.

Filters Tab
If global filters have been configured in the query, the dimensions for which they are configured are listed on this tab. Select the checkbox next to the ones you wish to display. Current filter values configured for the selected dimension are then displayed in the text element. You may also: •   Choose Select All to select all checkboxes and hence display all filters •   Choose Deselect All to deselect all checkboxes and hence disable display of all filters •   Select Display All Global Filter Values to display all the static filter values •   Select Display All Global Variable Values to display all the static variable values

Messages

Use the messages design item to reproduce messages in your worksheet. You configure which type of messages (warning, success, information, or any or all of these) you want to see. BEx Analyzer messages, including the latest messages generated from the system to which you are logged on, display in your worksheet. Unlike other design items, messages are not associated with one particular data provider. In analysis mode, the messages display in a list, each next to an icon representing its type. If there has been no message since you last refreshed your query results, nothing displays.

Figure 20: Messages

Configure properties for the messages on two different tabs of the Message Properties dialog box (see Inserting Design Items to learn about how to insert the message design item and access its properties):

- General – lets you configure cell range, which kinds of messages you'd like to see, and AutoFit options
- Clipping – lets you configure whether to clip or scroll the message display

Messages Properties

Property	Description
Message Name	Displays the name of the message item, which is generated automatically and is unique. This name is used on the BEx Analyzer ® Design Toolbar menu to refer to a particular instance of the message item.
General	
Range	Manipulate coordinates for a cell or cells in this field to move or resize the message item.
Display Warnings	Select this checkbox to display warning messages, for example: The value for the variable 'def' is incorrect; Access to Visual Basic project failed
Display Success Messages	Select this checkbox to display success messages, for example: The query 'abc' was successfully saved
Display Information	Select this checkbox to display informational messages, for example: The system will close down at 6pm today

AutoFit	When selected, the width of the cells in the message item expands as needed to accommodate the contents of the message
Clipping	
Use the clipping options to specifically define the vertical size of the message item using clipping or scroll bars. Clipping settings work together with cell coordinate settings in the Range field. Configure vertical clipping using the following options: • Clip – the message item can only extend as far down as the lower-most row defined in the Range field. The vertical display beyond that is clipped, or not displayed • Full Size – the message item begins display from the top-most row defined in the Range field, but expands down as many rows as the results dictate • Scroll – the message item displays within the upper and lower coordinates configured in the Range field, and a scroll bar allows you to scroll up or down within the results. You can scroll using the following icons in the scroll bar: Scroll to Bottom, Scroll Down, Scroll Up, Scroll to Top	

Workbook Settings

You use the workbook settings function to configure general functions or apply themes to individual workbooks. You configure general properties for the workbook on the General tab page, and configure themes on the Themes tab page. In this dialog box, you can also attach the macro required to use the BEx Analyzer API.

Settings Incude...
- Refresh on opening
- Process variables on refresh
- Allow drag and drop
- Protect
- Themes
- Macros
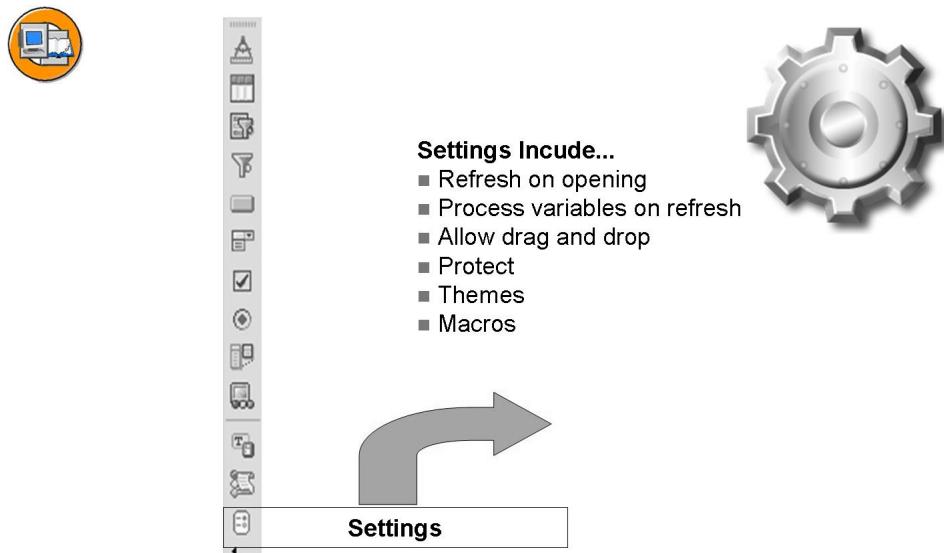
Settings

Figure 21: Workbook Settings

General

The following settings are available on the General tab page:

Refresh Workbook on Open If selected, when you open a workbook from the server, the query is automatically refreshed with values from the server. If not selected, of if you open a workbook from your local file system, you must manually refresh the query after you open it to retrieve the latest results from the server. .

Process Variables on Refresh If any query in your workbook contains variables, any current values set for the variables are part of the query view definition and are therefore stored in the workbook. If you refresh the workbook, these existing values are used to retrieve the data. If you want to enter new variable values whenever the workbook is refreshed, select this checkbox. Whenever you refresh your query, the Change Variable Values dialog box is also called and you can then set new values for the variables in the query.

Allow Drag and Drop This setting is selected by default. If you deselect this setting, you cannot use drag and drop functions to navigate your query in the analysis grid and navigation pane.

Protect Workbook If you select this setting, the BEx Analyzer password-protects all the sheets in your workbook against any changes you make with BEx Analyzer functions. When you attempt to change your workbook by inserting a design item, for example, the BEx Analyzer requests the password you specified here. The BEx Analyzer only accepts the changes if you enter the correct password. When the BEx Analyzer protects your workbook, you can still navigate in analysis

 SAP

mode, but you cannot enter data. Cells that are not locked with cell protection are not protected. Activate Enable Cell Protection in the Analysis Grid properties to protect the cells in the analysis grid as well.

Transfer Plan Values In a planning workbook, when you change the value of an input-ready cell and then navigate, the changed value is stored in the buffer on the BI server. However, you can change this behavior by selecting the following options:

*   Transfer Plan Value to the Server Before Navigation– The changed value is transferred automatically when you navigate. You do not need to execute the Transfer Plan Valuesfunction explicitly from the context menu when you change data in an input-ready cell.

*   Do Not Transfer Plan Values to the Server Before Navigation – The changed value is not transferred automatically when you navigate. You have to execute the Transfer Plan Valuesfunction explicitly from the context menu when you change data in an input-ready cell.

*   Confirm Transferring Plan Values to the Server Before Navigation – A message popup prompts you to confirm whether the changed value need to be transferred. You do not need to execute the Transfer Plan Values function explicitly from the context menu when you change data in an input-ready cell

Use Compression when Storing the Workbook You use this option to reduce the space required by your workbook if your workbook includes a large amount of information and metadata. Select this checkbox to compress the data in the worksheet on saving. You can use this option if, for example, you are working in formula mode and want to view the data in your result set without having to connect to the server. To do this, you must first save the result set in the workbook. If you do not do this, the data is not available in the cells the next time you open the workbook; only the formulas are displayed.

Themes

BEx themes, based on NetWeaver Portal styles, are shared services across the Business Explorer suite. In the BEx Analyzer, they are delivered as a set of Microsoft Excel style definitions for your workbook. Themes provide formatting information such as background color, font, and font size. To display the set of style definitions for a workbook, choose Format Style from the Microsoft Excel menu. The styles provided by the Business Explorer are prefixed with SAPBEX*. Themes are content objects that are stored on the server and that you can choose to activate. Themes are stylesheets in which you can make important changes to a theme. All workbooks in which the theme is used are updated accordingly.

Current Theme Displays the list of available BEx themes. The BEx theme that is currently applied to your workbook, if any, is selected

New/Delete/Apply/Save If the following special authorization is assigned to your user name, you can change and save themes on the server: AUTHORITY-CHECK OBJECT 'S_RS_TOOLS' ID 'COMMAND' FIELD 'THEMES'. If you have this authorization, these buttons appear:

- Create a new theme by choosing the New button
- Select a theme from the Current Theme list, and choose Delete to delete it. This deletes your theme locally.
- Select a theme from the Current Theme list, and choose Apply to apply currently configured styles to it. When you do this, the BEx Analyzer reads the Microsoft Excel style definitions of all the styles in the workbook that are prefixed with SAPBEX* and stores them in the selected theme. This allows you to edit the styles in Microsoft Excel and then create a theme with these customized styles
- To save all the themes to the server, choose Save

Reapply Theme Before Rendering Since BEx themes are saved on the server, they can be changed on the back end. Select this radio button if you want to retrieve the most up-to-date definition of the style from the server and apply it again.

Apply New Styles Select this radio button to update the current workbook with any style that has been added to the theme on the server since the theme was originally applied. If, for example, new styles have been added to a theme during a patch release, you can retrieve them from the server with this option.

Do Not Use a Theme Select this radio button to stop using the selected theme in your workbook.

Design Items

All design items that are inserted in the selected workbook are displayed on the Items tab page. You can display or modify the properties of a design item, as required

Data Provider

All data providers that are configured for the design items of the selected workbook are displayed on the Data Provider tab page. You can create a data provider or modify an existing data provider, as required.

Use the BEx Analyzer API

To use the BEx Analyzer API, you need a particular macro as well as references to the type libraries. Instead of manually maintaining the macro and references, you can do this using the following toggle button, which is available on both tab pages in this dialog box:

Attach Macros/Delete Macros Choose Attach Macros to attach the macro and references you need to use the BEx Analyzer API. To remove the references, choose Delete Macros.

2007/Q2   **51** SAP

Exercise 1: Analyzer Design Mode

Exercise Objectives

After completing this exercise, you will be able to:

- In the exercise you will learn how to work with design items in the BEx Analyzer design mode.
- You will learn how to combine Excel formatting features with formatting feature unique to BEx Analyzer in order to enhance the look and feel of workbooks.

Business Example

Users are demanding more from their BEx Analyzer workbooks. You have been asked to developed customized workbooks which present the users with flexible navigation options within a well formatted analysis environment.

Task 1:

You will create a customized workbook using a variety of design items from the design mode of the BEx Analyzer. The workbook is based on a pre-defined query. Once you have built the workbook you must test it to ensure the navigation options are working properly.

1. Open the BEx Analyzer and create a empty workbook.

2. Insert an analysis grid in cell A13 and assign the query *Divisional Performance* from InfoProvider *BW306 Customer (T_SDDEMO3)* which can be found in the path *BW Training → BW Customer Training → BW306 BI-Enterprise Reporting*. You will be prompted to log on to the assigned BI system as soon as you choose the analysis grid item. Select the *Autofit* property for the analysis grid. Remove the display of the *Division* key (you only want to see the text in the result - HINT: You must be in the analysis mode to do this part). Also remove the totals from the results. You do not want users to be able to use any navigation options directly from the results table so switch these off from the properties of the item.

3. Insert a checkbox group item in cell A8 assigned to the same dataprovider as the analysis grid. This checkbox group is used to allow the user to select the *Division* characteristic. You should display the checkboxes horizontally.

4. Insert a radio button group at cell E13 to allow the user to toggle between the two key figures available in the query. Make sure you display the choices horizontally. Leave the design mode and switch to analysis mode so that you can set the radio button initial position to key figure *Sales Volume EUR*. In cell D13 add the text *Performance By:* and set this to bold.

Continued on next page

5. Insert a dropdown box at cell E17 to allow the selection of a single *Calendar Year/Month*. Add the text *Select Month:* in cell D17 and make this text bold.

6. Insert a text item at cell E21 and select the filters for *Calendar Year/Month* and also *Division*. Add the text *You are looking at:* in cell D21 and make this bold.

7. Insert a text item at cell E23 to show the date and time of the last load of the InfoProvider. Add the text *For data at:* in cell E22 and make this text bold.

8. In cell E27 insert an exceptions list item to enable the user to toggle on/off the query exception. In cell D27 add the text *Show Missed Targets:* and make this text bold.

9. Insert a chart from the Excel chart options. Choose the chart type *Clustered Column with 3D Effect* and do not choose any other options form the chart wizard, simply press *Finish*. You need to assign the analysis grid item to this chart so that the data series is automatically aligned. To do this modify the properties of the analysis grid and select your chart from the *Associated Charts* tab.

    Now improve the appearance of the chart by removing the chart legend, also remove the chart borders and reduce the size of both axis text to be size 8. Finally reduce the overall size of the chart so that it fits into the space beneath the result table.

10. Add some graphics to the top area of the worksheet (you can find some nice graphics in workbook *Image Library* found under the role *BW306 BI-Reporting and Analysis*). Also add a report title using the text *Divisional Performance*, make this text bold and size 18.

11. To give the report a clean look, remove the worksheet gridlines.

12. So that users do not make any changes to the design of your beautiful workbook you must protect it (using SAP protection). Also make sure the workbook automatically refreshes on opening. Save your workbook to role *Unit 2 Analyzer Design Mode* with the name

    `GR## Design Mode Workbook 1` and close the workbook.

13. Open your workbook and test the interaction features you added. Try to modify the properties of any design item. What happens ?

Continued on next page

Task 2:

In this next task you will create a new workbook which contains command buttons. You will also need to use the feature of assigning multiple targets to a single navigation item. Finally you will work with formatting options to enhance the look and feel of the workbook and learn how to share your formatting with other workbooks.

1.   Create a empty workbook in the BEx Analyzer.

2.   Insert an analysis grid in cell A8 and assign the query view *Sales View 1 (SALES_VIEW_1)* from role *BW306 Reporting → Unit 2 BEx Analyzer for Business Experts*.

3.   Limit the worksheet area of the analysis grid to use the fixed cell range A8 to C15 only. Because you know that there will be more rows than can fit in this area you must offer vertical scroll options. Make sure the columns are automatically adjusted to fit the results. Return to the analysis mode to view the results and to observe the scroll buttons.

4.   You would like the user to be able to toggle between the query view you just assigned to the analysis grid and another useful query view. To do this insert two buttons in the workbook which will allow the user to choose the query views. You need to assign the query view *Sales View 1* to the first button - this button should be position in cell A6 and the button text should read *By Month*. The second button should be assigned to query view *Sales View 2* and should be positioned in cell C6 with the button text *By Division*.

5.   You should now add a second analysis item to the workbook so that you are able to view two different reports simultaneously on the same worksheet. The second analysis item should start at cell A20 and vertically extend no further than row 30 in the worksheet (you don't want to have to scroll the whole worksheet to see the results). Assign the query view *Sales View 3 (SALES_VIEW_3)* to this analysis item. Check your results before moving on to the next design task.

6.   You know that both reports share a common filter characteristic *Distribution Channel* and that there are only two possible distribution channel values to select from. You would like to offer these value selections via checkboxes but the selection should affect **both** reports simultaneously. You should also ensure the characteristic label is generated by the system so the user knows which characteristic they are selecting. View the results and test all navigation features to ensure they are working correctly.

7.   Remove the gridlines from the worksheet and add a suitable title for workbook and other graphics to improve the appearance. Graphics can be found in the workbook *Images Library* which is found in the *BW306 Reporting* role.

 SAP

8. You would now like to modify the formatting of the cells. Use you own ideas but suggestions include change the color, font and size of some of the result cells, remove the cell borders. Also you may want to change the color of the *Normal* style as this will change all non results cells. So that you can re-use this collection of formatting changes you should save them in a new theme called **BW306_THEME_G##**. Save your new theme to the server so that it can be used in a later exercise.

9. Save your workbook using the name **GR## Design Mode Workbook 2**.

56   2007/Q2

Solution 1: Analyzer Design Mode

Task 1:

You will create a customized workbook using a variety of design items from the design mode of the BEx Analyzer. The workbook is based on a pre-defined query. Once you have built the workbook you must test it to ensure the navigation options are working properly.

1. Open the BEx Analyzer and create a empty workbook.

    a) From the *Business Explorer* menu option choose *Analyzer*.

    b) Press the *New* icon from the Excel toolbar.

2. Insert an analysis grid in cell A13 and assign the query *Divisional Performance* from InfoProvider *BW306 Customer (T_SDDEMO3)* which can be found in the path *BW Training → BW Customer Training → BW306 BI-Enterprise Reporting*. You will be prompted to log on to the assigned BI system as soon as you choose the analysis grid item. Select the *Autofit* property for the analysis grid. Remove the display of the *Division* key (you only want to see the text in the result - HINT: You must be in the analysis

mode to do this part). Also remove the totals from the results. You do not want users to be able to use any navigation options directly from the results table so switch these off from the properties of the item.

a)   Click on cell A13 then select the analysis grid item from the Analyzer design toolbar.

b)   Log on to your assigned BI system using the details supplied by your instructor.

c)   Left click on the analysis grid design item to access the properties dialog and press the *Create Data Provider* icon. Click on the *Assign Query / Query View* icon and assign as the dataprovider the query *Divisional Performance* . You will find this under InfoProvider *BW306 Customer (T_SDDEMO3)*which is found in the InfoArea path *BW Training → BW Customer Training → BW306 BI-Enterprise Reporting.* Double-click on the query to select it and return to the *Create Data Provider* dialog.

Press the *OK* button to return to the *Analysis Grid* properties dialog.

d)   Select the checkbox *Autofit* . Press *OK* to leave the properties dialog.

e)   Switch to the analysis mode by selecting the icon *Exit Design Mode* in the design items toolbar. The results will now populate the table. Select any cells of the characteristic *Division* and right click to access the properties. From the *Presentation* dropdown list choose the option *Text.* From the drop down list *Supress Results Rows* choose *Always.* Click *OK* to leave the properties dialog.

f)   Now return to design mode by selecting the icon *Design Mode* from the design mode toolbar. Left click on the analysis grid item to re-access the properties then deselect the checkbox *Allow Navigation.* Press *OK* to leave the properties dialog.

3.   Insert a checkbox group item in cell A8 assigned to the same dataprovider as the analysis grid. This checkbox group is used to allow the user to select the *Division* characteristic. You should display the checkboxes horizontally.

a)   Click on cell A8 and then select the icon *Checkbox Group* from the design items toolbar.

b)   Right click on the item to access the properties dialog and select the checkbox *Display Dimension Horizontally.*

c)   Select the *Dimensions* tab and from the drop down list *Dimension* choose the characteristic *Division.* Press *OK* to leave the properties dialog.

4. Insert a radio button group at cell E13 to allow the user to toggle between the two key figures available in the query. Make sure you display the choices horizontally. Leave the design mode and switch to analysis mode so that you can set the radio button initial position to key figure *Sales Volume EUR*. In cell D13 add the text *Performance By:* and set this to bold.

    a) Click on cell E13 and then select the icon *Radio Button Group* from the design toolbar.

    b) Left-click on the item to access the properties and select the checkbox *Display Dimensions Horizontally.*

    c) Select the tab *Dimensions* and from the *Dimensions* drop down list select *Key Figures.* Press *OK* to leave the properties dialog.

    d) Press the icon *Exit Design Mode* in the design toolbar to go to analysis mode. The query results will display. Now select the radio button for the key figure *Sales Volume EUR.* This will make the key figure *Incoming Orders* disappear.

    e) Click in the cell D13 and type in the text *Performance By :* and press enter. Highlight the cell D13 and press the *Bold* icon from the standard Excel toolbar.

5. Insert a dropdown box at cell E17 to allow the selection of a single *Calendar Year/Month.* Add the text *Select Month:* in cell D17 and make this text bold.

    a) Click on cell E17 and the select the icon *Dropdown Box* from the design toolbar.

    b) Left click on the design item to access the properties dialog, then select the tab *Dimensions* and from the *Dimensions* drop down list select *Calendar Year/Month.* Press *OK* to leave the properties dialog.

    c) Click on cell D17 add type in the text *Select Month:*, then press enter. Highlight the cell D17 and press the *Bold* icon from the standard Excel toolbar.

6. Insert a text item at cell E21 and select the filters for *Calendar Year/Month* and also *Division*. Add the text *You are looking at:* in cell D21 and make this bold.

 SAP

        a)   Click on cell E21 and select the icon *Text*

        b)   Left click on the design item to access the properties dialog, then select the tab *Filters* and from the list select the checkboxes for *Calendar Year/Month* and *Division*. Press *OK* to leave the properties dialog.

        c)   Click on cell D21 add type in the text *You are looking at:*, then press enter. Highlight the cell D21 and press the *Bold* icon from the standard Excel toolbar.

7.   Insert a text item at cell E23 to show the date and time of the last load of the InfoProvider. Add the text *For data at:* in cell E22 and make this text bold.

        a)   Click on cell E23 and select the icon *Text.*

        b)   Left click on the design item to access the properties dialog, then select the tab *Constants* and from the list select the checkboxes for *Display Status of Data*. Press *OK* to leave the properties dialog.

        c)   Click on cell D23 add type in the text *For data at:*, then press enter. Highlight the cell D23 and press the *Bold* icon from the standard Excel toolbar.

8.   In cell E27 insert an exceptions list item to enable the user to toggle on/off the query exception. In cell D27 add the text *Show Missed Targets:* and make this text bold.

        a)   Click on cell E27 and select the icon *List of Exceptions.*

        b)   Left click on the design item to access the properties dialog to review the settings. You do not have to make any changes. Press *OK* to leave the properties dialog.

        c)   Click on cell D27 add type in the text *Show Missed Targets:*, then press enter. Highlight the cell D27 and press the *Bold* icon from the standard Excel toolbar.

9.   Insert a chart from the Excel chart options. Choose the chart type *Clustered Column with 3D Effect* and do not choose any other options form the chart wizard, simply press *Finish*. You need to assign the analysis grid item to this chart so that the data series is automatically aligned. To do this modify the properties of the analysis grid and select your chart from the *Associated Charts* tab.

Now improve the appearance of the chart by removing the chart legend, also remove the chart borders and reduce the size of both axis text to be size 8. Finally reduce the overall size of the chart so that it fits into the space beneath the result table.

a)   Select the standard Excel function from the menu *Insert → Chart*. Highlight the chart subtype *Clustered Column with 3D visual effect*. Click *Finish* to leave the chart dialog.

b)   Left click on the analysis grid item to access the properties, then select the tab *Associated Charts*. Select the check box for the chart and press *OK* to leave the dialog.

c)   Return to analysis mode to see the results of the chart.

d)   Carefully right click on the chart legend and from the menu choose the option *Clear*.

e)   Carefully right click on the vertical axis text (the axis values) and from the menu choose the option *Format Axis*. Select the *Font* tab and from the size drop down select *8* to reduce the size of the text. Do exactly the same for the horizontal axis.

f)   Right-click on the chart area and from the menu choose the option *Format Chart Area*. Select the *Patterns* tab and for the *Borders* option select *None*.

g)   Click on the chart to show the size control in the corners, click on one of them and slide the pointer to reduce the overall size of the chart area and make sure it fits under the results of the analysis grid.

10.   Add some graphics to the top area of the worksheet (you can find some nice graphics in workbook *Image Library* found under the role *BW306 BI-Reporting and Analysis*). Also add a report title using the text *Divisional Performance*, make this text bold and size 18.

a)   Open the workbook above and copy and paste any graphics which you would like (or find your own).

b)   Type in the text *Divisional Performance* somewhere suitable in the report and make sure it is formatted well (bold, size 18).

11.   To give the report a clean look, remove the worksheet gridlines.

a)   From the Excel menu *Tool → Option* deselect the checkbox *Gridlines*.

12.   So that users do not make any changes to the design of your beautiful workbook you must protect it (using SAP protection). Also make sure the workbook automatically refreshes on opening. Save your workbook to role *Unit 2 Analyzer Design Mode* with the name

           `GR## Design Mode Workbook 1` and close the workbook.

    a)   From the design mode toolbar select the icon *Workbook Settings*.

    b)   Select the checkbox *Refresh Workbook On Opening*.

    c)   Select the checkbox *Protect Workbook* and then enter a password of your choice in the *Password* field.

           Reconfirm the password by re-entering it in the *Confirm Password* field.

    d)   From the analysis toolbar of the BEx Analyzer select the *Save* icon. From the menu options choose *Save Workbook As*. Save your workbook to role *BW306 Reporting - Unit 2 BEx Analyzer for Business Experts* with the description of **`GR## Design Mode Workbook 1`**. Close the workbook by selecting the standard Excel menu option *File → Close*.

13.   Open your workbook and test the interaction features you added. Try to modify the properties of any design item. What happens ?

    a)   From the Bex Analyzer analysis toolbar select the icon *Open* and then *Open Workbook* to select your workbook from the *History* area. Once the workbook opens click on the icon *Design Mode*. You will be prompted to enter the password !

Task 2:

In this next task you will create a new workbook which contains command buttons. You will also need to use the feature of assigning multiple targets to a single navigation item. Finally you will work with formatting options to enhance the look and feel of the workbook and learn how to share your formatting with other workbooks.

1.   Create a empty workbook in the BEx Analyzer.

    a)   Press the *New* icon from the Excel toolbar.

2. Insert an analysis grid in cell A8 and assign the query view *Sales View 1 (SALES_VIEW_1)* from role *BW306 Reporting → Unit 2 BEx Analyzer for Business Experts*.

   a) Click on cell A8 then select the analysis grid item from the Analyzer design toolbar.

   b) Left click on the analysis grid design item to access the properties dialog and press the *Create Data Provider* icon. Click on the *Assign Query / Query View* icon and assign as the dataprovider the query view *Sales View 1 (SALES_VIEW_1)*. You will find this from role *BW306 Reporting → Unit 2 BEx Analyzer for Business Experts* Double-click on the query view to select it and return to the *Create Data Provider* dialog. Press *OK* to return to the properties dialog.

3. Limit the worksheet area of the analysis grid to use the fixed cell range A8 to C15 only. Because you know that there will be more rows than can fit in this area you must offer vertical scroll options. Make sure the columns are automatically adjusted to fit the results. Return to the analysis mode to view the results and to observe the scroll buttons.

   a) From the *Clipping* tab select the checkbox *Scroll* from the *Vertical* area.

   b) On the *General* tab enter **$A$8:$C$15** in the *Range* field.

   c) Select the *Autofit* checkbox then press *OK* to leave the properties dialog.

   d) Press the icon *Exit Design Mode* to view the results. Use the scroll button to move through the results.

4. You would like the user to be able to toggle between the query view you just assigned to the analysis grid and another useful query view. To do this insert two buttons in the workbook which will allow the user to choose the query views. You need to assign the query view *Sales View 1* to the first button -

                                       SAP

this button should be position in cell A6 and the button text should read *By Month*. The second button should be assigned to query view *Sales View 2* and should be positioned in cell C6 with the button text *By Division*.

a)   Click on cell A6 then select the *Insert Button* icon from the design toolbar.

b)   Left-click on the button item to access the command button wizard then select the radio button *Data Provider Specific Command*. Click the *Next* button.

c)   Select the radio button *Assign Query / Query View* then press the query icon below to assign the query view then in the *Open* dialog double click on the query view *Sales View 1*.

to select it. Click on the *Finish* button to access the properties dialog.

d)   In the *Button Text* field enter `By Month`, then press the *OK* button to complete the setup.

e)   Repeat the steps a) to d) except the cell will be C6 and the query view will be *Sales View 2* and the button text will `By Division`.

5.   You should now add a second analysis item to the workbook so that you are able to view two different reports simultaneously on the same worksheet. The second analysis item should start at cell A20 and vertically extend no further than row 30 in the worksheet (you don't want to have to scroll the whole worksheet to see the results). Assign the query view *Sales View 3 (SALES_VIEW_3)* to this analysis item. Check your results before moving on to the next design task.

a)   Click on cell A20 then select the analysis grid item from the Analyzer design toolbar.

b)   Left click on the analysis grid design item to access the properties dialog and press the *Create Data Provider* icon. Click on the *Assign Query / Query View* icon and assign as the dataprovider the query view*Sales View 3 (SALES_VIEW_3)*. You will find this from role *BW306 Reporting → Unit 2 BEx Analyzer for Business Experts*. Double-click on the query view to select it and return to the *Create Data Provider* dialog. Press *OK* to return to the properties dialog.

c)   From the *Clipping* tab select the checkbox *Scroll* from the *Vertical* area.

d)   On the *General* tab enter `$A$20:$C$30` in the *Range* field.

e)   Press *OK* to close the dialog then press the icon *Exit Design Mode* to view your results.

6.   You know that both reports share a common filter characteristic *Distribution Channel* and that there are only two possible distribution channel values to select from. You would like to offer these value selections via checkboxes

Continued on next page

64   2007/Q2

but the selection should affect **both** reports simultaneously. You should also ensure the characteristic label is generated by the system so the user knows which characteristic they are selecting. View the results and test all navigation features to ensure they are working correctly.

a)   Click on cell F6 and then select the icon *Insert Check Box Group*.

b)   Left click on the item to access the properties then select the checkbox *Display Label*.

c)   Click on the *Dimensions* tab and from the *Dimensions* dropdown list select *Distribution Channel*.

d)   Click on the *Target Data Provider* select the two dataproviders which you have previously used. They should be *DataProvider 1* and *DataProvider 2*.

e)   Click on the icon *Exit Design Mode* to view the results and test all the navigation features. Make sure the check boxes affect both results.

7.   Remove the gridlines from the worksheet and add a suitable title for workbook and other graphics to improve the appearance. Graphics can be found in the workbook *Images Library* which is found in the *BW306 Reporting* role.

a)   From the Excel menu *Tools → Options* de-select *Gridlines*.

b)   Add a title to the reports and make this text bold and size 28 using standard Excel formatting options in the toolbar. Add the SAP logo in the top left corner, the image is in the *Image Library* workbook.

8.   You would now like to modify the formatting of the cells. Use you own ideas but suggestions include change the color, font and size of some of the result cells, remove the cell borders. Also you may want to change the color of the *Normal* style as this will change all non results cells. So that you can

 SAP

re-use this collection of formatting changes you should save them in a new theme called **BW306_THEME_G##**. Save your new theme to the server so that it can be used in a later exercise.

a) Use the Excel menu option *Format → Style* to access the cell styles. For the style *Normal* click on *Modify*, then select tab *Pattern* and click on a pale color. Then click *OK*. Now select the style *SAPBexStdData* from the dropdown list and modify the pattern to any color you wish. Also make the font size 8. You may also want to remove the horizontal borders from the style *SAPBexStdItem*.

b) Click on the icon *Workbook Settings* on the design toolbar. Then click on the tab *Themes*.

c) Click on the *New* button and in the field type the name **BW306_THEME_GR##** and press OK.

d) Click on the *Apply* button to move the workbook styles you made earlier into the new theme.

e) Click on the *Save* button to save the theme to the server for re-use by others. Close the *Workbook Settings* dialog.

9. Save your workbook using the name **GR## Design Mode Workbook 2**.

a) From the analysis toolbar of the BEx Analyzer select the *Save* icon. From the menu options choose *Save Workbook As*. Save your workbook to role *BW306 Reporting - Unit 2 BEx Analyzer for Business Experts* with the description of **GR## Design Mode Workbook 2**.

Lesson Summary

You should now be able to:
- Design and build customized workbooks using the Design Mode of the BEx Analyzer.

Lesson: Integrating Excel Formulas and Formatting

Lesson Overview

In this lesson you will learn how to work with the formula mode of the BEx Analyzer. This opens up many opportunities for developing sophisticated workbooks which incorporate Excel standard functionality.

Lesson Objectives

After completing this lesson, you will be able to:

- Use native Excel formula to integrate Excel functionality with BI functions

Business Example

Excel users want to use native Excel formulas with the results from BI but the workbooks must still be refreshed with new BI data as required without destroying the references in the formula.

What is the formula mode ?

When you are working in formula mode the cell results are presented using standard (native) Excel formulas. This means that the presentation of the results cells including their formatting can be completely handed over to standard Excel settings rather than be handled by the Analyzer. So now you can access the results of a query by referencing the characteristic values together with the related key figure.

> **Hint:** It might help to think of the formula mode in this way : We are using the power and flexibility of the BEx Analyzer to manage data to BI and we are also using Excel standard formatting and formulas to manage the presentation of the results.

The diagram below shows a workbook which uses the results of a query and currency rates to calculate target totals in target currencies. Formulas were used throughout to reference the cells.
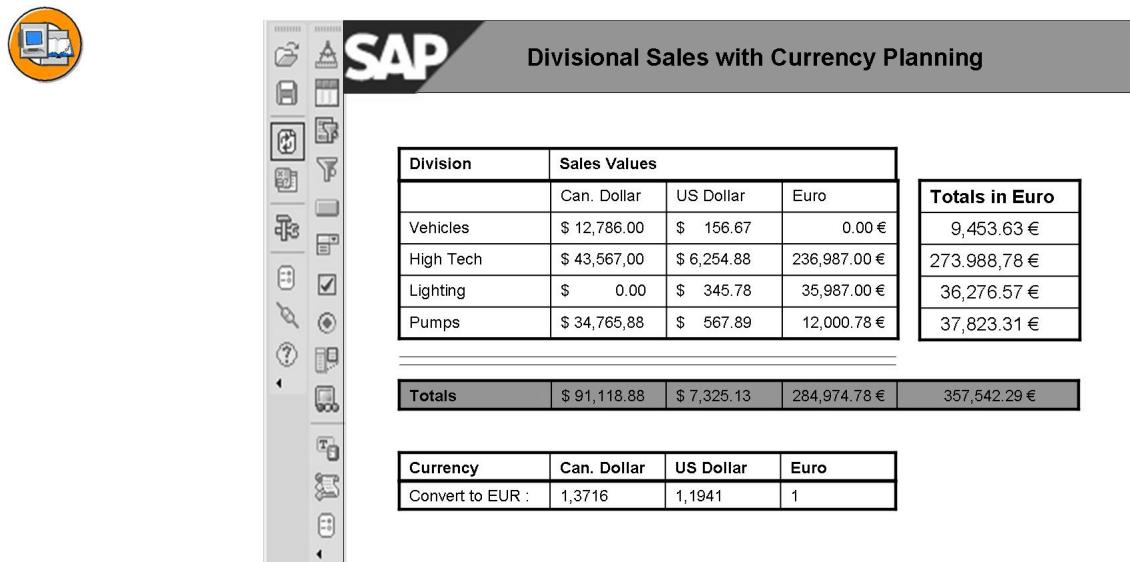
Figure 22: Example of workbook using formula mode

The next slides show the breakdown of the workbook, we are using a combination of standard Excel formula and the new **BExGetData** .
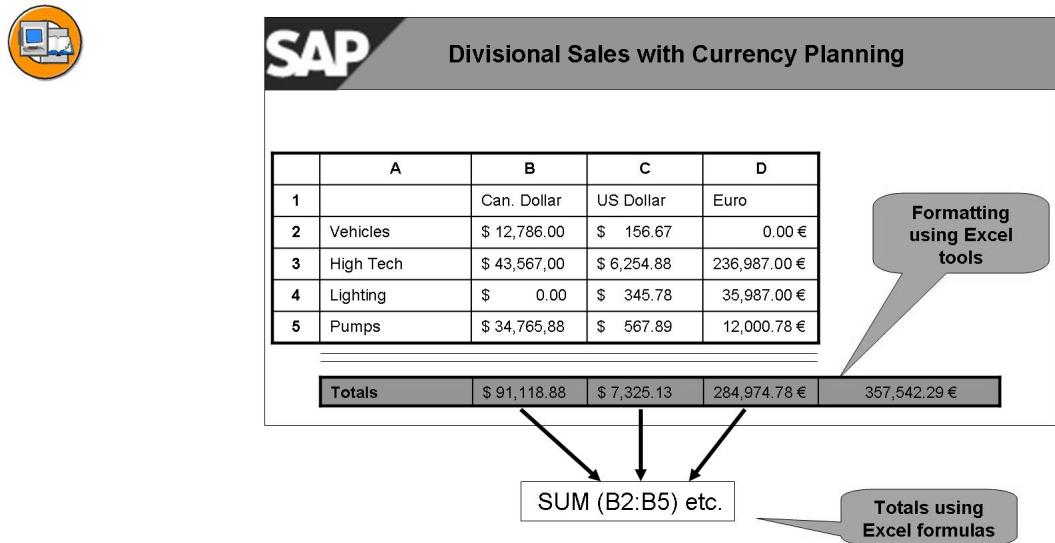
Figure 23: Breakdown of result (1)

Divisional Sales with Currency Planning

		Can. Dollar	C US Dollar	D Euro
1		Can. Dollar	US Dollar	Euro
2	Vehicles	$ 12,786.00	$ 156.67	0.00 €
3	High Tech	$ 43,567,00	$ 6,254.88	236,987.00 €
4	Lighting	$ 0.00	$ 345.78	35,987.00 €
5	Pumps	$ 34,765,88	$ 567.89	12,000.78 €
6				
7	Totals	$ 91,118.88	$ 7,325.13	284,974.78 €
8				
9	Currency	Can. Dollar	US Dollar	Euro
10	Convert to EUR :	1,3716	1,1941	1
11				

Results presented with BExGetData formula

Totals in Euro
9,453.63 €
273.988,78 €
36,276.57 €
37,823.31 €

357,542.29 €

=B7*B10 etc.

Standard Excel formula

Manually entered constant values

- Results positioning not affected by BEx refresh
- Formatting not affected by BEx refresh

Figure 24: Breakdown of result (2)

The formula syntax is very simple and very predictable. You can access the formula from the standard Excel formula dropdown list *Insert → Function*. Look in the category *User Defined* and find the formula **BExGetData**. There are three parameters to complete: **DataProvider, Key Figure, Characteristic.**. However you may need multiple characteristics in your formula depending on the number of characteristics being referenced, you simply supply the characteristics in sequence separated by a comma. The diagram below shows the breakdown of the formula, the formula can use either absolute values (as in this example) or cell references to locate the absolute values.

70   2007/Q2

formula name   key figure

=BExGetData("DATA_PROVIDER_1","Sales Volume","01.2005","SOUTH","UK")

source of data

characteristic values*

* as many as required in reference
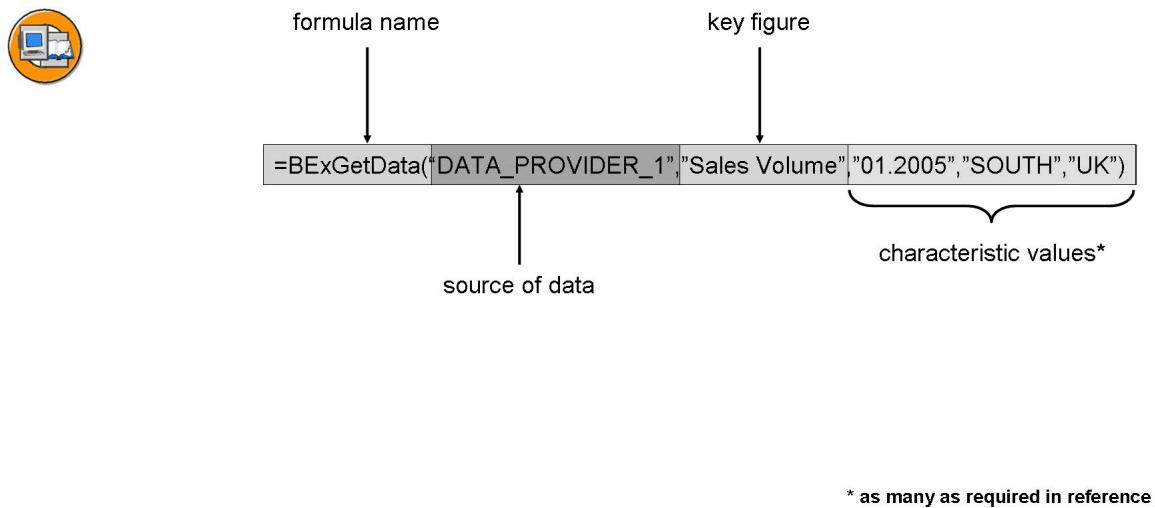
Figure 25: Anatomy of the BExGetData formula

The next slide illustrates the relationship between the InfoProvider, the query results, the formula and the results. Here we are using cell references to locate the characteristic values in the worksheet.

Query Results Data

Sales Volume EUR	01.2005	4182963.07
Sales Volume EUR	02.2005	3146942.96
Sales Volume EUR	03.2005	5121720.88
Sales Volume EUR		12451626.91

- **Query data made available to formula via dataprovider**

fx =BExGetData("DATA_PROVIDER_1",B$1,$A2)

- **Use cell references or absolute values**

	A	B
1	Calendar Year	Sales Volume EUR
2	01.2005	4182963.07
3	02.2005	3146942.96
4	03.2005	5121720.88

Figure 26: Data flow from InfoProvider to workbook cell

Working with formula mode

To switch to formula mode, you can click on a BI result cell and then use the menu option *Convert to Formula* in the context menu. This converts the results to Excel formulas and also deletes the number formats and deactivates navigation using drag and drop. Another approach is to insert an analysis grid item into the workbook and then select the checkbox *Use Formulas* in the properties dialog.
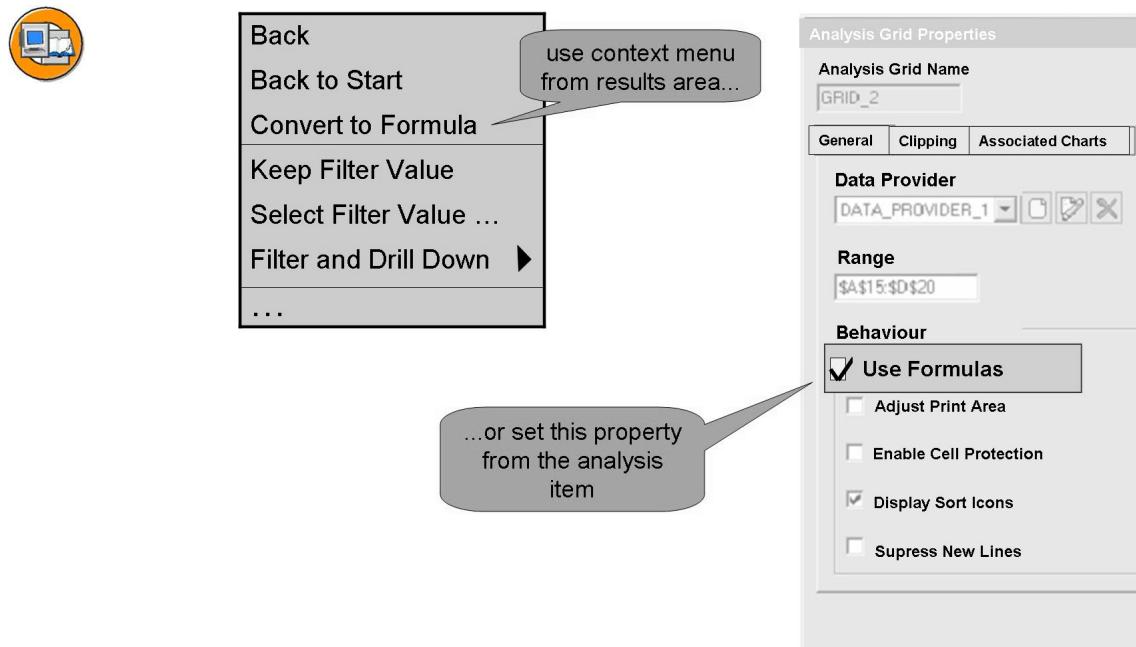
Figure 27: Converting to formulas mode

However the *Use Formulas* and *Convert to Formula* functions are not the same. Both functions generate the BExGetData formula in the data cells of the analysis grid, however, the *Use Formulas* function relates to a property of the analysis grid design item, this function is automatically switched on when you use the *Convert to Formula* function from the menu. Note that the *Use Formulas* property is just one part of the functions provided by the *Convert to Formula* context menu functions. It does **not** delete the analysis grid design item, and access to the context menu functions for the analysis grid and drag and drop remains active. When you set the *Use Formulas* property, standard formatting is also not overwritten permanently; when the workbook is refreshed, the standard formatting is restored. The *Convert to Formula* from the context menu function enables these actions in **one step**; this means it works as a type of shortcut.

If you open a workbook which contains results that are using the BEx formula you may find that you see the cell value **#NV**. This happens because the formula has failed to find the dataprovider in the workbook. To avoid this problem you need to save the results offline (with the workbook). To do this you must set the property *Provide the Results Offline* from the analysis grid item. This means

you can open the workbook (either offline or from the BI server) without having to refresh the query as the results are already contained within the workbook. Obviously if you want to add new values from the dataprovider you will need to establish a connection to the dataprovider by refreshing the workbook. If you do not set the property *Provide the Results Offline* then you must execute the query using the *Refresh*option in the workbook in order to re-establish the link between the dataprovider and the query results.
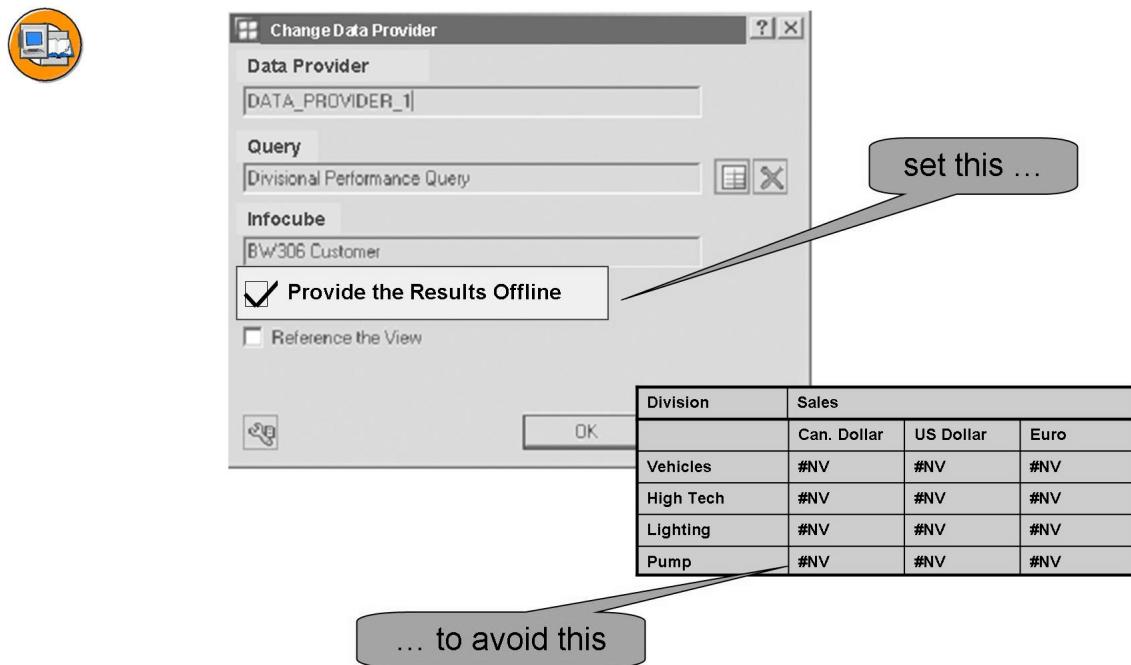
Figure 28: Saving Results Offline

Another alternative to converting results cells to formulas is to simply define a data provider using the *Settings* icon in the design toolbar and then type the formula directly into the cell (or use the menu option *Insert → Function*) and then supply the formula parameters manually. This is especially useful when you do not need the entire result set from the query, you can take exactly what you need from the dataprovider and type it directly into the formula. You need to know the exact syntax for the formula and of course this assumes you are familiar with the data available via the dataprovider, but for small amounts of data this could be useful.

Figure 29: Accessing DataProviders directly without using conversion

Exercise 2: Use Excel formatting and formulas with BEx Analyzer workbooks.

Exercise Objectives

After completing this exercise, you will be able to:
- You will understand how to integrate Excel formatting and formulas with the BEx Analyzer workbook.

Business Example

Users are requesting more and more integration between Excel features and BEx Analyzer workbook results. You need to develop your skills in this area, this exercise will help you to achieve this.

Task 1:

You would like to present the sales results for each of your four divisions. You would also like to model the bonus payouts for each division by entering different % values directly in the worksheet. You will begin by displaying a standard report using an analysis grid item, but you will then use you own formatting to improve the presentation whilst retaining the ability to refresh the sales values from the InfoProvider.

1. Create a empty workbook in the BEx Analyzer.

2. Insert an analysis grid in cell A7 and assign the query *Divisional Performance* from InfoProvider *BW306 Customer (T_SDDEMO3)* which can be found in the path *BW Training → BW Customer Training → BW306 BI-Enterprise Reporting.* You will be prompted to log on to the assigned BI system if you are not already logged on as soon as you choose the analysis grid item. Select the *Autofit* property for the analysis grid.

3. Convert the results to formulas using the context menu. How can you be sure this has worked ?

4. Remove the totals row.

5. In order to spread out the results you should drag the *Lighting* and *High Tech* divisions rows down to start at row 14. Do the same for the *Pumps* and *Motorcyles* so that these rows start at row 10.

6. Insert a strong bold underline at the bottom of the results, then add up the results of all divisions below the line you just added using Excel formulas.

Continued on next page

7. You want to allow the user to manually maintain the bonus percentage values per division. Insert the title **Bonus %** in cell F7. The percentage value should be entered in cells F9 and F10 and then F13 and F14. Enter any single digit number in each of these four cells.

8. Calculate the bonus payout value using the sales result from BI and the percentage you just entered for each division. The results should be in column H adjacent to each division. Use the title **Planned Bonus Payout**.

9. Format the *Planned Bonus Payout* result cells using Excel formatting so that the values appear in bold and as currency amount with the € (Euro) currency symbol to the **right** of the value.

10. Finish off the formatting using any of your own ideas, add a title, graphics, remove gridlines etc.

11. Save your workbook using the name **GR## Formula Mode Workbook 1** under the role *BW306 Reporting - Unit 2 BEx Analyzer for Business Experts*.

12. Close the workbook, then re-open it immediately. What do you notice? How can you solve this problem ?

Task 2:

You would like to develop a highly customized workbook by typing directly the characteristic values and key figure names into the cells and using formulas to access the dataprovider.

1. Create a empty workbook in the BEx Analyzer.

2. Using the *Settings* icon define a dataprovider and assign this to the query view *Sales View 1* from the role *BW306 Reporting → Unit 2 BEx Analyzer for Experts*. You will be prompted to log on to the assigned BI system if you are not already logged on as soon as you choose the settings icon. Make sure the results data will be available offline.

3. Add the heading **Sales Volume EUR** in cell B8.

4. Add the month / year value **'01.2005** in cell A9, the value **'02.2005** in cell A10 and the value **'03.2005** in cell A11.

5. Format these values by making the heading appear in bold.

6. In cell B9 enter the formula **=BExGet-Data("DATA_PROVIDER_1",B8,A9)**, then press enter, don't worry if you see **#NV**. All you need to do is to press the icon *Exit Design Mode* to process the results which should now appear.

7. Copy the formula you just entered and paste into cells B10 and B11. Modify the formula so it references the correct cells.

8. Add a total to the rows and make this bold and add a top and bottom cell border..

9. Add a title for the results in cell A6, use the heading **Q1 2005 Results** and format this so it is bold and large.

10. Copy all cells for the Q1 2005 results to an empty area to the right in the worksheet and adjust the title and month values to represent the quarter 2 2005.

11. Repeat the steps for quarter 3 and quarter 4 2005 using the space at the bottom of the worksheet.

12. Finish off the formatting using any of your own ideas, add a title, graphics, remove gridlines etc.

13. Save your workbook using the name **GR## Formula Mode Workbook 2** under the role *BW306 Reporting - Unit 2 BEx Analyzer for Business Experts*.

Solution 2: Use Excel formatting and formulas with BEx Analyzer workbooks.

Task 1:

You would like to present the sales results for each of your four divisions. You would also like to model the bonus payouts for each division by entering different % values directly in the worksheet. You will begin by displaying a standard report using an analysis grid item, but you will then use you own formatting to improve the presentation whilst retaining the ability to refresh the sales values from the InfoProvider.

1. Create a empty workbook in the BEx Analyzer.

   a) Press the *New* icon from the Excel toolbar.

2. Insert an analysis grid in cell A7 and assign the query *Divisional Performance* from InfoProvider *BW306 Customer (T_SDDEMO3)* which can be found in the path *BW Training → BW Customer Training → BW306 BI-Enterprise Reporting*. You will be prompted to log on to the assigned BI system if you are not already logged on as soon as you choose the analysis grid item. Select the *Autofit* property for the analysis grid.

   a) Click on cell A7 then select the analysis grid item from the Analyzer design toolbar.

   b) If you are not already logged onto the BI server you will now be prompted to do so now using the details supplied by your instructor.

   c) Left click on the analysis grid design item to access the properties dialog and press the *Create Data Provider* icon. Click on the *Assign Query / Query View* icon and assign as the dataprovider the query *Divisional Performance* . You will find this under the role *BW306 Reporting →Unit 2 BEx Analyzer for Business Experts*. Double-click on the query to select it and return to the *Create Data Provider* dialog.

   Press the *OK* button to return to the *Analysis Grid* properties dialog.

   d) Select the checkbox *Autofit* . Press *OK* to leave the properties dialog.

   e) Switch to the analysis mode by selecting the icon *Exit Design Mode* in the design items toolbar. The results will now populate the table.

3. Convert the results to formulas using the context menu. How can you be sure this has worked ?

   a) Right-click on any result cell and select the menu option *Convert to Formula*. Click on any key figure cell and you should notice the *BExGetData* formula appears in the formula field.

4. Remove the totals row.

    a)   Highlight the entire row 12 by pressing on the row button then use the
         menu path *Edit → Delete* to remove the results.

5. In order to spread out the results you should drag the *Lighting* and *High Tech*
    divisions rows down to start at row 14. Do the same for the *Pumps* and
    Motorcyles so that these rows start at row 10.

    a)   Highlight the entire results area of the *Lighting* and *High Tech*
         divisions and drag the outline so that the two rows are positioned
         starting at row 14.

    b)   Highlight the entire results area of the *Pumps* and *Motorcycles* divisions
         and drag the outline so that the two rows are positioned starting at row
         10. There should be two spaces between the sections and the title.

6. Insert a strong bold underline at the bottom of the results, then add up the
    results of all divisions below the line you just added using Excel formulas.

    a)   Highlight cells C17 to D17 then right click to access the menu option
         Format Cell. Under the *Border* tab click on the strongest line style then
         click on the top part of the border to add the line. Press *OK* to leave the
         dialog and return to the results.

    b)   Click on the cell C18 then enter the formula `=SUM(C10:C17)`.

    c)   With the cell C18 still highlighted, hold the bottom right small square
         and drag this across to cell D18. You should now have replicated the
         formula and also transposed the cells.

7. You want to allow the user to manually maintain the bonus percentage values
    per division. Insert the title `Bonus %` in cell F7. The percentage value
    should be entered in cells F9 and F10 and then F13 and F14. Enter any single
    digit number in each of these four cells.

    a)   Click on cell F7 and type in `Bonus %`.

    b)   Enter a single digit value in cells F9, F10, F13 and F14.

8. Calculate the bonus payout value using the sales result from BI and the
    percentage you just entered for each division. The results should be in
    column H adjacent to each division. Use the title `Planned Bonus
    Payout`.

    a)   Click on cell H7 and enter `Planned Bonus Payout`.

    b)   Click on cell H10 and enter the formula `=C10*F10`.

    c)   Copy the formula from H10 and paste this into cells H11, H14 and H15.

Continued on next page

9. Format the *Planned Bonus Payout* result cells using Excel formatting so that the values appear in bold and as currency amount with the € (Euro) currency symbol to the **right** of the value.

    a) Highlight the result cells H10 through to H15 then use the menu option *Format Cells* to reach the formatting properties dialog.

    b) On the *Font* tab select *Bold.*

    c) On the *Number* tab and under the category *Currency* select the entry *€ Euro (123 €)* from the *Symbol* drop down list. Press *OK* to return to the results and check your formatting looks okay.

10. Finish off the formatting using any of your own ideas, add a title, graphics, remove gridlines etc.

    a) Use you own ideas here. Don't forget to use the *Images Library* for graphics.

11. Save your workbook using the name **GR## Formula Mode Workbook 1** under the role *BW306 Reporting - Unit 2 BEx Analyzer for Business Experts*.

    a) From the analysis toolbar of the BEx Analyzer select the *Save* icon. From the menu options choose *Save Workbook As*. Save your workbook to role *BW306 Reporting - Unit 2 BEx Analyzer for Business Experts* with the description of **GR## Formula Mode Workbook 1**.

12. Close the workbook, then re-open it immediately. What do you notice? How can you solve this problem ?

    a) Use the Excel menu option *File → Close*, then use the BEx analysis toolbar icon *Open → Open Workbook* and in the *History* area double click on the your new workbook to open it. You notice that the results cells return the value **#NV** because you didn't make the results available offline! Now you think you can't go back to set the analysis grid property, right? It is true that you have lost the analysis grid design item forever but you can still fix this problem using the properties of the data provider within the workbook *Settings* .

Task 2:

You would like to develop a highly customized workbook by typing directly the characteristic values and key figure names into the cells and using formulas to access the dataprovider.

1. Create a empty workbook in the BEx Analyzer.

    a) Press the *New* icon from the Excel toolbar.

2. Using the *Settings* icon define a dataprovider and assign this to the query view *Sales View 1* from the role *BW306 Reporting → Unit 2 BEx Analyzer for Experts*. You will be prompted to log on to the assigned BI system if you are not already logged on as soon as you choose the settings icon. Make sure the results data will be available offline.

    a)   Click on the icon *Settings* in the design toolbar of the BEx Analyzer.

    b)   If you are not already logged onto the BI server you will now be prompted to do so now using the details supplied by your instructor.

    c)   Select the *Data Provider* tab and press the *Create* icon.

    d)   Press the *Assign Query / Query View* icon and assign as the dataprovider the query view *Sales View 1*. You will find this under the role *BW306 Reporting →Unit 2 BEx Analyzer for Business Experts*. Double-click on the query to select it and return to the *Create Data Provider* dialog. Don't forget to select the checkbox *Provide the Results Offline*. Close the *Settings* dialog and return to the worksheet.

3. Add the heading **Sales Volume EUR** in cell B8.

    a)   Click on cell B8 and type in the heading **Sales Volume EUR**. You must type this in **exactly** as you see it here !

4. Add the month / year value **'01.2005** in cell A9, the value **'02.2005** in cell A10 and the value **'03.2005** in cell A11.

    a)   Click on cell A9 and type in the value **'01.2005**. You must type this in **exactly** as you see it here including the apostrophe. Repeat this for the cells A10 and A11 typing in the next two months in sequence.

5. Format these values by making the heading appear in bold.

    a)   Highlight all cells were you have typed values then press the *Bold* icon in the Excel toolbar.

6. In cell B9 enter the formula **=BExGet-Data("DATA_PROVIDER_1",B8,A9)**, then press enter, don't worry if you see **#NV**. All you need to do is to press the icon *Exit Design Mode* to process the results which should now appear.

    a)   Click on cell B9 and enter the formula **=BExGet-Data("DATA_PROVIDER_1",B8,A9)**, then press enter. Press the icon *Exit Design Mode* to process the results. When you pressed *Settings* you put the workbook in *Design Mode* which is why the results did not immediately appear.

Continued on next page

7. Copy the formula you just entered and paste into cells B10 and B11. Modify the formula so it references the correct cells.

    a) Highlight cell B9, press ctrl+C then highlight cell B10 and press ctrl+V. Highlight cell B11 and press ctrl+V. Modify both formulas by clicking on each cell. The second part of the reference should be changed, so for the formula in cell B10 the formula should read `=BExGetData("DATA_PROVIDER_1",B8,A10)` and the formula in cell B11 should read `=BExGet-Data("DATA_PROVIDER_1",B8,A11)`.

8. Add a total to the rows and make this bold and add a top and bottom cell border..

    a) Click on cell B12 and press the *AutoSum* icon in the Excel toolbar. Also press the *Bold* icon, then use the icon *Borders* in the Excel toolbar and select the button which shows a line for the top and bottom.

9. Add a title for the results in cell A6, use the heading `Q1 2005 Results` and format this so it is bold and large.

    a) Click on cell A6 then type in `Q1 2005 Results` and make this bold and font size 14.

10. Copy all cells for the Q1 2005 results to an empty area to the right in the worksheet and adjust the title and month values to represent the quarter 2 2005.

    a) Highlight all cells in the range A6 to B12 then press ctrl+C, then highlight cell G6 and press ctrl+V to paste the results.

    b) Modify the title to read `Q2 2005 Results` and then adjust each of the month values individually in cells G9 to G11 to use `04.2005,05.2005` and `06.2005`.

11. Repeat the steps for quarter 3 and quarter 4 2005 using the space at the bottom of the worksheet.

    a) Follow the solution steps in 10 pasting into cells starting A16 for Quarter 3 and G16 for Quarter 4. Don't forget to adjust the month values and the headings !

12. Finish off the formatting using any of your own ideas, add a title, graphics, remove gridlines etc.

    a) Use you own ideas here. Don't forget to use the *Images Library* for graphics.

Continued on next page

13. Save your workbook using the name **GR## Formula Mode Workbook 2** under the role *BW306 Reporting - Unit 2 BEx Analyzer for Business Experts*.

a) From the analysis toolbar of the BEx Analyzer select the *Save* icon. From the menu options choose *Save Workbook As*. Save your workbook to role *BW306 Reporting - Unit 2 BEx Analyzer for Business Experts* with the description of **GR## Formula Mode Workbook 2**.

Lesson Summary

You should now be able to:
- Use native Excel formula to integrate Excel functionality with BI functions

Unit Summary

You should now be able to:

*   Design and build customized workbooks using the Design Mode of the BEx Analyzer.
*   Use native Excel formula to integrate Excel functionality with BI functions

Unit 3

BEx Report Designer

Unit Overview

The BEx Report Designer is a new member of the BEx Suite for creating highly formatted reports. These are often needed for special audiences or whenever more control over the layout of report elements is needed.

Unit Objectives

After completing this unit, you will be able to:

- Create highly formatted reports for displaying on the Web and also for printing using the new Report Designer making use of the basic functions of this tool.
- Further develop formatted reports using advanced features of the BEx Report Designer.

Unit Contents

Lesson: Getting Started with the Report Designer

Lesson Overview

This lesson introduces students to the Report Designer and provides them with an opportunity to develop simple reports using various formatting options. The students are also introduced to static vs dynamic sections.

Lesson Objectives

After completing this lesson, you will be able to:

- Create highly formatted reports for displaying on the Web and also for printing using the new Report Designer making use of the basic functions of this tool.

Business Example

Some of your business reports are difficult to interpret because they contain lots of complex information. You want to find ways to improve the presentation of complex reports for both web output and also for printing

Why do we need formatted reports ?

Sometimes it is important to present the business data with an emphasis on layout and formatting. Of course it has always been preferable to create reports that are visually appealing within an organization but as more and more business reports are exposed to outside parties, often via the web, the need to present a clear professional image is vital, especially when customers or potentials customers are viewing the reports. In addition there are many reports which contain complex business data, which if presented effectively would be easier to interpret. The diagram below gives a simple example of how the same business data can be presented quite differently using formatting.

Person	Product	Period	Bonus
232	12-A	01	3443
232	12-A	02	344
232	12-A	03	322
232	12-A	04	1278
232	12-A	05	1300
		TOTAL	234324
232	12-C	06	1322
232	12-C	07	1344
232	12-C	08	1366
232	12-C	09	1388
232	12-C	10	1410
232	12-C	11	1432
232	12-C	12	1454
232	12-C	13	1476
232	12-C	01	3844
		TOTAL	343242
232	23-S	02	3764
232	23-S	03	3684
232	23-S	04	3604
232	23-S	05	3524
232	23-S	06	3444
232	23-S	07	3364
232	23-S	08	3284
232	23-S	09	3204
232	23-S	10	3124
		TOTAL	677788

… from this

… to this

Figure 30: Formatted vs. unformatted reports

The Report Designer is a component of BEx Web, which is the family of tools used to develop web based content. The BEx Web is a component of the Business Explorer, which fits seamlessly into the landscape of the BI suite. The following diagram illustrates this positioning:
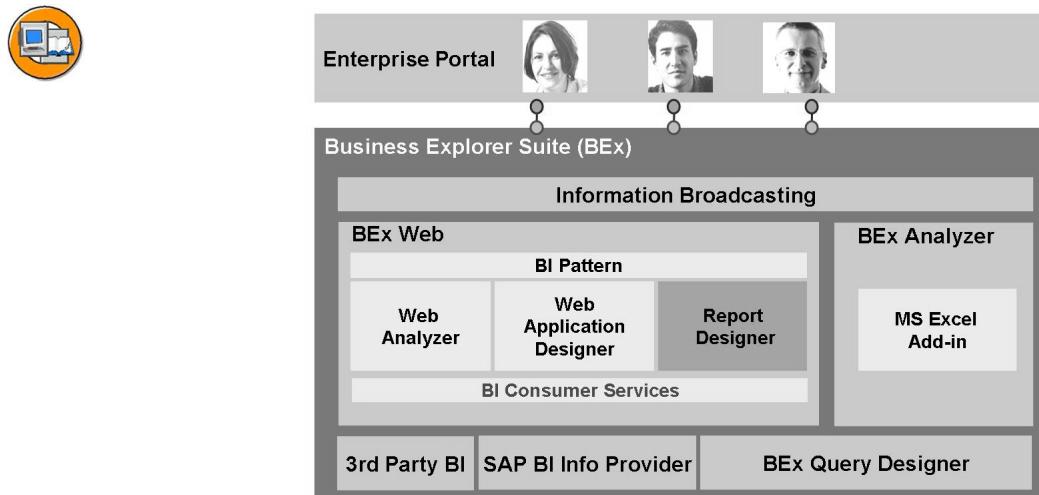
Figure 31: Position of Report Designer

With the Report Designer, SAP provides a user-friendly design tool that you can use to create formatted reports that are optimized for presentation and print. You can access a large number of formatting and layout functions. Reports created in

the BEx Report Designer can be converted to PDF documents using the connected Adobe server and saved or printed. You can also use the information broadcasting function to distribute formatted reports.
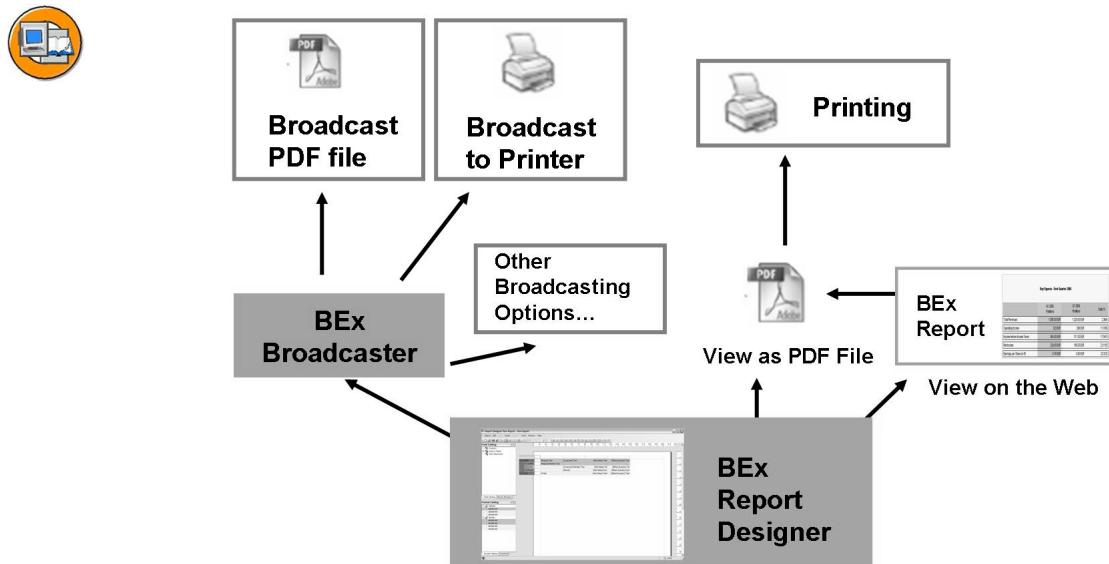
Figure 32: Report Designer output options

Report Designer Basics

You can access the Report Designer from a menu option in the Business Explorer program group. You can also access the Report Designer from the web item *Report* within the Web Application Designer. We will discuss the integration of the Report Designer in the Web Application Designer in a later section of this unit.

Once you start the Report Designer you will see that the screen is made up of sections as illustrated in the next two diagrams. Notice the use of tabs which allow you to switch between different types of information.
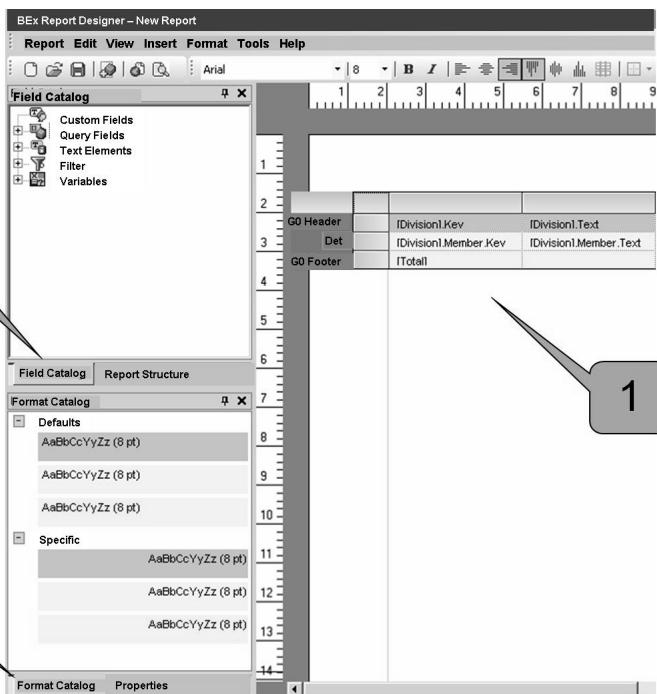
Figure 33: Layout of the Report Designer (1)
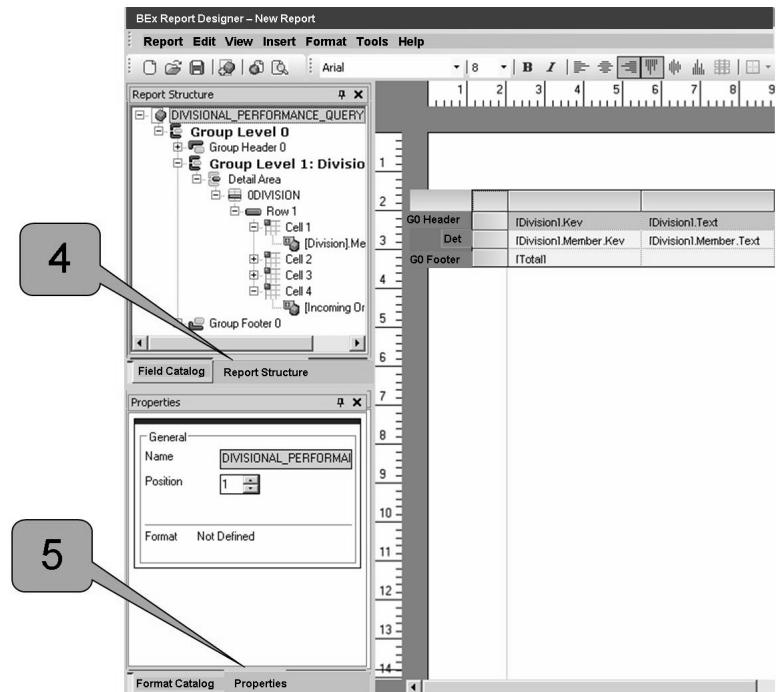
Figure 34: Layout of the Report Designer (2)

You use drag and drop to move report elements from the tab sections to the design area. The sections of the Report Designer are now described:

1. Design Area

The design area is a principal interface element of the Report Designer. You use it to create reports and to display them schematically. The design area displays report elements such as the page header and footer as well as report sections with and without data connection. You can use the context menu in the design area and the various functions provided by the menu bars and toolbars and by the other screen areas of the Report Designer to edit report elements and design your report.

2. Field Catalog

All query fields, filters, variables, and text elements for the data provider (query or query view) on which the report is based are displayed in the field catalog. The field catalog also contains the list of custom texts created by users, which were inserted into the header area of the report, for example. Using the field catalog, you can insert or reuse query fields or text elements for the data provider that were deleted from the report, for example, at any time with drag and drop.

3. Report Structure

The report structure hierarchically presents the report and all its visible and hidden components. As in the design area, the report structure also has a context menu with which you can work. In contrast to the design area, in which you cannot select objects such as group levels, you can select and edit these types of objects in the report structure

4. Format Catalog

The format catalog provides an overview of the formats used in the report. Both standard formats and user-defined formats are displayed. You can transfer the formats to report elements (rows or cells) using Drag & Drop.

5. Properties

In the properties area, you can specify the properties of individual report elements (fields, rows, or cells). To do this, you select the required field, row, or cell and make the required settings in the properties area. This allows you to specify, for example, the format and position of text fields and the height and width of rows and columns.

Developing your first report

Before you can begin designing your new formatted report you have to identify the source of the data. The Report Designer is able to use the a query or a query view developed with the Business Explorer as the source of data which will be presented in the report. It is possible to use multiple queries and query views in a single formatted report, we will discuss this capability later in the Unit.
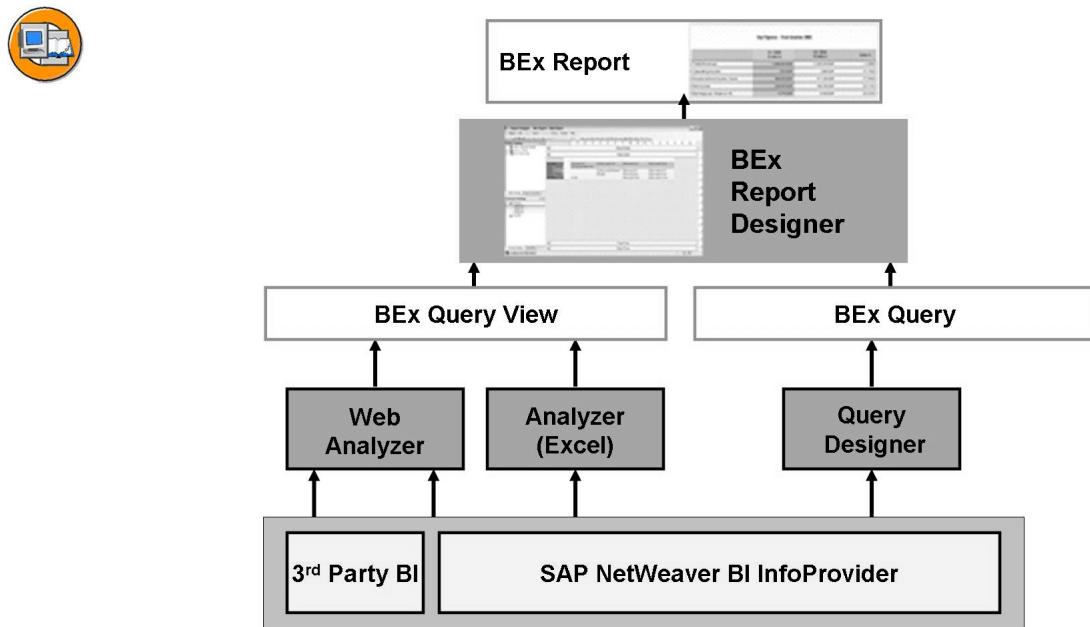
Figure 35: Data Providers available to Report Designer

There are four key steps in the process of creating a report and viewing the results. The steps are described here and are also indicated on the following graphic.
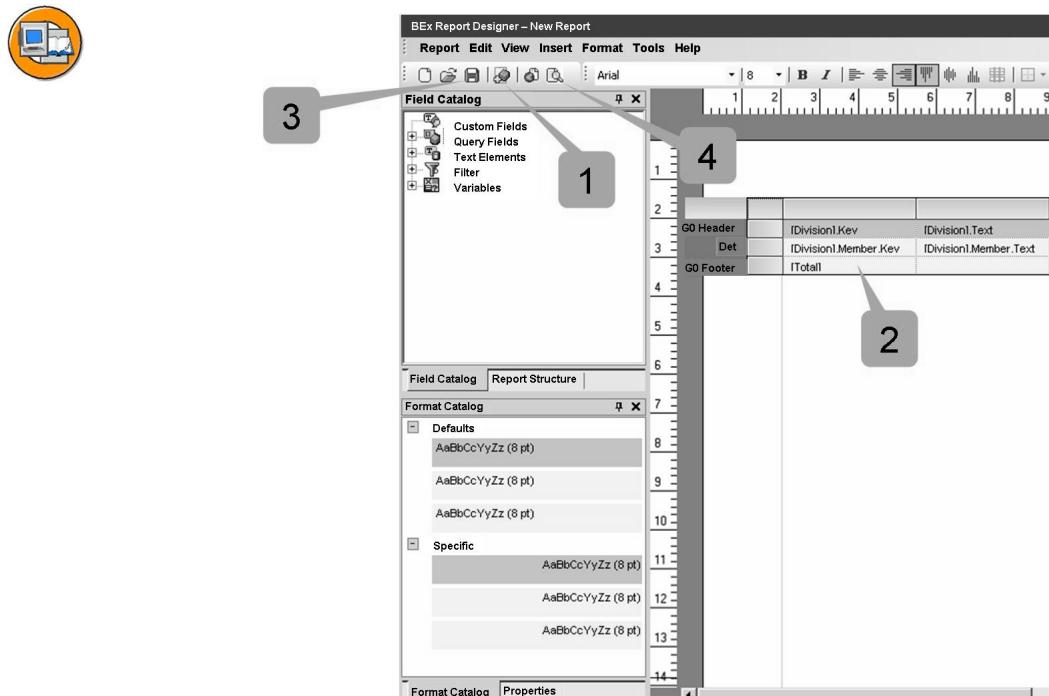
Figure 36: Steps to create your first report

Step 1 - Inserting a Data Provider You use the Insert Data Provider function to select a suitable query or query view as the data provider. The selected data provider is automatically embedded in a report section and is displayed together with the associated row patterns in the design area of the Report Designer. If you want to create a new query or change an existing one, call the BEx Query Designer from the Tools menu in the menu bar for the Report Designer. If you want to create a new query view or change an existing one, call the BEx Web Analyzer from the Tools menu in the menu bar for the Report Designer (note : you could also create or change a query view from the Analyzer, however there is no link to this tool directly from the Report Designer).

Step 2 - Designing the Report You create the design of the report. There are many options available for this purpose. You can use standard formatting functions such as selecting the color, font, and border for individual report elements (cells or fields). You can make layout changes, for example, adapt the height and width of rows and columns as required, insert custom text, and rearrange query fields and text elements. You can design the header and footer areas of the page and the report. You can insert images and custom texts into these areas or integrate totals fields for the query into the header area using drag and drop, for example.

Step 3 - Saving the Report You save the report using the save dialog. Once you have saved the report, you can also use it in Web applications into which you have inserted the Report Web item.

Step 4 - Executing the Report You execute the report in the portal (Web). The report is displayed in the standard Web template for reports. You can also create a print version of the report directly in the Report Designer. The report is converted into a PDF document, which you can then print.

Basic Formatting Options

Formatting options can be broken into groups. There are formatting settings which affect the **cell** such as :

-      Alignment
-      Background color
-      Word wrap

There are formatting options which affect the text such as :

-      Font (ariel, comic sans etc)
-      Font style (bold, italic etc)
-      Font size
-      Font color

There are formatting options which affect the borders such as :

• Line style, width and color
• Cell padding

There are general formatting options such as :

• Adding new columns and rows
• Adjusting column and row widths
• Overwriting text headings
• Moving cells to alternative positions in the layout

You access the formatting options from a number of places. You can access many formatting options directly from the context menu of the cells. You can also access the formatting options by highlighting the cells you want to change and then using the toolbar buttons or the *Format* menu.

If you would like to apply the same formatting choices to multiple areas of your report, then you should make use of the *Format Catalog*. This features allows you to collect a number of formatting changes and then apply then using drag and drop on to the cells in the reports. A format is created automatically for each unique set of formats you build.

> **Note:** Be careful ! Dragging a format from the format catalog overwrites any changes you have made earlier to cells. For example, you manually change the font size of a cell in the report from the context menu. You then drag a format from the format catalog over this cell. The font size will immediately be adjusted to the setting in the format catalog and your earlier change will be overwritten. Make sure the predefined format in the catalog contains all the formatting you require, alternatively make sure you add any additional formatting to the cells **after** you have dragged a format from the catalog.

As well as changing individual cells you are able to change entire rows, columns or even the entire report at one time, by pressing the grey buttons around the edge of the report in the design area. You can also use the *ctrl* button to mark the individual cells you would like to make a group change to. These cells do not need to be contiguous.

It is possible to change text in any heading cell. This means you can override any characteristic description or key figure name. Changing text is done in two steps. First you may want to delete any existing contents in the field (unless you are appending your own text), however deleting the contents is not mandatory. To delete contents use the *Delete Contents* option from the context menu. You then use the option *Add Custom Text* from the menu option to add your own text. This creates a placeholder for your text. The final step is to use the context menu to select the option *Edit Text ...* so that you may enter your own customized text.

 SAP
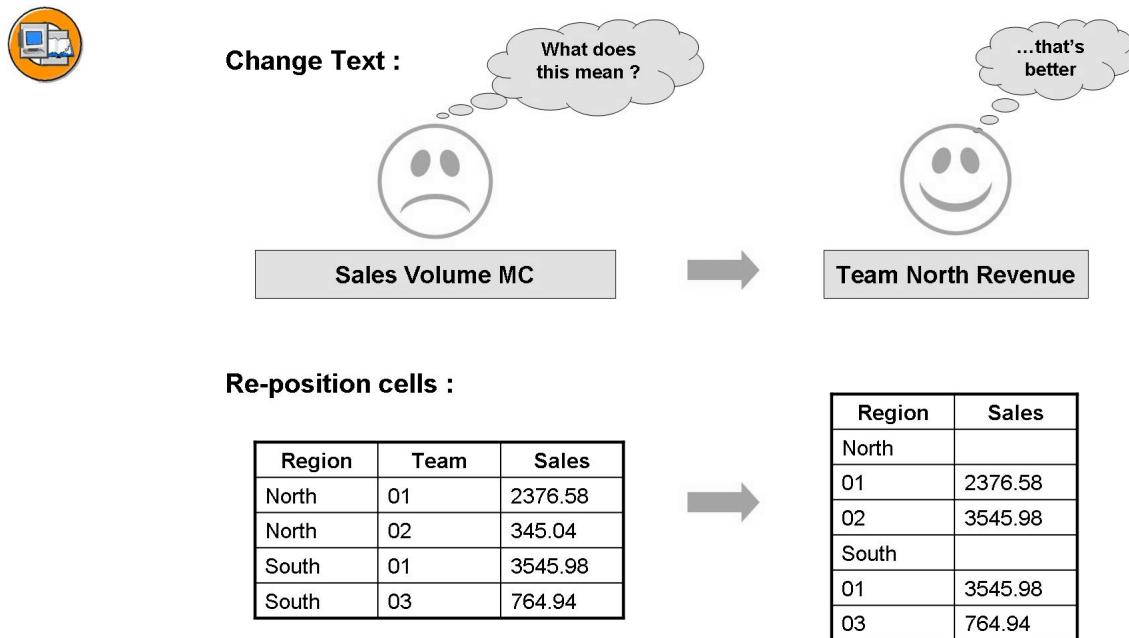
Figure 37: Formatting and re-positioning cells

You can change the position of cells in order to improve the presentation of your results. The options available for re-positioning cell are governed by the type of **section** you are working with in the report. A section is used to pass the data provider results to the report. There are two types of section, **static section** and **dynamic section**. You do not specify the type of section in the Report Designer yourself, the section type is automatically set depending on the type of data provider you are using. A query (or query view) which contains two structures is the basis for a static section. There must be one structure in the rows and one structure in the columns. With a static section you can freely position the results cells. This is possible because each cell of the result is unique defined and so the formatting and positioning information is related directly to an individual cell.

Static Section

- Every result cell is uniquely defined
- Offers most flexible cell re-positioning options

	[Struct.].Text	Incoming Orders	Sales Volume
	01.2005	[Incoming Orders/01.2005].Val	Sales Volume EUR/01.2005].Val
	02.2005	[Incoming Orders/02.2005].Val	Sales Volume EUR/02.2005].Val
	03.2005	[Incoming Orders/03.2005].Val	Sales Volume EUR/03.2005].Val
	04.2005	[Incoming Orders/04.2005].Val	Sales Volume EUR/04.2005].Val
	05.2005	[Incoming Orders/05.2005].Val	Sales Volume EUR/05.2005].Val
	06.2005	[Incoming Orders/06.2005].Val	Sales Volume EUR/06.2005].Val
	07.2005	[Incoming Orders/07.2005].Val	Sales Volume EUR/07.2005].Val
	08.2005	[Incoming Orders/08.2005].Val	Sales Volume EUR/08.2005].Val
	09.2005	[Incoming Orders/09.2005].Val	Sales Volume EUR/09.2005].Val
	10.2005	[Incoming Orders/10.2005].Val	Sales Volume EUR/10.2005].Val
	11.2005	[Incoming Orders/11.2005].Val	Sales Volume EUR/11.2005].Val
	12.2005	[Incoming Orders/12.2005].Val	Sales Volume EUR/12.2005].Val

Figure 38: Static Section

A query (or query view) with only one structure is the basis for a dynamic section. Within the dynamic section, a **group level** is provided to represent each characteristic in the query. The cell re-positioning options are limited with a dynamic section because the cells are not all uniquely defined. For example, with a single structure query you do not know what will appear in a particular row of the report as this depends on the data available from the data provider. The Report Designer could not possible let you commit to formatting or moving a cell when it doesn't know itself what the cell will contain ! Re-positioning cells can only be done by moving a cell from an external group level to an internal group level. For example, you can move a cell from group level 1 to group level 2, but you cannot move the other way. You can also move a cell within its own group, i.e. horizontally to change to column order.

Dynamic Section

- We don't know the result of each cell until runtime (the cells are dynamic)
- Re-positioning only from external group to internal group (or within same group)
- We can't format each result row individually, we have to format the group
- We can format each column within a group individually

Group Level 0
Group Level 1
Group Level 2

Figure 39: Dynamic Section

The diagram above shows a dynamic section with three group levels, level 1 and level 2 relate to the two characteristics in the query, group level 0 relates to the header (there will always be a group 0)

Note: If a query (or query view) contain two structures and also one or more characteristics then a dynamic section will be used. The dynamic section will generate individual cells for each intersection of the two structures, this means you will be able to re-position those cells freely. Also you cannot use a query or query view which contains more than one structure in the columns.

- Within group level 1 there are three row types
- Define a pattern (font, color etc.) for each row type within a group level
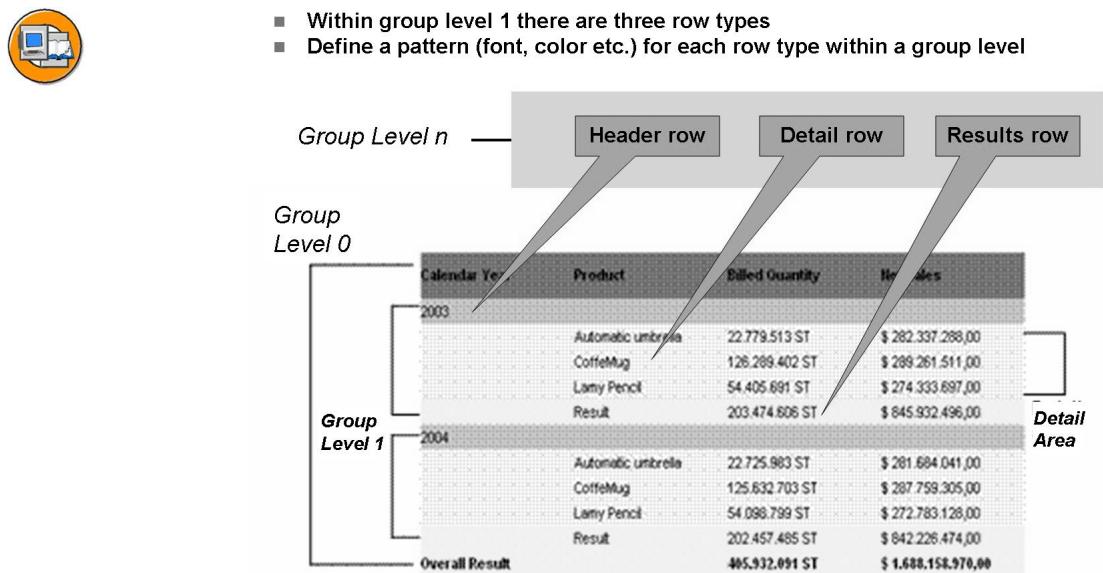
Figure 40: Row Pattern within Group Levels

The diagram above illustrates how, for every group level, there are three row types, header, detail and footer. For each of the row types you can apply unique formatting. We call the set of formatting choices a row pattern.

Once you have developed your formatted report you have various output options. You can present the output on the web. The Report Designer uses a standard web template to organize the standard features such as the print and broadcast buttons (your formatting from the Report Designer is not modified by the template). Another output option is printing. Printing is handled by the Adobe document service. You can press the *Print Version* button from within the Report Designer or you can also press the same button from the web output. Regardless of which option you choose you will be presented with a PDF output of the report initially, you can they decide is you would like to save the PDF to you own specified location or you can print the report to your specified printer.

- **Web page**
- **PDF**
- **Printout**
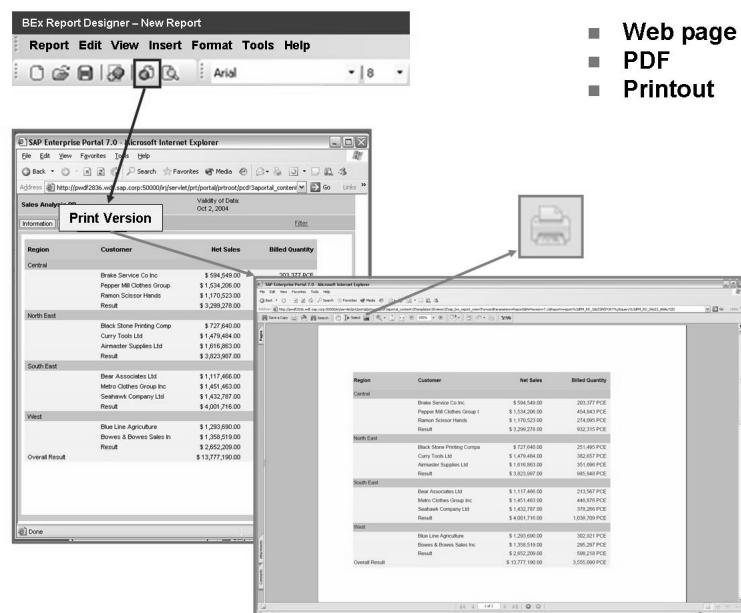
Figure 41: Execution Options

Exercise 3: Create a basic report using the Report Designer

Exercise Objectives

After completing this exercise, you will be able to:

- Develop a formatted report using the basic formatting features of the Report Designer.

Business Example

You need to improve the presentation of a sales report so that it can be published on your intranet with a professional image. The report will present sales results in monthly summary form and also in a more detailed breakdown by distribution channel and division. You will use the Report Designer to achieve this.

Task 1:

Develop a report which contains a static section for the monthly summary of sales figures. You will change the layout of the results cells and improve the presentation by using multiple formatting options. View the results on the web.

> ⚠️ **Caution:** Make sure your portal is set to the desired language before you begin this exercise. Your instructor may already have asked you to do this. If not then you can change the portal language by simply logging on to the assigned training system using the SAPGUI (use the Windows Start button shortcut) and then use the *Personalize Portal* menu option. Log onto the portal using your assigned user id and password and then press the button *'Ändern'* (this is the German word for Modify). Look for the drop down list '*Sprache'* and find the entry *English*, finally press the button *'Sichern'* to save your changes. Close the browser.

1. Open the Report Designer and log on using your assigned systems details.

2. Insert a data provider based on query *Summary Static* which you will find under the role *BW306_Reporting → Unit 3: BEx Report Designer*.

3. You do not wish to see the key figure *Sales Volume EUR* so remove this from the report.

4. You would like to improve the layout by moving the months *July 2005* to *December 2005* to appear alongside the first six months. You will need to move both the month descriptions and the key figure values for each month. Remove any empty cells that remain.

5.   Add a small amount of vertical space between the two halves of the year 2005 by inserting an empty column and reducing its width.

6.   Change the cell background color for all the month values (which you should see in the first and fourth columns of the report). Choose any color you wish and also make the text bold. Also make sure the values are left aligned in the cell. **Important** : Make all the changes using the context menu of the first cell, but then drag this newly created format from the format catalog to the remaining cells so they are formatted in the same way.

7.   Overwrite the heading text for the month values with the custom text **Sales Period**. Important : Type in the custom text into the heading for the first month value column, **but** you must not manually enter the custom text for the second month value column - use the Field Catalog and drag the custom text into the second heading.

8.   Format all the headings so that all text is bold and size 14, horizontally and vertically centered, **but** make these changes using the toolbar icons, not the context menu of the cells. You should also increase the height of the heading row.

9.   Add a background color to all the heading cells, choose any color you like, **but** do this using the *Report Structure* tab.

10. Save your report under the role path *BW306_Reporting → Unit 3 BEx Report Designer* using the name **GR## BW306 Report 1** and technical name **GR##BW306REPORT1**.

11. Execute the report on the web and admire your formatted report. You will need to log onto the portal using your assigned user id and password. Are you able to filter on the month values ?

Task 2:

You now want to add a breakdown of the sales results by division and distribution channel directly beneath the month summary. You will do this by inserting a new section, but this will be a **dynamic** section. You will make changes to the layout of the report and through the exercise discover how dynamic sections and static section offer different layout change possibilities.

1.   Make sure your report **GR## BW306 Report 1** is displayed in the Report Designer.

2.   You want to create some space between the results so insert a new section at the bottom of the report.

3.   Add a title for the new section, use the text **Breakdown by Product Line and Channel**. Format this text using your own ideas, but make this text really stand out.

Continued on next page

4. Insert a new data provider using the query *Breakdown Dynamic*.

5. Increase the row space for the heading you entered earlier and make sure the heading text is vertically aligned towards the bottom of the cell.

6. You do not want to show they key values for either *Distribution Channel* or *Division* so remove these columns from the report.

7. You want the *Division* to appear underneath the *Distribution Channel* in the same column.

8. You should now remove the unused column which now only contains the heading *[Division].Text*

9. Remove the subtotal rows, you do not want to see a total for each *Distribution Channel*. Do not delete the final result total.

10. Save your report under the role path *BW306_Reporting → Unit 3 BEx Report Designer* using the new name **GR## BW306 Report 2** and technical name **GR##BW306REPORT2**.

11. Execute the report on the web and admire your formatted report. You will need to log onto the portal using your assigned user id and password.

12. View the PDF of your output directly from the web.

Solution 3: Create a basic report using the Report Designer

Task 1:

Develop a report which contains a static section for the monthly summary of sales figures. You will change the layout of the results cells and improve the presentation by using multiple formatting options. View the results on the web.

> ⚠️ **Caution:** Make sure your portal is set to the desired language before you begin this exercise. Your instructor may already have asked you to do this. If not then you can change the portal language by simply logging on to the assigned training system using the SAPGUI (use the Windows Start button shortcut) and then use the *Personalize Portal* menu option. Log onto the portal using your assigned user id and password and then press the button *'Ändern'* (this is the German word for Modify). Look for the drop down list '*Sprache'* and find the entry *English*, finally press the button *'Sichern'* to save your changes. Close the browser.

1. Open the Report Designer and log on using your assigned systems details.

   a) Using the path *Start → Programs → Business Explorer → Report Designer* open the Report Designer. You will be prompted to log on, use the logon details assigned to you by your instructor.

2. Insert a data provider based on query *Summary Static* which you will find under the role *BW306_Reporting → Unit 3: BEx Report Designer*.

   a) Press the icon *Insert Data Provider* from the toolbar and select the query *Summary Static* which you will find under the role *BW306_Reporting → Unit 3: BEx Report Designer*.

3. You do not wish to see the key figure *Sales Volume EUR* so remove this from the report.

   a) Click on the grey button above the heading *Sales Volume EUR* to highlight the entire column then right-click to access the context menu and select the option *Delete Column*.

4. You would like to improve the layout by moving the months *July 2005* to *December 2005* to appear alongside the first six months. You will need to move both the month descriptions and the key figure values for each month. Remove any empty cells that remain.

   a) Insert two new columns by clicking on the button above the key figure *Incoming Orders* to highlight the entire column. Then right-click to access the context menu and select the menu option *Insert Column (right)*. Repeat this step to add a second new column.

   b) Click once on the cell containing the value *07.2005* then drag the content of this cell to appear alongside *January 2005* in the first empty column. Repeat this steps for each of the remaining months up to *December 2005*. You should now have an equal number of months side by side.

   c) Move the key figures for the months *July 2005* to *December 2005* to appear alongside the month descriptions, using the same technique as described in the previous step.

   d) Highlight the first empty row at the bottom of the report by pressing the grey row button and then right-click to access the context menu. Select the option *Delete Row*. Repeat this step for the remaining rows.

5. Add a small amount of vertical space between the two halves of the year 2005 by inserting an empty column and reducing its width.

   a) Highlight the second column then right-click to access the context menu, then select *Insert Column (right)*.

   b) Put the cursor carefully on the edge of the column until you see the cursor pointer change to a double arrow, then drag the line to reduce the width.

6. Change the cell background color for all the month values (which you should see in the first and fourth columns of the report). Choose any color you wish and also make the text bold. Also make sure the values are left aligned in the

Continued on next page

2007/Q2   **105** SAP

cell. **Important** : Make all the changes using the context menu of the first cell, but then drag this newly created format from the format catalog to the remaining cells so they are formatted in the same way.

a)   Right-click on the first month value cell and from the context menu select the option *Cell Format → Background Color* and choose any light color. Press *OK* to return to the results.

b)   Right-click on the same cell to select the menu option *Text Format → Bold.*

c)   Right-click on the same cell and select the menu option *Cell Format → Horizontal Alignment → Left Align*.

d)   Locate the new format in the Format Catalog which will have been created automatically. Drag this format and drop it onto each of the remaining month value cells so the formatting is applied. All month value cells should now look the same.

7.   Overwrite the heading text for the month values with the custom text **Sales Period**. Important : Type in the custom text into the heading for the first month value column, **but** you must not manually enter the custom text for the second month value column - use the Field Catalog and drag the custom text into the second heading.

a)   Right-click on the heading cell for the month values and select the menu option *Delete Contents* then right-click again to select the menu option *Insert Custom Text.* Right-click to select the menu option *Edit Text* then add the text **Sales Period**. (you may need to click outside the cell then click back in before this menu option appears).

b)   If you look under Field Catalog you will find the custom text you just entered available in the Custom Text node. Drag this text into the heading cell for the second column of month values. You will need to delete the original contents of the cell using the context menu.

8. Format all the headings so that all text is bold and size 14, horizontally and vertically centered, **but** make these changes using the toolbar icons, not the context menu of the cells. You should also increase the height of the heading row.

    a)   Highlight the entire heading row of the report by pressing the button to the left of the first cell, this will make all cells in the row appear darker.

    b)   Click on the icon *Bold* then from the *Font Size* drop down list choose size 14.

    c)   Press on the icon *Centered* to center the text in the cells, then press the icon *Vertically Centered*.

    d)   Increase the height of the row to accommodate the new larger text by dragging the horizontal row bar.

9. Add a background color to all the heading cells, choose any color you like, **but** do this using the *Report Structure* tab.

    a)   Click on the tab *Report Structure*.

    b)   Expand the structure by clicking on the + symbols so that you reach the following component : *Group Level 0 → Group Header 0 → Headers → Row 1*.

    c)   Right-click on the component *Row 1* and select the menu option *Cell Format → Background Color*. Select any color you like then press *OK*.

10. Save your report under the role path *BW306_Reporting → Unit 3 BEx Report Designer* using the name **GR## BW306 Report 1** and technical name **GR##BW306REPORT1**.

    a)   Click on the icon *Save* from the toolbar or use the menu option *Report → Save*, then press the *Roles button* and follow the path *BW306_Reporting → Unit 3 BEx Report Designer*. Enter the report name **GR## BW306 Report 1** and a technical name **GR##BW306REPORT1** and press *OK* .

11. Execute the report on the web and admire your formatted report. You will need to log onto the portal using your assigned user id and password. Are you able to filter on the month values ?

    a)   Press the icon *Execute* from the toolbar and when prompted enter your user id and password assigned to you by the instructor. If the portal logon screen appears in the wrong language this means you have not set the language under the personalization settings correctly. See the instructions at the beginning of this exercise.

    b)   YES - you can filter on the month values using the *Filter* settings.

Continued on next page

Task 2:

You now want to add a breakdown of the sales results by division and distribution channel directly beneath the month summary. You will do this by inserting a new section, but this will be a **dynamic** section. You will make changes to the layout of the report and through the exercise discover how dynamic sections and static section offer different layout change possibilities.

1. Make sure your report **GR## BW306 Report 1** is displayed in the Report Designer.

   a) If you need to re-open this report you will find it under the role path *BW306_Reporting → Unit 3 BEx Report Designer*.

2. You want to create some space between the results so insert a new section at the bottom of the report.

   a) Right-click anywhere in the empty space below the results cells and from the context menu select the option *Insert Report Section*.

3. Add a title for the new section, use the text **Breakdown by Product Line and Channel**. Format this text using your own ideas, but make this text really stand out.

   a) Click on the empty cell of the new section (there will only be one cell in the section) and right-click to choose the context menu option *Insert Custom Text*. Click on the cell again and choose the context menu option *Edit Text* and enter **Breakdown by Product Line and Channel**.

   b) Click on the row button for the new text and then press the *Bold* icon in the toolbar, also use the font size drop down list to choose size 13.

4. Insert a new data provider using the query *Breakdown Dynamic*.

   a) Right-click anywhere in the empty space at bottom of the report and from the context menu choose the option *Insert Data Provider*. From the Open dialog choose the query *Breakdown Dynamic* which you will find under the role *BW306_Reporting → Unit 3: BEx Report Designer*. Click *OK*.

5. Increase the row space for the heading you entered earlier and make sure the heading text is vertically aligned towards the bottom of the cell.

   a) Drag the line immediately below the heading text cell (you should see the pointer change to a double arrow) so that the cell in double the height.

   b) Click on the row button for the heading and from the toolbar press the icon *Align Bottom*.

6. You do not want to show they key values for either *Distribution Channel* or *Division* so remove these columns from the report.

    a)   Locate the column where the *[Distribution Channel] Key* appears (this should be the first column). Right-click on the column button and from the context menu select *Delete Column*. Do this also for the column containing *[Division] Key*.

7. You want the *Division* to appear underneath the *Distribution Channel* in the same column.

    a)   Click once on the cell *[Division] Member.Text* until you see the cell appear grey with a rectangle within the cell, this means you have selected the cell.

    b)   Drag the cell to and drop it into the cell immediately to the left, this should now appear directly underneath the cell *[Distribution Channel].Member.Text*.

8. You should now remove the unused column which now only contains the heading *[Division].Text*

    a)   Right-click on the column button where you see *[Division].Text* and from the context menu select *Delete Column*.

9. Remove the subtotal rows, you do not want to see a total for each *Distribution Channel*. Do not delete the final result total.

    a)   Right-click on the row button for G1 Footer and from the context menu select *Delete Row*.

10. Save your report under the role path *BW306_Reporting → Unit 3 BEx Report Designer* using the new name **GR## BW306 Report 2** and technical name **GR##BW306REPORT2**.

    a)   From the toolbar use the menu option *Report → Save As* then press the *Roles button* and follow the path *BW306_Reporting → Unit 3 BEx Report Designer*. Enter the report name **GR## BW306 Report 2** and a technical name **GR##BW306REPORT2** and press *OK* .

11. Execute the report on the web and admire your formatted report. You will need to log onto the portal using your assigned user id and password.

    a)   Press the icon *Execute* from the toolbar and when prompted enter your user id and password assigned to you by the instructor.

12. View the PDF of your output directly from the web.

    a)   From the web output press the button *Print Version* and when prompted press the *Open* button to view the PDF.

2007/Q2   **109** SAP

Lesson Summary

You should now be able to:

- Create highly formatted reports for displaying on the Web and also for printing using the new Report Designer making use of the basic functions of this tool.

Lesson: Detailed Settings and Formatting Options

Lesson Overview

This lesson introduces students to the advanced features of the Report Designer

Lesson Objectives

After completing this lesson, you will be able to:

*   Further develop formatted reports using advanced features of the BEx Report Designer.

Business Example

You can now create formatted reports using the basic features of the Report Designer. You now want to learn more about the advanced features of the tool to see how your reports can be further improved.

Hierarchy Formatting Options

As well as formatting tabular reports using the Report Designer you can also format reports which present the results using hierarchies. A unique format can be applied to each individual node level of the hierarchy so that the results can be emphasized to make it easier for the user to interpret the report. All the usual formatting options are available for the hierarchy levels including :

*   Text size
*   Text color
*   Text font type
*   Bold / italic
*   Cell height / width
*   Cell color
*   Border style

		Sales
Region Sales		**$22000**
Region	North	$14500
Team	*A*	*$6500*
	Hi Tech Inc	$2000
	Arrow Corp.	$3500
	Innovations Inc	$1000
Team	*B*	*$8000*
	New Wave Corp	$3000
	Dazzlezone Inc	$3000
	Edgefire	$2000
Region	South	$7500
Team	*C*	*$7500*
	Bluetube Corp	$7000
	Toplink	$500

Node level formatting for :

- Text color
- Text size
- Font type
- Bold
- Italic
- Cell color
- Cell height, width
- Borders

Figure 42: Formatting Hierarchies

Inserting Images

Images can be inserted into a report. For example you may want to add you company logo in the header, or a strapline graphic in the footer. The graphics available to the Report Designer must be stored in the MIME Repository of BI. The MIME Repository is a central store for all graphical objects used in the BEx and can handle all type of graphical object types such as .gif, .jpg, .bmp etc. but for images you want to use in the Report Designer they must be **.gif** file formats. Graphic objects are placed into the MIME Repository via a check-in procedure. You access the MIME Repository with transaction *S02_MIME_REPOSITORY* or you can find the link from the menu path *Business Explorer → MIME Repository* from the SAP standard BI menu.
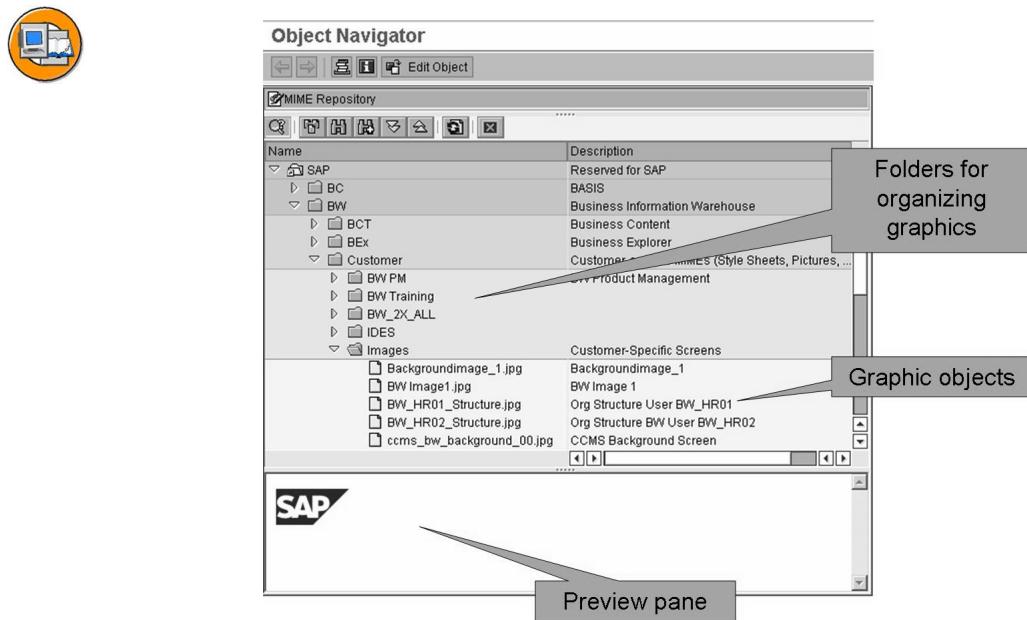
Figure 43: MIME Repository in BI

If you would like to add you own images to the MIME Repository simply right-click on the folder which should contain the image and use the menu option *Import MIME Objects*. You will be prompted for the location of the file. Once you have specified the location of the file you will then be asked to provide a name for the MIME object as it will be know in BI. If you should ever need to modify the image then you should use the *Downoad* menu option which you will find when you right-click on the image file in the MIME Repository.

To upload the same image back once it has been modified you should right-click on the same image in the MIME Repository and use the menu option *Upload/Download → Upload and Replace*.

The procedure for inserting an image into a report is very simple.

1. Right-click on any cell where you would like to image to appear and from the context menu select *Insert Image*.
2. You will now see a small icon in the cell, simply click on this icon and you will see the icon disappears and the cell then contains a grey box with the word *Image* in the center.
3. The *Properties* tab should now be showing the setting for the image. You need to supply the image file name (and path) in the MIME field.
4. Consider changing the image size in the properties pane.
5. You need to save then execute the report to actually see the image, it does not appear in the report design layout area.

specify image location

choose size

make sure you select this tab
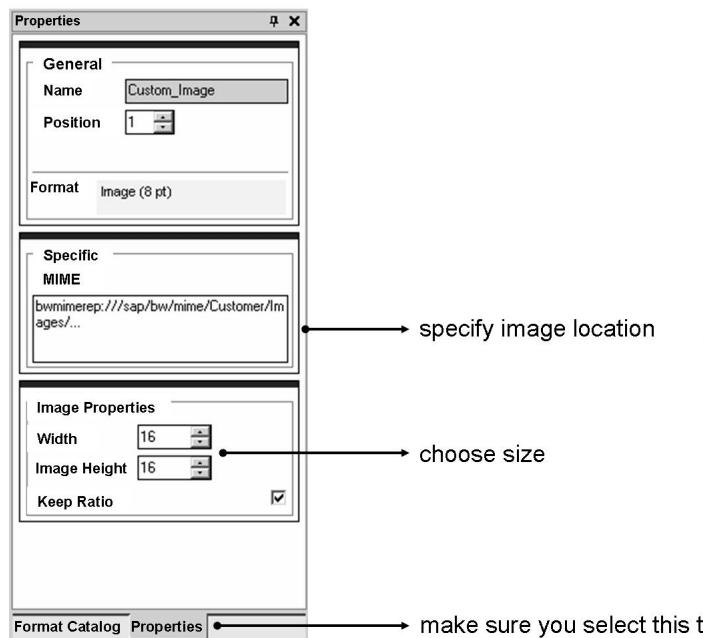
Figure 44: Inserting an Image into a Report

Headers, Footers and Sections

Inserting headers and footers in your report provides a great way to add a professional look and feel, as well as providing space to add additional useful information. From the Report Designer main menu use the option *Insert → Page Header* or *Insert → Page Footer* to begin. Once you have used one of these options you will notice an extra section of the report has appeared with one empty cell to get you started. Within the cell you can insert :

- Custom text
- Images
- Text fields
- Filter values
- Query fields
- Variable values

You can also insert more cell if you need them and, of course, re-size the cells. The cells and any content can be formatted using any of the regular formatting options found in the Report Designer.
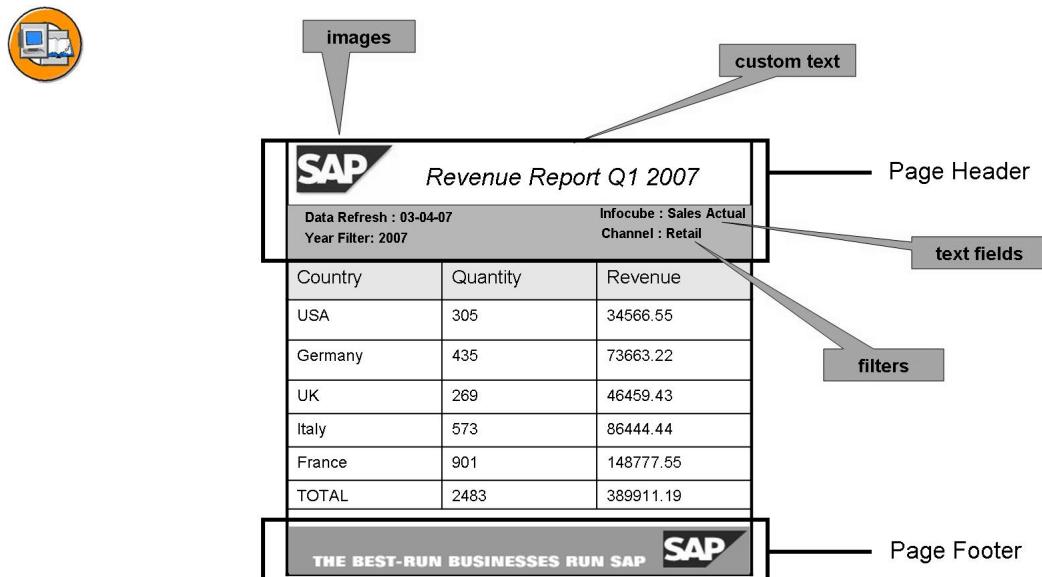
Figure 45: Headers and Footers

As well as headers and footers you can also add a **section**. A section is automatically inserted when you add a data provider, but it is possible to add a section without a data provider, or to put it another way, an empty section. Use the Report Designer main menu *Insert → Report Section* to do this. You can insert as many sections as you require into the report. Sections provide you with more report cells so you can add more custom text, text fields etc.

Field Catalog

Apart from the report results, a well formatted report needs to present information about the general runtime environment. Information such as the date of the data refresh, a reminder of the current filter values and variable values, the name of the data provider and many more fields are available in the field catalog. Most of the fields from the field catalog can be inserted into any report cell, though typically many of these fields will be inserted into either the header or footer areas. Don't forget you can apply formatting to these fields.

Hint: Within the header or footer, format a range of empty cells with borders to create the impression of a table. Include a description of each field in one column then insert the field in another column. This will ensure you fields are presented in a tidy way.

You can also insert more than one field into a cell, the fields will sit side by side. Simply click on the tab to open the field catalog pane and you will see the field have been divided into five categories:
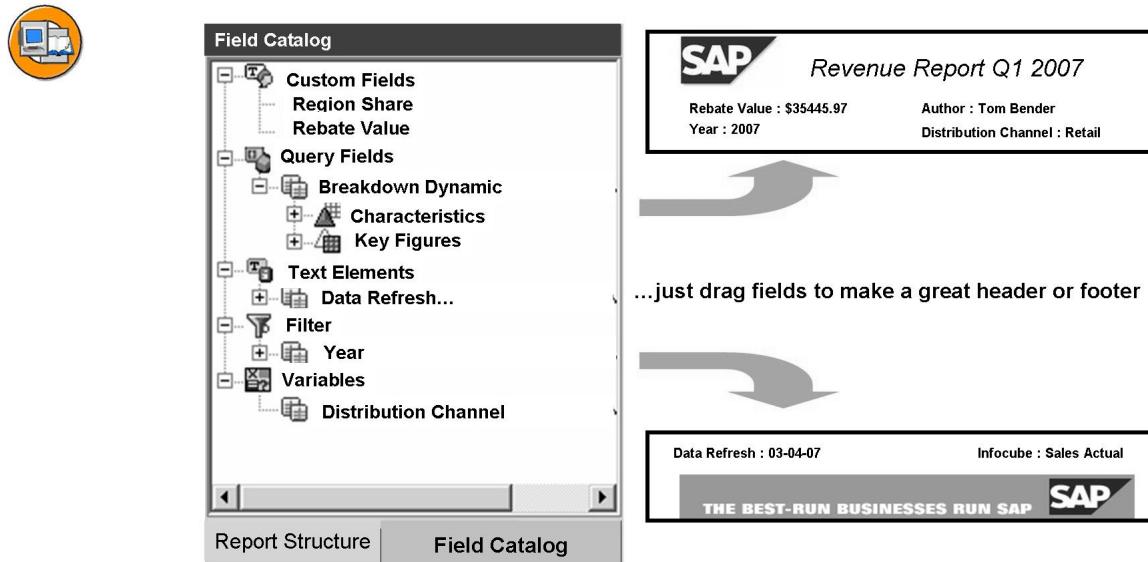
Figure 46: Field Catalog

Query Fields

The query fields represent all the values of characteristics and key figures available in the data provider. This means you have access to every characteristics member description and key as a field which you can use in a header or footer. You also have access to every key figure for each member and also the totals of each key figure cumulated to every characteristic. Imagine being able to present the total sales revenue amount in the header of the report, before the results have even appeared ? Well you can.

It is important to note that there are restrictions which apply to the placement of the individual characteristic members. These can only be placed in the group level of the characteristic (or a more internal group level). For example, if your report contains *Month* and *Region* (in that order) you can only place the query field for region (key or description) in the group level for the region. You could not place this in the group level for month and certainly not in the header and footer sections. This, of course, makes sense as each characteristic member and its related key figures are only known at the time of row generation for the group level. All other query fields, such as the final result of the query, can be placed anywhere in the report, including within any group level.

Text Elements

The text elements represent a comprehensive range of informational parameters which can be used to provide the user with useful supporting information to the report. These include **Date of data refresh, query name, author** and many others.

There are no restrictions for placement of the these text elements in the report.

Filters

Filter fields are used to remind the user of the current filter values applied to the report. This is especially useful if the filters are not obvious as they may be background filters. There are no restrictions for placement of these text elements in the report.

Variables

Variable fields are just like filter fields except they are used to specifically remind the user of the variable values chosen. There are no restrictions for placement of these text elements in the report.

Custom Fields

Custom fields are automatically generated whenever you create custom text in the report. Every unique custom text you define in the report cells is automatically added to the custom text fields list. You can then re-use the custom text fields and place them wherever you want in the report.

Conditional Formatting

Using conditional formatting, you can format selected characteristics (such as customer XY) differently. In the executed report, these characteristic values have the formatting you specify and are set apart from the other characteristic values in your group level. Conditional formatting only applies to dynamic sections.

> **Note:** Conditional formatting is not the same as query conditions, be careful not to confuse these two BEx features.

Karaoke Superstore (Philadelphia) - Sales Q1 2007

Month	Product	Orders	Return
January	Karaoke Unit AC65	4755	3
	Microphone Basic W32	**422**	**0**
	CDG Soft Rock Hits	3422	12
February	CDG 1980's Hits	234	
	Karaoke Unit R44	3358	8
	Microphone Basic W32	**234**	**0**
March	CDG Country Hits	4389	54
	Karaoke Unit Y763	32	0
	Microphone Wireless A2	44	2

Microphones are this months free gift !

- Emphasize any characteristics using formatting
- This is a check of the characteristic value not the key figure value !

Figure 47: Conditional Formatting

The steps required to define conditional formatting are as follows:

1. Insert a data provider with a dynamic section.

2. In the design area, select a row in the group level of the characteristic for which you want to apply conditional formatting to its characteristic value(s).

3. In the menu bar for the Report Designer, choose *Format → Conditional Formatting*. The input help dialog appears.

4. In the input help dialog, select a characteristic value for which you want to create the conditional formatting. In the design area, an additional row is displayed for the selected characteristic value beneath the selected characteristic row. This new row has its own row pattern.

5. Select the new row created for the characteristic value and format as required using the formatting functions.

6. To add more characteristic values simply repeat steps 3 to 5.

7. Save the report and execute it to check the required characteristic value(s) are formatted as you expected.

Themes

Of course the main purpose of the Report Designer is to provide you with the opportunity to format your report contents by providing specific cell level formatting options. However it is also worth remembering that the Report Designer uses **themes** to provide a starting point (also known as default formats). A theme provides a set of formatting instructions to your report cells, but the important things to bear in mind is that the formatting supplied by the Report Designer custom settings take priority over the default settings supplied by the theme. The theme is supplied by SAP as Business Content objects (look for the themes starting with '0'). The theme can also be developed by you in the BEx Analyzer. The workbook styles are passed to the Report Designer via the them, however it is important to note that only three of the workbook styles are used by the Report Designer and these are mapped to the areas of the report as follows:

Style	Report Designer area affected
Standard	Header
SAPBExstdData	Detail
SAPBExaggData	Results
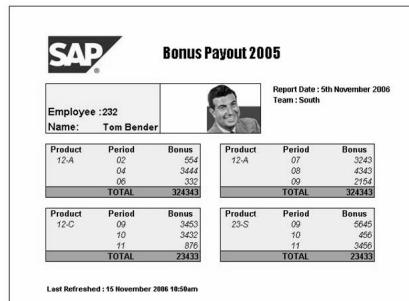
- Choose your look and feel
- Themes shared with Analyzer
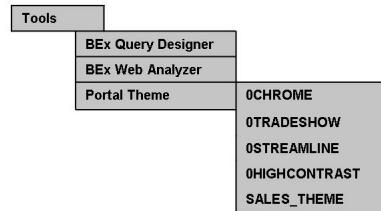
Theme (Yours* or SAP Standard)

Default ⟶ Default Override

Custom Value
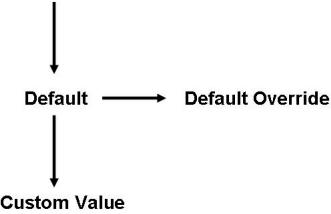
* Your theme provides formatting rules for:

Style	Affects Report
Standard	Header
SAPBExstdData	Detail
SAPBExaggData	Results

Figure 48: Themes in the Report Designer

Exercise 4: Advanced features of the Report Designer

Exercise Objectives

After completing this exercise, you will be able to:

*   Develop a formatted report using the advanced features of the Report Designer.

Business Example

You would like to present a hierarchically structured report to the business and you would like to enhance the presentation of the sections of the hierarchy using formatting features of the Report Designer. The report will also use a variable so you need to see how a formatted report will handle variables. Finally you will add an image to the report and also add headers and footers containing useful background information related to the report.

Task 1:

Develop a report which shows product sales presented hierarchically. Format the sections of the hierarchy so that the presentation is clear. The report will also use a variable so you need to observe how the variable will be presented to the users at run time.

1. Create a new report and insert a data provider based on query *Sales Hierarchy* (SALES_HIERARCHY) which you will find under the role *BW306_Reporting → Unit 3: BEx Report Designer*.

2. You do not wish to see the key of the characteristic *Material* so remove this from the report.

3. You would like to use color to emphasize all the 3 node levels of the hierarchy, **but** use the *Edit Format* dialog accessible from the cell context menu to select the colors, use any colors you desire.

4. The text for the leaf (the actual material values) should be displayed using the font *Comic Sans MS*.

5. Save your report under the role path *BW306_Reporting → Unit 3: BEx Report Designer* using the name **GR## BW306 Report 3** and technical name **GR##BW306REPORT3**.

6. Execute the report on the web and check the results. You will need to log onto the portal using your assigned user id and password. When prompted for a hierarchy node variable choose only the node *Hardware*.

2007/Q2   **121** SAP

Task 2:

You would now like to add a header and a footer to the report. In the header you would like to include the name of the data provider and a logo. In the footer you want to add the author of the report and the InfoCube name.

1. Add a page header and within the header you will insert an image. You can use any image of your own or you can use one of the images already in the MIME repository. Make sure the image is sized correctly.

2. You want the description of the data provider to appear alongside the image. Make sure the text is large enough to stand out.

3. Add a page footer and within the footer cell add the text element *Infocube* and *Author*.

4. Save your report under the role path *BW306_Reporting → Unit 3: BEx Report Designer* using the new name **GR## BW306 Report 4** and technical name **GR##BW306REPORT4**.

5. Execute the report on the web and check the results. You will need to log onto the portal using your assigned user id and password. When prompted for a hierarchy node variable choose only the node *Hardware*.

Task 3:

Your company likes to offer a free item with any computer bundles. In the sales report you would like to highlight the free item. Use conditional formatting to place more emphasis on this product.

1. Create a new report and insert a data provider based on query *Conditional Format* (CONDITIONAL_FORMAT) which you will find under the role *BW306_Reporting → Unit 3: BEx Report Designer*.

2. You would like a specific material to stand out in bold. The material is *Sunny Sunny 01*, the technical name for this material is *M-01*.

3. Save your report under the role path *BW306_Reporting → Unit 3 BEx Report Designer* using the new name **GR## BW306 Report 4** and technical name **GR##BW306REPORT4**.

4. Execute the report on the web and check the results. You will need to log onto the portal using your assigned user id and password.

Task 4:

Your colleagues have developed some themes using the Analyzer. You would like to try them out with your reports developed with the Report Designer to see if they improve the presentation.

1.   Create a new report and insert a data provider based on query *Breakdown Dynamic (BREAKDOWN_DYNAMIC)* which you will find under the role *BW306_Reporting → Unit 3: BEx Report Designer*.

2.   You would like to apply the theme which you developed in a previous exercise. The theme should be called *GR##_THEME_BW306*.

3.   Save your report under the role path *BW306_Reporting → Unit 3 BEx Report Designer* using the name **GR## BW306 Report 5** and technical name **GR##BW306REPORT5**.

4.   Execute the report on the web and check the results. You will need to log onto the portal using your assigned user id and password.

Solution 4: Advanced features of the Report Designer

Task 1:

Develop a report which shows product sales presented hierarchically. Format the sections of the hierarchy so that the presentation is clear. The report will also use a variable so you need to observe how the variable will be presented to the users at run time.

1. Create a new report and insert a data provider based on query *Sales Hierarchy* (SALES_HIERARCHY) which you will find under the role *BW306_Reporting → Unit 3: BEx Report Designer*.

    a)   Press the icon *Insert Data Provider* from the toolbar and select the query *Sales Hierarchy (SALES_HIERARCHY)* which you will find under the role *BW306_Reporting → Unit 3: BEx Report Designer*.

2. You do not wish to see the key of the characteristic *Material* so remove this from the report.

    a)   Click on the grey button above the heading *[Material].Key* to highlight the entire column, then right-click to access the context menu and select the option *Delete Column*.

3. You would like to use color to emphasize all the 3 node levels of the hierarchy, **but** use the *Edit Format* dialog accessible from the cell context menu to select the colors, use any colors you desire.

    a)   Right-click on the cell which contains *Prod.hier.level 1* and from the context menu select the option *Edit Format*.

    b)   From the *Cell* tab select the push button next to the *Background Color* and choose any color from the pallet. Press *OK* to close the dialog.

    c)   Repeat this step for the remaining node levels using different colors for each level.

4. The text for the leaf (the actual material values) should be displayed using the font *Comic Sans MS*.

    a)   Left-click on the cell which contains *Product Hierarchy for Material MARA Leaf* and from the font selection drop down list in the toolbar select the font *Comic Sans MS*.

5. Save your report under the role path *BW306_Reporting → Unit 3: BEx Report Designer* using the name **GR## BW306 Report 3** and technical name **GR##BW306REPORT3**.

    a)   Click on the icon *Save* from the toolbar or use the menu option *Report → Save*, then press the *Roles button* and follow the path *BW306_Reporting → Unit 3: BEx Report Designer*. Enter the report name **GR## BW306 Report 3** and a technical name **GR##BW306REPORT3** then press *OK*.

6. Execute the report on the web and check the results. You will need to log onto the portal using your assigned user id and password. When prompted for a hierarchy node variable choose only the node *Hardware*.

    a)   Press the icon *Execute* from the toolbar and when prompted enter your user id and password assigned to you by the instructor. When the variable pop-up appears press the button for the values selection. Open the root node and select the node *Hardware* and press the button *Add* Now press *OK* twice to execute the report.

Task 2:

You would now like to add a header and a footer to the report. In the header you would like to include the name of the data provider and a logo. In the footer you want to add the author of the report and the InfoCube name.

1. Add a page header and within the header you will insert an image. You can use any image of your own or you can use one of the images already in the MIME repository. Make sure the image is sized correctly.

    a)   Select the menu option *Insert → Page Header*.

    b)   Right click in the header cell and from the context menu choose *Insert Image*. Then increase the height of the cell by dragging the cell border down so it is at least twice the original size.

    c)   Click on the image icon in the header cell, then press the tab for *Properties* at the bottom left of the screen. In the field *MIME* add the file name **SAP_LOGO.gif** to the end of the image path. (you need to remove the two period symbols first)

    d)   In the fields *Image Properties* change the width to 40 and the height also to 40.

Continued on next page

2.   You want the description of the data provider to appear alongside the image. Make sure the text is large enough to stand out.

      a)   From the tab *Field Catalog* expand the node *Text Elements* and drag the field *<Query Description>* to the header field to the right of the image icon. Click on the text field in the header cell and from the text size drop down list select size 24.

3.   Add a page footer and within the footer cell add the text element *Infocube* and *Author.*

      a)   From the Report Designer menu select *Insert → Page Footer*.

      b)   From the *Field Catalog* expand the node *Text Elements* and drag the field *<Author>* and *<Infocube>* into the footer cell.

4.   Save your report under the role path *BW306_Reporting → Unit 3: BEx Report Designer* using the new name **GR## BW306 Report 4** and technical name **GR##BW306REPORT4**.

      a)   From the toolbar use the menu option *Report → Save As* then press the *Roles button* and follow the path *BW306_Reporting → Unit 3: BEx Report Designer*. Enter the report name **GR## BW306 Report 4** and a technical name **GR##BW306REPORT4** and press *OK* .

5.   Execute the report on the web and check the results. You will need to log onto the portal using your assigned user id and password. When prompted for a hierarchy node variable choose only the node *Hardware*.

      a)   Press the icon *Execute* from the toolbar and when prompted enter your user id and password assigned to you by the instructor. When the variable pop-up appears press the button for the values selection. Open the root node and select the node *Hardware* and press the button *Add* Now press *OK* twice to execute the report.

Task 3:

Your company likes to offer a free item with any computer bundles. In the sales report you would like to highlight the free item. Use conditional formatting to place more emphasis on this product.

1.   Create a new report and insert a data provider based on query *Conditional Format* (CONDITIONAL_FORMAT) which you will find under the role *BW306_Reporting → Unit 3: BEx Report Designer*.

      a)   Press the icon *New* then press the icon *Insert Data Provider* from the toolbar and select the query *Conditional Format (CONDITIONAL_FORMAT)* which you will find under the role *BW306_Reporting → Unit 3:BEx Report Designer*.

2. You would like a specific material to stand out in bold. The material is *Sunny Sunny 01*, the technical name for this material is *M-01*.

   a) Highlight the entire row containing the material members (this should be *Det* row) and then from the main menu select the option *Format → Conditional Formatting*.

   b) Once the *Select Values for Material* dialog appears use the drop down to switch the view to *Single Values*. In the field *Direct Input* enter the material **M-01**.

   c) You will now see a new row has been added to the layout, right-click on the cell *Sunny Sunny 01* and from the context menu select the option *Text Format → Bold*.

3. Save your report under the role path *BW306_Reporting → Unit 3 BEx Report Designer* using the new name **GR## BW306 Report 4** and technical name **GR##BW306REPORT4**.

   a) From the toolbar use the menu option *Report → Save As* then press the *Roles button* and follow the path *BW306_Reporting → Unit 3 BEx Report Designer*. Enter the report name **GR## BW306 Report 4** and a technical name **GR##BW306REPORT4** then press *OK* .

4. Execute the report on the web and check the results. You will need to log onto the portal using your assigned user id and password.

   a) Press the icon *Execute* from the toolbar and if prompted enter your user id and password assigned to you by the instructor.

Task 4:

Your colleagues have developed some themes using the Analyzer. You would like to try them out with your reports developed with the Report Designer to see if they improve the presentation.

1. Create a new report and insert a data provider based on query *Breakdown Dynamic (BREAKDOWN_DYNAMIC)* which you will find under the role *BW306_Reporting → Unit 3: BEx Report Designer*.

   a) Press the icon *New* then press the icon *Insert Data Provider* from the toolbar and select the query *Breakdown Dynamic (BREAKDOWN_DYNAMIC)* which you will find under the role *BW306_Reporting → Unit 3:BEx Report Designer*.

2. You would like to apply the theme which you developed in a previous exercise. The theme should be called *GR##_THEME_BW306*.

   a) From the main Report Designer toolbar choose the menu option *Tools → Portal Theme*. From the drop down list select the theme *GR##_THEME_BW306*.

3. Save your report under the role path *BW306_Reporting → Unit 3 BEx Report Designer* using the name **GR## BW306 Report 5** and technical name **GR##BW306REPORT5**.

    a)   Click on the icon *Save* from the toolbar or use the menu option *Report → Save* then press the *Roles button* and follow the path *BW306_Reporting → Unit 3 BEx Report Designer*. Enter the report name **GR## BW306 Report 5** and a technical name **GR##BW306REPORT5** then press *OK* .

4. Execute the report on the web and check the results. You will need to log onto the portal using your assigned user id and password.

    a)   Press the icon *Execute* from the toolbar and if prompted enter your user id and password assigned to you by the instructor.

128   2007/Q2

Lesson Summary

You should now be able to:
*   Further develop formatted reports using advanced features of the BEx Report Designer.

Unit Summary

You should now be able to:

- Create highly formatted reports for displaying on the Web and also for printing using the new Report Designer making use of the basic functions of this tool.

- Further develop formatted reports using advanced features of the BEx Report Designer.

Unit 4

BEx Web Application Designer

Unit Overview

Delivering BI information through sophisticated web applications is becoming more common in businesses today. With the BEx Web Application Designer, you have a tool that makes it possible to develop flexibly, highly informative web applications with no HTML programming expertise required.

Unit Objectives

After completing this unit, you will be able to:

- Access the Web Application Designer
- Identify the main components of the tool
- Create a simple web application
- Change the properties of web items
- Execute the web application and navigate in the results
- Align objects as desired in the web application
- Add texts, images and tabs to your web application
- List the impact of portal themes on your web application
- Add a range of content to your web application with the Report, Ticker and Information Field web items
- Enhance the navigation capabilities of your web applications with the Navigation Pane, Exceptions, Conditions, Checkbox, Radio Buttons and Context Menu web items
- Present data in chart format in your web applications
- Describe the process of geocoding for characteristics
- Present data in map format in your web applications
- Explain the basic operation of the Command Wizard
- Insert commands into the web application using various web items
- Use sequences of commands to accomplish multiple tasks
- Efficiently develop web applications by reusing modules such as header and footer web templates
- Use the BI Pattern Wizard to modify the SAP-delivered patterns.

 SAP

- Use the features of the XHTML editor
- Generate parameterized URLs
- Use JavaScript functions to enhance web applications

Unit Contents

Lesson: Web Framework and General Settings

Lesson Overview

You will be designing and creating web applications for your company. You want to become familiar with the Web Application Designer in BI and understand the layout of the various parts of the tool.

Lesson Objectives

After completing this lesson, you will be able to:

- Access the Web Application Designer
- Identify the main components of the tool

Business Example

You will be designing and creating web applications for your company. You want to become familiar with the Web Application Designer in BI and understand the layout of the various parts of the tool.

Web Applications and Their Architecture

This section covers the various components of a web application and how those components are related to each other. A good understanding of this architecture is important for the effective use of the BEx Web Application Designer.

 SAP
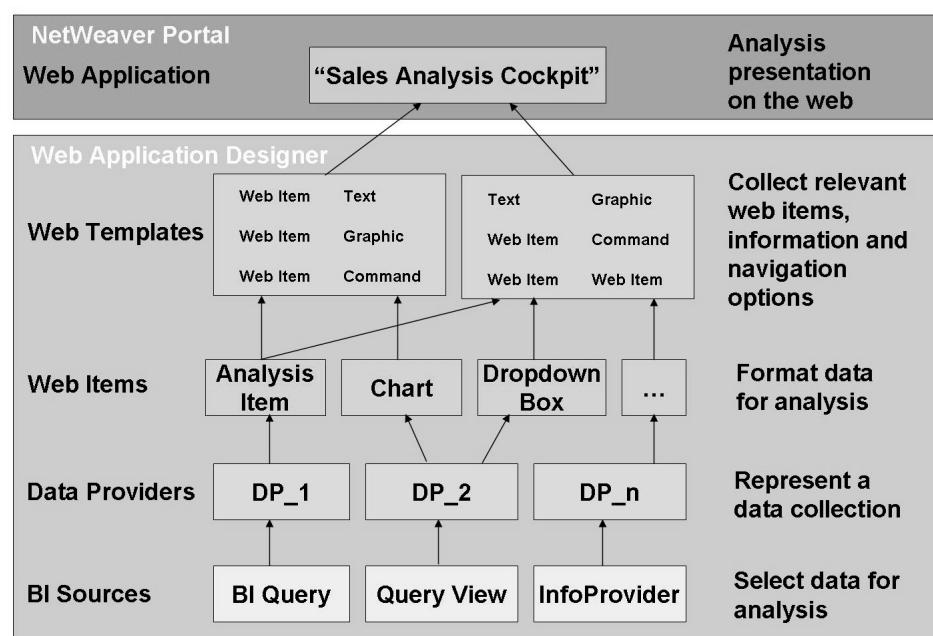
Figure 49: Web Application Architecture

The purpose of a web application is to convey information to the user in a web environment in an effective manner. To do that, effective design principles must be used while constructing the web template or templates that comprise the web application. Additionally, the designer of the web application must have a thorough knowledge of the sources of data that will be used.

BI sources supply the data to a web application. These sources can be:

- BI Queries
- BI Query Views
- BI InfoProviders

Having these sources of BI data available means that virtually any data that is accessible in the BI data warehouse can be portrayed in a web application. Of the three types of BI sources, BI queries and query views are the most commonly used. BI queries are created using the BEx Query Designer. BI query views are typically created using the BEx Web Analyzer or the BEx Analyzer, or even from your own web applications.

In the Web Application Designer, the BI sources are linked into the web framework through **data providers**. A data provider provides a logical connection to the data supplied by a BI source. There is usually a one-to-one correspondence of BI source to data provider. However, each data provider is independent and so it is possible to use the same BI source for multiple data providers if that is necessary.

In general, **web items** are responsible for formatting the data received from a data provider. For example, a *Chart* web item would show the data as a chart while the *Navigation Pane* web item would list the components of the data provider in the format of a navigation window. The assignment of a data provider to a web item is known as **data binding**.

Not every web item formats data for the user of the web application. The *XML - Data Provider Information* web item, for example, makes the data and metadata of the web application available in XML format for use with JavaScript routines. Other web items, such as the *Container* or *Context Menu* do not use any data binding at all.

Web items, because of their ability to represent the data provider data in so many different formats, serve as the basic building blocks of a **web template**. A web template is the collection of web items, formatting objects, texts, images and parameters that define a web page. Web templates can be very simple or very complex, depending on the purpose they serve. The main activity for a user of the BEx Web Application Designer is the creation and maintenance of web templates.

When web templates are executed in the NetWeaver Portal, they are referred to as **web applications**. A web application might consist of one web template or many web templates linked together, but their purpose is to provide the information from the BI sources in a logical, understandable and visually interesting way to the user.

Web Application Designer Components

The BEx Web Application Designer is the tool used to create web templates in BI. It is accessed via the menu path *Start → Programs → Business Explorer → Web Application Designer*. With NetWeaver 2004s BI, you have access to two versions of the tool. The path just mentioned accesses the BI version of the Web Application Designer. If you want to execute the BW 3.x version of the Web Application Designer, the menu path is *Start → Programs → Business Explorer → Business Explorer (SAP BW 3.x) → Web Application Designer (SAP BW 3.x)*. The focus of this lesson is the BI version of the tool.

 SAP
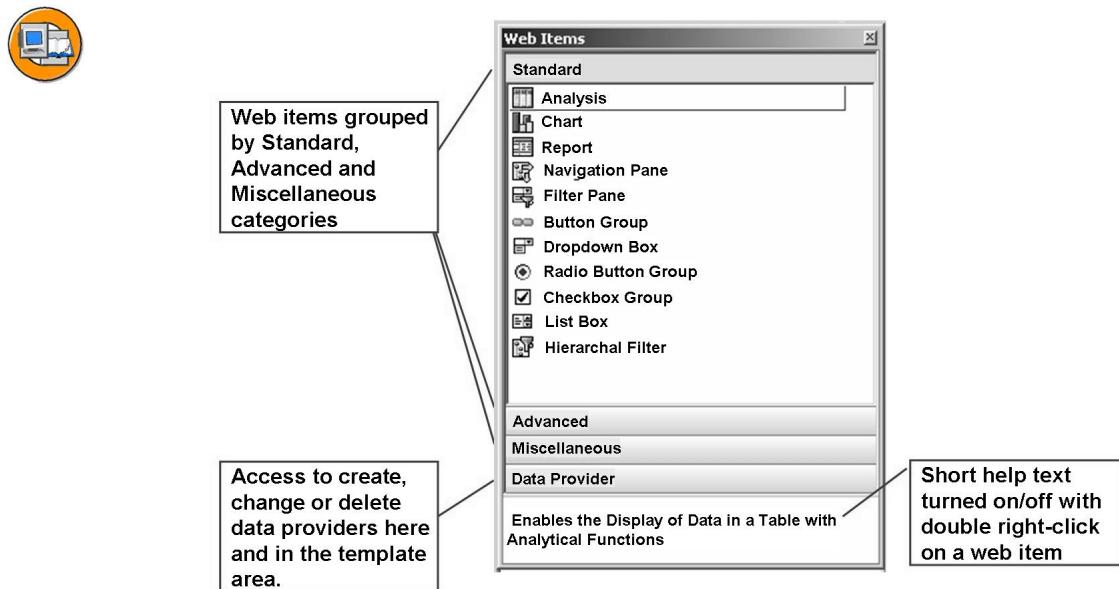
Figure 50: Web Items Screen Area

When you start the Web Application Designer, the workspace is divided into several screen areas. One of the areas is the **Web Items Screen Area**. This screen area provides access to all the available web items, data providers and a small help section. The web items are segmented into three groupings – Standard, Advanced and Miscellaneous. Clicking on the group heading will display the contents of that group. Since the web items are the basic building blocks of the web templates, it is very common to select a web item from this screen area and drag it into the template screen area to define the template.
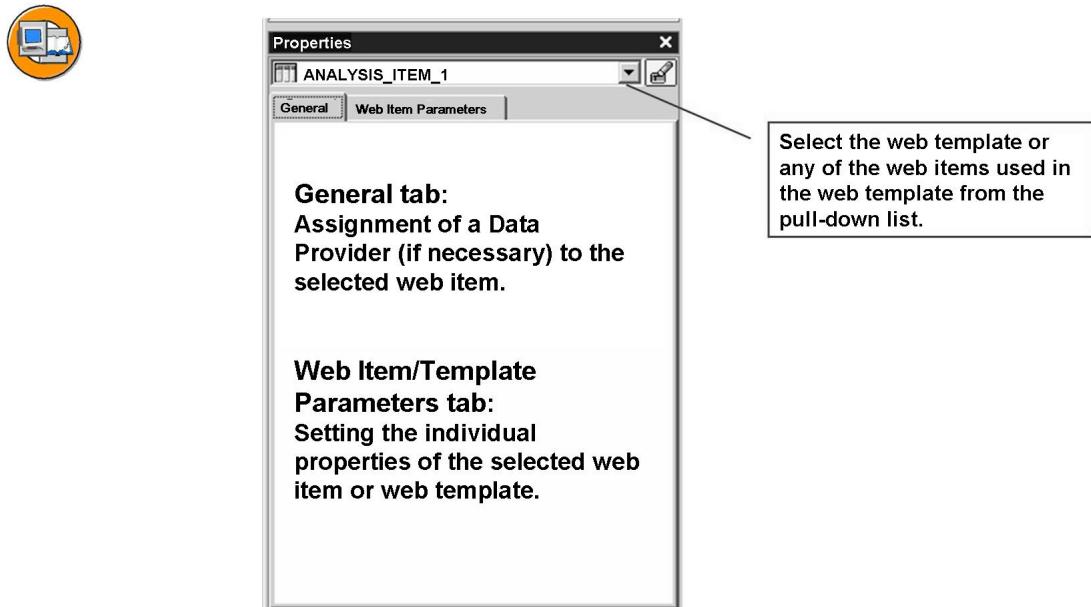
Figure 51: Properties Screen Area

The **Properties Screen Area** provides access to all the properties of the selected web item or the web template itself. Depending on the component selected, there may be a few or many properties to set for the proper execution of that component. The properties screen area uses two tabs to segment the properties. The *General* tab provides access to the data provider assignment, if necessary. The *Web Item Parameters* or *Web Template Parameters* tab, depending on which type of component is chosen from the dropdown box at the top of the screen area, shows a listing of the configuration settings of that component.

Thus, it is very common when building a web template to first select a web item and drag it into the web template, then come to the properties screen area to make the necessary data binding and parameter settings for that web item.

The Layout tab of the Template Area shows the web items, texts, graphics and other elements of the web template.
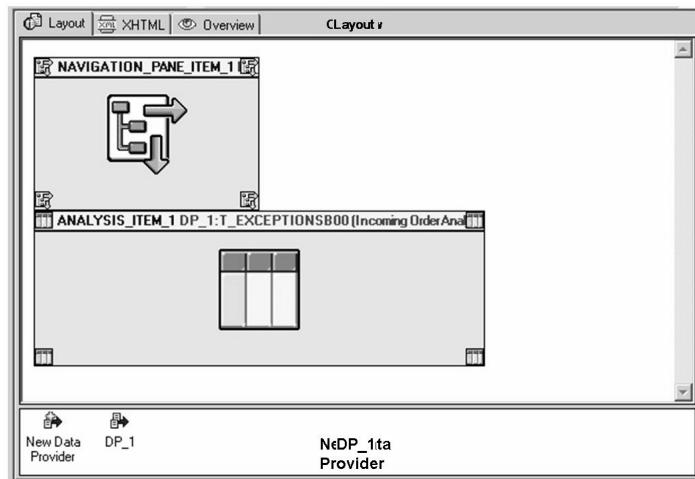
Figure 52: Template Area - Layout Tab

The **Template Area** is where the actual web template is constructed. As described above, web items are dragged into position in the template area. The web template is defined by the collection of web items and other objects placed in this area.

The template area is segmented into three different tabs. The first tab shown above is the **Layout Tab**. This work area shows the web item contents as a collection of graphical icons with each icon representing a specific instance of a web item. The template area is not an exact representation of the placement or spacing of the individual web items when the web application is executed. Rather it is a logical grouping of items that makes it easy to see the relationship of one item to another. This is the work area most commonly used to construct a web template.

■ **The XHTML tab of the Template Area shows the HTML
code generated when items are added to the Layout tab.
This code can be maintained directly in this view.**

Figure 53: Template Area - XHTML Tab

The second tab in the template area is the **XHTML Tab**. This work area shows the
resulting HTML code that is automatically generated as web items are placed in
the web template and their parameters are set. The developer of the web template
can make direct coding additions or enhancements in this work area. They may
also choose to begin their work with the Web Application Designer then continue
editing the HTML code with another web development tool. This is possible
since the HTML code generated by the Web Application Designer complies with
industry standards.

Many HTML programming functions are provided on the XHTML tab and those
will be discussed in detail in a later lesson.

- **The Overview tab of the Template Area shows the web items, data providers and commands of the web template.**
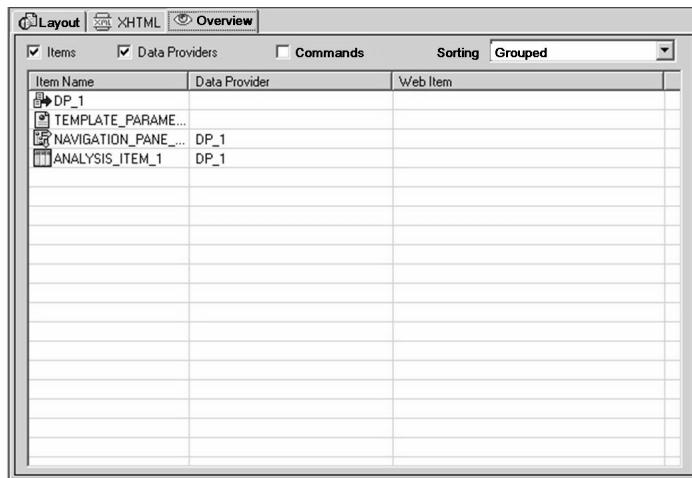
Figure 54: Template Area - Overview Tab

The **Overview Tab** is the third tab of the layout area and it shows an inventory of the web items, data providers and commands in the web template. This list of web template components can be filtered as well as sorted several different ways, depending on the needs of the user. This work area is very helpful to verify that the data binding of the web items is correct. The logical names of the web items as well as their properties can be changed from this work area. To do so, simply right click on a web item to bring up the context menu and choose either *Rename* or *Edit*.

- **Web Items can be saved as Reusable Web Items in the Favorites or Roles folders. Data provider assignments are not saved with the reusable web item.**
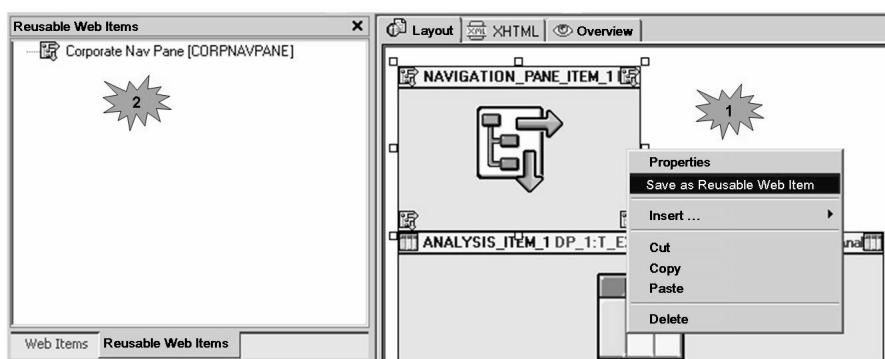
Figure 55: Reusable Web Items

It is a very good practice for web application design to be consistent in the 'look and feel' of a collection of web templates. For example, if a *Navigation Pane* web item is used on several web templates that comprise a complete web application, it makes sense that the user would see a consistent rendering of the size, content and arrangement of the items in the navigation pane from one template to the next.

To help promote this consistency, web items can be configured, then saved in roles or the Favorites folder as **reusable web items**. Once saved, they can be accessed from the *Reusable Web Items* tab in the *Web Items* screen area. It is important to note that not all properties of the web item are saved with a reusable web item. The data binding of a web item is not retained in the reusable web item. Data binding must be done individually for each relevant web item in the template where it is used.

In summary, the Web Application Designer provides a rich environment for the development of robust web applications. In the lessons that follow, you will explore this environment and learn to use the many functions provided.

 SAP

Lesson Summary

You should now be able to:
• Access the Web Application Designer
• Identify the main components of the tool

Related Information

• Additional useful information about the concepts presented in this lesson can be found in the SAP NetWeaver online documentation which can be accessed at http://help.sap.com.

142   2007/Q2

Lesson: Basic Web Applications

Lesson Overview

This lesson teaches the basic operations required to create, change and execute a simple web application.

Lesson Objectives

After completing this lesson, you will be able to:

- Create a simple web application
- Change the properties of web items
- Execute the web application and navigate in the results

Business Example

In order to understand the basic process of web application development, you want to create a simple web template to quickly gain the needed experience.

Creating Basic Web Applications

The steps for creating any web application with the Web Application Designer are very simple. This lesson will cover that process using a simple web template as an example.
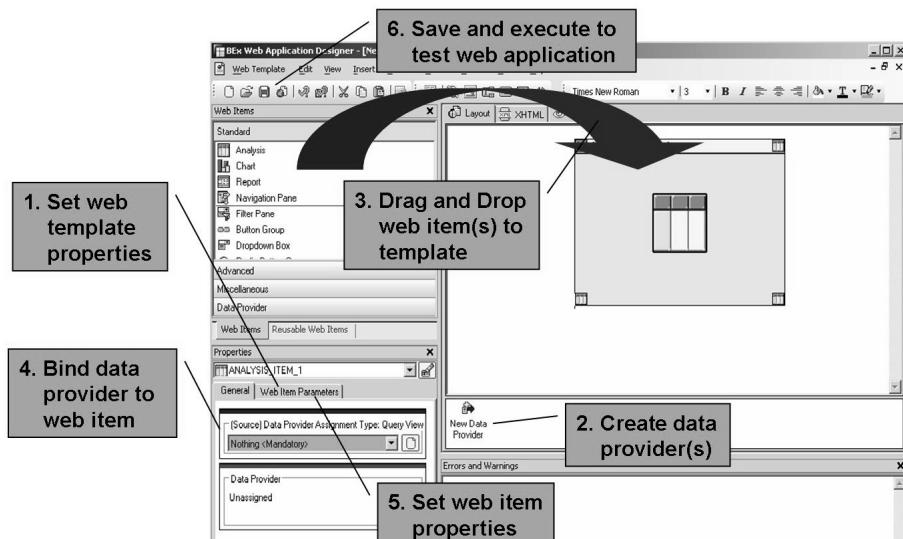

Figure 56: Creating a Web Template

Once you have started the Web Application Designer, you are ready to create a new web template. The opening screen of the *Layout* area will offer you a history of web templates you have accessed before as well as a button labeled *Create New Web Template*. You can use that button or the *New* ⬜ icon from the toolbar to start your new web template.

Once you have created a blank web template the typical steps for web template development are:

1.   Set the web template properties
2.   Create the necessary data provider(s)
3.   Drag and drop the necessary web item(s) to the template
4.   Perform the necessary data binding for the relevant web items
5.   Set the properties of the web items in the template
6.   Save the web template and execute it to test the functionality

Since a web template will likely have several web items, it is not necessary that you follow these steps in exactly this order. You are free to work on one part of the template then another as you see fit.

The major steps above will be discussed in more detail in the remainder of this lesson.

Set web template properties to influence the overall behavior of the web template.

- ■ **Internal Display**
  - ◆ **Which messages (info, warning, system) are displayed**
- ■ **Behavior**
  - ◆ **Variable and variable screen behavior**
  - ◆ **Report/Report Interface behavior**
  - ◆ **Personalization**
  - ◆ **Template initialization commands**
- ■ **Data Binding**
  - ◆ **Variable variants**
  - ◆ **Level for saving documents**
- ■ **Dialog Binding**
  - ◆ **Mode for starting new dialogs**
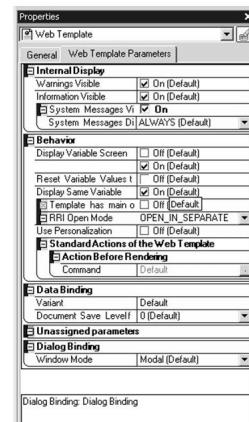
Figure 57: Web Template Properties

Web template properties are settings that influence the entire web template. The properties themselves are segmented into the groupings mentioned in the graphic above. Very often, the default values for these properties will be sufficient for the

proper operation of the web application. However, you may have requirements that differ from the default values, so it pays to be familiar with the impact of each of the web template properties.

Internal Display Properties

This property controls which, if any, system messages should be displayed to the user of the web application.

Behavior Properties

Properties in this grouping deal with the interaction of the web application with the user. Whether the variable screen is shown before the template results, whether the report/report interface launches in a new browser window or whether personalized values for the template are stored and used are just a few of the parameters available here.

Data Binding Properties

The variant, if any, that the web application will use and the level (query level or InfoProvider level) at which InfoProvider data documents are created are the two properties found in this segment.

> **Note:** Document Integration is covered in the *BW305 - BI Enterprise Reporting, Query and Analysis (Part I)* course.

Dialog Binding Properties

Whether new dialogs start in new browser windows or not and if so, whether the user can return to the original window is determined by the setting for this property. This would be significant if the web application had navigation, such as links or buttons, that launched other web templates.

2007/Q2   **145** SAP
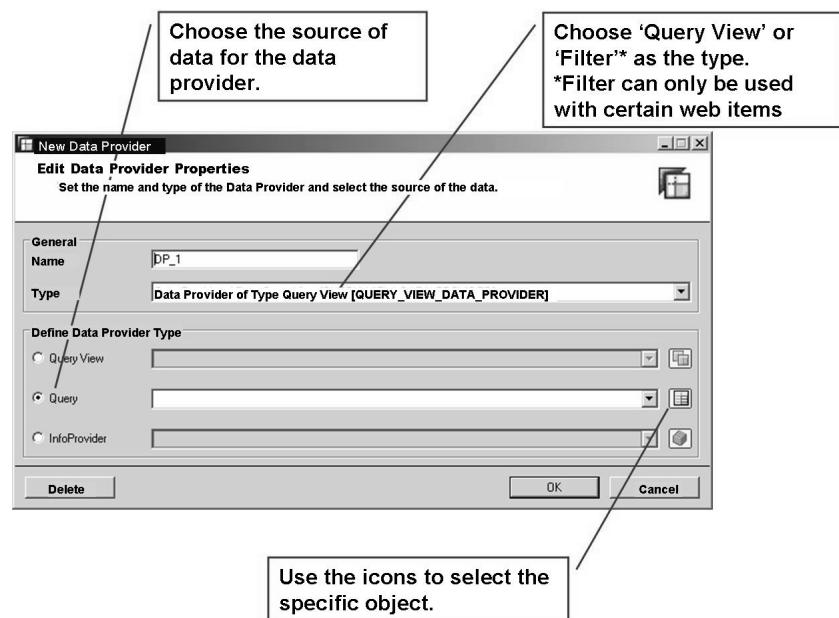
Figure 58: Creating Data Providers

Almost every web template will have at least one **data provider**. Data providers can be created in the *Web Items* screen area or in the *Template - Layout Tab* screen area.

Data Providers can be either a *Filter* or *Query View* type. Data providers of the filter type usually provide data for web items that are related to filtering data, such as the Dropdown Box and Radio Button Group web items. They cannot be used with web items that display the results of a query or query view, such as the Analysis or Chart web items.

More common are the Query View type of data provider. These can reference queries, query views or InfoProviders. They can be used with any web item that is relevant for data binding.

If you are creating a web application and you discover that the necessary queries have not been created yet, you can access the BEx Query Designer directly from the Web Application Designer using the path *Tools → BEx Query Designer*. You can also call the BEx Web Analyzer from the *Tools* menu if you need to create a query view.

Data Binding
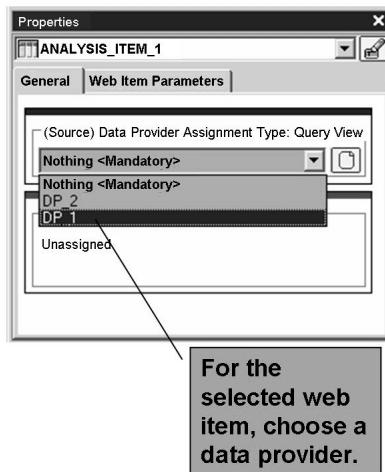
■ **Assigning a data provider to a web item is known as <u>data binding</u>.**

■ **The data and the list of elements from the data provider are made available to the web item through this process.**

■ **The web item formats the data and elements according to its definition. For example, a chart web item will present the data in the format of a chart, but a dropdown box will list the values of a characteristic selected from the data provider's elements.**

Figure 59: Data Binding

As described before, **data binding** is the process of assigning a data provider to a web item or command. This process can be done in the *Properties* screen area either on the *General* tab or the *Web Item Parameters* tab. For commands, data binding will usually take place in the *Command Wizard* screens.

Note: The Command Wizard is covered fully in a later lesson in this unit.

Set web item properties to influence the specific behavior of the web item.

Example: Analysis Item

■ **Display**
  ◆ Width and height settings

■ **Internal Display**
  ◆ Presentation format and visibility of data
  ◆ Selection of specific rows and columns

■ **Behavior**
  ◆ Type of navigation allowed
  ◆ Rows/columns selectable/not selectable

■ **Data Binding**
  ◆ Data provider assigned
  ◆ Affected data providers on navigation

■ **Paging**
  ◆ Settings for scrolling operations

■ **Cell Content**
  ◆ Display of document icons
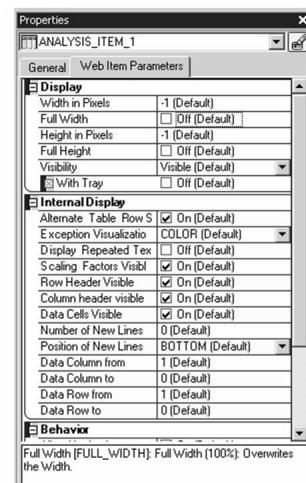
Figure 60: Web Item Properties

Often the most intensive portion of the web application development process is setting the **web item properties**. Some web items have only a few properties while others have many properties. The defaulted value of each property may be sufficient for your purposes, but you will likely need to change several properties to accomplish the goal of the web application.

Web item properties are displayed in the *Properties* screen area in various groupings as shown in the graphic above. Not every web item uses all the grouping categories. As specific web items are introduced in the later lessons, the key properties will be discussed for each web item.

	Default:	Web item navigation affects all web items with same data binding.
	Scenario:	'Affected Data Providers' property on the Dropdown Box is set to DP_1 <u>and</u> DP_2.
	Result:	Filtering from the Dropdown Box will affect both Analysis Items. Navigation on either Analysis Item will not affect the other.

Dropdown Box
Data Binding: DP_1
Affected Data Providers: DP_1 & DP_2

Country: [Germany   ▼]

Country ⇕	Sold-to Party ⇕	Incoming Orders ⇕ January 2007	Incoming Orders ⇕ February 2007	Incoming Orders ⇕ March 2007
		EUR	EUR	EUR
Germany	1001	108.132,47	484.220,26	166.092,05
	1033	227.107,85	117.555,80	72.552,05
	1172	146.595,21	112.900,10	175.481,54
	1174	200.425,68	16.959,49	19.942,86
	1175	666.953,58	467.188,17	475.340,17
	1300	390.855,93	162.555,96	452.067,27
	1320		683.252,16	742.536,24
	1321	749.136,99		
	1360	174.301,77	76.343,66	71.259,61
	1460	111.659,09	172.400,12	160.460,30

Page 1 of 3

Analysis Item 1
Data Binding: DP_1

Country ⇕	Material ⇕	Incoming Orders ⇕ January 2007	Incoming Orders ⇕ February 2007	Incoming Orders ⇕ March 2007
		EUR	EUR	EUR
Germany	1400-100	22.547,89	23.959,05	
	1400-200	6.764,37	7.193,85	8.451,62
	1400-300		483.169,05	567.531,90
	1400-310	200.425,68		250.532,10
	1400-400	51.538,03	54.759,16	
	1400-750	9.190,44	9.765,64	11.491,24
	AZ2-730	0,00		
	DPC1002	76.644,42	71.564,24	70.901,61
	DPC1003	84.447,21	78.817,40	78.384,34
	DPC1004	89.211,18	84.759,10	83.866,90

Page 1 of 9

Analysis Item 2
Data Binding: DP_2

Figure 61: Affected Data Providers

Web items can interact with one another in several ways. One of the easiest ways to cause navigation actions taken in one web item to affect the results in another web item is with the *Affected Data Providers* property.

By default, navigating in a web item affects all other web items assigned to the same data provider. However, with the *Affected Data Providers* property, you can specify which data providers will be impacted by navigation steps taken in the chosen web item.

The graphic above shows a typical example where in the properties of the *Dropdown Box*, the affected data providers is set to both DP_1 and DP_2. Each of the *Analysis* web items is assigned to only one of the data providers. Navigation actions taken in the *Dropdown Box* will thus affect the navigation state of both the *Analysis* web items.

This is helpful when you want to synchronize the actions of multiple items in a web application.

Exercise 5: Basic Web Applications

Exercise Objectives

After completing this exercise, you will be able to:

- Create a simple web application.

Business Example

You need to learn how to use the BEx Web Application Designer in order to construct various web applications for your company.

Task 1: Create a Basic Web Application

You will create a simple web application containing a dropdown box and two analysis items.

1. Start the BEx Web Application Designer and create a new web template.

2. To begin, create a new data provider and use the query **T_COQ4** to supply the data.

3. Next, choose the *Dropdown Box* web item from the Standard group of web items and drag it into your template. This web item allows you to select items from a dropdown list of items.

4. Set the properties of the Dropdown box as follows:

Property/Parameter	Value
Data Binding > Data Binding Type	`Char/Structure Member`
%NM% (Selection of Characteristic)	`Click on button at end of row`
Selection of Characteristic > Data Provider	`DP_1`
Selection of Characteristic > Characteristic	`0CALMONTH` (use button at end of row)

5. Next, choose the *Analysis* web item from the Standard group of web items and drag it into your template to the right of the Dropdown Box. This web item allows you to display data in a table format.

6. By default, both web items in your template will have been automatically assigned to *DP_1*, the first data provider. Now you need to create a second data provider. Use the query view **T_COQ4V1** as the source for this new data provider.

Continued on next page

2007/Q2   **151** SAP

7.   Next, add a second *Analysis* web item from the Standard group of web items to your template to the right of the first Analysis item.

8.   Check that the second Analysis item has automatically been assigned to data provider **DP_2**. If not, change the data provider assigned to the second Analysis item *(ANALYSIS_ITEM_2)* to be **DP_2**.

9.   Save your web template using the values below. Then execute the template and navigate by selecting values from the Dropdown Box to filter the results. What do you notice about the results as you navigate?

When you are finished, exit the results and return to the Web Application Designer for the next task.

Field	Value
Description	**GR## Basic Template**
Technical Name	**GR##BASIC**

Task 2: Affected Data Providers

In the previous task, only the first Analysis item responded to the choices made from the Dropdown Box. Now you will link the second Analysis item to the Dropdown box.

1.   In the Web Application Designer, access the properties of the *Dropdown Box* in your web template. Make the following changes to the properties:

Property/Parameter	Value
Data Binding > DP_1 (Selection of Characteristic	**Click on button at end of row**
Selection of Characteristic > Affected Data Providers	**DP_1**
Selection of Characteristic > Affected Data Providers	**DP_2** Selection made on the next row.

2.   Save and execute your web template. Navigate using the Dropdown box and notice the effect on the results.

Solution 5: Basic Web Applications

Task 1: Create a Basic Web Application

You will create a simple web application containing a dropdown box and two analysis items.

1. Start the BEx Web Application Designer and create a new web template.

   a) Start the BEx Web Application Designer using the path *Start →*
      Programs → Business Explorer → Web Application Designer.

   b) Logon to the system with the user ID and password provided by your instructor.

   c) Select *"Create new Web Template"*.

2. To begin, create a new data provider and use the query **T_COQ4** to supply the data.

   a) Double click on *New Data Provider* in the design area.

   b) Select the data provider type *Query* using the radio button.

   c) Enter **T_COQ4** as the name of the query and select *OK*.

3. Next, choose the *Dropdown Box* web item from the Standard group of web items and drag it into your template. This web item allows you to select items from a dropdown list of items.

   a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

   b) Drag the *Dropdown Box* web item to your web template as the first web item.

4. Set the properties of the Dropdown box as follows:

2007/Q2   **153** SAP

Property/Parameter	Value
Data Binding > Data Binding Type	`Char/Structure Member`
%NM% (Selection of Characteristic)	`Click on button at end of row`
Selection of Characteristic > Data Provider	`DP_1`
Selection of Characteristic > Characteristic	`0CALMONTH` (use button at end of row)

    a)   In the *Properties* work area, set the properties for the fields as shown in the table above.

> **Hint:** The 'Dot' button at the end of some property fields will open a new window where additional selections can be made.

    b)   When finished, select *OK*.

5.   Next, choose the *Analysis* web item from the Standard group of web items and drag it into your template to the right of the Dropdown Box. This web item allows you to display data in a table format.

    a)   In the Web Items area of the Web Application Designer, open the *Standard* web item category.

    b)   Drag the *Analysis* web item to your web template and drop it to the right of the Dropdown Box. It will automatically position itself below the Dropdown box.

6.   By default, both web items in your template will have been automatically assigned to *DP_1*, the first data provider. Now you need to create a second data provider. Use the query view **T_COQ4V1** as the source for this new data provider.

    a)   Double click on *New Data Provider* in the design area.

    b)   Select the data provider type *Query View* using the radio button.

    c)   Enter **T_COQ4V1** as the name of the query view and select *OK*.

7. Next, add a second *Analysis* web item from the Standard group of web items to your template to the right of the first Analysis item.

   a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

   b) Drag the *Analysis* web item to your web template and drop it to the right of the first Analysis item. It will automatically position itself below the first Analysis item.

8. Check that the second Analysis item has automatically been assigned to data provider **DP_2**. If not, change the data provider assigned to the second Analysis item *(ANALYSIS_ITEM_2)* to be **DP_2**.

   a) In the *Properties* work area, make sure that *ANALYSIS_ITEM_2* is selected.

   b) On the *General* tab, check that the *Source* field is set to **DP_2**. Change it if necessary.

9. Save your web template using the values below. Then execute the template and navigate by selecting values from the Dropdown Box to filter the results. What do you notice about the results as you navigate?

When you are finished, exit the results and return to the Web Application Designer for the next task.

Field	Value
Description	`GR## Basic Template`
Technical Name	`GR##BASIC`

   a) In the Web Application Designer, click on the *Save* 💾 icon.

   b) Enter the values shown above for the *Description* and *Technical Name* fields.

   c) Select *OK* when finished.

   d) Select the *Execute* 🔲 icon to execute your web application.

   e) Logon to the NetWeaver portal using your assigned user ID and password.

   f) From the displayed results, navigate by choosing various values for *Calendar Year/Month*. You'll notice that only the first Analysis item responds to the choices made in the Dropdown Box.

   g) Exit the results and return to the Web Application Designer.

 SAP

Task 2: Affected Data Providers

In the previous task, only the first Analysis item responded to the choices made from the Dropdown Box. Now you will link the second Analysis item to the Dropdown box.

1. In the Web Application Designer, access the properties of the *Dropdown Box* in your web template. Make the following changes to the properties:

Property/Parameter	Value
Data Binding > DP_1 (Selection of Characteristic	`Click on button at end of row`
Selection of Characteristic > Affected Data Providers	`DP_1`
Selection of Characteristic > Affected Data Providers	`DP_2` Selection made on the next row.

    a) Select the *Dropdown Box* in the layout of your template.

    b) In the *Properties* work area, scroll to the end of the list of properties to find the *Data Binding* properties.

    c) Proceed as directed in the table above to set the *Affected Data Providers* properties of the Dropdown box. You will make the settings so that **both** the DP_1 and DP_2 data providers are affected by navigation done on the Dropdown box.

    d) Select *OK* when finished making the property settings.

2. Save and execute your web template. Navigate using the Dropdown box and notice the effect on the results.

    a) In the Web Application Designer, click on the *Save* 💾 icon.

    b) Select the *Execute* 🖳 icon to execute your web application.

    c) Navigate by choosing values from the Dropdown Box. Now, both Analysis items are affected by the chosen filter value.

          2007/Q2

Lesson Summary

You should now be able to:
- Create a simple web application
- Change the properties of web items
- Execute the web application and navigate in the results

Related Information

- Additional documentation is available for all the topics mentioned in this lesson at http://help.sap.com in the NetWeaver 2004s online documentation for Business Intelligence. This can be particularly helpful for the web item properties.

Lesson: Enhancing Web Template Layouts

Lesson Overview

This lesson focuses on the tools and techniques for enhancing the look and feel of your web applications. The placement and alignment of items as well as the addition of text and graphics can all make a positive impact on the user's experience.

Lesson Objectives

After completing this lesson, you will be able to:

- Align objects as desired in the web application
- Add texts, images and tabs to your web application
- List the impact of portal themes on your web application

Business Example

Your web applications are difficult to interpret because the content seems disorganized. You need to learn more about the options available for improving the look and feel of the web application.

Web Item Alignment

Creating web applications where the content is arranged haphazardly on the screen leads to user complaints, misinterpretation of results and low usage by the users. This section will discuss several easy techniques to insure that your web applications are well-arranged and easier to interpret.

- An HTML Table serves as an alignment grid for other web items.
- To create, choose *Table → Insert Table* or use the Insert Table icon.
- Table cells can be merged and split to enhance the layout.
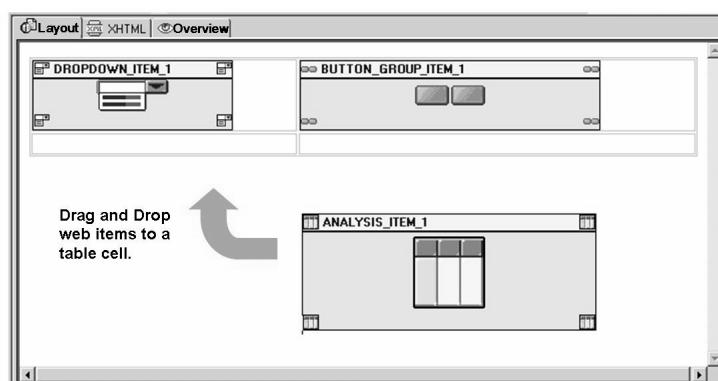
Figure 62: Using HTML Tables

One of the easiest ways to align a collection of web items is with the **HTML table**. The HTML table serves as a grid into which web items are placed. This provides an effective method for placing web items side-by-side or in a vertical arrangement on the web page.

To create an HTML table, use the menu path *Table → Insert Table* or simply click on the *Table* 📇 icon on the toolbar. You are then prompted to choose the number of rows and columns for the table and to set other global properties such as the height and width in pixels.

The table will be inserted into your web template at the current cursor location. At any time, you can use the context menu in the table (right mouse click inside the table) to make changes. From the context menu you can:

- Add or delete rows or columns or delete the table itself
- Edit the properties (for example: horizontal and vertical alignment of items) in a cell, row or the table itself
- Split and merge cells

In order to merge cells, follow these steps:

1. Place some text in each of the cells to be merged.
2. Select the text with the mouse.
3. Choose *Merge Cells* from the context menu or the *Table* menu.

To split cells that have been merged, follow the same procedure as for merging cells, but select *Split Cells* from the menu.

The purpose of both the Container and the Container Layout web items is to group other objects together under one technical name.
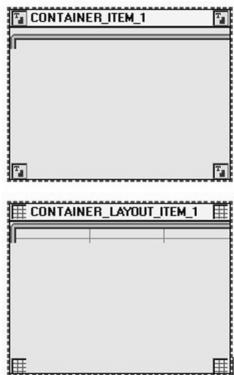
The Container web item is simply an object within which are placed other web items, text, graphics or HTML code. No method of aligning the items is provided. This item is useful to place multiple web items on one tab page, for example.

The Container Layout web item makes it possible to arrange other web items in a row and column grid, much like the HTML table. But unlike the HTML table, technical functions such as accessibility features are automatically applied to the Container Layout.

Figure 63: Container versus Container Layout

The *Container* and *Container Layout* web items play an important role in the design of web applications. These web items are used to group or combine other web items together. This is useful, for example, with other web items like the

Tab Pages web item. The Tab Pages web item only allows one web item to be assigned to each tab. If your design calls for the inclusion of a chart and some buttons to invoke changes to the chart's properties, you could not assign both the chart and the buttons to the same tab. The solution is to first place the chart and the buttons inside a Container or Container Layout web item, then you can assign that web item to the tab.

As the graphic above points out, the difference in the two web items is that the Container Layout web item provides alignment properties that let you easily arrange the contents into various row and column layouts. The Container web item does not have those properties.

If you have users that make use of the accessibility features of BI (cursor key navigation in lieu of using a mouse, for example), the Container Layout web item supports those features, while the HTML table does not.

- **The Tab Pages web item is used to organize template content on various tab pages.**
- **The sequence in which items are added to the Tab Page web item determines the sequence of the tab pages.**
- **Only one web item can be assigned to each tab.**
- **To assign multiple items to a tab, first assign the items to a Container web item. Then assign the Container to the tab.**
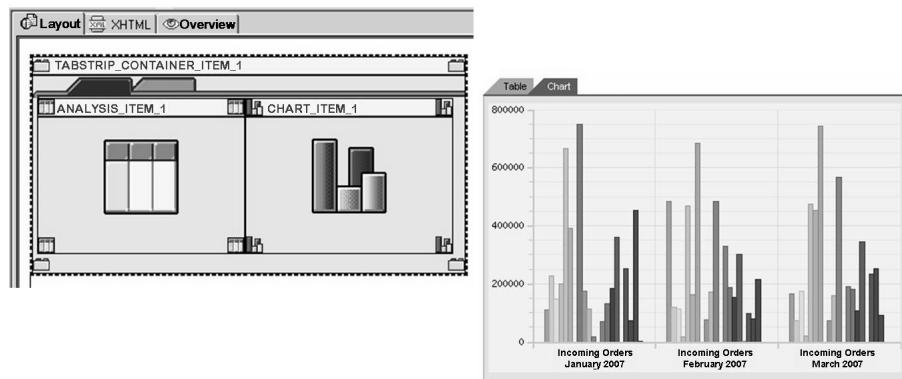
Figure 64: Using Tab Pages

The *Tab Page* web item provides a professional looking arrangement of items on a web application. The advantage of using tabs is that the user can easily navigate to a variety of data sets by simply clicking on the appropriate tab. In addition, using tabs is a good way to save display space in a web application. Many analyses can be displayed in a relatively small area.

To use the Tab page web item, you should first place it in your web template, then place the items that are to be assigned to the tabs inside the Tab Page web item. You can only assign an item such as an Analysis item or Chart item to a tab if the item is already placed inside the Tab Page web item.

Height and width parameters are set here. Note: The width and height parameters of the items assigned to the tabs can also influence the size displayed in the web application.

Tabs can be added or deleted with these buttons.

Enable or disable navigation here.

The active tab for the initial display is set here.

The button at the end of each tab in the 'Tab Panel' accesses the parameters screen.

There, you maintain the tab's:
- **caption**
- **icon**
- **commands**
- **associated web item.**

Figure 65: Tab Page Properties

Once you have placed all the relevant content into the Tab Pages web item, it is time to set the properties of the Tab Pages. This is done on the *Properties* screen area as shown in the graphic above.

The first set of properties deals with the overall height and width of the Tab Pages web item as it is initially displayed. Keep in mind that the height and width parameters of the content of the tab will override the Tab Pages setting.

The *Internal Display* properties are the most important ones for the Tab Pages. Here you define the actual tabs that will appear. The *Initial Tab Index* property determines the tab number that will be initially displayed when the web application is executed. The tabs are numbered starting with 1. Clicking on the button at the end of each tab number will access the parameters for assigning which web item will appear on the tab, the caption and any icon you want to appear on the tab as well as any commands that you want to execute when a tab is selected or deselected. These command properties make it possible to do very sophisticated navigations by only clicking on the tab.

Using Text, Images and Styles

This section discusses how to enhance your web application through the use of text and images and the role that style settings play in conjunction with the portal themes.

- **Text can be entered directly on the Layout view of the template.**
- **The Text Editing toolbar facilitates changing the text as desired.**
- **The Language Dependent Text button can be used for language dependent and independent texts.**
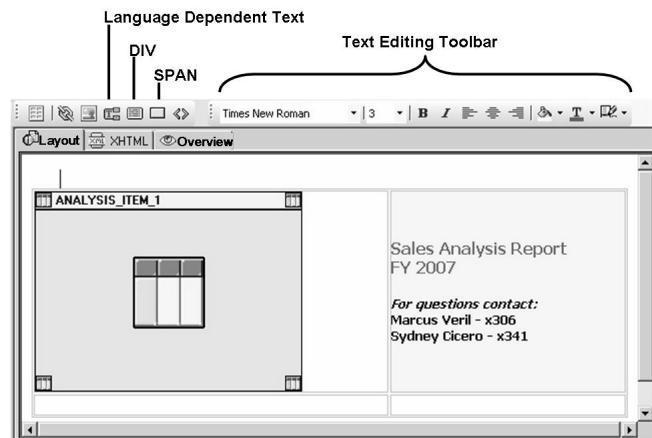
Figure 66: Using Text in the Web Template

Web applications are not simply presentations of data. You will often need to provide some textual information to help the user put the data in context, or to provide additional direction as to how the web application is to be used. For these reasons, the Web Application Designer makes it easy to insert text directly into the web template. Simply place the cursor at the starting position of your text and begin to type.

The *Text Toolbar* in the Web Application Designer lets you modify the following properties of the text:

- Font and font size
- Bold and/or italic
- Alignment of the selected text
- Background color of the web template
- Text color
- Text background color

The *<DIV>* and *<SPAN>* HTML tags can also be used to specify properties of textual information. The <DIV> tag is used to specify a container within which you can put text and apply various properties, such as the font, color, size and alignment of the text. This is useful for formatting text in different areas of the header of your web application, for example.

The <SPAN> tag defines an in-line text container and is often used to apply specific CSS (Cascading Style Sheet) styles to parts of a text block. For example, you could use the <SPAN> tag to insert a style to change the color and style of the text if the user hovered over the text with the cursor.

Hint: You can access Microsoft's HTML documentation directly from the Web Application Designer. This documentation is helpful for seeing working examples of any HTML tag. The documentation is available by a link when you choose *Tag <BODY> Properties* from the context menu of any text field.

This dialog is available wherever text can be entered:
- Web items such as Tabs, Buttons or Group for captions
- Text entered directly in the Layout view of the template
- In the Data Binding property of the Text web item
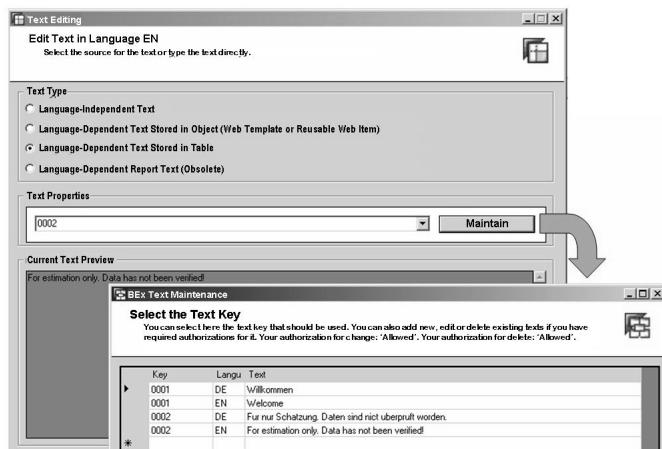
Figure 67: Language-Dependent Text Dialog

If your environment must support users with different languages, you can use language-dependent text in your web application. Since the web application is executed in the NetWeaver portal, it is the logon language of the NetWeaver portal user ID that determines which text language is displayed. The dialog screen shown above offers the following choices for text input:

- Language independent text

  This text will always show in the web application as it is entered, regardless of the logon language of the user.

- Language dependent text stored in object (Web template or reusable web item)

  With this option, the text can only be entered in the current logon language of the web template developer. Thus, if a text was needed in two languages, the web template developer would logon in the first language, enter the text in that language, then logon in the second language to enter the text in the second language. While text maintenance may be somewhat cumbersome because of the need for multiple logons, the text is stored with the template, and so is automatically transported with the template object.

- Language dependent text stored in table

  This option allows the web template developer to enter the text in as many languages as needed in the same session. Each text has a language key to identify it. However, the texts are stored in a separate table (*RSBEXTEXTS*) and it must be transported separately from the web templates.

- Language dependent report text (Obsolete)

  When you select this text type, you can select the texts of an ABAP report program that you have created. These texts have a translation connection. They are not transported with the web template must be created in the relevant ABAP report program in the BI system.

Menu path: *Insert → Image*

MIME Repository file path:
bwmimerep:///sap/bw/mime/Customer/
Images/<filename>

Sales Analysis Report
Q1/2007

For questions contact:
Marcus Veril - x306
Sydney Cicero - x341

Country ⇕	Sold to party ⇕	Incoming Orders ⬩ January 2007 ⬩ EUR	Incoming Orders ⬩ February 2007 ⬩ EUR	Incoming Orders ⬩ March 2007 ⬩ EUR
Germany	1001	108.132,47	484.220,26	166.092,05
	1033	227.107,85	117.555,80	72.552,05
	1172	146.595,21	112.900,10	175.481,54

Edit HTML Element

Tag <IMG> </IMG>
Open linked documentation...

Custom | Attributes | CSS Style

Image Source
Source [bwmimerep:///sap/bw/mime/Customer/Images/consultant.bmp] [...]

Images from other
locations are accessible
with this button.

Alternate Text
Text [null]

Layout
Alignment [▼]
Border Thickness [0] [⇕]

Here, the image file *consultant.bmp* from the MIME
Repository is specified.

You can import your own images into the MIME
Repository in the *SAP/BW/Customer/Images* folder.

Spacing
Horizontal [0] [⇕]
Vertical [0] [⇕]

Access to the MIME Repository is through
transaction code SE80 or via *SAP Menu → Business
Explorer → Mime Repository.*

Figure 68: Inserting Images

You can easily enhance the look of your web applications with the use of relevant
graphic images. Graphic files with the extensions **.bmp, .jpg and .gif** can be
used in the web applications. The MIME Repository on the BI system is used to
store these graphic files. Access to the MIME Repository is through the SAP BI
menu using the path *SAP Menu → Business Explorer → Mime Repository* or
with transaction code SE80.

Within the MIME Repository there is a standard folder structure supplied for
storing a variety of object types. For graphic files, the folder path is *SAP → BW
→ Customer → Images* and you can easily import your own graphic images to
this folder and then reference them in your web templates.

In the Web Application Designer, inserting an image is initiated by the menu
path *Insert → Image* or by clicking on the *Image* 🖼 icon on the toolbar. The
dialog window shown in the graphic above opens and you can add the name of
your image file in the MIME Repository to the end of the defaulted file path,
replacing the ellipses with the file name. The image will only have a graphic
placeholder in the Web Application Designer, but will render properly when the
web application is executed.

- **In NetWeaver 2004s, web templates are not assigned cascading style sheets (CSS) as they were in NetWeaver 2004. Instead, portal themes developed in the NetWeaver Portal, are adopted by the web application.**
- **In the Web Application Designer, the CSS attributes can be changed on individual web items and will override the styles of the portal theme in that specific instance.**
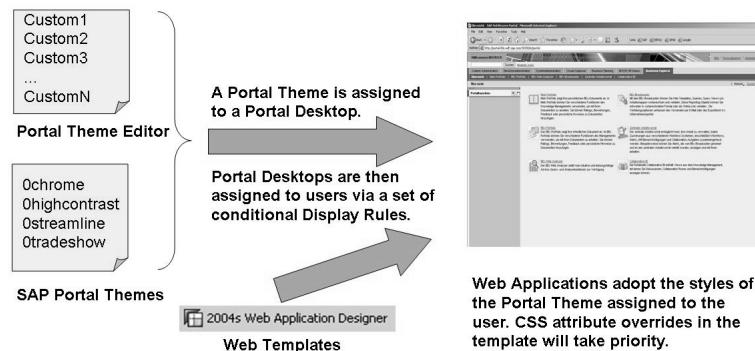
Figure 69: Using Themes and Styles

Since all BI web applications are displayed on the NetWeaver portal, the portal themes are used to apply HTML-like styles to the objects in the web application. Prior to NetWeaver 2004s BI, a cascading style sheet was assigned to each web template.

In the graphic above, you can see that the portal themes replace the cascading style sheets. These portal themes serve the same purpose, but are created and maintained with the *Theme Editor* in the NetWeaver Portal. These themes, in turn, are assigned to a portal desktop. Portal desktops are a collection of parameters that define the look of a particular portal environment. Portal users are assigned to a portal desktop and thus the web applications they execute will take on the look defined by the connected portal theme.

In the Web Application Designer, it is possible to override the styles from the portal theme for any of the tags in your web template. For example, for text you can simply select the text you wish to change, right click to bring up the context menu and choose *Tag <P> Properties* to display the *Edit HTML* dialog window. There you can access any of the CSS styles from the *CSS Styles* tab.

Exercise 6: Enhancing Web Template Layouts

Exercise Objectives

After completing this exercise, you will be able to:

- Use HTML tables to align objects in the web template
- Enhance your web templates with text, images and tab pages
- Use the Theme Editor to influence the styles applied to your web application

Business Example

You want the web templates that you develop to be well-designed and visually pleasing so that the users will be able to find the information provided in a clear and logical way. You also want to influence the color scheme of certain portions of your web applications.

Task 1: Using HTML Tables

In this task you will use an HTML table to align the web items in your template.

1. Start the BEx Web Application Designer and create a new web template.

2. To begin, create a new data provider and use the query **T_COQ4** to supply the data.

3. Insert an HTML table into your template. The table should have two rows and two columns.

4. Insert another HTML table inside the upper left-hand cell of the first HTML table. The table should have one row and two columns.

5. You want to assure that items placed in the cells are aligned the same. Using the context menu in the last row of the first HTML table, set the vertical alignment of the entire row to `Top`.

6. Choose the *Dropdown Box* web item from the Standard group of web items and drag it into your second HTML table in the left-hand cell.

7. Add another *Dropdown Box* web item into your second HTML table in the right-hand cell.

8. Add an *Analysis* web item into your first HTML table in the lower left-hand corner cell of the HTML table.

9.   Add a *Chart* web item into your HTML table in the lower right-hand corner cell of the first HTML table.

> **Hint:** The right-hand cells of the HTML table will probably be very narrow at this time. Use care to drop the web item into the right-hand cell and not add it to the left-hand cell.

10.   As a check, your template arrangement should now look like the table below:

Dropdown Box 1	Dropdown Box 2	
Analysis Item 1		Chart 1

11.   Set the properties of the first Dropdown box as follows:

Property/Parameter	Value
Data Binding > Data Binding Type	`Char/Structure Member`
%NM% (Selection of Characteristic)	`Click on button at end of row`
Selection of Characteristic > Data Provider	`DP_1`
Selection of Characteristic > Characteristic	`0CALMONTH` (use button at end of row)
Selection of Characteristic > Label Visible	`On (checked)`

12.   Set the properties of the second Dropdown box as follows:

Property/Parameter	Value
Data Binding > Data Binding Type	`Char/Structure Member`
%NM% (Selection of Characteristic)	`Click on button at end of row`
Selection of Characteristic > Data Provider	`DP_1`
Selection of Characteristic > Characteristic	`0DISTR_CHAN` (use button at end of row)
Selection of Characteristic > Label Visible	`On (checked)`

13.   Save your web template using the values below.

Continued on next page

Field	Value
Description	`GR## Layout 1`
Technical Name	`GR##LAYOUT1`

Then execute the template and navigate by selecting values from the Dropdown Boxes to filter the results. What do you notice about the alignment of the web items in the web application?

When you are finished, exit the results and return to the Web Application Designer for the next task.

Task 2: Using Text, Images and the Tab Pages Web Item

To make navigation between various elements easy for the user, you want to insert tabs into your web application.

1. In the Web Application Designer, create a new web template.

2. To begin, create a new data provider and use the query **T_COQ4** to supply the data.

3. Insert an HTML table at the top of your template. The HTML table should have 1 row and 2 columns.

4. Insert a logo inside the left-hand cell of the HTML table. Use the *sap_logo.gif* file.

5. Insert some free text inside the right-hand cell of the HTML table, such as **Q4 Sales Analysis**. Change the font and font size as desired. Apply a background color of your choice to the text.

6. Choose the *Tab Pages* web item from the Advanced group of web items and drag it into web template below the HTML table.

7. Add an *Analysis* web item into the *Tab Pages* web item.

8. Add a *Container Layout* web item into the *Tab Pages* web item to the right of the *Analysis* web item.

9. Add **two** *Chart* web items into the *Container Layout* web item.

10. Create another data provider (DP_2) based on the query view **T_COQ4V1** and assign it to the second *Chart* web item.

11. Now it is time to configure the tab pages. In the *Properties* work area, make the following settings:

Continued on next page

Property/Parameter	Value
Internal Display > 1 ?? (Tab Panel)	`Click on button at end of row`
Tab Panel > Caption	`Table`
Tab Panel > Subordinate Web Item	`ANALYSIS_ITEM_1`

12. Configure the second tab to show the contents of the *Container Layout* web item by making the following settings:

Property/Parameter	Value
Internal Display > 2 ?? (Tab Panel)	`Click on button at end of row`
Tab Panel > Caption	`Charts`
Tab Panel > Subordinate Web Item	`CONTAINER_LAYOUT_ITEM_1`

13. Now it is time to configure the *Container Layout* web item. In the *Properties* work area, select the *CONTAINER_LAYOUT_ITEM_1* and make the following settings:

Property/Parameter	Value
Internal Display > Row	`Click on button at end of row`
Row (0) > * Column	`Click on button at end of row`
Column > Subordinate Web Item	`CHART_ITEM_1`, then choose *OK*.
Row (1) > * Column	`Click on button at end of row`
Column > Subordinate Web Item	`CHART_ITEM_2`, then choose *OK* two times.

14. Save your web template using the values below.

Field	Value
Description	`GR## Layout 2`
Technical Name	`GR##LAYOUT2`

Then execute the template and navigate by selecting the tabs to change views.

Continued on next page

Task 3: Using the Theme Editor

You want to make some changes to the standard portal theme currently assigned to your user regarding how *Analysis* items are displayed.

1. Logon to the assigned BI system and display your User Menu. From the User Menu, choose *Access Portal*. Logon to the NetWeaver Portal using your user ID and password.

2. In the NetWeaver Portal, access the *Theme Editor* from the *System administration* menu. In the Theme Editor window, locate your group's theme using the table below. Click on your assigned portal theme to begin the editing process.

Group Number	Portal Theme		Group Number	Portal Theme
00 or 20	BI_THEME_A		10 or 30	BI_THEME_K
01 or 21	BI_THEME_B		11 or 31	BI_THEME_L
02 or 22	BI_THEME_C		12 or 32	BI_THEME_M
03 or 23	BI_THEME_D		13 or 33	BI_THEME_N
04 or 24	BI_THEME_E		14 or 34	BI_THEME_O
05 or 25	BI_THEME_F		15 or 35	BI_THEME_P
06 or 26	BI_THEME_G		16 or 36	BI_THEME_Q
07 or 27	BI_THEME_H		17 or 37	BI_THEME_R
08 or 28	BI_THEME_I		18 or 38	BI_THEME_S
09 or 29	BI_THEME_J		19 or 39	BI_THEME_T

3. In the Theme Editor window, scroll down and locate the *Complex Elements* group. Within that group, select *Tables*. For the Tables element, make the following changes to its styles:

Style	Action
Alternating Table Cells > Background Color of Alternating Table Cell > Options	Select a color and choose **Save**. Make a note of the color chosen.
Semantic Colors > Background Color of Subtotal Cells > Options	Select a color and choose **Save**. Make a note of the color chosen.
Semantic Colors > Background Color of Total Cells > Options	Select a color and choose **Save**. Make a note of the color chosen.
Save your theme changes	Select **Save**.

2007/Q2   **171**

4. Logoff from the NetWeaver Portal. Close any remaining portal windows or portal logon windows.

5. Logoff from the BI system.

6. In order to flush the browser cache, you must also logoff from your Citrix session.

7. Now, logon to your Citrix session and then to the BI system. Once in the BI system, go to your *Favorites* folder and execute your *GR## Layout 1* web application. Observe how the colors have changed on the various parts of the *Analysis* web item.

Solution 6: Enhancing Web Template Layouts

Task 1: Using HTML Tables

In this task you will use an HTML table to align the web items in your template.

1.   Start the BEx Web Application Designer and create a new web template.

      a)   Start the BEx Web Application Designer using the path *Start → Programs → Business Explorer → Web Application Designer*.

      b)   Logon to the system with the user ID and password provided by your instructor.

      c)   Select *"Create new Web Template"*.

2.   To begin, create a new data provider and use the query **T_COQ4** to supply the data.

      a)   Double click on *New Data Provider* in the design area.

      b)   Select the data provider type *Query* using the radio button.

      c)   Enter **T_COQ4** as the name of the query and select *OK*.

3.   Insert an HTML table into your template. The table should have two rows and two columns.

      a)   Click on the *Insert Table* 🔲 icon or from the context menu in your template choose *Insert... → Table*.

      b)   In the *Edit HTML Element* screen, change the *Column* value to **2** and choose *OK*.

4.   Insert another HTML table inside the upper left-hand cell of the first HTML table. The table should have one row and two columns.

      a)   Click inside the upper left-hand cell of the HTML table to position the cursor there. Then, click on the *Insert Table* 🔲 icon or from the context menu inside the upper left-hand cell choose *Insert... → Table*.

      b)   In the *Edit HTML Element* screen, change the *Row* value to **1** and the *Column* value to **2**. Choose *OK*.

Continued on next page

5. You want to assure that items placed in the cells are aligned the same. Using the context menu in the last row of the first HTML table, set the vertical alignment of the entire row to **Top**.

    a) Right-click inside the lower left-hand cell of the first HTML table access the context menu for the table. Then using the path *Table → Edit → Edit Row <tr>* set the *Vertical* field to **Top**.

    b) Choose *OK*.

6. Choose the *Dropdown Box* web item from the Standard group of web items and drag it into your second HTML table in the left-hand cell.

    a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

    b) Drag the *Dropdown Box* web item to your web template into the left-hand cell of the second HTML table.

7. Add another *Dropdown Box* web item into your second HTML table in the right-hand cell.

    a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

    b) Drag the second *Dropdown Box* web item to your web template into the right-hand cell of the second HTML table.

8. Add an *Analysis* web item into your first HTML table in the lower left-hand corner cell of the HTML table.

    a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

    b) Drag an *Analysis* web item to your web template into the lower left-hand corner cell of the first HTML table.

9. Add a *Chart* web item into your HTML table in the lower right-hand corner cell of the first HTML table.

    💡 **Hint:** The right-hand cells of the HTML table will probably be very narrow at this time. Use care to drop the web item into the right-hand cell and not add it to the left-hand cell.

    a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

    b) Drag a *Chart* web item to your web template into the lower right-hand corner cell of the first HTML table.

10. As a check, your template arrangement should now look like the table below:

Dropdown Box 1	Dropdown Box 2	
Analysis Item 1		Chart 1

a)   -

11. Set the properties of the first Dropdown box as follows:

Property/Parameter	Value
Data Binding > Data Binding Type	`Char/Structure Member`
%NM% (Selection of Characteristic)	`Click on button at end of row`
Selection of Characteristic > Data Provider	`DP_1`
Selection of Characteristic > Characteristic	`0CALMONTH` (use button at end of row)
Selection of Characteristic > Label Visible	`On (checked)`

a)   In the *Properties* work area, set the properties for the fields as shown in the table above.

> **Hint:** The 'Dot' button at the end of some property fields will open a new window where additional selections can be made.

b)   When finished, select *OK*.

12. Set the properties of the second Dropdown box as follows:

Property/Parameter	Value
Data Binding > Data Binding Type	`Char/Structure Member`
%NM% (Selection of Characteristic)	`Click on button at end of row`

Selection of Characteristic > Data Provider	**DP_1**
Selection of Characteristic > Characteristic	**0DISTR_CHAN** (use button at end of row)
Selection of Characteristic > Label Visible	**On (checked)**

     a)   In the *Properties* work area, set the properties for the fields as shown in the table above.

     b)   When finished, select *OK*.

13.   Save your web template using the values below.

Field	**Value**
Description	**GR## Layout 1**
Technical Name	**GR##LAYOUT1**

Then execute the template and navigate by selecting values from the Dropdown Boxes to filter the results. What do you notice about the alignment of the web items in the web application?

When you are finished, exit the results and return to the Web Application Designer for the next task.

     a)   In the Web Application Designer, click on the *Save* 💾 icon.

     b)   Enter the values shown above for the *Description* and *Technical Name* fields.

     c)   Select *OK* when finished.

     d)   Select the *Execute* 🗗 icon to execute your web application.

     e)   Logon to the NetWeaver portal using your assigned user ID and password.

     f)   From the displayed results, navigate by choosing various values for *Calendar Year/Month* and *Distribution Channel*. You'll notice that the Analysis item and the Chart item are side-by-side, and that the two Dropdown Boxes are aligned over the Analysis item results.

     g)   Exit the results and return to the Web Application Designer.

Task 2: Using Text, Images and the Tab Pages Web Item

To make navigation between various elements easy for the user, you want to insert tabs into your web application.

1. In the Web Application Designer, create a new web template.

    a)   In the Web Application Designer, select *"Create new Web Template"*.

2. To begin, create a new data provider and use the query **T_COQ4** to supply the data.

    a)   Double click on *New Data Provider* in the design area.

    b)   Select the data provider type *Query* using the radio button.

    c)   Enter **T_COQ4** as the name of the query and select *OK*.

3. Insert an HTML table at the top of your template. The HTML table should have 1 row and 2 columns.

    a)   Click on the *Insert Table* [icon] icon or from the context menu inside the template choose *Insert...* → *Table*.

    b)   In the *Edit HTML Element* screen, change the *Row* value to **1** and the *Column* value to **2**. Choose *OK*.

4. Insert a logo inside the left-hand cell of the HTML table. Use the *sap_logo.gif* file.

    a)   Click inside the left-hand cell of the HTML table to position the cursor there. Click on the *Image* [icon] icon or from the context menu inside the template choose *Insert...* → *Image*.

    b)   In the *Edit HTML Element* screen, replace the **...** values at the end of the file path in the *Source* field with **sap_logo.gif** .

    c)   Select *OK* to save your setting.

5. Insert some free text inside the right-hand cell of the HTML table, such as **Q4 Sales Analysis**. Change the font and font size as desired. Apply a background color of your choice to the text.

    a)   Place the cursor inside the right-hand cell of the HTML table and type a suitable title for the analysis.

    b)   Select the text, then apply your chosen settings for the font, font size and background color using the *Text toolbar*.

2007/Q2   **177** SAP

6. Choose the *Tab Pages* web item from the Advanced group of web items and drag it into web template below the HTML table.

    a) In the Web Items area of the Web Application Designer, open the *Advanced* web item category.

    b) Drag the *Tab Pages* web item into your web template.

7. Add an *Analysis* web item into the *Tab Pages* web item.

    a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

    b) Drag an *Analysis* web item to your web template and drop it inside of the *Tab Pages* web item.

8. Add a *Container Layout* web item into the *Tab Pages* web item to the right of the *Analysis* web item.

    a) In the Web Items area of the Web Application Designer, open the *Advanced* web item category.

    b) Drag a *Container Layout* web item to your web template and drop it inside of the *Tab Pages* web item to the right of the *Analysis* web item.

9. Add **two** *Chart* web items into the *Container Layout* web item.

    a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

    b) Drag a *Chart* web item to your web template and drop it inside of the *Container Layout* web item. Repeat the process with another *Chart* web item.

10. Create another data provider (DP_2) based on the query view **T_COQ4V1** and assign it to the second *Chart* web item.

    a) Double click on *New Data Provider* in the design area.

    b) Select the data provider type *Query View* using the radio button.

    c) Enter **T_COQ4V1** as the name of the query view and select *OK*.

    d) Select the second *Chart* web item. Then on the *General* tab of the *Properties* work area, assign data provider **DP_2**.

11. Now it is time to configure the tab pages. In the *Properties* work area, make the following settings:

Property/Parameter	Value
Internal Display > 1 ?? (Tab Panel)	`Click on button at end of row`
Tab Panel > Caption	`Table`
Tab Panel > Subordinate Web Item	`ANALYSIS_ITEM_1`

    a)   In the *Properties* work area for the *TABSTRIP_CONTAINER_ITEM_1* web item, set the properties for the fields as shown in the table above.

    b)   When finished, select *OK*.

12.   Configure the second tab to show the contents of the *Container Layout* web item by making the following settings:

Property/Parameter	Value
Internal Display > 2 ?? (Tab Panel)	`Click on button at end of row`
Tab Panel > Caption	`Charts`
Tab Panel > Subordinate Web Item	`CONTAINER_LAYOUT_ITEM_1`

    a)   In the *Properties* work area for the *TABSTRIP_CONTAINER_ITEM_1* web item, set the properties for the fields as shown in the table above.

    b)   When finished, select *OK*.

13.   Now it is time to configure the *Container Layout* web item. In the *Properties* work area, select the *CONTAINER_LAYOUT_ITEM_1* and make the following settings:

Property/Parameter	Value
Internal Display > Row	`Click on button at end of row`
Row (0) > * Column	`Click on button at end of row`

Continued on next page

Column > Subordinate Web Item	**CHART_ITEM_1**, then choose *OK*.
Row (1) > * Column	**Click on button at end of row**
Column > Subordinate Web Item	**CHART_ITEM_2**, then choose *OK* two times.

a) In the *Properties* work area for the *TABSTRIP_CONTAINER_ITEM_1* web item, set the properties for the fields as shown in the table above.

b) When finished, select *OK*.

14. Save your web template using the values below.

Field	**Value**
Description	**GR## Layout 2**
Technical Name	**GR##LAYOUT2**

Then execute the template and navigate by selecting the tabs to change views.

a) In the Web Application Designer, click on the *Save* 🔲 icon.

b) Enter the values shown above for the *Description* and *Technical Name* fields.

c) Select *OK* when finished.

d) Select the *Execute* 🔲 icon to execute your web application.

e) Logon to the NetWeaver portal using your assigned user ID and password.

f) From the displayed results, navigate by selecting each tab to change the displayed results.

Continued on next page

Task 3: Using the Theme Editor

You want to make some changes to the standard portal theme currently assigned to your user regarding how *Analysis* items are displayed.

1. Logon to the assigned BI system and display your User Menu. From the User Menu, choose *Access Portal*. Logon to the NetWeaver Portal using your user ID and password.

    a) If you are not already logged on to the BI system, then logon using the menu path *Start → SAP Logon* to display the SAP Logon window.

    b) Select your BI system from the screen and choose *Log on*.

    c) Complete the logon process using your assigned user ID and password.

    d) Switch to the *User Menu*, if necessary. There choose *Access Portal*.

    e) Logon to the NetWeaver Portal using your user ID and password.

2. In the NetWeaver Portal, access the *Theme Editor* from the *System administration* menu. In the Theme Editor window, locate your group's theme using the table below. Click on your assigned portal theme to begin the editing process.

Group Number	Portal Theme		Group Number	Portal Theme
00 or 20	BI_THEME_A		10 or 30	BI_THEME_K
01 or 21	BI_THEME_B		11 or 31	BI_THEME_L
02 or 22	BI_THEME_C		12 or 32	BI_THEME_M
03 or 23	BI_THEME_D		13 or 33	BI_THEME_N
04 or 24	BI_THEME_E		14 or 34	BI_THEME_O
05 or 25	BI_THEME_F		15 or 35	BI_THEME_P
06 or 26	BI_THEME_G		16 or 36	BI_THEME_Q
07 or 27	BI_THEME_H		17 or 37	BI_THEME_R
08 or 28	BI_THEME_I		18 or 38	BI_THEME_S
09 or 29	BI_THEME_J		19 or 39	BI_THEME_T

    a) In the NetWeaver Portal, select the *System administration* tab from the top level menu. The *Theme Editor* should open automatically and display various portal themes in the workspace to the right.

    b) Using the table above, determine the portal theme that has been assigned to your group. Locate it and then click on it to begin the editing process.

3. In the Theme Editor window, scroll down and locate the *Complex Elements* group. Within that group, select *Tables*. For the Tables element, make the following changes to its styles:

Style	**Action**
Alternating Table Cells > Background Color of Alternating Table Cell > Options	Select a color and choose **Save**. Make a note of the color chosen.
Semantic Colors > Background Color of Subtotal Cells > Options	Select a color and choose **Save**. Make a note of the color chosen.
Semantic Colors > Background Color of Total Cells > Options	Select a color and choose **Save**. Make a note of the color chosen.
Save your theme changes	Select **Save**.

a) In the *Theme Editor* workspace, scroll in the list of elements until you locate the *Complex Elements* group. Within that group, locate and click on **Tables** to display its styles.

b) Locate and change the styles as specified in the table above. Remember to make a note of the colors you have selected for the specified styles. When finished, select **Save** to save the changes.

c) Respond to the popup message with **OK**.

4. Logoff from the NetWeaver Portal. Close any remaining portal windows or portal logon windows.

a) Select *Log Off → Yes* to exit the portal. Close any remaining portal windows.

5. Logoff from the BI system.

a) Select *System → Log off → Yes* from the BI menu.

6. In order to flush the browser cache, you must also logoff from your Citrix session.

a) Select *Start → Log Off → Log Off* to exit the Citrix session.

182   2007/Q2

7.   Now, logon to your Citrix session and then to the BI system. Once in the BI system, go to your *Favorites* folder and execute your *GR## Layout 1* web application. Observe how the colors have changed on the various parts of the *Analysis* web item.

a)   Start your Citrix session again as you normally would.

b)   Logon to the assigned BI system from the SAP Logon screen as you normally would.

c)   In the BI *Favorites* folder, locate the *GR## Layout 1* web application that you created earlier and double click on it to execute it.

d)   When prompted, logon to the NetWeaver Portal with your user ID and password.

e)   Observe the changes that have occurred on the table presentation of the results.

Lesson Summary

You should now be able to:
- Align objects as desired in the web application
- Add texts, images and tabs to your web application
- List the impact of portal themes on your web application

Lesson: Other Web Items

Lesson Overview

In this lesson we will explore some commonly-used web items that can add significant options to your web applications.

Lesson Objectives

After completing this lesson, you will be able to:

- Add a range of content to your web application with the Report, Ticker and Information Field web items
- Enhance the navigation capabilities of your web applications with the Navigation Pane, Exceptions, Conditions, Checkbox, Radio Buttons and Context Menu web items

Business Example

You have been successful building basic web templates, but you want to be able to add variety with additional web items.

Other Web Items

Web items offer many options for the presentation of data and navigation. In this lesson you will explore a mixture of both types that can be used to make your web applications more flexible and interesting.

- **Adds formatted reports to the web template**
- **Key properties:**
  - Internal Display → Report Design
    - Specify the technical name of the BEx Report
    - Button to launch BEx Report Designer

Sales Analysis – FY 2006		
Country	Sales Revenue	Cost of Sales
Germany	359.901.456 EUR	195.899.266 EUR
France	1.066.520 EUR	801.450 EUR
Great Britain	12.911.200 EUR	3.105.807 EUR
Italy	11.261.750 EUR	3.272.340 EUR
Norway	18.338.000 EUR	4.398.421 EUR
United States	41.357.559 EUR	24.976.608 EUR
Grand Total	**444.836.486 EUR**	**232.453.892 EUR**

Figure 70: Web Item - Report

With the *Report* web item you can insert BEx reports into the web application. These formatted reports are created using the *BEx Report Designer*.

In the properties of the Report web item, the *Internal Display → Report Design* property is used to specify which BEx report object will be displayed. You can enter the technical name directly in the field provided, or select the pushbutton beside the field to call the Report Designer. In the Report Designer you can create a new BEx report or open an existing one. When you close the Report Designer, the name of the BEx report will be automatically inserted in the web item along with the data providers that it references.
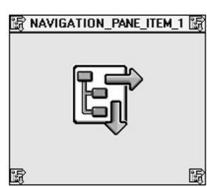
■ **Adds drag and drop-enabled navigation window to the web template**
■ **Key properties:**
   Internal Display → Navigation Pane Contents
      - Specify the data provider objects from the columns, rows, free characteristics or all objects
   Data Binding
      - Data provider specification
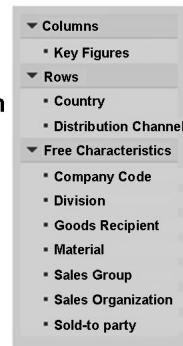
Figure 71: Web Item - Navigation Pane

With the *Navigation Pane* web item you can insert a navigation window into the web application. This is the same navigation window format that is used in the BEx Web Analyzer and enables drag and drop navigation as well as context menu navigation.

In the properties of the Navigation Pane web item, the *Internal Display → Navigation Pane Contents* property allows you to choose which elements of the assigned data provider will be shown in the web application. You can be very specific or easily select all the data provider elements using simple checkboxes.

Data binding is required for this web item and can be done on either the *General* tab or on the *Web Item Parameters* tab.

■ **Adds scrolling data to the web template**
■ **Key properties:**
Internal Display → Ticker Width in Characters
- **Specify the length of the display area in characters**
Behavior → Delay and Speed in Milliseconds
- **Start up delay and data scroll speed**
Data Binding
- **Data provider specification**

Figure 72: Web Item - Ticker

The *Ticker* web item creates a scrolling list of information from the assigned data provider and is very handy when you want to draw attention to summary statistics. This is the same navigation window format that is used in the BEx Web Analyzer and enables drag and drop navigation as well as context menu navigation.

The properties of the Ticker web item are very straightforward. In the *Internal Display → Ticker Width in Characters* property defines the length of the ticker text that will be displayed. The *Behavior → Delay in Milliseconds* defines the amount of time from the start of the web application execution that the ticker will wait before the data begins to move. The *Behavior → Speed in Milliseconds* property determines how many milliseconds will elapse before the data moves one character to the right. Lower numbers in this field result in a faster scroll rate.

Data binding is required for this web item and can be done on either the *General* tab or on the *Web Item Parameters* tab.

EXCEPTIONS_LIST_ITEM_1

■ **Adds list of exceptions to the web template**
■ **Key properties:**
     Data Binding
       - Data provider specification
■ **The Conditions web item has similar properties.**

Exception ⇕	Status ⇕	Definition ⇕
Margin % < 45%	Inactive	Bad 7: <45
Revenue < 15 Mill	Active	Bad 9: < 15.000

Add | Details | Toggle State | Delete

Sold-to Party Country ⇕		Sales Volume EUR ⇕	Cost of Sales ⇕	Margin % ⇕
		* 1.000 EUR	* 1.000 EUR	%
DE	Germany	359.901	195.899	46
FR	France	1.067	801	25
GB	Great Britain	12.911	3.106	76
IT	Italy	11.262	3.272	71
NO	Norway	18.338	4.398	76
US	United States	41.358	24.977	40
Overall Result		444.836	232.454	48

Figure 73: Web Item - Exceptions

If the data providers for an *Analysis* or *Chart* web item contain exceptions, then the *List of Exceptions* web item provides access to the exceptions to change easily activate or deactivate them. You can also use this web item to display the definition of any existing exception in the web application or to create new, ad hoc exceptions. Ad hoc exceptions are not saved as part of the web application definition.

The major property of this web item is the *Data Binding* property.

Much like the *List of Exceptions*, the *List of Conditions web item displays any conditions that are defined for the assigned data provider. The two web items are often used together where appropriate.*

■ **Adds selection boxes to the web template**
■ **Key properties:**
    Internal Display → Number of Columns
      - **Choose '1' to align entries vertically**
    Data Binding → Characteristic
      - **Specify which characteristic values or structure**
        members will be shown with checkboxes.

Figure 74: Web Item - Checkbox

The *Checkbox* web item is an easy to use filtering method where the number of choices is small. For example, if the characteristic *Customer Group* were assigned to a checkbox web item, then each customer group value would be displayed with a checkbox for selection purposes.

The checkbox web item allows multiple values to be selected simultaneously, as opposed to other filtering web items like the *Radio Button Group* that only allow single values to be selected.

The major properties of the checkbox are the *Data Binding* property where a characteristic or structure member is selected and the *Internal Display → Number of Columns* property that is used to determine the physical arrangement of the checkboxes in the web application.
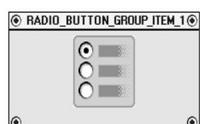
■ **Adds radio buttons to the web template**
■ **Key properties:**
    Internal Display → Number of Columns
      - **Select number of entries to be arranged**
        horizontally.
    Data Binding → Characteristic
      - **Specify which characteristic values or structure**
        members will be shown with radio buttons.

Figure 75: Web Item - Radio Button Group

The *Radio Button* web item provides a simple way to select single filter values for characteristics or to filter a list of key figures. Formatting options like the number of columns to display, the maximum number of values and whether or not to offer an *All* entry are available in the properties definition.

Figure 76: Web Item - Information Field

The *Information Field* web item provides a method to display any or all of the text elements associated with the data provider. Additionally, the filter values can be displayed.

The default properties of the web item display all available information in a list. You can, however, select groupings of text elements such as:

- Headers
- General text
- Static filters
- Dynamic filters
- Variable values

If specific text elements are required, then they can be named individually in the properties definition.

- **Controls available context menu choices in the web template.**
- **Key properties:**
    - Behavior
        - Each possible context menu choice can be enabled or disabled.
            - This facilitates extending or restricting the options available to users.

Figure 77: Web Item - Context Menu

The *Context Menu* web item allows the specification of which context menu choices are available to the user of the web template. In the *Properties* area for the web item each of the possible context menu choices are marked either as *On* or *Off*.

Marking a context menu choice as *On* does not ensure that it will be displayed in the context menu. Only relevant context menu choices for the chosen object are displayed. For example, you choose to enable the context menu choice *Hierarchy Drilldown*. In the web application, you right click on an *Analysis* item that is displaying the data from a query in which no hierarchy is assigned. In that case, the *Hierarchy Drilldown* choice will not be present in the context menu.

Exercise 7: Other Web Items

Exercise Objectives

After completing this exercise, you will be able to:

• Add a variety of functions to your web applications with various web items

Business Example

In order to meet sophisticated analysis needs, you want to learn how to use several other web items.

Task 1: Radio Buttons, Checkbox and Information Fields

You will create a new web template using radio buttons, checkboxes and various information fields.

1. Start the BEx Web Application Designer and create a new web template.

2. To begin, create a new data provider and use the query **T_COQ4** to supply the data.

3. Next, choose the *Radio Button Group* web item from the Standard group of web items and drag it into your template. This web item allows you to select one item from a list of items.

4. Set the properties of the Radio Button Group as follows:

Property/Parameter	Value
Internal Display > Label Visible	`Checked (On)`
Internal Display > Number of Columns	`5`
Data Binding > Characteristic	`Select the button at the end of the row`
Value Help Selector Screen	Select **0CALMONTH** from the list at the bottom

5. Next, choose the *Checkbox Group* web item from the Standard group of web items and drag it into your template after the Radio Button Group. This web item allows you to select multiple items from a list of items.

6. Set the properties of the Checkbox Group as follows:

Continued on next page

Property/Parameter	Value
Internal Display > Label Visible	**Checked (On)**
Internal Display > Number of Columns	**2**
Data Binding > Characteristic	**Select the button at the end of the row**
Valuehelp Selector Screen	Choose **Key Figures** from the list at the bottom of screen.

7.   Next, choose the *Analysis* web item from the Standard group of web items and drag it into your template after the Checkbox Group.

8.   Set the properties of the Analysis item as follows:

Property/Parameter	Value
Paging > Number of Data Rows Displayed at Once	**6**
Paging > Paging Area on Top of Table Visible	**Unchecked (Off)**

9.   Next, choose the *Information Field* web item from the Advanced group of web items and drag it into your template after the Analysis web item. This web item allows you to display informational text fields.

10. Set the properties of the Information Field as follows:

Property/Parameter	Value
Display > Width in Pixels	**300**
Data Binding > List of Text Elements to Display (0) > * General Text Elements	From the list choose **Query Description**.
Data Binding > List of Text Elements to Display (1) > * General Text Elements	From the list choose **Last Data Update**.

11. Save your web template using the values below. Then execute the template and navigate by selecting values from the Checkbox Group and Radio Button Group to filter the results.

     When you are finished, exit the results and return to the Web Application Designer for the next task.

Field	Value
Description	`GR## Web Items 1`
Technical Name	`GR##WEBITEMS1`

Task 2: Navigation Pane, Ticker and List of Exceptions

You will create a new web template using the Navigation Pane, Ticker and List of Exceptions web items. You will also learn how to merge cells in an HTML table.

1. Return to the BEx Web Application Designer and create a new web template.

2. To begin, create a new data provider and use the query **T_DCMGA** to supply the data.

3. Insert an HTML table into your template. The table should have two rows and three columns.

4. Change the properties of the second row of the HTML table so that items in the row are vertically aligned at the top of the cell. Use the context menu from the second row of the HTML table to access the properties of the row.

5. Now you will merge the first two cells of the first row of the HTML table to form one larger cell. First, type some text in the first and second cells of the first row of the HTML table. Then, select the text in both cells by dragging over the text with the mouse. This selects the first cell and the second cell of that row. Next, select the menu path *Table → Merge Cells*. You will notice that the first two cells have now been merged into one cell.

6. Next, choose the *Ticker* web item from the Miscellaneous group of web items and drag it into the first cell in the first row of the HTML table. This web item will scroll the information from its data provider inside a window.

7. Set the properties of the Ticker as follows:

Property/Parameter	Value
Display > Width in Pixels	`500`

8. Next, choose the *Navigation Pane* web item from the Standard group of web items and drag it into your template in the first cell of the second row of the HTML table. This web item allows you to navigate using either the context menu or drag and drop navigation. Use the default properties of this web item.

9. Next, choose the *Analysis* web item from the Standard group of web items and drag it into the second cell of the second row of the HTML table. Use the default property values for this web item.

Continued on next page

10. Next, choose the *List of Exceptions* web item from the Miscellaneous group of web items and drag it to the third cell of the second row of the HTML table. This web item allows you to access any of the exceptions defined for the assigned data provider. Accept the default values for this web item.

11. Save your web template using the values below. Then execute the template. Use drag and drop navigation from the Navigation Pane. Notice what happens to the information in the Ticker when you do. Activate and deactivate exceptions in the results.

    When you are finished, exit the results and return to the Web Application Designer for the next task.

Field	Value
Description	`GR## Web Items 2`
Technical Name	`GR##WEBITEMS2`

Task 3: Using Reports

You will create another web template to compare data in an *Analysis* web item to the same results in a *BEx Report* web item.

1. Return to the BEx Web Application Designer and create a new web template.

2. To begin, create a new data provider and use the query **SUMMARY_STATIC** to supply the data.

3. Insert an HTML table into your template. The table should have one row and two columns.

4. Change the properties of the only row of the HTML table so that items in the row are vertically aligned at the top of the cell. Use the context menu from the row of the HTML table to access the properties of the row.

5. Next, choose the *Analysis* web item from the Standard group of web items and drag it into the first cell of the HTML table. Use the default property values for this web item.

6. Next, choose the *Report* web item from the Standard group of web items and drag it into the second cell of the HTML table. Use the default property values for this web item.

7. Set the properties of the Report as follows:

Property/Parameter	Value
Internal Display > Report Design > %NM%	`Replace %NM% with your report GR##BW306REPORT1`
Data Binding > * Default	`DP_1`

8. Save your web template using the values below. Then execute the template. Compare the format of the data in the *Analysis* web item with the same data in the *Report* web item.

    When you are finished, exit the results and return to the Web Application Designer for the next task.

Field	Value
Description	`GR## Web Items 3`
Technical Name	`GR##WEBITEMS3`

Solution 7: Other Web Items

Task 1: Radio Buttons, Checkbox and Information Fields

You will create a new web template using radio buttons, checkboxes and various information fields.

1. Start the BEx Web Application Designer and create a new web template.

   a) Start the BEx Web Application Designer using the path *Start → Programs → Business Explorer → Web Application Designer*.

   b) Logon to the system with the user ID and password provided by your instructor.

   c) Select *"Create New Web Template"*.

2. To begin, create a new data provider and use the query **T_COQ4** to supply the data.

   a) Double click on *New Data Provider* in the design area.

   b) Select the data provider type *Query* using the radio button.

   c) Enter **T_COQ4** as the name of the query and select *OK*.

3. Next, choose the *Radio Button Group* web item from the Standard group of web items and drag it into your template. This web item allows you to select one item from a list of items.

   a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

   b) Drag the *Radio Button Group* web item to your web template as the first web item.

4. Set the properties of the Radio Button Group as follows:

Property/Parameter	Value
Internal Display > Label Visible	`Checked (On)`
Internal Display > Number of Columns	`5`
Data Binding > Characteristic	`Select the button at the end of the row`
Value Help Selector Screen	Select **0CALMONTH** from the list at the bottom

   a)   In the *Properties* work area, set the properties for the fields as shown in the table above.

> **Hint:** The 'Dot' button at the end of some property fields will open a new window where additional selections can be made.

   b)   When finished, select *OK*.

5.   Next, choose the *Checkbox Group* web item from the Standard group of web items and drag it into your template after the Radio Button Group. This web item allows you to select multiple items from a list of items.

   a)   In the Web Items area of the Web Application Designer, open the *Standard* web item category.

   b)   Drag the *Checkbox Group* web item to your web template and drop it to the right of the Radio Button Group.

6.   Set the properties of the Checkbox Group as follows:

Property/Parameter	Value
Internal Display > Label Visible	`Checked (On)`
Internal Display > Number of Columns	`2`
Data Binding > Characteristic	`Select the button at the end of the row`
Valuehelp Selector Screen	Choose **Key Figures** from the list at the bottom of screen.

   a)   In the *Properties* work area, set the properties for the fields as shown in the table above.

   b)   When finished, select *OK*.

7.   Next, choose the *Analysis* web item from the Standard group of web items and drag it into your template after the Checkbox Group.

    a)   In the Web Items area of the Web Application Designer, open the *Standard* web item category.

    b)   Drag the *Analysis* web item to your web template and drop it to the right of the Checkbox Group.

8.   Set the properties of the Analysis item as follows:

Property/Parameter	Value
Paging > Number of Data Rows Displayed at Once	`6`
Paging > Paging Area on Top of Table Visible	`Unchecked (Off)`

    a)   In the *Properties* work area, set the properties for the fields as shown in the table above.

    b)   When finished, select *OK*.

9.   Next, choose the *Information Field* web item from the Advanced group of web items and drag it into your template after the Analysis web item. This web item allows you to display informational text fields.

    a)   In the Web Items area of the Web Application Designer, open the *Advanced* web item category.

    b)   Drag the *Information Field* web item to your web template and drop it to the right of the Analysis web item.

10. Set the properties of the Information Field as follows:

Property/Parameter	Value
Display > Width in Pixels	`300`
Data Binding > List of Text Elements to Display (0) > * General Text Elements	From the list choose `Query Description`.
Data Binding > List of Text Elements to Display (1) > * General Text Elements	From the list choose `Last Data Update`.

    a)   In the *Properties* work area, set the properties for the fields as shown in the table above.

    b)   When finished, select *OK*.

11. Save your web template using the values below. Then execute the template and navigate by selecting values from the Checkbox Group and Radio Button Group to filter the results.

   When you are finished, exit the results and return to the Web Application Designer for the next task.

Field	Value
Description	`GR## Web Items 1`
Technical Name	`GR##WEBITEMS1`

   a) In the Web Application Designer, click on the *Save* 💾 icon.

   b) Enter the values shown above for the *Description* and *Technical Name* fields.

   c) Select *OK* when finished.

   d) Select the *Execute* 🖼 icon to execute your web application.

   e) Logon to the NetWeaver portal using your assigned user ID and password.

   f) From the displayed results, navigate by choosing various values from the Checkbox Group and the Radio Button Group.

   g) Exit the results and return to the Web Application Designer.

Task 2: Navigation Pane, Ticker and List of Exceptions

You will create a new web template using the Navigation Pane, Ticker and List of Exceptions web items. You will also learn how to merge cells in an HTML table.

1. Return to the BEx Web Application Designer and create a new web template.

   a) In the Web Application Designer, select *"Create New Web Template"*.

2. To begin, create a new data provider and use the query **T_DCMGA** to supply the data.

   a) Double click on *New Data Provider* in the design area.

   b) Select the data provider type *Query* using the radio button.

   c) Enter **T_DCMGA** as the name of the query and select *OK*.

3. Insert an HTML table into your template. The table should have two rows and three columns.

    a) Click on the *Insert Table* 🔲 icon or from the context menu in your template choose *Insert... → Table*.

    b) In the *Edit HTML Element* screen, change, if necessary the *Row* value to **2** and the *Column* value to **3**. Choose *OK*.

4. Change the properties of the second row of the HTML table so that items in the row are vertically aligned at the top of the cell. Use the context menu from the second row of the HTML table to access the properties of the row.

    a) Place your cursor in the second row of the HTML table. From the context menu choose *Table... → Edit... → Edit Row <tr> ...*.

    b) From the Edit HTML Element screen, go to the *Vertical* field and select **Top**.

    c) Select *OK* to finish.

5. Now you will merge the first two cells of the first row of the HTML table to form one larger cell. First, type some text in the first and second cells of the first row of the HTML table. Then, select the text in both cells by dragging over the text with the mouse. This selects the first cell and the second cell of that row. Next, select the menu path *Table → Merge Cells*. You will notice that the first two cells have now been merged into one cell.

    a) Enter a text character in each of the first two cells of the first row of the HTML table.

    b) Drag your cursor across the text in both cells with the mouse.

    c) Select the menu path *Table → Merge Cells*.

6. Next, choose the *Ticker* web item from the Miscellaneous group of web items and drag it into the first cell in the first row of the HTML table. This web item will scroll the information from its data provider inside a window.

    a) In the Web Items area of the Web Application Designer, open the *Miscellaneous* web item category.

    b) Drag the *Ticker* web item to your web template and place it in the first cell of the first row of the HTML table.

7. Set the properties of the Ticker as follows:

Property/Parameter	Value
Display > Width in Pixels	`500`

    a)   In the *Properties* work area, set the properties for the fields as shown in the table above.

8.   Next, choose the *Navigation Pane* web item from the Standard group of web items and drag it into your template in the first cell of the second row of the HTML table. This web item allows you to navigate using either the context menu or drag and drop navigation. Use the default properties of this web item.

    a)   In the Web Items area of the Web Application Designer, open the *Standard* web item category.

    b)   Drag the *Navigation Pane* web item to your web template and place it in the first cell of the second row of the HTML table.

9.   Next, choose the *Analysis* web item from the Standard group of web items and drag it into the second cell of the second row of the HTML table. Use the default property values for this web item.

    a)   In the Web Items area of the Web Application Designer, open the *Standard* web item category.

    b)   Drag the *Analysis* web item to the second cell of the second row of the HTML table.

10.   Next, choose the *List of Exceptions* web item from the Miscellaneous group of web items and drag it to the third cell of the second row of the HTML table. This web item allows you to access any of the exceptions defined for the assigned data provider. Accept the default values for this web item.

    a)   In the Web Items area of the Web Application Designer, open the *Miscellaneous* web item category.

    b)   Drag the *List of Exceptions* web item to the third cell on the second row of the HTML table..

11.   Save your web template using the values below. Then execute the template. Use drag and drop navigation from the Navigation Pane. Notice what happens to the information in the Ticker when you do. Activate and deactivate exceptions in the results.

When you are finished, exit the results and return to the Web Application Designer for the next task.

Field	Value
Description	`GR## Web Items 2`
Technical Name	`GR##WEBITEMS2`

a)   In the Web Application Designer, click on the *Save* 💾 icon.

b)   Enter the values shown above for the *Description* and *Technical Name* fields.

c)   Select *OK* when finished.

d)   Select the *Execute* 📲 icon to execute your web application.

e)   Logon to the NetWeaver portal using your assigned user ID and password.

f)   From the displayed results, navigate as described in the exercise and view the changes to the results.

g)   Exit the results and return to the Web Application Designer.

Task 3: Using Reports

You will create another web template to compare data in an *Analysis* web item to the same results in a *BEx Report* web item.

1.   Return to the BEx Web Application Designer and create a new web template.

a)   In the Web Application Designer, select *"Create New Web Template"*.

2.   To begin, create a new data provider and use the query **SUMMARY_STATIC** to supply the data.

a)   Double click on *New Data Provider* in the design area.

b)   Select the data provider type *Query* using the radio button.

c)   Enter **SUMMARY_STATIC** as the name of the query and select *OK*.

3.   Insert an HTML table into your template. The table should have one row and two columns.

a)   Click on the *Insert Table* ▦ icon or from the context menu in your template choose *Insert...* → *Table*.

b)   In the *Edit HTML Element* screen, change the *Row* value to **1** and the *Column* value to **2**. Choose *OK*.

4. Change the properties of the only row of the HTML table so that items in the row are vertically aligned at the top of the cell. Use the context menu from the row of the HTML table to access the properties of the row.

   a) Place your cursor in the row of the HTML table. From the context menu choose *Table... → Edit... → Edit Row <tr> ....*

   b) From the Edit HTML Element screen, go to the *Vertical* field and select **Top**.

   c) Select *OK* to finish.

5. Next, choose the *Analysis* web item from the Standard group of web items and drag it into the first cell of the HTML table. Use the default property values for this web item.

   a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

   b) Drag the *Analysis* web item to the first cell of the HTML table.

6. Next, choose the *Report* web item from the Standard group of web items and drag it into the second cell of the HTML table. Use the default property values for this web item.

   a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

   b) Drag the *Analysis* web item to the second cell of the second row of the HTML table.

7. Set the properties of the Report as follows:

Property/Parameter	Value
Internal Display > Report Design > %NM%	`Replace %NM% with your report GR##BW306REPORT1`
Data Binding > * Default	`DP_1`

   a) In the *Properties* work area, set the properties for the fields as shown in the table above.

8. Save your web template using the values below. Then execute the template. Compare the format of the data in the *Analysis* web item with the same data in the *Report* web item.

   When you are finished, exit the results and return to the Web Application Designer for the next task.

Field	Value
Description	`GR## Web Items 3`
Technical Name	`GR##WEBITEMS3`

a)   In the Web Application Designer, click on the *Save* 🖫 icon.

b)   Enter the values shown above for the *Description* and *Technical Name* fields.

c)   Select *OK* when finished.

d)   Select the *Execute* 🪟 icon to execute your web application.

e)   Logon to the NetWeaver portal using your assigned user ID and password.

f)   Compare the results displayed as an *Analysis* web item and a *Report* web item.

g)   Exit the results and return to the Web Application Designer.

Lesson Summary

You should now be able to:

- Add a range of content to your web application with the Report, Ticker and Information Field web items
- Enhance the navigation capabilities of your web applications with the Navigation Pane, Exceptions, Conditions, Checkbox, Radio Buttons and Context Menu web items

Related Information

- [Enter an optional reference using the URL or CrossReference tag to additional information that learner may find useful. Examples include websites or whitepapers. Delete if not used.]

Lesson: Charts and Maps

Lesson Overview

This lesson presents the concept and the architecture of BEx Maps. It also discusses the integration of static and dynamic geo-characteristics and their use in reporting.

Lesson Objectives

After completing this lesson, you will be able to:

- Present data in chart format in your web applications
- Describe the process of geocoding for characteristics
- Present data in map format in your web applications

Business Example

In your enterprise some query results are to be displayed on geographic maps. You want to use some static geo-characteristics immediately and learn how to maintain the SAPBWKEY.

In addition, you want to learn how to indicate location-relevant information on a map.

Designing Charts

Since a well-designed chart can add significant visual interest to a web application, it is not surprising to see charts used in many different formats in typical web applications. To effectively communicate their information, however, the designer of the chart must understand how to choose from a wide array of options and formats.

Adding charts and maps to your web application can
- **Provide more visual interest**
- **Show complex data relationships more clearly**
- **Speed the interpretation of the results**
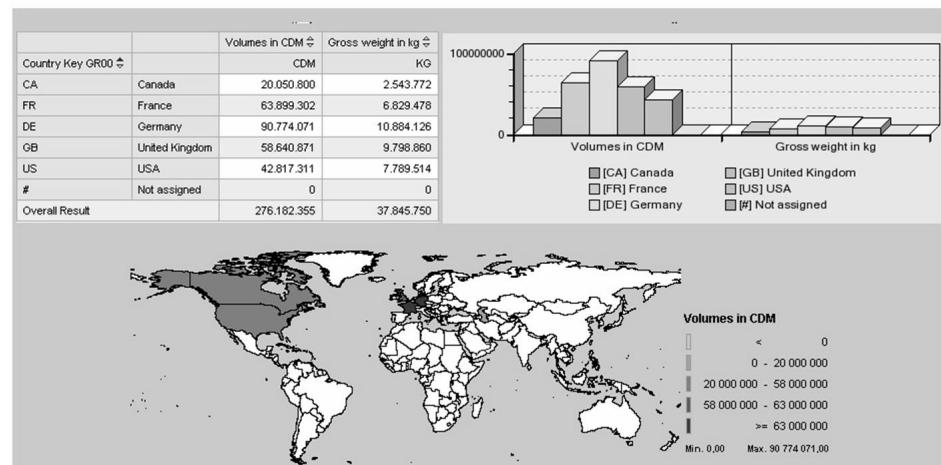
Figure 78: Using Charts and Maps

Often the purpose of the web application is to provide a 'analysis cockpit' or 'dashboard' where the user receives information in graphical form for quick interpretation of the results. Graphs are excellent ways to provide interesting visualizations of large amounts of data. Maps are useful if the data has some geographic relevance.

- **Adds charts of all types to the web template**
- **Key properties:**
  - Internal Display → Edit Chart
    - **- Button calls the Chart Wizard**
  - Data Binding → Data Provider
    - **- Specify the source of data to be charted**
  - Chart Text → Axis Description/Label
    - **- Enter language dependent axis descriptions**
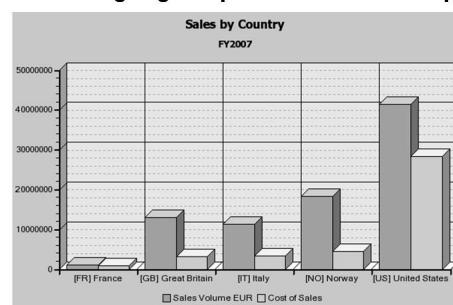
Figure 79: Web Item - Chart

In the BEx Web Application Designer, the *Chart* web item is used for any graphical display of data. One key property is the *Edit Chart* property under the *Internal Display* property grouping. By editing the chart web item you call the *Chart Wizard* which provides a step-by-step guide to producing useful charts of all types. Additionally, chart text can be made language-dependent to meet world-wide requirements.
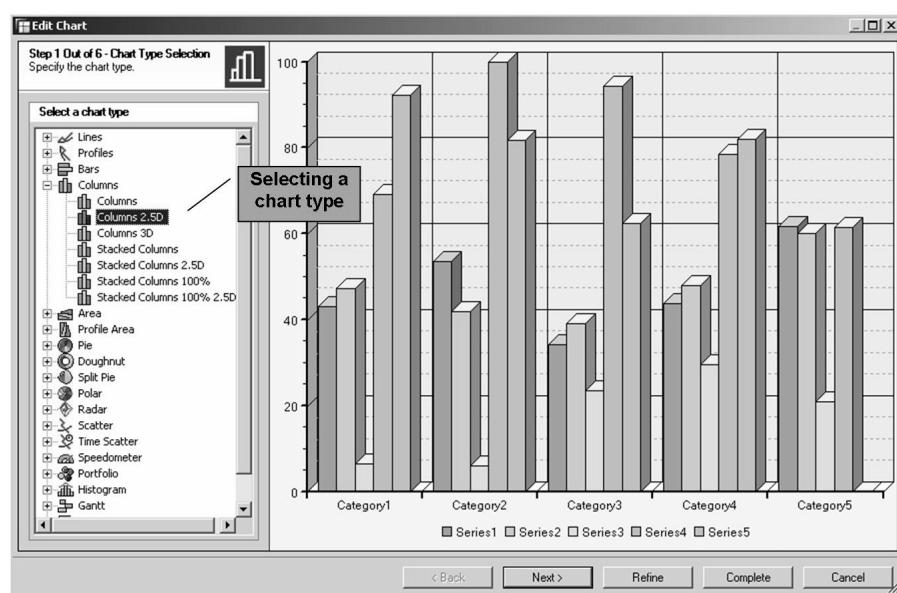
Figure 80: Chart Wizard - Step 1

In the Chart Wizard, you are guided through a six step process for choosing the major properties of a chart. Step one begins with selecting a chart type from an extensive array of charts. The selection screen is interactive, showing the user an example of the chart type selected.

At any step in the Chart Wizard, the user can call the Chart Designer by selecting the *Refine* button.
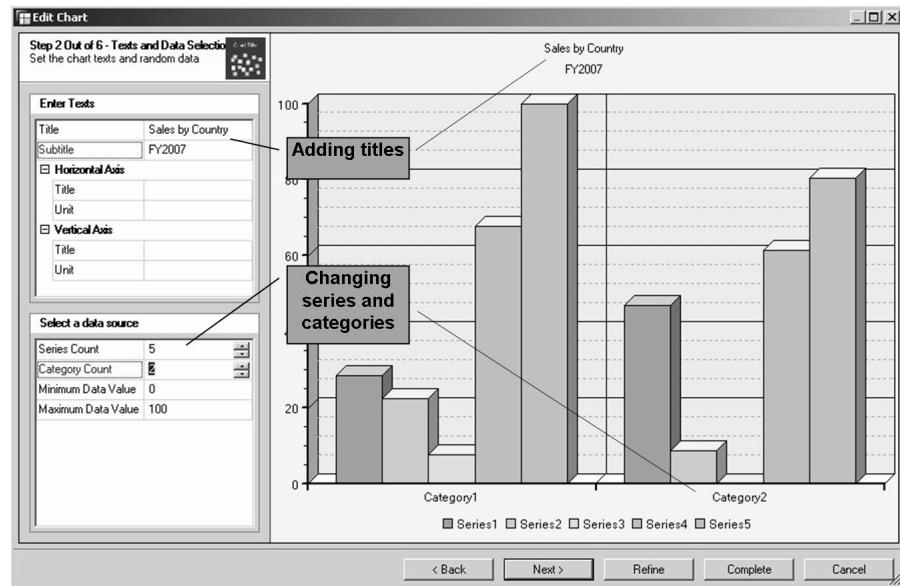
Figure 81: Chart Wizard - Step 2

In the second step of the Chart Wizard the task is to provide various texts (titles, subtitles and axis titles) and to set the number of data series that will be displayed. Again, the user sees the effect of any change to make verification of the design easy.
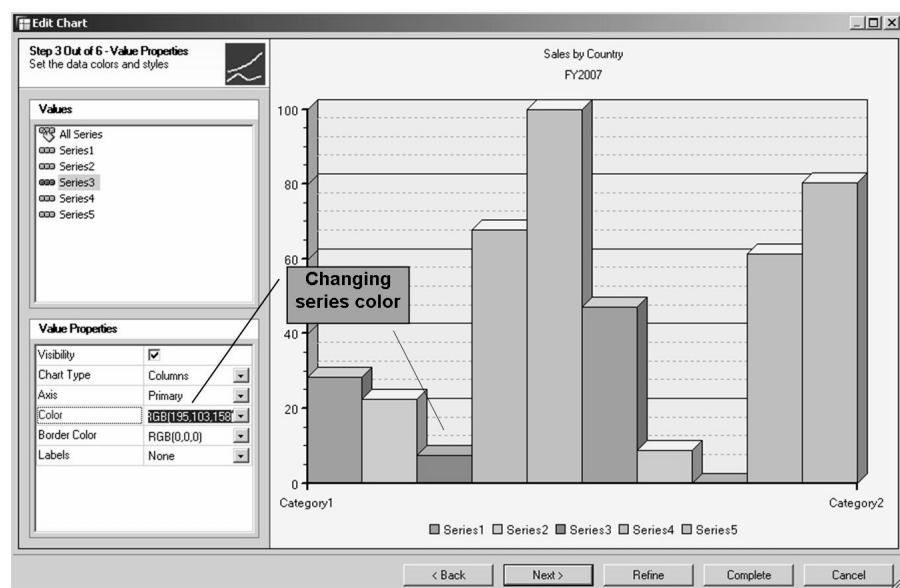
Figure 82: Chart Wizard - Step 3

Setting the properties of the series is performed in the third step of the Chart Wizard. The color, border color, labels and even the type of chart can be changed for each series independently. The *Visibility* property allows you to hide certain series as well.
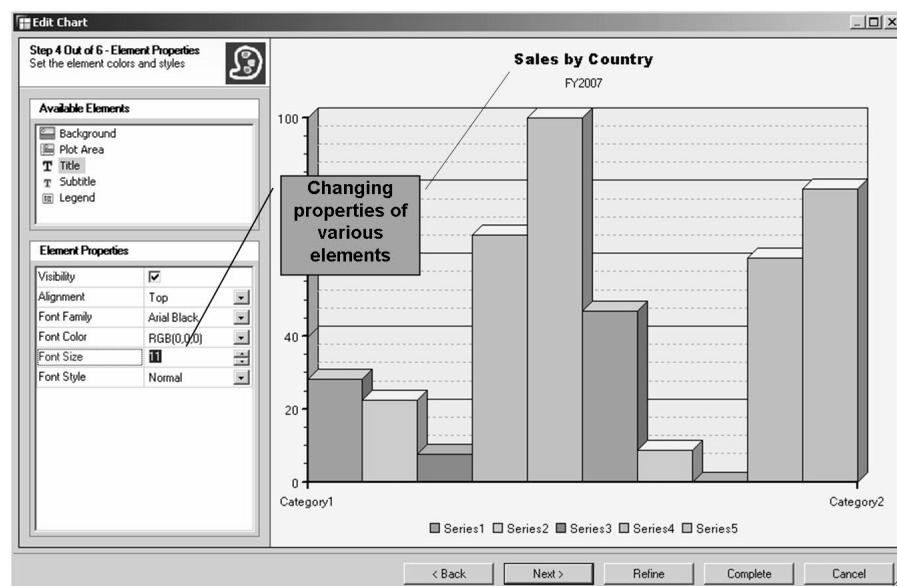
Figure 83: Chart Wizard - Step 4

In the fourth step of the Chart Wizard, you can change the properties of various elements such as the chart background, the plot area, the legends and the titles. For example, the graphic above shows that after selecting the element *Title*, the user can easily change the font family, color, size and style to achieve the desired effect on the chart.

■ **The Chart Editor provides access to all chart properties. You can switch between the Wizard and the Editor as needed.**
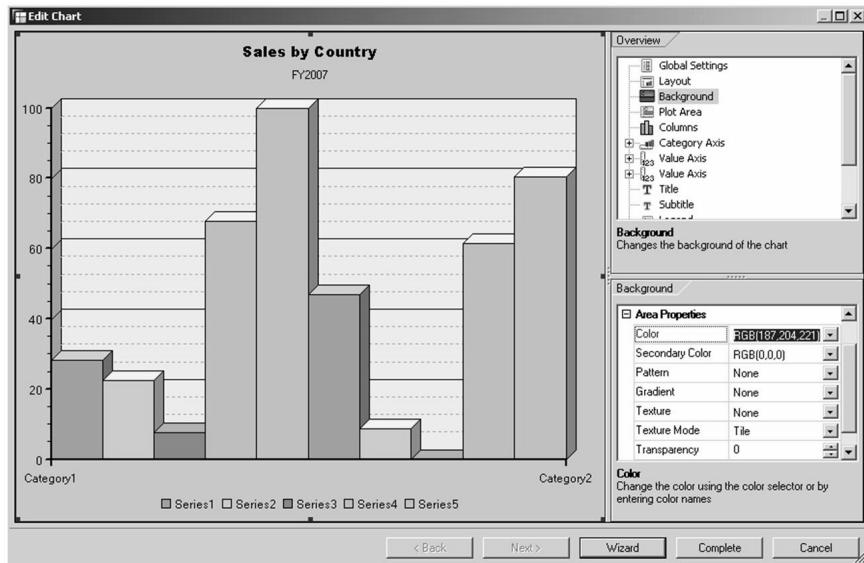
Figure 84: Chart Editor

The Chart Editor provides access to all chart properties and is useful for fine tuning the design and function of the chart. While in the Chart Editor, you can easily return to the Chart Wizard using the *Wizard* button. Therefore both tools are frequently used as needed for designing a chart.

With the Chart Editor, the user selects the property of interest from the *Overview* window, then accesses the parameters of this property in the window below. This allows for precise control over every parameter available for the chart.

Using Maps

Key Figures	Order Volume (Value)			
Distribution channel Division	Electronic Parts	High Tech	Service	Overall Result
Direct Sales		$ 83.870.034,75		$ 83.870.034,75
Final Customer Sales	$ 208.556.532,70	$ 307.167.314,82		$ 515.723.847,52
Sold for resale	$ 53.071.818,47	$ 246.148.942,29		$ 299.220.760,76
Service			$ 16.420.591,77	$ 16.420.591,77
Overall Result	$ 261.628.351,17	$ 637.186.291,86	$ 16.420.591,77	$ 915.235.234,80

Figure 85: Mapping Results

Numerous characteristics of the SAP BI possess geographic meaning, for example, customer, sales region, country. The geographic information, as well as the associated relevant characteristic values, can be evaluated in the BEx Map. The BEx Map is the geographical information system (GIS) of SAP BI. It is integrated into the Business Explorer.

Maps can be used to visualize economic connections and distributions. You can clearly represent, for example, specific characteristic numbers as hues, circle or bar charts for a country, a region or a city. You can also alter the layout of a map by either zooming in or out around detailed or region-specific information. In this way the marketing study of a certain product becomes related to a certain region or a country. Market opportunities or dangers can be recognized in one view.

In addition to using tables or diagrams to visualize information, you can represent geographically relevant information on a map using BEx Map.

Within the BEx Map, you can use different ways to visualize information:

- You can assign different colors to certain areas of maps.
- You can add diagrams (bar charts and/or pie charts).
- You can display location information (for example, customer addresses) in the form of points, triangles, squares or crosses on the map.
- You can use any query that contains geo-relevant characteristics for mapping.

Customer

Sales Office

Sales Personnel

Market potential

Regions

Countries
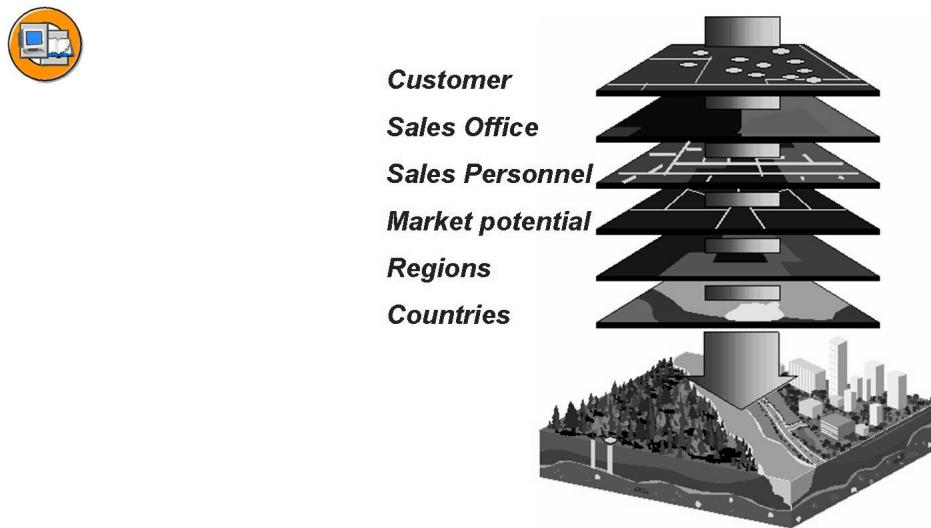
Figure 86: Map Layers

Maps of different degrees of detail can be layered on top of each other as indicated by different levels of characteristic values.

For example, if a characteristic number in a query is used on the country, region and postal code levels, three layers in the BEx Map can be indicated. The respective characteristic for the geographical analysis must be activated.

A higher level covers an underlying level if it has a larger, opaque surface over the map level below it.

Connections (or correlations) between data provide the basis for planning and actions because they are more easily recognizable when visualized on a map.
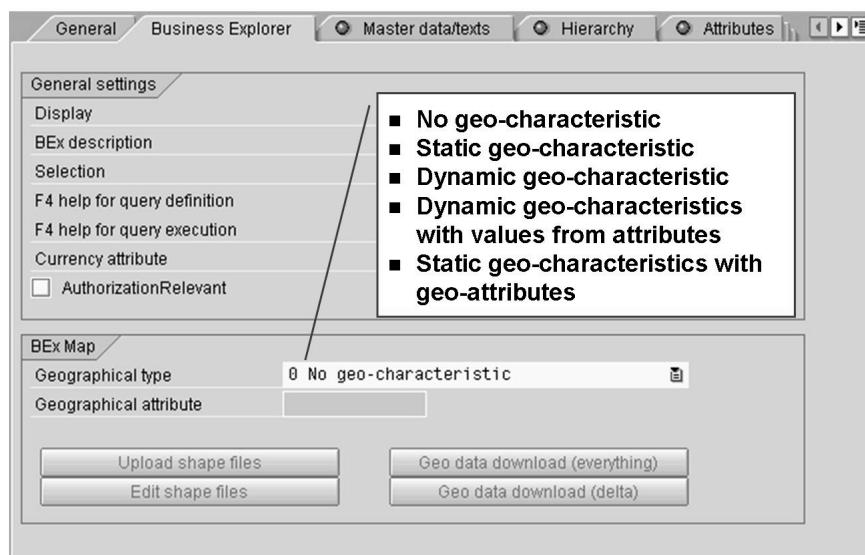
Figure 87: Geo-characteristic Types

To use an InfoObject in BEx Map, you must first define it as a geo-characteristic. This is done in the InfoObject maintenance of the Administrator Workbench.

In the listing of the InfoObjects, select the desired InfoObject and open the InfoObject in maintenance mode. On the *Business Explorer* tab, indicate the desired type of geo-characteristic in the *Geographical Type* field.

Activate the InfoObject to save the assignment.

Shape files make the static geo-characteristics available on the map.

The Structure of the Shape Files

- The Shape files consist of three differently formatted files:
    - ***.shp** – Geo-data that forms the map
    - ***.shx** – Geo-index to reduce map access time
    - ***.dbf** – Attributes for the geo-element, such as country or region

Static and dynamic geo-characteristics describe data with geographic relevance (for example, characteristics such as customer, sales region, country). This geo-relevant data is represented and evaluated by maps.

There are four different types of geo-characteristics: static, dynamic, dynamic
with values as attributes, and static with geo-characteristics.

- **Static geo-characteristics**

    – Characteristics that do not change over a long period of time (for
      example, country, state)

- **Dynamic geo-characteristics**

    – Characteristics that are subject to change (for example, customer, sales
      office)

- **Dynamic geo-characteristic with values as attributes**

    – Characteristics that are subject to change and whose geo-coordinates are
      derived from a referenced static geo-characteristic with geo-attributes
      (for example, customers referenced with the geo-characteristic *postal
      code*).

- **Static geo-characteristic with geo-attributes**

    – Static geo-characteristics are enhanced with geo-coordinates to
      determine geographic location; used with dynamic geo-characteristics
      that have values as attributes (for example, postal code).

Static geo-characteristic

A static geo-characteristic has geographic coordinates that do not often
change and describe a surface (polygon). Country or region are examples of
static geo-characteristics. Attributes of static geo-characteristics are defined
by the geometry of surfaces or polygons whose data are stored in shape files.

Dynamic geo-characteristic

A dynamic geo-characteristic has geographic coordinates that change
more frequently and describe a place (location information). Customer or
workplace are examples of dynamic geo-characteristics because they are
settled at a geographic point, which can be described by an address. The
address data of these characteristics is subject to frequent change. SAP BW
adds a set of standard attributes to this geo-characteristic. The geographic
coordinates of the appropriate object are stored for each line in the master
data table. A geographic coordinate includes the following geo-attributes:

Technical Name	Description	Data type	Length
LONGITUDE	geographic length of the place	DEC	15
LATITUDE	geographic width of the place	DEC	15

Technical Name	Description	Data type	Length
ALTITUDE	height of the place (compared to sea level)	DEC	17
PRECISEID	ID for accuracy of the data	NUMC	4
SRCID	ID for data source	CHAR	4

> **Hint:** Only the attributes LONGITUDE and LATITUDE are being used at present. ALTITUDE, PRECISID and SRCID are reserved for future use. If you change the geographic type of a characteristic to *No geo-characteristic*, these attributes in the InfoObject definition are deleted.

Dynamic geo-characteristic with values as attributes

To avoid having to geo-code each dynamic geo-characteristic separately, a dynamic geo-characteristic can take its geo-attributes (length, width, height) from another geo-coded dynamic characteristic (for example, postal code). Customer or workplace are examples of this type, *Dynamic geo-characteristic with values as attributes*(Type 3).

This geo-characteristic is treated in the system like a regular dynamic geo-characteristic, which describes a place (location information). On the database level, the master data table is not extended by the geo-attributes specified above. Instead, the geo-coordinates of a regular attribute of a characteristic are stored in the master data table.

> **Hint:** You want to define a dynamic geo-characteristic for workplace, which has postal code as attribute. The geo-coordinates are generated at run time from the postal code master data table.

> **Hint:** With the help of this technology, you avoid redundant entries in the master data table.

Static geo-characteristic with geo-attributes

A static geo-characteristic can refer to a geo-characteristic of type 3, which possesses geo-attributes (length, height, width). For example, postal code can be used as a static geo-characteristic with geo-attributes.

> **Note:** You can use 0POSTCD_GIS (postal code) as an attribute of the dynamic geo-characteristic 0BPARTNER (business partner), which receives its coordinates from this attribute. In this way the map reference for the business partner on its level is stored by postal code areas.
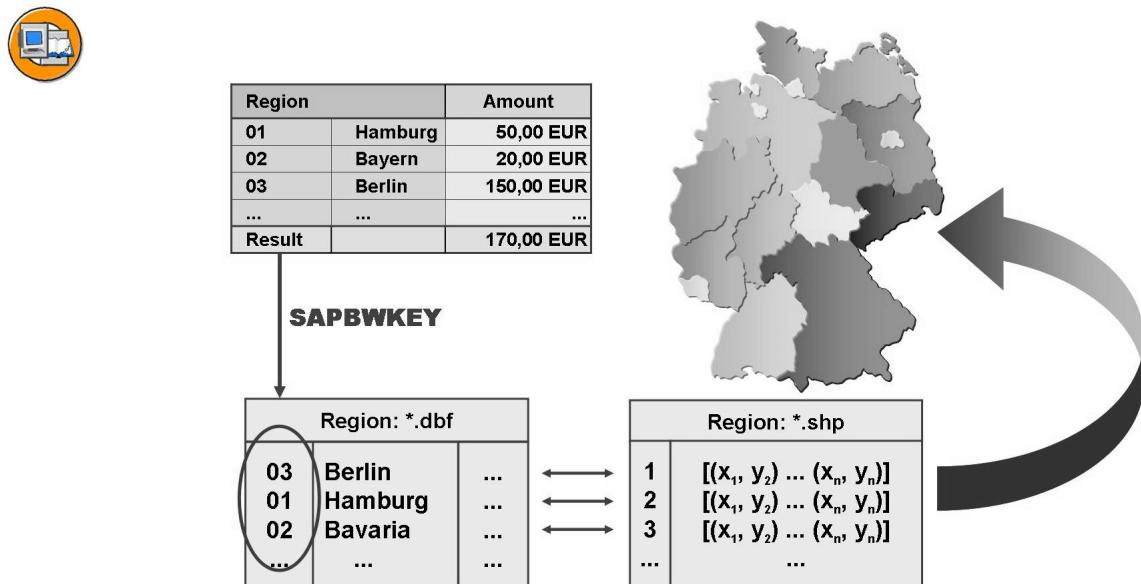
Figure 88: Linking Data to a Map

Static geo-characteristics are connected to shape file map coordinates using the SAPBWKEY.

When implementing a query for this geo-characteristic, BEx Map transfers the data for every area on the map.

To be able to maintain the SAPBWKEY, you have to load the three files for the shape file to your local hard disk. However, you can maintain the SAPBWKEY in the **dbf** file only. While it is possible to maintain the SAPBWKEY in Microsoft Excel or in ArcView, SAP recommends that you use Excel for small amounts of data because it is easy to operate.

- **The geo-coordinates are maintained during geo-coding using ArcView and subsequently uploaded as a CSV file into BI.**
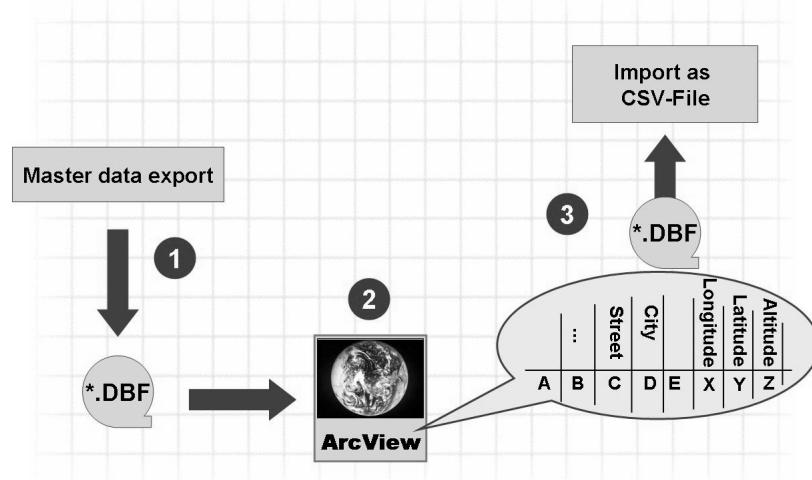
Figure 89: Dynamic Geo-characteristic Maintenance

If an InfoObject (on tab page *Business Explorer* in InfoObject maintenance) is marked as a dynamic geo-characteristic, the three attributes of the geographical height, length and width for your InfoObject are automatically defined. These attributes are loaded into a dbf file in ArcView where the geo-coding is done: i.e., you connect material geographical coordinates with the appropriate geo-attributes of the InfoObjects.

Then this data is loaded in form of a csv file into your BW system. The loading procedure is similar to loading data from flat file into the BW with a defined InfoSource.
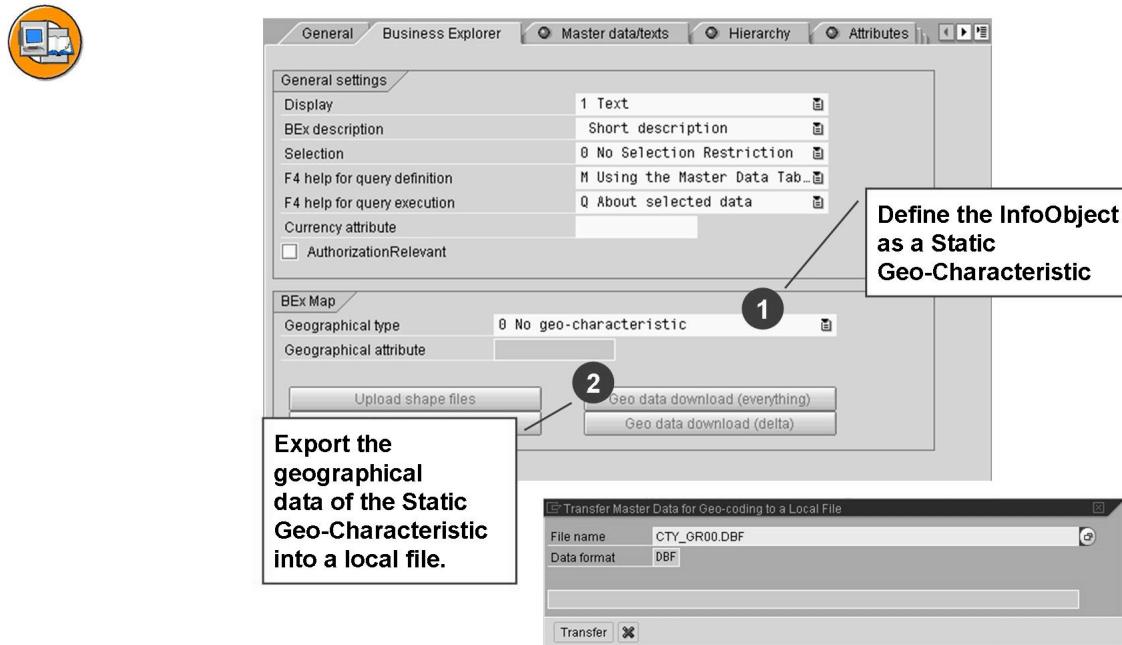
Figure 90: Geo Data Download

By choosing the *Geo Data Download (Everything)* button, you are prompted to save the three geocoding files to a location of your choice. Once the files have been downloaded, the **DBF** file can then be maintained with the SAPBWKEY values. This maintenance can be done with the ArcView tool or with Microsoft Excel, as shown in the next graphic.

Figure 91: SAPBWKEY Maintenance with Excel

Usually you use the shape files from the appropriate CD which is received after the purchase of BI. Load the three shape files into the same folder of your local hard drive. This would be the same folder in which you stored the geographical data of your geo-characteristic (e.g. SAPWorkDir). Open the file with the geographical data of the geo-characteristic and the DBF shape file with Excel. Transfer the SAPBWKEY value from the file of the geo-characteristic into the appropriate column of the DBF shape file. It is important to take care in maintaining the SAPBWKEY with Excel and in no case change the order of the rows in the DBF shape file! A correct correlation of the geo data to the polygons on the map would no longer be possible if the sequence is changed. Finally save your inputs and close the file.
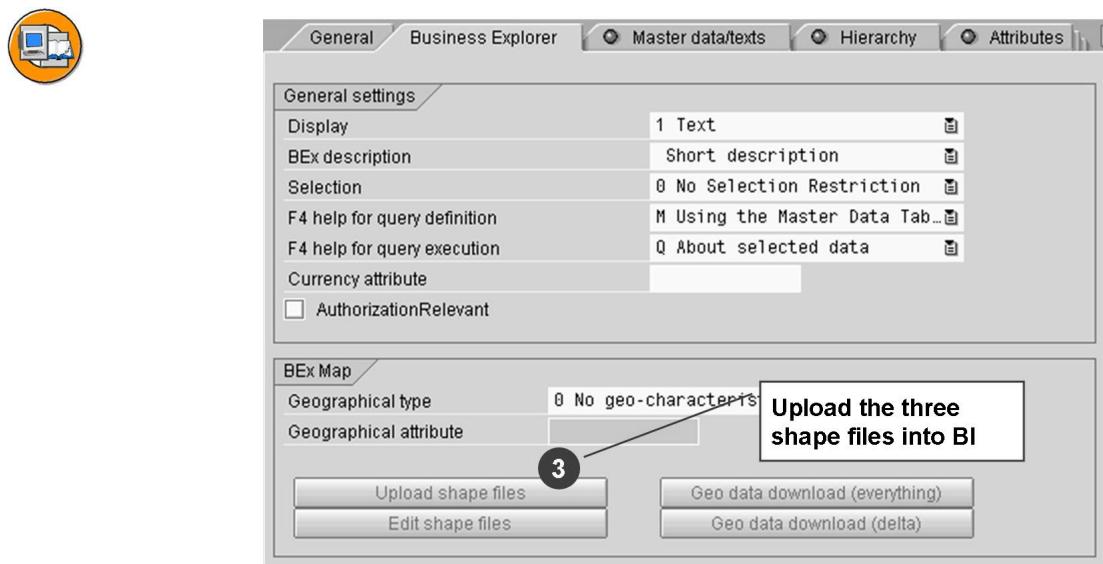
Figure 92: Geo Data Upload

Once the DBF file has been geocoded, you can reload the geographical information back into the InfoObject. Use the *Upload shape files* button in the InfoObject definition. You will be prompted for the location of each of the three shape files in sequence. Once the third shape file is reloaded, you are ready to use the geocoded InfoObject for mapping.
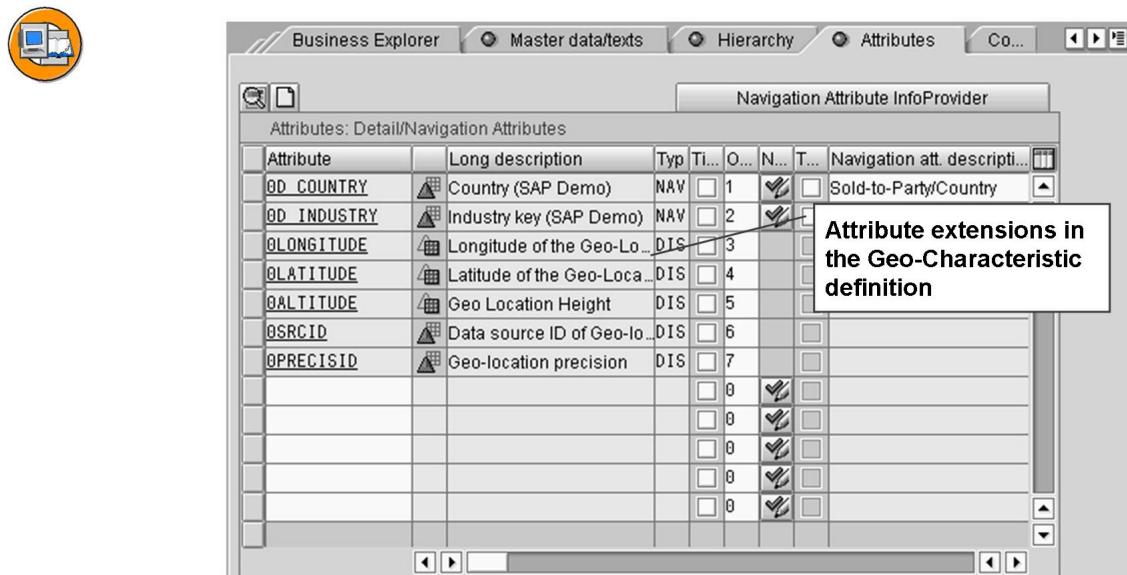
Figure 93: Attribute Extensions

If you define an InfoObject as a dynamic geo-characteristic, the geographical length, latitude and height are automatically added as additional attributes for your InfoObject. They can be seen on the *Attribute* tab page . Before the geo-coding, these attributes do not contain data.
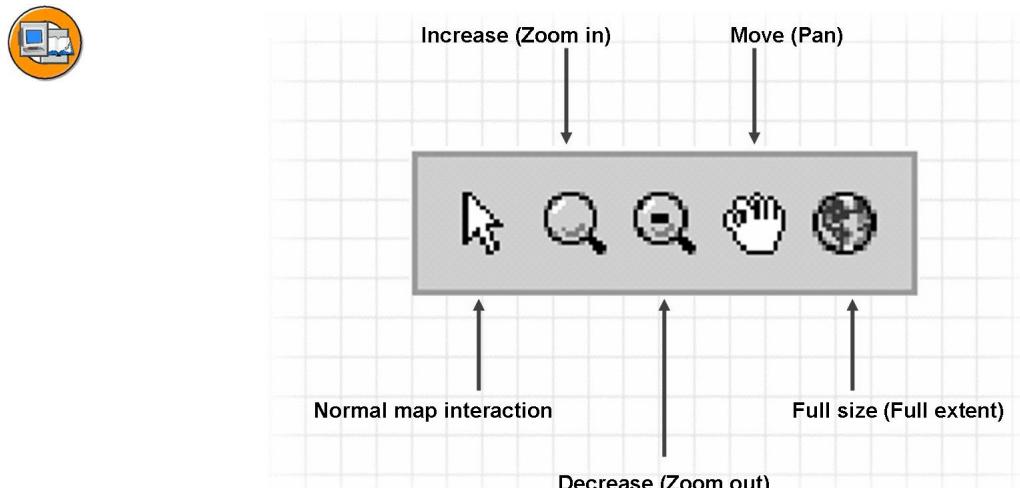
Figure 94: BExMap Functions

When a web application containing a map is executed, you can enable the parameter of the web item that displays a navigational toolbar for the user. With this toolbar, the user can zoom, pan and interact with the elements of the map.

Exercise 8: Charts and Maps

Exercise Objectives

After completing this exercise, you will be able to:

- • Create charts using the Chart Wizard and Chart Editor.
- • Perform simple geo-coding of maps for your web application.

Business Example

You want to be able to represent data graphically in your web applications using both charts and maps.

Task 1: Using the Chart Wizard

You will create some simple charts using the functionality of the Chart Wizard.

1. Start the BEx Web Application Designer and create a new web template.

2. To begin, create a new data provider and use the query **T_COQ4** to supply the data.

3. Next, choose the *Chart* web item from the Standard group of web items and drag it into your template. This web item allows you to show data in the format of a chart.

4. Set the properties of the Chart as follows:

Property/Parameter	Value
Display > Width in Pixels	`500`
Display > Height in Pixels	`400`
Internal Display > Edit Chart	`Select the button at the end of the row to launch the Chart Wizard`

5. In Step 1 of the Chart Wizard, change the chart type to **Columns 2.5D**. Click on *Next* when finished.

6. In Step 2 of the Chart Wizard, make the following settings:

Property/Parameter	Value
Enter Texts > Title	`GR## Revenue and Cost`
Enter Texts > Vertical Axis > Unit	`EUR`
Select a data source > Series Count	`6`
Select a data source > Category Count	`2`

Click on *Next* when finished.

7. In Step 3 of the Chart Wizard, accept all the default settings.

   Click on *Next* to proceed.

8. In Step 4 of the Chart Wizard, make the following settings:

Property/Parameter	Value
Background > Color > select the down arrow	Choose the `Light Yellow` color from the samples.
Plot Area > Color > select the down area	Choose a `Darker Yellow` color from the samples.
Title > Font Size	`11`
Title > Font Style	`Bold`

Click on *Next* when finished.

9. In Step 5 of the Chart Wizard, accept all the default settings.

   Click on *Next* to proceed. Then choose *Complete* to save your settings and exit the Chart Wizard

10. Save your web template using the values below. Then execute the template to view your chart.

    When you are finished, exit the results and return to the Web Application Designer for the next task.

Field	Value
Description	`GR## Charts`
Technical Name	`GR##Charts`

Task 2: Using the Chart Designer

You want to exert finer control over the settings for your charts. To do that, you will need to use the Chart Designer. The particular chart you want to create is a column chart that shows the sales by quarter for a selected year and the amount of change for the same quarter in the next year. Since the amount of change can be positive or negative, you want to start the change amount at the end of the prior year amount and have the bar go up or down as appropriate for the change.

1. Return to the BEx Web Application Designer and create a new data provider (DP_2) for your *GR## Charts* web template. Use the query **T_CONTROL** to supply the data. This query uses a variable to prompt for a *Country* value.

2. Change the web template properties to force the variable screen to display before the web template executes.

3. Add an *Analysis* web item to your web template below the first chart.

4. Create a new data provider (DP_3) for your *GR## Charts* web template. Use the query **T_QTRCOMP** to supply the data. This query provides a quarterly comparison of sales data for 2000 and 2001.

5. Add another *Chart* web item to your web template below the Analysis web item.

6. Set the properties of the Chart as follows:

Property/Parameter	Value
Display > Width in Pixels	`800`
Display > Height in Pixels	`400`
Internal Display > Edit Chart	`Select the button at the end of the row to launch the Chart Wizard`

7. In Step 1 of the Chart Wizard, set the chart type to *Stacked Columns*. Switch from the Chart Wizard to the *Chart Designer* by choosing the *Refine* button.

8. In the *Overview* section of the screen, select the *Legend* segment. Then in the lower part of the screen, set the *Visibility* property to **unchecked**.

9. In the *Overview* section of the screen, expand the *Series* segment. Then select *Series 1* in order to change its properties. In the lower portion of the screen, locate the *Area Properties* section and expand it. Locate the *Transparency* property and set the value to **100**.

10. Finish your work with the Chart Designer by selecting the *Complete* button and return to the Web Application Designer.

11. Save then execute your web template. When prompted for a country value, enter **US** and notice the effect on the second chart for the third quarter.

    When you are finished, exit the results and return to the Web Application Designer for the next task.

12. **Optional:** You can view a more complete example of this exercise using template *T_DELTACHART* if you have time. The link to execute the template is in your user menu in folder *Unit 4: BEx Web Application Designer*. Much of the effort to make this chart display possible is done in the two queries *T_QTRCOMP* and *T_CONTROL*.

Task 3: Geocoding and Map Web Items

You will perform some simple geocoding for a static geo-characteristic, then use that characteristic's data in a web template using the *Map* web item.

1. Start in the *Data Warehousing Workbench* in your BI system and open the InfoObject *Country Key GR## (CTY_GR##)* in change mode. On the *Business Explorer* tab, look at the type of geo-characteristic that has been set for this InfoObject. Remain in the InfoObject definition for the next step.

2. Download the geographic data of the geo-characteristic into your local network drive using the *Geo data download (everything)* button. The local network drive is typically drive N: in a WTS environment. When prompted, select *DAT File (for Double-Byte)*. Leave the InfoObject definition open when finished.

    💡 **Hint:** Ask your instructor which drive is available in your training course environment.

3. To begin the process of maintaining the SAPBWKEY, load the three shape files into the same N: drive where you downloaded the *CTY_GR##.DBF* file. The three shape files are in the Shared Folders area of the SAP office in the BI system. Start a new session in BI. From the SAP menu, choose *Office → Folders → Shared Folders → Training: Material for BW Training → BEXMAP: BExMap Scenarios*. Open the message titled *BExMap Geocoding Files* and export the three shape files successively to your local network drive (N:).

4. Start Microsoft Excel or use the BEx Analyzer. Open the *CTY_GR##.DBF* file with the geographic information of your geo-characteristic (CTY_GR##) and the DBF version of the shape files *(cntry299.dbf)*. Transfer the SAPBWKEY value from the CTY_GR##.DBF file into the SAPBWKEY column in the cntry299.dbf file.

> ⚠ **Caution:** Do not change the order of the rows in the cntry299.dbf
> file!

Save the cntry299.dbf file and close the two files.

5. Return to the BI session with the definition of your InfoObject CTY_GR##.

Load the three shape files (cntry299.shx, cntry299.shp, cntry299.dbf) from
the N: drive into the business document service of SAP BI. To do this, select
Upload Shapefiles. Follow the prompts which will automatically proceed
through each of the files. Exit the InfoObject session when finished.

6. Open the Web Application Designer and create a web template using the
Analysis and *Map* web items. Set the *Map* web item a width and height of
900 pixels.

7. Start the BEx Query Designer from the *Tools* menu of the Web Application
Designer. Create a query using the InfoCube *BExMap-Cube (ZBW207GIS)*.
Place the geo-characteristic *Country Key GR* ## into the Rows and the key
figure *Volumes in CDM* into the Columns. Save the query with a description
of `GR## GIS Query` and a technical name of `GR##GISQ`. Then exit the
BEx Query Designer and return to the Web Application Designer.

8. Create a new data provider and use the query **GR##GISQ** to supply the data.
Assign the data provider to the *Analysis* web item.

9. Set the properties of the Map as follows:

Property/Parameter	Value
Internal Display > List of Maplayers (0) > * Maplayer	`Select the button at the end of the row`.
Maplayer Settings > Data Provider	Select `DP_1`, then select `OK`.

10. Save your web template using the values below. Then execute the template
to view your chart.

When you are finished, exit the results and return to the Web Application
Designer for the next task.

Field	Value
Description	`GR## Map`
Technical Name	`GR##MAP`

Solution 8: Charts and Maps

Task 1: Using the Chart Wizard

You will create some simple charts using the functionality of the Chart Wizard.

1. Start the BEx Web Application Designer and create a new web template.

    a) Start the BEx Web Application Designer using the path *Start →
        Programs → Business Explorer → Web Application Designer*.

    b) Logon to the system with the user ID and password provided by your
        instructor.

    c)   Select *"Create New Web Template"*.

2. To begin, create a new data provider and use the query **T_COQ4** to supply
    the data.

    a) Double click on *New Data Provider* in the design area.

    b) Select the data provider type *Query* using the radio button.

    c) Enter **T_COQ4** as the name of the query and select *OK*.

3. Next, choose the *Chart* web item from the Standard group of web items
    and drag it into your template. This web item allows you to show data in
    the format of a chart.

    a) In the Web Items area of the Web Application Designer, open the
        Standard web item category.

    b) Drag the *Chart* web item to your web template as the first web item.

4. Set the properties of the Chart as follows:

Property/Parameter	Value
Display > Width in Pixels	`500`
Display > Height in Pixels	`400`
Internal Display > Edit Chart	`Select the button at the end of the row to launch the Chart Wizard`

    a) In the *Properties* work area, set the properties for the fields as shown
        in the table above.

> **Hint:** The 'Dot' button at the end of some property fields will
> open a new window where additional selections can be made.
> In this case, it starts the Chart Wizard.

5. In Step 1 of the Chart Wizard, change the chart type to `Columns 2.5D`.
    Click on *Next* when finished.

    a) In the Chart Wizard under the *Select a Chart* window, expand the
        Columns chart group and select `Columns 2.5D`.

    b) Click on *Next* to proceed to the next screen.

6. In Step 2 of the Chart Wizard, make the following settings:

Property/Parameter	Value
Enter Texts > Title	`GR## Revenue and Cost`
Enter Texts > Vertical Axis > Unit	`EUR`
Select a data source > Series Count	`6`
Select a data source > Category Count	`2`

    Click on *Next* when finished.

    a) In the second step of the Chart Wizard, make the settings as shown
        in the table above.

    b) Click on *Next* when finished.

7. In Step 3 of the Chart Wizard, accept all the default settings.

    Click on *Next* to proceed.

    a) In the third step of the Chart Wizard, accept the default settings.

    b) Click on *Next* to proceed.

8. In Step 4 of the Chart Wizard, make the following settings:

Property/Parameter	Value
Background > Color > select the down arrow	Choose the `Light Yellow` color from the samples.
Plot Area > Color > select the down area	Choose a `Darker Yellow` color from the samples.
Title > Font Size	`11`
Title > Font Style	`Bold`

2007/Q2   **231** SAP

Click on *Next* when finished.

a)   In the fourth step of the Chart Wizard, make the settings as shown in the table above.

b)   Click on *Next* when finished

9.   In Step 5 of the Chart Wizard, accept all the default settings.

Click on *Next* to proceed. Then choose *Complete* to save your settings and exit the Chart Wizard

a)   In the fifth step of the Chart Wizard, accept the default settings.

b)   Click on *Next* to proceed.

c)   Click on *Complete* to save and exit the Chart Wizard.

10.   Save your web template using the values below. Then execute the template to view your chart.

When you are finished, exit the results and return to the Web Application Designer for the next task.

Field	Value
Description	`GR## Charts`
Technical Name	`GR##Charts`

a)   In the Web Application Designer, click on the *Save* 💾 icon.

b)   Enter the values shown above for the *Description* and *Technical Name* fields.

c)   Select *OK* when finished.

d)   Select the *Execute* 🖼 icon to execute your web application.

e)   Logon to the NetWeaver portal using your assigned user ID and password.

f)   View your chart results.

g)   Exit the results and return to the Web Application Designer.

Task 2: Using the Chart Designer

You want to exert finer control over the settings for your charts. To do that, you will need to use the Chart Designer. The particular chart you want to create is a column chart that shows the sales by quarter for a selected year and the amount of change for the same quarter in the next year. Since the amount of change can be positive or negative, you want to start the change amount at the end of the prior year amount and have the bar go up or down as appropriate for the change.

1. Return to the BEx Web Application Designer and create a new data provider (DP_2) for your *GR## Charts* web template. Use the query **T_CONTROL** to supply the data. This query uses a variable to prompt for a *Country* value.

    a) Return to the Web Application Designer and double click on *New Data Provider* in the design area.

    b) Select the data provider type *Query* using the radio button.

    c) Enter **T_CONTROL** as the name of the query and select *OK*.

2. Change the web template properties to force the variable screen to display before the web template executes.

    a) In the *Properties* screen area, select the *Web Template* and change the properties as show below.

Property/Parameter	Value
Behavior > Display Variable Screen	**On (checked)**

3. Add an *Analysis* web item to your web template below the first chart.

    a) Select the *Analysis* web item from the Standard group of web items and drag it into your template below the first chart.

4. Create a new data provider (DP_3) for your *GR## Charts* web template. Use the query **T_QTRCOMP** to supply the data. This query provides a quarterly comparison of sales data for 2000 and 2001.

    a) Double click on *New Data Provider* in the design area.

    b) Select the data provider type *Query* using the radio button.

    c) Enter **T_QTRCOMP** as the name of the query and select *OK*.

5. Add another *Chart* web item to your web template below the Analysis web item.

    a) Select the *Chart* web item from the Standard group of web items and drag it into your template.

6. Set the properties of the Chart as follows:

Property/Parameter	Value
Display > Width in Pixels	`800`
Display > Height in Pixels	`400`
Internal Display > Edit Chart	`Select the button at the end of the row to launch the Chart Wizard`

   a) In the *Properties* work area, set the properties for the fields as shown in the table above.

7. In Step 1 of the Chart Wizard, set the chart type to *Stacked Columns*. Switch from the Chart Wizard to the *Chart Designer* by choosing the *Refine* button.

   a) In the Chart Wizard, select *Stacked Columns* as the chart type. Then select the *Refine* button to start the Chart Designer.

8. In the *Overview* section of the screen, select the *Legend* segment. Then in the lower part of the screen, set the *Visibility* property to **unchecked**.

   a) Set the *Visibility* property of the legend to **unchecked** as described in the exercise step.

9. In the *Overview* section of the screen, expand the *Series* segment. Then select *Series 1* in order to change its properties. In the lower portion of the screen, locate the *Area Properties* section and expand it. Locate the *Transparency* property and set the value to **100**.

   a) Set the *Transparency* value of Series 1 to **100** as described in the exercise step.

10. Finish your work with the Chart Designer by selecting the *Complete* button and return to the Web Application Designer.

   a) Select the *Complete* button to return to the Web Application Designer.

11. Save then execute your web template. When prompted for a country value, enter **US** and notice the effect on the second chart for the third quarter.

   When you are finished, exit the results and return to the Web Application Designer for the next task.

   a) In the Web Application Designer, click on the *Save* 💾 icon.

   b) Select the *Execute* 🗗 icon to execute your web application.

   c) Logon to the NetWeaver portal using your assigned user ID and password.

   d) Enter **US** as the country value when prompted. View your chart results.

   e) Exit the results and return to the Web Application Designer.

Continued on next page

12. **Optional:** You can view a more complete example of this exercise using template *T_DELTACHART* if you have time. The link to execute the template is in your user menu in folder *Unit 4: BEx Web Application Designer*. Much of the effort to make this chart display possible is done in the two queries *T_QTRCOMP* and *T_CONTROL*.

   a) Go to your user menu on BI and execute the *T_DELTACHART* web application. Then use the BEx Query Designer to display the definition of the two queries used as data providers.

Task 3: Geocoding and Map Web Items

You will perform some simple geocoding for a static geo-characteristic, then use that characteristic's data in a web template using the *Map* web item.

1. Start in the *Data Warehousing Workbench* in your BI system and open the InfoObject *Country Key GR## (CTY_GR##)* in change mode. On the *Business Explorer* tab, look at the type of geo-characteristic that has been set for this InfoObject. Remain in the InfoObject definition for the next step.

   a) In your BI system, use this menu path to locate your InfoObject: *SAP Easy Access Menu → SAP menu → Modeling → Data Warehousing Workbench: Modeling → InfoObjects*. Select the InfoArea*BW Training → BW Customer Training → BExMAP Scenarios → GIS → Country Key GR##*.

   b) Open the InfoObject in change mode by right clicking on the InfoObject and choosing *Change*.

   > **Hint:** Select the *Navigator on/off* 🖳 icon to display your InfoObject definition full screen.

   c) Within the *Business Explorer* tab page, scroll down to the *BExMap* section and observe that the Geographical Type is set as *Static Geo-characteristic*.

   d) Remain on that screen.

2.   Download the geographic data of the geo-characteristic into your local
     network drive using the *Geo data download (everything)* button. The local
     network drive is typically drive N: in a WTS environment. When prompted,
     select *DAT File (for Double-Byte)*. Leave the InfoObject definition open
     when finished.

> **Hint:** Ask your instructor which drive is available in your training
> course environment.

a)   Select *Geo data download (everything)*. When prompted, select the
     radio button for *DAT File (for Double-Byte)*.

b)   On the next prompt *(Transfer Master Data for Geo-coding to a Local
     File)*, choose the 'pull down' button at the end of the *File Name* field
     to bring up the *Save As* dialog window. From the dropdown box at
     the top of the window, choose your *N: drive*. In the *File name:* field
     at the bottom of the screen, enter **CTY_GR##.DBF** and then choose
     the *Save* button.

c)   Back on the *(Transfer Master Data for Geo-coding to a Local File)*
     screen, select the *Transfer* button.

3.   To begin the process of maintaining the SAPBWKEY, load the three shape
     files into the same N: drive where you downloaded the *CTY_GR##.DBF* file.
     The three shape files are in the Shared Folders area of the SAP office in the
     BI system. Start a new session in BI. From the SAP menu, choose *Office
     → Folders → Shared Folders → Training: Material for BW Training →
     BEXMAP: BExMap Scenarios*. Open the message titled *BExMap Geocoding
     Files* and export the three shape files successively to your local network
     drive (N:).

a)   Start a new session in your BI system.

b)   From the SAP menu, choose *Office → Folders → Shared Folders →
     Training: Material for BW Training → BEXMAP: BExMap Scenarios*.

c)   Double click on the message *BExMap Geocoding Files* to open it.

d)   Select the *Attachments* tab.

e)   Right click on the *cntry299.dbf* file and choose *Export attachment*.

f)   On the *Export File* screen, choose your N: drive from the dropdown
     box at the top of the screen.

g)   Choose the *Save* button.

h)   Repeat the export process for the other two files.

i)   Return to the SAP Easy Access menu by selecting the green*Back arrow*
     twice.

Continued on next page

4. Start Microsoft Excel or use the BEx Analyzer. Open the *CTY_GR##.DBF* file with the geographic information of your geo-characteristic (CTY_GR##) and the DBF version of the shape files *(cntry299.dbf)*. Transfer the SAPBWKEY value from the CTY_GR##.DBF file into the SAPBWKEY column in the cntry299.dbf file.

> ⚠️ **Caution:** Do not change the order of the rows in the cntry299.dbf file!

Save the cntry299.dbf file and close the two files.

a) Choose *Start → Programs → Business Explorer → Analyzer*.

b) Open the two files using the Excel menu *File → Open*. Select your N: drive and specify **All files** in the *Files of type:* field. and *File → Open → SAPWorkDir → cntry299.dbf*.

c) Open both the *CTY_GR##.DBF* file and the *cntry299.dbf* file.

d) Enter the *SAPBWKEY* value found in the *CTY_GR##.dbf* file into the *SAPBWKEY* column of the *cntry299.dbf* file next to the appropriate country. Do not change the order of rows in the *cntry299.dbf* file! Save and close the two files.

e) Return to the SAP Easy Access menu.

5. Return to the BI session with the definition of your InfoObject CTY_GR##.

Load the three shape files (cntry299.shx, cntry299.shp, cntry299.dbf) from the N: drive into the business document service of SAP BI. To do this, select *Upload Shapefiles*. Follow the prompts which will automatically proceed through each of the files. Exit the InfoObject session when finished.

a) Return to the InfoObject maintenance session for CTY_GR##.

b) On the *Business Explorer* tab page, select *Upload Shapefiles*. Load the files in succession by selecting the requested file from the N: drive and then choosing *Open*. Confirm the entry on the *Business Document Service* popup screen.

c) Click on the green *Back arrow* twice to leave the Data Warehousing Workbench.

6. Open the Web Application Designer and create a web template using the *Analysis* and *Map* web items. Set the *Map* web item a width and height of 900 pixels.

    a) Choose *Start → Programs → Business Explorer → Web Application Designer*. Create a new web template and drag both the *Analysis* web item and the *Map* web item into the web template.

    b) In the *Properties* window, select the *Map* web item. Set both the width and height of the map to 900 pixels.

7. Start the BEx Query Designer from the *Tools* menu of the Web Application Designer. Create a query using the InfoCube *BExMap-Cube (ZBW207GIS)*. Place the geo-characteristic *Country Key GR* ## into the Rows and the key figure *Volumes in CDM* into the Columns. Save the query with a description of **GR## GIS Query** and a technical name of **GR##GISQ**. Then exit the BEx Query Designer and return to the Web Application Designer.

    a) To create a new query, choose *Tools → BEx Query Designer*.

    b) Select the *New* icon to create a new query. In the next dialog box, select the InfoCube BExMap-Cube under *InfoArea → BW Training → BW Customer Training → BExMap Scenarios*.

    c) Drag the characteristic *Country key GR##* to the *Rows* and the key figure *Volumes in CDM* to the *Columns*.

    d) Save your query with a description of **GR## GIS Query** and a technical name of **GR##GISQ**.

    e) Exit the BEx Query Designer and return to the Web Application Designer.

8. Create a new data provider and use the query **GR##GISQ** to supply the data. Assign the data provider to the *Analysis* web item.

    a) Double click on *New Data Provider* in the design area.

    b) Select the data provider type *Query* using the radio button.

    c) Enter **GR##GISQ** as the name of the query and select *OK*.

    d) On the *General* tab in the *Properties* screen area, assign the new data provider to the *Analysis* web item.

9. Set the properties of the Map as follows:

238   2007/Q2

Property/Parameter	Value
Internal Display > List of Maplayers (0) > * Maplayer	**Select the button at the end of the row.**
Maplayer Settings > Data Provider	Select **DP_1**, then select **OK**.

a)   In the *Properties* work area, set the properties for the fields as shown in the table above.

10.   Save your web template using the values below. Then execute the template to view your chart.

When you are finished, exit the results and return to the Web Application Designer for the next task.

Field	Value
Description	**GR## Map**
Technical Name	**GR##MAP**

a)   In the Web Application Designer, click on the *Save* 🖫 icon.

b)   Enter the values shown above for the *Description* and *Technical Name* fields.

c)   Select *OK* when finished.

d)   Select the *Execute* 🖳 icon to execute your web application.

e)   Logon to the NetWeaver portal using your assigned user ID and password.

f)   View your map results. Notice the different color shading applied to the countries based on the value of the mapped key figure.

g)   Exit the results and return to the Web Application Designer.

Lesson Summary

You should now be able to:
- Present data in chart format in your web applications
- Describe the process of geocoding for characteristics
- Present data in map format in your web applications

Lesson: Command Wizard

Lesson Overview

This lesson introduces the capabilities of the Command Wizard and how commands are implemented in a variety of web items.

Lesson Objectives

After completing this lesson, you will be able to:

- Explain the basic operation of the Command Wizard
- Insert commands into the web application using various web items
- Use sequences of commands to accomplish multiple tasks

Business Example

You want to add specific and sometimes complex operations to your web application, but you want to avoid using HTML as this is not widely understood by the development team. You are hoping that commands will meet your requirements.

Web API and the Command Wizard

While you are able to create many functional web applications using just the basic parameters of various web items, many of your requirements go beyond the basics. To answer these requirements, you need to extend the navigational and processing power of the web items. To do this, the Web Application Designer offers the Web API and the Command Wizard.

The Web Design API **provides the following functions:**

- **Creating commands for data providers, planning applications, web items and web templates. Any context menu navigation can be replaced with a command.**
- **Parameterization of web items.**
- **The main tool for generating the commands is the** Command Wizard **of the BEx Web Application Designer. The Command Wizard enables you to easily create commands using a step-by-step procedure.**

Figure 95: Web Design API

While web items in a web application provide many opportunities for the effective presentation of information, there is far more that can be done. The Web Design API is provided to enhance web templates, web items, data providers and planning applications with commands. These commands extend the interaction and capabilities of objects to enhance the interaction and integration of web-based analysis objects.

With NetWeaver 2004s, commands are generated with the Command Wizard. This Web Design API tool is accessible from anywhere that a command would be relevant and guides the user through the necessary parameters without the need to master complex HTML syntax. Any navigation that can be done with the context menu in a web application can be done as well with a command. This makes if practical to develop web applications that guide the user with on-screen navigation opportunities.

■ **Command Wizard:**
  ◆ **Leveraging the power of the WEB API in an easy and intuitive fashion**
  ◆ **Reduces the need for customer JavaScripting**
  ◆ **Invoked from relevant web items and links in the template**
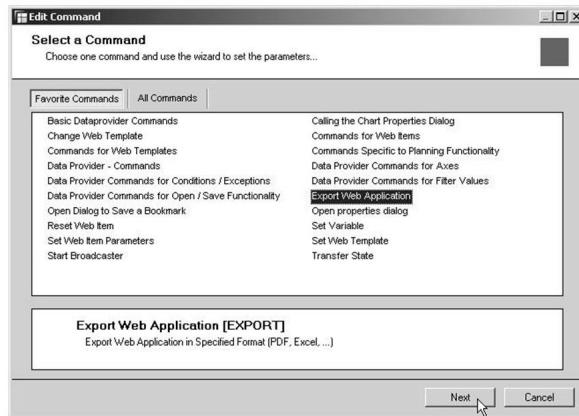
Figure 96: Command Wizard - Initial View

The Command Wizard is typically called from the Properties screen area of the Web Application Designer. On its initial view two tabs are visible. The first is the Favorites tab where a web template designer can place any frequently-used commands for easy recall. When a command is selected, a brief help text is displayed on the bottom portion of the screen.

Commands can often reduce the need for creating JavaScripts to perform complex sequences of operations within a web application.

■ **Selecting commands with the checkbox places that command on the 'Favorite Commands' tab for easy access later.**
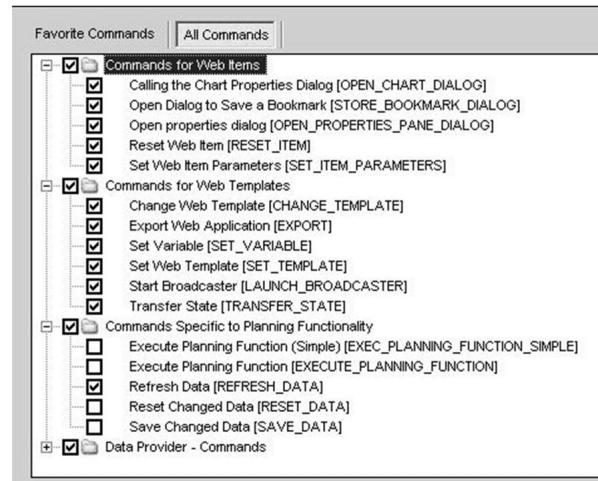
Figure 97: Command Wizard - All Commands Tab

The All Commands tab provides full access to all the available commands. Commands are grouped by their relevance into four major categories:

• Data Provider Commands

• Commands for Web Templates

• Commands for Web Items

• Commands Specific to Planning Functionality

Commands for Data Providers comprises the largest collection of commands with subcategories forming the list

The checkbox next to each command is used to place that command on the *Favorite Commands* tab. To use a command, select it and click on the *Next* button at the bottom of the screen.

- ■ **Once a command is chosen, specify the parameters.**
- ■ **Commands can be grouped using the 'Next Command' button.**
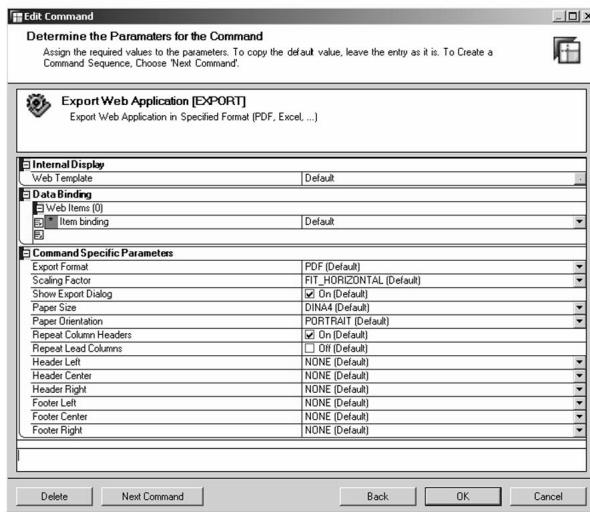
Figure 98: Command Wizard - Command Parameters

Once a command is selected, the parameters for that command are displayed for the user. Parameter choices are made either by direct entry or selected from dropdown lists. Many command parameters have additional subscreens for entering additional parameters.

Command sequences are easily constructed by using the *Next Command* button at the bottom of the screen. For example, you could assign a command sequence to a button that first changes the data provider, then sets specific filter values for a series of characteristics and finally sets various display properties of the *Analysis* item used to show the results.

- ■ **Adds a group of buttons where each one can execute one or more commands (e.g. filter by region)**
- ■ **Key properties:**
  Internal Display → List of Buttons
  - Button on each row calls the parameters screen.
  - Here, under Button → Action you can invoke the Command Wizard to specify the button's action.

Figure 99: Web Item - Button Group

One of the most commonly used items for invoking commands is the **Button Group** web item. Buttons in a web application are a good way to offer the user specific pre-configured navigational options. The web item is rendered as one or multiple buttons, depending on the chosen parameters.

In the properties of the Button Group, the most important property is the *List of Buttons* in the *Internal Display* parameters group. Here the designer of the web template defines each button and what its function will be. A command or command sequence is linked to each button to determine its action.

You can insert any text (language-dependent and language-independent) for the button labels, assign a quick link to them, and specify their design.

- **Adds a dropdown box to the web template**
- **Key properties:**
  Data Binding → Data Binding Type
  - Char/Structure Member, Query View Selection, Variable Selection or Fixed List of Options
  Data Binding → (Type dependent)
  - Depending on the Data Binding Type chosen, specify the necessary properties. All but Variable Selection offer opportunities for commands.

Dropdown box using two commands. Each command changes the query assigned to the data provider for the analysis item.

		Sales Volume EUR ⇕	Cost of Sales ⇕	Margin % ⇕
		* 1.000 EUR	* 1.000 EUR	%
DE	Germany	359.901	195.899	46
FR	France	1.067	801	25
GB	Great Britain	12.911	3.106	76
IT	Italy	11.262	3.272	71
NO	Norway	18.338	4.398	76
US	United States	41.358	24.977	40
Overall Result		444.836	232.454	48

Figure 100: Web Item - Dropdown Box

The **Dropdown Box** web item is used where the designer wishes to give the web application user a list of items from which to choose. The list can be one of the following:

- CharacteristicValues for Filtering

  Using the *Dropdown Box* web item, you can easily filter one or more connected data providers by a characteristic value. When you select an entry from the dropdown box, the connected data provider is filtered according to this value. If a different method is used to select a filter value for the characteristic in the dropdown box, the current filter value appears in the dropdown box.

- Query View Selection

  By selecting an entry from the dropdown box, you can switch, in one step, from a data provider and its display (in a table, for example) to another data provider that may be displayed in a different way (in a chart, for example).

- Fixed List of Options

  When you select an entry from the dropdown box, an associated command from the Web Design API is executed.

- Fixed List of Options: Manual Update

  When you select an entry from the dropdown box, an associated command from the Web Design API is executed. It is also possible to trigger a manual update.

- Variable Selection

  When you select an entry from the dropdown box, the data displayed depends on the variable selection.

On the Properties screen area, the key properties are found in the *Data Binding* parameter group. The *Data Binding Type* is used to make the choice from the five types of lists described above. Depending on the choice for that property, the actual *Data Binding* property is set. All Data Binding Types except *Variable Selection* offer access to the Command Wizard.

As an example, you could create a dropdown list of characteristics to add to the drilldown of an analysis. Depending on the characteristic chosen, the display properties of the analysis could be changed appropriately. Another common example is to develop several different views for a data provider and then create a dropdown list of the views to allow easy navigation to pre-defined navigational states.
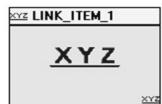
■ **Adds active links to the web template**

■ **Key properties:**

Behavior → Action

◆ **Select to have the link execute a command or a JavaScript**

Behavior → Action → Command

◆ **Invoke the Command Wizard**

Example of links to filter the chart and to call the chart properties dialog
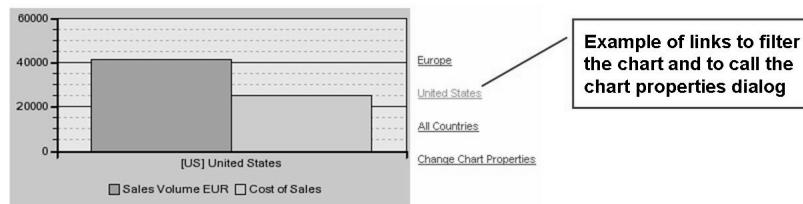
Figure 101: Web Item - Link

The **Link** web item is used to place an active link in the web template that calls a command, command sequence or JavaScript. This provides functionality much like the Button Group, but with a different, more compact visual representation.

The main properties of this web item are found in the *Behavior* parameter group. There the choice of command or JavaScript is made with the *Action* parameter. If Command is chosen here, then the *Action Command* parameter provides access to the Command Wizard where the action is defined.

You can add any text (language dependent and language independent) to the link and assign a quick link to it.

■ **Adds cascading menus to the web template**

■ **Key properties:**

Internal Display → Menu Bar → Menu

◆ **Use 'Grouping of Menu Entries' to add another layer of menu choices**

◆ **Use 'Trigger an Action' to invoke another parameter screen where the Command Wizard is called to define the action.**

Grouping of Menu Entries (Queries)
▶ Grouping of Menu Entries (Sales Queries)
   ▶ Trigger an Action (Sales Org. Analysis)
      ▶ Command (Data Provider Properties)

Figure 102: Web Item - Menu Bar

With the **Menu Bar** web item the designer can construct multilevel cascading menus for the web template. This is a good choice for the initial screen of a complex web application where you want to offer many choices in a logical hierarchy without consuming too much screen space.

Menu entries are of two types:

•   Grouping of Menu Entries

•   Trigger an Action

The *Grouping of Menu Entries* enables the creation of multiple menu levels by specifying which choices are found below a specific menu item. This nesting of menu entries creates the cascading effect of the menu. Nesting is restricted to a maximum of three levels.

Trigger an Action is typically found at the end of a menu expansion and is used to invoke a command. In the menu shown in the graphic above, the menu entry *Sales Org. Analysis* is given type *Trigger an Action*. This enables the button for calling the Command Wizard. Here, the command is the *Data Provider Properties* command that lets the developer specify which data provider will be called for a specific web item like a Chart or Analysis item.
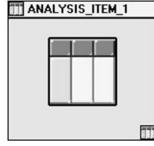
- **Adds analysis data in a tabular format to the web template**
- **Key properties:**
  Behavior → Row/Column Selection
    - Enables command usage when a row or column is selected or deselected.

Figure 103: Web Item - Analysis Item

The **Analysis** web item is a common way to display data provider results in a web template. However, it can also be used to execute commands when the *Behavior → Row/Column Selection* property is invoked. This is desirable when the designer of the web template wants to provide an easy and intuitive way to filter on one or multiple values from an analysis.

In the example above, the *Row Selection* property has been set for the Analysis web item. Additionally, a command sequence is invoked when any of the rows are selected. This sequence makes a hidden Analysis web item on the right visible and then passes a characteristic value of the selected row as a filter value to the second Analysis web item.

■ **Adds a visual backdrop for other web items with caption, toolbar and content areas.**

■ **Key properties:**

Internal Display → Subordinate Web Item
- **Assign a web item to the content area.**
Internal Display → Caption / Toolbar
- **Choose to display a caption and/or toolbar.**
- **For the toolbar, a Button Group is normally used.**

				Sales Volume EUR ⇕	Cost of Sales ⇕	Margin % ⇕
Sold-to Party Country⇕	Sales Organization ⇕			* 1.000 EUR	* 1.000 EUR	%
DE	Germany	0020	Balkans	85	84	1
		1000	Germany Frankfurt	354.145	188.278	47
		R100	Retail Germany	5.436	5.448	-0
		2000	UK Heathrow/Hayes	0	0	DIV/0
		3020	USA Denver	0	0	DIV/0
		3000	USA Philadelphia	235	2.089	-788
		Result		359.901	195.899	46

Caption → Sales Analysis
Toolbar → Back to Start │ Sales Org. Detail │ Dist. Chan. Detail │ Print
Content →

Figure 104: Web Item - Group

The **Group** web item is useful to provide a consistent, compartmentalized look to a web application. An analysis cockpit with multiple Group web items, each with its own toolbar for navigation options can be quickly assembled.

The Group web item consists of three components:

• Caption

• Toolbar

• Content Area

The *Caption* and *Toolbar* areas are optional and can easily be selected in the Properties of the Group web item. The toolbar area is assigned another web item, typically the Button Group, to define the toolbar. As described earlier in the discussion of the Button Group, commands are used to define the action of each button. However, other web items (or multiple web items if enclosed in a Container, for example) can be assigned to the Toolbar area. A Filter, Checkbox or Radio Button Group web item could easily provide toolbar functionality.

The *Content* area holds the main display of information and is usually a Report, Chart or Analysis web item.

Exercise 9: Command Wizard

Exercise Objectives

After completing this exercise, you will be able to:

- Use the Command Wizard from a wide range of web items.

Business Example

Production web applications often make use of commands to carry out sophisticated operations. It is important that you learn how to make use of this important functionality in the Web Application Designer.

Task 1: Using Button Groups and Dropdown Boxes

To increase the sophistication of your web applications, you will enhance them with commands called with buttons and selectable query views from a dropdown box.

1. Start the BEx Web Application Designer and create a new web template.

2. To begin, create a new data provider and use the query **T_COQ4** to supply the data.

3. For alignment purposes, insert an HTML table into your template with two rows and two columns. In each of the rows, set the vertical alignment to `Top`.

4. Choose the *Dropdown box* web item from the Standard group of web items and drag it into web template inserting it in the left cell of the first row of the HTML table.

5. Add a *Button Group* web item into the right cell of the first row of the HTML table.

6. Add a *Navigation Pane* web item into the left cell of the second row of the HTML table.

7. Finally, add an *Analysis* web item into the right cell of the second row of the HTML table.

8. Now, configure the *Dropdown Box*. You want the dropdown list to show all the query views that are available for the selected data provider. In the *Properties* work area, choose the *Dropdown Box* and make the following settings:

Property/Parameter	Value
Data Binding > Data Binding Type	`Query View Selection`
Data Binding > Selection of Query Views	`Select the button at the end of the row`
Selection of Query Views > Data Provider	`Select DP_1`
Selection of Query Views > Specific List of Query Views	`Off (unchecked)`
Selection of Query Views > Target Web Item	`ANALYSIS_ITEM_1`

9. Now, configure the *Button Group*. You want to have a button for saving new views and a button to print the current analysis to a PDF file. In the *Properties* work area, choose the *BUTTON_GROUP_ITEM_1* web item and make the following settings:

Property/Parameter	Value
Internal Display > List of Buttons (1) > 1 ?? (Button)	`Select the button at the end of the row`
Button > Caption	`Save View`
Button > Action > Command	`Select the button at the end of the row`
Edit Command window	Select the `All Commands` tab.
Commands for Data Provider > Data Provider Commands for Open/Save Functions	Expand the list and choose `Call Save Dialog [SAVE_AS]`. Then choose `Next`.
Command Target > Data Provider Affected	Select `DP_1`, then choose `Next`.
Command Specific Parameters > Layout of the Open-Save Dialog	`Display only Dataprovider Persistency Options [DATAPROVIDER]`
Finish the properties for this button	Select `OK` twice.
Internal Display > List of Buttons (1) > * Button	`Select the button at the end of the row`
Button > Caption	`Print to PDF`
Button > Action > Command	`Select the button at the end of the row`

Continued on next page

Edit Command window	Select the **All Commands** tab.
Commands for Web Templates	Expand the list and choose **Export Web Application [EXPORT]**. Then choose **Next**.
Data Binding > Web Items (0) > Item Binding	Select **ANALYSIS_ITEM_1**
Command Specific Parameters > Show Export Dialog	**Off (unchecked)**
Finish the properties of this button	Select **OK** twice.

10. Save your web template using the values below.

Field	Value
Description	**GR## Commands 1**
Technical Name	**GR##COMMANDS1**

Then execute the template. From the dropdown box select from the views that are available to change the analysis results. Using the navigation pane, create a new view by adding or rearranging characteristics.

Then save the view using the *Save View* button. Choose your own description and technical name. Then refresh the browser window and locate your new view in the dropdown list.

Finally, view the results in PDF format using the *Print to PDF* button.

Task 2: Using Groups and Table Selection Commands

You will create a web template that allows users to select a row of data and automatically show a more detailed analysis for the selected value. You will also create a button to reset the results to the original view.

1. Return to the BEx Web Application Designer and create a new web template.

2. To begin, create a new data provider and use the query **T_CTYSALES** to supply the data.

3. Create a second data provider and use the query **T_REGSALES** to supply the data.

4. Choose the *Group* web item from the Advanced group of web items and drag it into web template. This web item provides for a Caption area, a Toolbar area and a Content area.

5. Choose the *Button Group* web item from the Standard group of web items and drag it into web template inside the *Group* web item.

6. Choose the *Container* web item from the Advanced group of web items and drag it into web template to the right of the *Button Group* web item.

7. For alignment purposes, insert an HTML table inside the *Container* web item with one row and two columns. In the row, set the vertical alignment to **Top**.

8. Add an *Analysis* web item to the left cell of the HTML table. Then add an *Analysis* web item to the right cell of the HTML table.

9. Now, configure the *Group* web item. In the *Properties* work area, choose the *Group* and make the following settings:

Property/Parameter	Value
Display > Width in Pixels	**950**
Display > Height in Pixels	**400**
Internal Display > Subordinate Web Item	Select **CONTAINER_ITEM_1**.
Internal Display > With Caption > Caption Type	Select **Text [CAPTION]**.
Internal Display > With Caption > Caption	**Country/Region Analysis**
Internal Display > With Toolbar	**On (checked)**
Internal Display > With Toolbar > Subordinate Web Item	Select **BUTTON_GROUP_ITEM_1**

10. Next, configure the *Analysis* web item in the right cell of the HTML table. You will configure it to be hidden when the web application starts. It should be bound to data provider *DP_2*.

   In the *Properties* work area, choose the *ANALYSIS_ITEM_2* item and make the following settings:

Property/Parameter	Value
Display > Visibility	**Hidden**
Data Binding > Data Provider	**DP_2**

11. Next, configure the *Analysis* web item in the left cell of the HTML table. You will configure it to first allow single rows of data to be selected and then pass that *Country* value on to the second Analysis item as a filter value using a command. As the next command, the second Analysis area should then be made visible.

Continued on next page

The Analysis item in the left cell of the HTML table should be bound to data provider *DP_1*.

In the *Properties* work area, choose the *ANALYSIS_ITEM_1* item and make the following settings:

Property/Parameter	Value
Data Binding > Data Provider	Select **DP_1**.
Behavior > Row Selection	Select **Single with Commands**.
Behavior > Row Selection > Activation Action > Command	**Select the button at the end of the row**.
Edit Command Screen	Choose **All Commands** tab.
Commands for Data Provider > Data Provider Commands for Filter Values	Choose **Set Filter Values by Different Sources [SET_SELECTION_STATE_BY_BINDING]**. Then select **Next**.
Command Target > Target Data Providers (0) > Data Provider Affected	Select **DP_2**, then **Next**.
Data Binding > Selection Data Bindings (0) > * Selection Binding	**Select the button at the end of the row.**
Selection Binding > Characteristic	**Select the button at the end of the row.**
Valuehelp Selector Screen	Select **Sold-to Party Country (0SOLD_TO__0COUNTRY)** from the list and the bottom of the screen. Then select **OK**.
Selection Binding > Binding Type	Select **Web Item Selection**.
Selection Binding > Item Selection > Item binding	Select **ANALYSIS_ITEM_1**.
Selection Binding > Item Selection > Characteristic	Select **Sold-to Party Country (0SOLD_TO__0COUNTRY)**. Then choose **OK**.
Edit Command screen	Select the **Next Command** button.
Command List Editing screen	Select the **Insert** button.
Edit Command screen	Choose **All Commands** tab.

Continued on next page

 SAP

Commands for Web Items	Choose `Set Web Item Parameters [SET_ITEM_PARAMETERS]`. Select `Next`.
Command Target > Target Web Item	Select `ANALYSIS_ITEM_2`, then `Next`.
Display > Visibility	Select `Visible`.
Scroll down to Data Binding > Data Provider	Select `DP_2`.
Finish Command Sequence	Select `OK` twice.

12. Finally, configure the *Button Group*. You want to have a button for removing the detailed analysis after you have selected a particular country. In the *Properties* work area, choose the *BUTTON_GROUP_ITEM_1* web item and make the following settings:

Property/Parameter	Value
Internal Display > List of Buttons (1) > 1 ?? (Button)	`Select the button at the end of the row`
Button > Caption	`Remove Details`
Button > Action (Command Triggered) > Command	`Select the button at the end of the row`
Edit Command window	Select the `All Commands` tab.
Commands for Web Items **(This command will hide ANALYSIS_ITEM_2)**	Choose `Set Web Item Parameters [SET_ITEM_PARAMETERS]`. Select `Next`.
Command Target > Target Web Item	Select `ANALYSIS_ITEM_2`, then `Next`.
Display > Visibility	Select `Hidden`. (Do not use the defaulted value.)
Scroll down to Data Binding > Data Provider	Select `DP_2`. Then select the `Next Command` button.
Start the next command	Select the `Insert` button.
Edit Command window	Select the `All Commands` tab.

Continued on next page

Commands for Web Items **(This command will reset the row selection property of ANALYSIS_ITEM_1.)**	Choose `Set Web Item Parameters [SET_ITEM_PARAMETERS]`. Select **Next**.
Command Target > Target Web Item	Select **ANALYSIS_ITEM_1**, then **Next**.
Behavior > Row Selection	Select **None**.
Scroll down to Data Binding > Data Provider	Select **DP_1**. Then select the **Next Command** button.
Edit Command screen	Select the **All Commands** tab.
Commands for Web Items **(This command will turn on row selection for ANALYSIS_ITEM_1. It will have two other commands embedded in it.)**	Choose `Set Web Item Parameters [SET_ITEM_PARAMETERS]`. Select **Next**.
Command Target > Target Web Item	Select **ANALYSIS_ITEM_1**, then **Next**.
Behavior > Row Selection	Select `Single with Commands`. (Do not use the default value.)
Behavior > Row Selection > Activation Action > Command	`Select the button at the end of the row`.
Edit Command screen	Choose `All Commands tab`
Commands for Data Provider > Data Provider Commands for Filter Values	Choose `Set Filter Values by Different Sources [SET_SELECTION_STATE_BY_BINDING]`. Select **Next**.
Command Target > Target Data Providers (0) > Data Provider Affected	Select **DP_2**, then **Next**.
Data Binding > Selection Data Bindings (0) > * Selection Binding	`Select the button at the end of the row`.
Selection Binding > Characteristic	`Select the button at the end of the row`.

Valuehelp Selector screen	Select **Sold-to Party Country (0SOLD_TO__0COUNTRY)** from the list and the bottom of the screen. Then select **OK**.
Selection Binding > Binding Type	Select **Web Item Selection**.
Selection Binding > Item Selection > Item binding	Select **ANALYSIS_ITEM_1**.
Selection Binding > Item Selection > Characteristic	Select **Sold-to Party Country (0SOLD_TO__0COUNTRY)**. Then choose **OK**.
Edit Command screen	Select the **Next Command** button.
Command List Editing screen	Select the **Insert** button.
Edit Command screen	Choose **All Commands** tab.
Commands for Web Items	Choose **Set Web Item Parameters [SET_ITEM_PARAMETERS]**. Select **Next**.
Command Target > Target Web Item	Select **ANALYSIS_ITEM_2**, then **Next**.
Display > Visibility	Select **Visible**.
Scroll down to Data Binding > Data Provider	Select **DP_2**.
Finish Command Sequence	Select **OK** five times.

13. Save your web template using the values below.

Field	Value
Description	**GR## Commands 2**
Technical Name	**GR##COMMANDS2**

Then execute the template. From the initial view of the web application, choose a country using the selection buttons at the beginning of each row. Notice that the additional details for that country are shown in the area to the right. Choosing the *Remove Details* button will return the application to its starting point. You also can choose the selection button for the *Results* row to see details for all countries.

Continued on next page

Task 3: Using Commands in Links

You will build a simple web application and use a series of links to control a chart presentation.

1. Return to the BEx Web Application Designer and create a new web template.

2. To begin, create a new data provider and use the query **T_CTYSALES** to supply the data.

3. For alignment purposes, insert an HTML table inside the template with one row and two columns. Set the table width to **500 pixels**. In the row, set the vertical alignment to **Top**.

4. Choose the *Chart* web item from the Standard group of web items and drag it into web template inserting it in the left cell of the HTML table.

5. Add 4 *Link* web items into the right cell of the first row of the HTML table. Insert a blank line after each one in order to have the link items stacked vertically in the right cell of the HTML table.

6. Configure the *Chart* web item. In the *Properties* work area, choose the *CHART_ITEM_1* item and make the following settings:

Property/Parameter	Value
Width in Pixels	**400**

7. Now, configure the first *Link* web item to filter the chart to **Germany**. In the *Properties* work area, choose *LINK_ITEM_1* and make the following settings:

Property/Parameter	Value
Internal Display > Text	**Germany**
Behavior > Action (Command Triggered) > Command	**Select the button at the end of the row**.
Edit Command screen	Select the **All Commands** tab.
Commands for Data Provider > Data Provider Commands for Filter Values	Select the **Set Filter Value for a Characteristic** command, then choose **Next**.
Command Target > Target Data Providers (0) > Data Provider Affected	Select **DP_1**, then choose **Next**.
Data Binding > Characteristic	**Select the button at the end of the row**.

Continued on next page

Valuehelp Selector screen	Choose **Sold-to Party Country** from the values at the bottom of the screen.
Command Specific Parameters > Operator > Equals > Member Name	**Select the button at the end of the row**.
Selection screen	Choose **Germany**, then choose **OK**.
Finish the command settings	Choose **OK**.

8. Configure the second*Link* web item to filter the chart to exclude **Germany**. In the *Properties* work area, choose *LINK_ITEM_2* and make the following settings:

Property/Parameter	Value
Internal Display > Text	**Exclude Germany**
Behavior > Action (Command Triggered) > Command	**Select the button at the end of the row**.
Edit Command screen	Select the **All Commands** tab.
Commands for Data Provider > Data Provider Commands for Filter Values	Select the **Set Filter Value for a Characteristic** command, then choose **Next**.
Command Target > Target Data Providers (0) > Data Provider Affected	Select **DP_1**, then choose **Next**.
Data Binding > Characteristic	**Select the button at the end of the row**.
Valuehelp Selector screen	Choose **Sold-to Party Country** from the values at the bottom of the screen.
Command Specific Parameters > Sign	Select **Excluding**
Command Specific Parameters > Operator > Equals > Member Name	**Select the button at the end of the row**.
Selection screen	Choose **Germany**, then choose **OK**.
Finish the command settings	Choose **OK**.

9. Configure the third *Link* web item to clear any filter values for *Country*. In the *Properties* work area, choose *LINK_ITEM_3* and make the following settings:

Property/Parameter	Value
Internal Display > Text	`All Countries`
Behavior > Action (Command Triggered) > Command	`Select the button at the end of the row`.
Edit Command screen	Select the `All Commands` tab.
Commands for Data Provider > Data Provider Commands for Filter Values	Select the `Remove Filter Values for a Characteristic` command, then choose `Next`.
Command Target > Target Data Providers (0) > Data Provider Affected	Select `DP_1`, then choose `Next`.
Data Binding > Characteristic	Choose `Sold-to Party Country`, then select `OK`.
Finish the command settings	Choose `OK`.

10. Finally, configure the fourth *Link* web item to call up the *Chart Properties Dialog*. In the *Properties* work area, choose *LINK_ITEM_4* and make the following settings:

Property/Parameter	Value
Internal Display > Text	`Chart Properties`
Behavior > Action (Command Triggered) > Command	`Select the button at the end of the row`.
Edit Command screen	Select the `All Commands` tab.
Commands for Web Items	Select the `Call Chart Properties Dialog` command, then choose `Next`.
Command Target > Target Web Item	Select `CHART_ITEM_1`, then choose `Next`.
Finish the command settings	Choose `OK`.

11. Save your web template using the values below.

Field	Value
Description	`GR## Commands 3`
Technical Name	`GR##COMMANDS3`

Then execute the template. Use the links to change the results in the chart.

Solution 9: Command Wizard

Task 1: Using Button Groups and Dropdown Boxes

To increase the sophistication of your web applications, you will enhance them with commands called with buttons and selectable query views from a dropdown box.

1. Start the BEx Web Application Designer and create a new web template.

    a) Start the BEx Web Application Designer using the path *Start →
        Programs → Business Explorer → Web Application Designer*.

    b) Logon to the system with the user ID and password provided by your
        instructor.

    c)   Select *"Create New Web Template"*.

2. To begin, create a new data provider and use the query **T_COQ4** to supply
    the data.

    a) Double click on *New Data Provider* in the design area.

    b) Select the data provider type *Query* using the radio button.

    c) Enter **T_COQ4** as the name of the query and select *OK*.

3. For alignment purposes, insert an HTML table into your template with two
    rows and two columns. In each of the rows, set the vertical alignment to **Top**.

    a) Click on the *Insert Table* icon or from the context menu in your
        template choose *Insert... → Table*.

    b) In the *Edit HTML Element* screen, change the *Row* value to **2** and the
        Column value to **2**. Choose *OK* to close the window.

    c) Right-click inside a cell of the first row of the HTML table to access
        the context menu for the table. Then using the path *Table → Edit →
        Edit Row <tr>* set the *Vertical* field value to **Top**.

    d) Choose *OK*.

    e) Repeat the process for the second row of the HTML table.

4. Choose the *Dropdown box* web item from the Standard group of web items
    and drag it into web template inserting it in the left cell of the first row of
    the HTML table.

    a) In the Web Items area of the Web Application Designer, open the
        Standard web item category.

    b) Drag the *Dropdown box* web item into your web template and drop it in
        the left cell of the first row of the HTML table.

5. Add a *Button Group* web item into the right cell of the first row of the HTML table.

    a) In the Web Items area of the Web Application Designer, open the *Advanced* web item category.

    b) Drag a *Button Group* web item to your web template and drop it inside the right cell of the first row of the HTML table.

6. Add a *Navigation Pane* web item into the left cell of the second row of the HTML table.

    a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

    b) Drag a *Navigation Pane* web item to your web template and drop it inside the left cell of the second row of the HTML table.

7. Finally, add an *Analysis* web item into the right cell of the second row of the HTML table.

    a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

    b) Drag an *Analysis* web item to your web template and drop it inside the right cell of the second row of the HTML table.

8. Now, configure the *Dropdown Box*. You want the dropdown list to show all the query views that are available for the selected data provider. In the *Properties* work area, choose the *Dropdown Box* and make the following settings:

Property/Parameter	Value
Data Binding > Data Binding Type	`Query View Selection`
Data Binding > Selection of Query Views	`Select the button at the end of the row`
Selection of Query Views > Data Provider	`Select DP_1`
Selection of Query Views > Specific List of Query Views	`Off (unchecked)`
Selection of Query Views > Target Web Item	`ANALYSIS_ITEM_1`

    a) In the *Properties* work area for the *DROPDOWN_ITEM_1* web item, set the properties for the fields as shown in the table above.

    b) When finished, select *OK*.

264 2007/Q2

9. Now, configure the *Button Group*. You want to have a button for saving new views and a button to print the current analysis to a PDF file. In the *Properties* work area, choose the *BUTTON_GROUP_ITEM_1* web item and make the following settings:

Property/Parameter	Value
Internal Display > List of Buttons (1) > 1 ?? (Button)	`Select the button at the end of the row`
Button > Caption	`Save View`
Button > Action > Command	`Select the button at the end of the row`
Edit Command window	Select the `All Commands` tab.
Commands for Data Provider > Data Provider Commands for Open/Save Functions	Expand the list and choose `Call Save Dialog [SAVE_AS]`. Then choose `Next`.
Command Target > Data Provider Affected	Select `DP_1`, then choose `Next`.
Command Specific Parameters > Layout of the Open-Save Dialog	`Display only Dataprovider Persistency Options [DATAPROVIDER]`
Finish the properties for this button	Select `OK` twice.
Internal Display > List of Buttons (1) > * Button	`Select the button at the end of the row`
Button > Caption	`Print to PDF`
Button > Action > Command	`Select the button at the end of the row`
Edit Command window	Select the `All Commands` tab.
Commands for Web Templates	Expand the list and choose `Export Web Application [EXPORT]`. Then choose `Next`.

Continued on next page

 SAP

Data Binding > Web Items (0) > Item Binding	Select **ANALYSIS_ITEM_1**
Command Specific Parameters > Show Export Dialog	**Off (unchecked)**
Finish the properties of this button	Select **OK** twice.

a)   In the *Properties* work area for the *BUTTON_GROUP_ITEM_1* web item, set the properties for the fields as shown in the table above.

b)   When finished, select *OK* twice to return to your template.

10.   Save your web template using the values below.

Field	**Value**
Description	**GR## Commands 1**
Technical Name	**GR##COMMANDS1**

Then execute the template. From the dropdown box select from the views that are available to change the analysis results. Using the navigation pane, create a new view by adding or rearranging characteristics.

Then save the view using the *Save View* button. Choose your own description and technical name. Then refresh the browser window and locate your new view in the dropdown list.

Finally, view the results in PDF format using the *Print to PDF* button.

a) In the Web Application Designer, click on the *Save* 💾 icon.

b) Enter the values shown above for the *Description* and *Technical Name* fields.

c) Select *OK* when finished.

d) Select the *Execute* 🔵 icon to execute your web application.

e) Logon to the NetWeaver portal using your assigned user ID and password.

f) From the displayed results, choose a different query view from the dropdown box to change the results.

g) Add one or two characteristics from the *Free Characteristics* area of the Navigation Pane to the Analysis area. When you are satisfied with the result, select the *Save View* button. Enter a description and technical name for the query view and then select *OK*.

h) In your browser window, select the *Refresh* button. Then see if your newly created query view is in the list of choices from the dropdown box.

i) Use the *Print to PDF* button to output the current results to a PDF file. When prompted, select *Open* to view the file.

Task 2: Using Groups and Table Selection Commands

You will create a web template that allows users to select a row of data and automatically show a more detailed analysis for the selected value. You will also create a button to reset the results to the original view.

1. Return to the BEx Web Application Designer and create a new web template.

a) In the Web Application Designer, select *"Create New Web Template"*.

2. To begin, create a new data provider and use the query **T_CTYSALES** to supply the data.

a) Double click on *New Data Provider* in the design area.

b) Select the data provider type *Query* using the radio button.

c) Enter **T_CTYSALES** as the name of the query and select *OK*.

Continued on next page

3. Create a second data provider and use the query **T_REGSALES** to supply the data.

    a) Double click on *New Data Provider* in the design area.

    b) Select the data provider type *Query* using the radio button.

    c) Enter **T_REGSALES** as the name of the query and select *OK*.

4. Choose the *Group* web item from the Advanced group of web items and drag it into web template. This web item provides for a Caption area, a Toolbar area and a Content area.

    a) In the Web Items area of the Web Application Designer, open the *Advanced* web item category.

    b) Drag the *Group* web item into your web template.

5. Choose the *Button Group* web item from the Standard group of web items and drag it into web template inside the *Group* web item.

    a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

    b) Drag the *Button Group* web item into your web template and place it inside the *Group* web item.

6. Choose the *Container* web item from the Advanced group of web items and drag it into web template to the right of the *Button Group* web item.

    a) In the Web Items area of the Web Application Designer, open the *Advanced* web item category.

    b) Drag the *Container* web item into your web template and drop it to the right of the *Button Group* web item.

7. For alignment purposes, insert an HTML table inside the *Container* web item with one row and two columns. In the row, set the vertical alignment to **Top**.

    a) Click on the *Insert Table* ⊞ icon or from the context menu in your template choose *Insert... → Table*.

    b) In the *Edit HTML Element* screen, change the *Row* value to **1** and the *Column* value to **2**. Choose *OK* to close the window.

    c) Right-click inside the first row of the HTML table to access the context menu for the table. Then using the path *Table → Edit → Edit Row <tr>* set the *Vertical* field value to **Top**.

    d) Choose *OK*.

8. Add an *Analysis* web item to the left cell of the HTML table. Then add an *Analysis* web item to the right cell of the HTML table.

a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

b) Drag an *Analysis* web item into the left cell of the HTML table.

c) Repeat the process for the right cell of the HTML table.

9. Now, configure the *Group* web item. In the *Properties* work area, choose the *Group* and make the following settings:

Property/Parameter	Value
Display > Width in Pixels	`950`
Display > Height in Pixels	`400`
Internal Display > Subordinate Web Item	Select `CONTAINER_ITEM_1`.
Internal Display > With Caption > Caption Type	Select `Text [CAPTION]`.
Internal Display > With Caption > Caption	`Country/Region Analysis`
Internal Display > With Toolbar	`On (checked)`
Internal Display > With Toolbar > Subordinate Web Item	Select `BUTTON_GROUP_ITEM_1`

a) In the *Properties* work area for the *GROUP_ITEM_1* web item, set the properties for the fields as shown in the table above.

10. Next, configure the *Analysis* web item in the right cell of the HTML table. You will configure it to be hidden when the web application starts. It should be bound to data provider *DP_2*.

In the *Properties* work area, choose the *ANALYSIS_ITEM_2* item and make the following settings:

Property/Parameter	Value
Display > Visibility	`Hidden`
Data Binding > Data Provider	`DP_2`

a) In the *Properties* work area for the *ANALYSIS_ITEM_2* web item, set the properties for the fields as shown in the table above.

2007/Q2   **269** SAP

11. Next, configure the *Analysis* web item in the left cell of the HTML table.
     You will configure it to first allow single rows of data to be selected and then
     pass that *Country* value on to the second Analysis item as a filter value
     using a command. As the next command, the second Analysis area should
     then be made visible.

     The Analysis item in the left cell of the HTML table should be bound to
     data provider *DP_1*.

     In the *Properties* work area, choose the *ANALYSIS_ITEM_1* item and make
     the following settings:

Property/Parameter	Value
Data Binding > Data Provider	Select **DP_1**.
Behavior > Row Selection	Select **Single with Commands**.
Behavior > Row Selection > Activation Action > Command	**Select the button at the end of the row**.
Edit Command Screen	Choose **All Commands** tab.
Commands for Data Provider > Data Provider Commands for Filter Values	Choose **Set Filter Values by Different Sources [SET_SELECTION_STATE_BY_BINDING].** Then select **Next**.
Command Target > Target Data Providers (0) > Data Provider Affected	Select **DP_2**, then **Next**.
Data Binding > Selection Data Bindings (0) > * Selection Binding	**Select the button at the end of the row.**
Selection Binding > Characteristic	**Select the button at the end of the row.**
Valuehelp Selector Screen	Select **Sold-to Party Country (0SOLD_TO__0COUNTRY)** from the list and the bottom of the screen. Then select **OK**.
Selection Binding > Binding Type	Select **Web Item Selection**.
Selection Binding > Item Selection > Item binding	Select **ANALYSIS_ITEM_1**.
Selection Binding > Item Selection > Characteristic	Select **Sold-to Party Country (0SOLD_TO__0COUNTRY)**. Then choose **OK**.

Edit Command screen	Select the **Next Command** button.
Command List Editing screen	Select the **Insert** button.
Edit Command screen	Choose **All Commands** tab.
Commands for Web Items	Choose **Set Web Item Parameters [SET_ITEM_PARAMETERS].** Select **Next**.
Command Target > Target Web Item	Select **ANALYSIS_ITEM_2**, then **Next**.
Display > Visibility	Select **Visible**.
Scroll down to Data Binding > Data Provider	Select **DP_2**.
Finish Command Sequence	Select **OK** twice.

a)   In the *Properties* work area for the *ANALYSIS_ITEM_2* web item, set the properties for the fields as shown in the table above.

12. Finally, configure the *Button Group*. You want to have a button for removing the detailed analysis after you have selected a particular country. In the *Properties* work area, choose the *BUTTON_GROUP_ITEM_1* web item and make the following settings:

Property/Parameter	Value
Internal Display > List of Buttons (1) > 1 ?? (Button)	**Select the button at the end of the row**
Button > Caption	**Remove Details**
Button > Action (Command Triggered) > Command	**Select the button at the end of the row**
Edit Command window	Select the **All Commands** tab.
Commands for Web Items **(This command will hide ANALYSIS_ITEM_2)**	Choose **Set Web Item Parameters [SET_ITEM_PARAMETERS].** Select **Next**.
Command Target > Target Web Item	Select **ANALYSIS_ITEM_2**, then **Next**.
Display > Visibility	Select **Hidden**. (Do not use the defaulted value.)
Scroll down to Data Binding > Data Provider	Select **DP_2**. Then select the **Next Command** button.

Continued on next page

Start the next command	Select the **Insert** button.
Edit Command window	Select the **All Commands** tab.
Commands for Web Items **(This command will reset the row selection property of ANALYSIS_ITEM_1.)**	Choose **Set Web Item Parameters [SET_ITEM_PARAMETERS].** Select **Next**.
Command Target > Target Web Item	Select **ANALYSIS_ITEM_1**, then **Next**.
Behavior > Row Selection	Select **None**.
Scroll down to Data Binding > Data Provider	Select **DP_1**. Then select the **Next Command** button.
Edit Command screen	Select the **All Commands** tab.
Commands for Web Items **(This command will turn on row selection for ANALYSIS_ITEM_1. It will have two other commands embedded in it.)**	Choose **Set Web Item Parameters [SET_ITEM_PARAMETERS].** Select **Next**.
Command Target > Target Web Item	Select **ANALYSIS_ITEM_1**, then **Next**.
Behavior > Row Selection	Select **Single with Commands**. (Do not use the default value.)
Behavior > Row Selection > Activation Action > Command	**Select the button at the end of the row**.
Edit Command screen	Choose **All Commands tab**
Commands for Data Provider > Data Provider Commands for Filter Values	Choose **Set Filter Values by Different Sources [SET_SELEC- TION_STATE_BY_BINDING].** Select **Next**.
Command Target > Target Data Providers (0) > Data Provider Affected	Select **DP_2**, then **Next**.
Data Binding > Selection Data Bindings (0) > * Selection Binding	**Select the button at the end of the row.**
Selection Binding > Characteristic	**Select the button at the end of the row.**

Valuehelp Selector screen	Select **Sold-to Party Country (0SOLD_TO__0COUNTRY)** from the list and the bottom of the screen. Then select **OK**.
Selection Binding > Binding Type	Select **Web Item Selection**.
Selection Binding > Item Selection > Item binding	Select **ANALYSIS_ITEM_1**.
Selection Binding > Item Selection > Characteristic	Select **Sold-to Party Country (0SOLD_TO__0COUNTRY)**. Then choose **OK**.
Edit Command screen	Select the **Next Command** button.
Command List Editing screen	Select the **Insert** button.
Edit Command screen	Choose **All Commands** tab.
Commands for Web Items	Choose **Set Web Item Parameters [SET_ITEM_PARAMETERS]**. Select **Next**.
Command Target > Target Web Item	Select **ANALYSIS_ITEM_2**, then **Next**.
Display > Visibility	Select **Visible**.
Scroll down to Data Binding > Data Provider	Select **DP_2**.
Finish Command Sequence	Select **OK** five times.

a) In the *Properties* work area for the *BUTTON_GROUP_ITEM_1* web item, set the properties for the fields as shown in the table above.

b) When finished, select *OK* five times to return to your template.

13. Save your web template using the values below.

Field	**Value**
Description	**GR## Commands 2**
Technical Name	**GR##COMMANDS2**

Continued on next page

Then execute the template. From the initial view of the web application, choose a country using the selection buttons at the beginning of each row. Notice that the additional details for that country are shown in the area to the right. Choosing the *Remove Details* button will return the application to its starting point. You also can choose the selection button for the *Results* row to see details for all countries.

a)   In the Web Application Designer, click on the *Save* 🖫 icon.

b)   Enter the values shown above for the *Description* and *Technical Name* fields.

c)   Select *OK* when finished.

d)   Select the *Execute* 🖳 icon to execute your web application.

e)   Logon to the NetWeaver portal using your assigned user ID and password.

f)   From the displayed results, the selection buttons at the beginning of each row to display detailed information for that country. Use the *Remove Details* button to reset the application to its original state.

Task 3: Using Commands in Links

You will build a simple web application and use a series of links to control a chart presentation.

1.   Return to the BEx Web Application Designer and create a new web template.

    a)   In the Web Application Designer, select *"Create New Web Template"*.

2.   To begin, create a new data provider and use the query **T_CTYSALES** to supply the data.

    a)   Double click on *New Data Provider* in the design area.

    b)   Select the data provider type *Query* using the radio button.

    c)   Enter **T_CTYSALES** as the name of the query and select *OK*.

3. For alignment purposes, insert an HTML table inside the template with one row and two columns. Set the table width to **500 pixels**. In the row, set the vertical alignment to **Top**.

   a) Click on the *Insert Table* ⊞ icon or from the context menu in your template choose *Insert... → Table*.

   b) In the *Edit HTML Element* screen, change the *Row* value to **1** and the *Column* value to **2**. Enter **500** for the width of the table. Choose *OK* to close the window.

   c) Right-click inside the first row of the HTML table to access the context menu for the table. Then using the path *Table → Edit → Edit Row <tr>* set the *Vertical* field value to **Top**.

   d) Choose *OK*.

4. Choose the *Chart* web item from the Standard group of web items and drag it into web template inserting it in the left cell of the HTML table.

   a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

   b) Drag the *Chart* web item into your web template and drop it in the left cell of the HTML table.

5. Add 4 *Link* web items into the right cell of the first row of the HTML table. Insert a blank line after each one in order to have the link items stacked vertically in the right cell of the HTML table.

   a) In the Web Items area of the Web Application Designer, open the *Miscellaneous* web item category.

   b) Drag the first *Link* web item to your web template and drop it inside the right cell of the HTML table. After you add the first *Link* web item, place the cursor at the end of the item and press *Enter* to insert a blank line before inserting the next *Link* web item. Repeat this process until all four links are stacked vertically in the right cell of the HTML table.

6. Configure the *Chart* web item. In the *Properties* work area, choose the *CHART_ITEM_1* item and make the following settings:

Property/Parameter	Value
Width in Pixels	**400**

   a) In the *Properties* work area for the *DROPDOWN_ITEM_1* web item, set the properties for the fields as shown in the table above.

   b) When finished, select *OK*.

Continued on next page

7.   Now, configure the first *Link* web item to filter the chart to `Germany`. In the *Properties* work area, choose *LINK_ITEM_1* and make the following settings:

Property/Parameter	Value
Internal Display > Text	`Germany`
Behavior > Action (Command Triggered) > Command	`Select the button at the end of the row`.
Edit Command screen	Select the `All Commands` tab.
Commands for Data Provider > Data Provider Commands for Filter Values	Select the `Set Filter Value for a Characteristic` command, then choose `Next`.
Command Target > Target Data Providers (0) > Data Provider Affected	Select `DP_1`, then choose `Next`.
Data Binding > Characteristic	`Select the button at the end of the row`.
Valuehelp Selector screen	Choose `Sold-to Party Country` from the values at the bottom of the screen.
Command Specific Parameters > Operator > Equals > Member Name	`Select the button at the end of the row`.
Selection screen	Choose `Germany`, then choose `OK`.
Finish the command settings	Choose `OK`.

a)   In the *Properties* work area for the *LINK_ITEM_1* web item, set the properties for the fields as shown in the table above.

b)   When finished, select *OK*.

8.   Configure the second*Link* web item to filter the chart to exclude `Germany`. In the *Properties* work area, choose *LINK_ITEM_2* and make the following settings:

Property/Parameter	Value
Internal Display > Text	`Exclude Germany`
Behavior > Action (Command Triggered) > Command	`Select the button at the end of the row`.
Edit Command screen	Select the `All Commands` tab.

Commands for Data Provider > Data Provider Commands for Filter Values	Select the `Set Filter Value for a Characteristic` command, then choose `Next`.
Command Target > Target Data Providers (0) > Data Provider Affected	Select `DP_1`, then choose `Next`.
Data Binding > Characteristic	`Select the button at the end of the row`.
Valuehelp Selector screen	Choose `Sold-to Party Country` from the values at the bottom of the screen.
Command Specific Parameters > Sign	Select `Excluding`
Command Specific Parameters > Operator > Equals > Member Name	`Select the button at the end of the row`.
Selection screen	Choose `Germany`, then choose `OK`.
Finish the command settings	Choose `OK`.

a)   In the *Properties* work area for the *LINK_ITEM_2* web item, set the properties for the fields as shown in the table above.

b)   When finished, select *OK*.

9.   Configure the third *Link* web item to clear any filter values for *Country*. In the *Properties* work area, choose *LINK_ITEM_3* and make the following settings:

Property/Parameter	Value
Internal Display > Text	`All Countries`
Behavior > Action (Command Triggered) > Command	`Select the button at the end of the row`.
Edit Command screen	Select the `All Commands` tab.
Commands for Data Provider > Data Provider Commands for Filter Values	Select the `Remove Filter Values for a Characteristic` command, then choose `Next`.

Continued on next page

Command Target > Target Data Providers (0) > Data Provider Affected	Select **DP_1**, then choose **Next**.
Data Binding > Characteristic	Choose **Sold-to Party Country**, then select **OK**.
Finish the command settings	Choose **OK**.

a) In the *Properties* work area for the *LINK_ITEM_3* web item, set the properties for the fields as shown in the table above.

b) When finished, select *OK*.

10. Finally, configure the fourth *Link* web item to call up the *Chart Properties Dialog*. In the *Properties* work area, choose *LINK_ITEM_4* and make the following settings:

Property/Parameter	Value
Internal Display > Text	**Chart Properties**
Behavior > Action (Command Triggered) > Command	**Select the button at the end of the row**.
Edit Command screen	Select the **All Commands** tab.
Commands for Web Items	Select the **Call Chart Properties Dialog** command, then choose **Next**.
Command Target > Target Web Item	Select **CHART_ITEM_1**, then choose **Next**.
Finish the command settings	Choose **OK**.

a) In the *Properties* work area for the *LINK_ITEM_4* web item, set the properties for the fields as shown in the table above.

b) When finished, select *OK*.

11. Save your web template using the values below.

Field	Value
Description	**GR## Commands 3**
Technical Name	**GR##COMMANDS3**

Then execute the template. Use the links to change the results in the chart.

a) In the Web Application Designer, click on the *Save* 💾 icon.

b) Enter the values shown above for the *Description* and *Technical Name* fields.

c) Select *OK* when finished.

d) Select the *Execute* 🖼 icon to execute your web application.

e) Logon to the NetWeaver portal using your assigned user ID and password.

f) From the displayed results, use the links to navigate to different versions of the chart.

Lesson Summary

You should now be able to:
- Explain the basic operation of the Command Wizard
- Insert commands into the web application using various web items
- Use sequences of commands to accomplish multiple tasks

Related Information

- In the NetWeaver 2004s BI online documentation, each parameter of each web item is documented. Visit http://help.sap.com to locate the documentation.

Lesson: Modular Web Application Design

Lesson Overview

The focus of this lesson is creating reusable web template modules in order to speed the development of sophisticated web applications.

Lesson Objectives

After completing this lesson, you will be able to:

- Efficiently develop web applications by reusing modules such as header and footer web templates
- Use the BI Pattern Wizard to modify the SAP-delivered patterns.

Business Example

You realize that many of your web templates use the same components, so you want to investigate options for reusing existing web templates to lessen your development time and promote a consistent look and feel for all your web applications.

BI Pattern Wizard

SAP delivers several *patterns* for common web template components. These patterns can be edited and customized using the *BI Pattern Wizard.*

- **The Pattern Wizard is part of the Web Application Designer.**
- **SAP supplies several BI Patterns that can be customized and copied using the Pattern Wizard.**
- **Patterns provide for dropdowns, buttons, filters, variables, etc.**
- **Using patterns can help ensure consistency between web applications.**

Example of Information Consumer Pattern

Figure 105: BI Pattern Wizard Features

Within the Web Application Designer you can access SAP-supplied BI Patterns and maintain them for your own use using the BI Pattern Wizard. Each supplied pattern is a full analysis template designed for a specific purpose. For example, the BI Consumer pattern provides a common analysis workbench complete with filters, buttons, dropdown boxes, variables and analysis areas.

The BI Patterns and the BI Pattern Wizard are accessed from the *Select a Template* screen that is displayed initially when the Web Application Designer is started. Click on the *New* button and then choose *All Patterns and Web Templates*.

The BI Pattern Wizard is accessed by choosing *Web Template* → *New* → *All Patterns and Web Templates* in the Web Application Designer.

The steps for using the BI Pattern Wizard are:

1. Choose a BI Pattern to copy.

2. Choose the style and content of the toolbar.

3. Choose the content and settings for the dropdown box using reusable web items.

4. Choose characteristics to appear in the filter area.

5. Make general choices about titles, size of items, variable screen, RRI, header and footer templates.

Figure 106: BI Pattern Wizard Steps

Using the BI Pattern Wizard can lower the cost of developing a series of web templates. Choices of toolbar content, dropdown box usage and content, filter characteristics and other miscellaneous items are offered in an easy-to-use sequence of steps.

Once the result has been saved, it can be further enhanced directly in the Web Application Designer using any of the capabilities of this design tool.

Modular Web Applications

■ **Often, many parts of your web applications will have the same 'look and feel'.**

■ **In this case, you should consider creating common components for these items that can be reused in other web templates.**

■ **Common examples are:**
   ◆ **Corporate header with logo and text**
   ◆ **Standard footer with text elements**
   ◆ **Standard set of links for user assistance**
   ◆ **Standard buttons for common actions**
   ◆ **Standard charts for cockpit usage**

■ **The Web Template web item makes it easy to reuse component templates in other web templates…" web template nesting".**
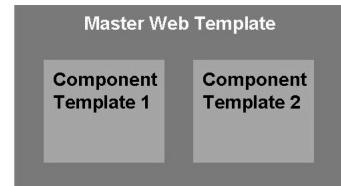
Figure 107: Creating Web Application Modules

Frequently an enterprise will want to adopt a set of standards for their web applications. Doing so helps the user in adopting new analysis applications since the learning curve will be shorter if it follows the pattern established by earlier analysis applications.

To assist in establishing and following these standards, you can create web application modules that can then be reused wherever needed in production web templates. Common examples of these web application modules are:

• Corporate header with logo and text

• Standard footer with common text elements

• Standard set of links for user assistance

• Standard button set to for common actions

• Standard chart representation for cockpit usage

Once a module is developed, it is linked into another template with the *Web Template* web item.
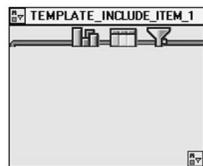
■ **Embeds a web template in the web template**
■ **Key properties:**

Internal Display → Web Template Name
- **Specify the technical name of the embedded web template.**
- **Multiple levels of web template nesting possible**

Master Web Template

Header Template

Body Template with
Standard Components

Analysis Template   Chart Template

Other Web Items

Footer Template
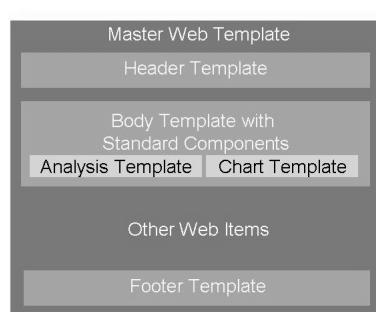
Figure 108: Web Item - Web Template

The Web Template web item has a simple purpose. It links other web templates into the template in which it is placed. This makes it possible to construct web templates using web application modules stored as web templates.

In the Properties screen area, the *Internal Display → Web Template Name* is the main property to be set.

There is no limit to the number of levels of web template "nesting" that can be used.

While creating reusable web application modules it may be necessary to create and assign data providers in order to create commands that must reference data providers as part of their operation. However, the query, query view or InfoProvider supplying the data to the data provider is unlikely to be the source in every template that uses this module. So, the developer can take two approaches to saving the web application module.

The first approach is to leave the data providers used in the design in place and then change them in each template that uses the module.

The second approach is to delete the data providers in the reusable web application module before saving it. You will receive error messages in the Web Application Designer, but this does not prohibit saving the module. In this case, when reusing the module it will only be necessary to create the appropriate data providers for the production web application. As long as the logical name of the data provider (i.e.; DP_1) is not changed, then no changes will be required for embedded commands referencing this data provider.

In both cases, if the command in the module references characteristic or key figure InfoObjects, the developer will need to validate that the same InfoObjects are included in the actual data providers used in the production web template.

Scenario: You have created a standard corporate template with a header, body and footer as shown.

Problem: You want to change the footer information as the user changes analyses with the tabs.

Solution: Assign the same data provider to the tab items and the footer text elements. Then put a command on each tab to change the source assigned to the data provider.

The texts will change with each tab selection.

		Header Template			
	Tabs w/ two queries				

Country Analysis	Region Analysis				
		Sales Vol. EUR	Cost of Sales	Margin %	
Sold-to Party Country		* 1,000 EUR	* 1,000 EUR	%	
DE	Germany	359,901	195,899	46	
FR	France	1,067	801	25	
GB	Great Britain	12,911	3,106	76	
IT	Italy	11,262	3,272	71	
NO	Norway	18,338	4,398	76	
US	United States	41,358	24,977	40	
Overall Result		444,836	232,454	48	

InfoProvider Technical Name:	T_SDC05	
InfoProvider Description:	Customer Sales	**Footer Template**
Last data update:	09.06.2006 12:07:24	
Query Description:	Country Sales Analysis	

Figure 109: Web Templates and Commands

In the graphic above, an example is given where a sophisticated web application used to analyze several sources of data can suffice with only one data provider. In this case, a command is used for each of the tabs to change the data bound to the data provider by changing the underlying query or query view. This in ensures that the footer text items, also assigned to the same data provider, will show the appropriate values for the data on the selected tab.

Exercise 10: Modular Web Application Design

Exercise Objectives

After completing this exercise, you will be able to:

- Create reusable modules that can be used in other web applications.

Business Example

As you build web applications for your enterprise, you will want to insure consistency between similar web templates and reuse objects to reduce your development effort. Building reusable modules is an easy way to accomplish these goals.

Task 1: Creating Reusable Modules

You will create header and footer modules that can be used in subsequent web applications.

1. Start the BEx Web Application Designer and create a new web template.

2. This web template will be the header template. It will contain a logo and company slogan and will be used in all of your templates where you need this information. If your slogan ever changes, you will only need to change it once in this template.

   Insert the image *sap_logo.gif* into the template. Or, you may use another image from the MIME Repository if you imported one in an earlier exercise.

3. Place your cursor at the end of the image placeholder and enter a few spaces before typing a company slogan such as **The Best-Run Businesses Run SAP**. Select the text and change the font to **Tahoma** and the font size to **4**. Insert a blank line after the text for spacing purposes.

4. Save your web template using the values below.

Field	Value
Description	**GR## Header**
Technical Name	**GR##HEADER**

   Then execute the template to check the results.

5. Return to the BEx Web Application Designer and create a new web template.

6. To begin, create a new data provider and use the query **T_DUMMY** to supply the data.

Continued on next page

7. Choose the *Information Field* web item from the Advanced group of web items and drag it into web template.

8. Now, configure the *Information Field* web item. In the *Properties* work area, choose *INFO_FIELD_ITEM_1* and make the following settings:

Property/Parameter	Value
Data Binding > List of Text Elements to Display (0) > * General Text Elements	Select **Query Description**.
Data Binding > List of Text Elements to Display (1) > * General Text Elements	Select **InfoProvider Description**.
Data Binding > List of Text Elements to Display (2) > * General Text Elements	Select **Last Data Update**.

9. Before saving the template, delete data provider *DP_1* from the list of data providers. Ignore any error messages you may receive. Since this web template will later be referenced in other web templates, you will assign the appropriate source for DP_1 then.

10. Save your web template using the values below. Ignore any messages you may receive.

Field	Value
Description	**GR## Footer**
Technical Name	**GR##FOOTER**

Task 2: Create a Master Template

1. Return to the BEx Web Application Designer and create a new web template.

2. To begin, create a new data provider and use the query **T_DUMMY** to supply the data.

3. Now you will add a reference to your *GR##HEADER* web template into this web template. Choose the *Web Template* web item from the Advanced group of web items and drag it into web template.

4. Now, configure the *Web Template* web item. In the *Properties* work area, choose *TEMPLATE_INCLUDE_ITEM_1* and make the following settings:

Continued on next page

Property/Parameter	Value
Internal Display > Web Template	Select **Select the button at the end of the row**.
Open screen	Select template **GR##HEADER** and choose **Open**.

5. Choose the *Tab Pages* web item from the Advanced group of web items and drag it into web template to the right of the *Web Template* web item.

6. Choose the *Analysis* web item from the Standard group of web items and drag it into web template inside of the *Tab Pages* web item.

7. Add a second *Analysis* web item from the Standard group of web items and place it in the *Tab Pages* web item.

8. Now, configure the *Tab Pages* web item. In the *Properties* work area, choose *TABSTRIP_CONTAINER_ITEM_1* and make the following settings:

Property/Parameter	Value
Internal Display > Tab Panel List (0) > * Tab Panel	**Select the button at the end of the row**.
Tab Panel > Subordinate Web Item	Select **ANALYSIS_ITEM_1**.
Tab Panel > Activation Action > Command	**Select the button at the end of the row**.
Edit Command screen	Select the **All Commands** tab.
Commands for Data Provider > Basic Data Provider Commands	Select **Set Data Provider Parameters**, then select **Next**.
Command Target > Data Provider Affected	Select **DP_1**, then select **Next**.
Edit Command screen	Choose **OK** twice to return to template.
Internal Display > Tab Panel List (1) > * Tab Panel	**Select the button at the end of the row**.
Tab Panel > Subordinate Web Item	Select **ANALYSIS_ITEM_2**.
Tab Panel > Activation Action > Command	**Select the button at the end of the row**.
Edit Command screen	Select the **All Commands** tab.

 SAP

Commands for Data Provider > Basic Data Provider Commands	Select **Set Data Provider Parameters**, then select **Next**.
Command Target > Data Provider Affected	Select **DP_1**, then select **Next**.
Edit Command screen	Choose **OK** twice to return to template.

9. Finally, you will add a reference to your *GR##FOOTER* web template into this web template. Choose the *Web Template* web item from the Advanced group of web items and drag it into web template as the last item in the template.

10. Now, configure the *Web Template* web item. In the *Properties* work area, choose *TEMPLATE_INCLUDE_ITEM_2* and make the following settings:

Property/Parameter	Value
Internal Display > Web Template	Select **Select the button at the end of the row**.
Open screen	Select template **GR##FOOTER** and choose **Open**.

11. Before saving the template, delete data provider *DP_1* from the list of data providers. Ignore any error messages you may receive. Since this web template will later be used as the basis of other web templates, you will assign the appropriate source for data provider DP_1 then.

12. Save your web template using the values below. Ignore any messages you may receive.

Field	Value
Description	**GR## Master**
Technical Name	**GR##MASTER**

Task 3: Create a Web Application from the Master Template

Now you will use your master template as the basis for your real web application. This will require creating a data provider, adding captions to the tabs and choosing the actual data providers for the tab results.

1. If it is not already open, then open your template *GR##MASTER* in the Web Application Designer.

Continued on next page

2. From the menu bar, select *Web Template* → *Save as...* and save a copy of the master template using the values below.

Field	Value
Description	`GR## Modular Application`
Technical Name	`GR##MODULAR`

3. Now you can make some quick changes to create your new web application. First, create a data provider for the data you want to appear on the first tab. In this case, use query *T_CTYSALES*.

4. Configure the *Tab Pages* web item. In the *Properties* work area, choose *TABSTRIP_CONTAINER_ITEM_1* and make the following settings:

Property/Parameter	Value
Internal Display > Tab Panel List (2) > 1 Tab Panel	`Select the button at the end of the row`.
Tab Panel > Caption	Enter `Country Analysis`, then select **OK**.
Internal Display > Tab Panel List (2) > 2 Tab Panel	`Select the button at the end of the row`.
Tab Panel > Caption	Enter `Q4 Analysis`.
Tab Panel > Activation Action > Command	`Select the button at the end of the row`.
Edit Command screen	Select **Next**.
Data Binding > Query	`Select the button at the end of the row`.
Open screen	Select query `T_COQ4`, then select **Open**.
Edit Command screen	Choose **OK** twice to return to template.

5. Finish by saving your new modular application again, then executing it to see the results.

Click on the tabs to change the data that is displayed. Notice that the information fields change to correspond to the tab choice.

Solution 10: Modular Web Application Design

Task 1: Creating Reusable Modules

You will create header and footer modules that can be used in subsequent web applications.

1. Start the BEx Web Application Designer and create a new web template.

    a) Start the BEx Web Application Designer using the path *Start → Programs → Business Explorer → Web Application Designer*.

    b) Logon to the system with the user ID and password provided by your instructor.

    c) Select *"Create New Web Template"*.

2. This web template will be the header template. It will contain a logo and company slogan and will be used in all of your templates where you need this information. If your slogan ever changes, you will only need to change it once in this template.

    Insert the image *sap_logo.gif* into the template. Or, you may use another image from the MIME Repository if you imported one in an earlier exercise.

    a) From the menu bar, choose the *Image* icon or use the menu path *Insert → Image*.

    b) From the *Edit HTML Element* screen, replace the **...** at the end of the defaulted path with **sap_logo.gif** or the name of another image file in the MIME Repository, if desired.

    c) Select *OK* to return to your template.

3. Place your cursor at the end of the image placeholder and enter a few spaces before typing a company slogan such as **The Best-Run Businesses Run SAP**. Select the text and change the font to **Tahoma** and the font size to **4**. Insert a blank line after the text for spacing purposes.

    a) Place the cursor immediately after the image placeholder.

    b) With the spacebar, enter three or four spaces.

    c) Type the company slogan suggested in the exercise, or one of your own choosing.

    d) Drag over the text to select it, then use the text toolbar to change the font to **Tahoma**.

    e) With the text still selected, change the font size to **4**.

    f) With the cursor at the end of the text, select the *Enter* key to insert a blank line in the template.

4. Save your web template using the values below.

Field	Value
Description	**GR## Header**
Technical Name	**GR##HEADER**

Then execute the template to check the results.

    a) In the Web Application Designer, click on the *Save* 💾 icon.

    b) Enter the values shown above for the *Description* and *Technical Name* fields.

    c) Select *OK* when finished.

    d) Select the *Execute* 🔲 icon to execute your web application.

    e) Logon to the NetWeaver portal using your assigned user ID and password.

    f) Check the displayed results for accuracy, then exit the web application.

5. Return to the BEx Web Application Designer and create a new web template.

    a) In the Web Application Designer, select *"Create New Web Template"*.

2007/Q2   **293** SAP

6. To begin, create a new data provider and use the query **T_DUMMY** to supply the data.

    a) Double click on *New Data Provider* in the design area.

    b) Select the data provider type *Query* using the radio button.

    c) Enter **T_DUMMY** as the name of the query and select *OK*.

7. Choose the *Information Field* web item from the Advanced group of web items and drag it into web template.

    a) In the Web Items area of the Web Application Designer, open the *Advanced* web item category.

    b) Drag the *Information Field* web item into your web template.

8. Now, configure the *Information Field* web item. In the *Properties* work area, choose *INFO_FIELD_ITEM_1* and make the following settings:

Property/Parameter	Value
Data Binding > List of Text Elements to Display (0) > * General Text Elements	Select **Query Description**.
Data Binding > List of Text Elements to Display (1) > * General Text Elements	Select **InfoProvider Description**.
Data Binding > List of Text Elements to Display (2) > * General Text Elements	Select **Last Data Update**.

    a) In the *Properties* work area for the *INFO_FIELD_ITEM_1* web item, set the properties for the fields as shown in the table above.

9. Before saving the template, delete data provider *DP_1* from the list of data providers. Ignore any error messages you may receive. Since this web template will later be referenced in other web templates, you will assign the appropriate source for DP_1 then.

    a) Right click on data provider *DP_1* and from the context menu choose *Delete*. Respond to the pop-up message with **OK**.

10. Save your web template using the values below. Ignore any messages you may receive.

Field	Value
Description	**GR## Footer**
Technical Name	**GR##FOOTER**

a)   In the Web Application Designer, click on the *Save* 💾 icon. Ignore any messages you may receive.

b)   Enter the values shown above for the *Description* and *Technical Name* fields.

c)   Select *OK* when finished.

Task 2: Create a Master Template

1.   Return to the BEx Web Application Designer and create a new web template.

a)   In the Web Application Designer, select *"Create New Web Template"*.

2.   To begin, create a new data provider and use the query **T_DUMMY** to supply the data.

a)   Double click on *New Data Provider* in the design area.

b)   Select the data provider type *Query* using the radio button.

c)   Enter **T_DUMMY** as the name of the query and select *OK*.

3.   Now you will add a reference to your *GR##HEADER* web template into this web template. Choose the *Web Template* web item from the Advanced group of web items and drag it into web template.

a)   In the Web Items area of the Web Application Designer, open the *Advanced* web item category.

b)   Drag the *Web Template* web item into your web template.

4.   Now, configure the *Web Template* web item. In the *Properties* work area, choose *TEMPLATE_INCLUDE_ITEM_1* and make the following settings:

Property/Parameter	Value
Internal Display > Web Template	Select **Select the button at the end of the row**.
Open screen	Select template **GR##HEADER** and choose **Open**.

a)   In the *Properties* work area for the *TEMPLATE_INCLUDE_ITEM_1* web item, set the properties for the fields as shown in the table above.

5.   Choose the *Tab Pages* web item from the Advanced group of web items and drag it into web template to the right of the *Web Template* web item.

        a)   In the Web Items area of the Web Application Designer, open the *Advanced* web item category.

        b)   Drag the *Tab Pages* web item into your web template and drop it to the right of the *Web Template* web item.

6.   Choose the *Analysis* web item from the Standard group of web items and drag it into web template inside of the *Tab Pages* web item.

        a)   In the Web Items area of the Web Application Designer, open the *Standard* web item category.

        b)   Drag the *Analysis* web item into your web template and drop it inside of the *Tab Pages* web item..

7.   Add a second *Analysis* web item from the Standard group of web items and place it in the *Tab Pages* web item.

        a)   In the Web Items area of the Web Application Designer, open the *Standard* web item category.

        b)   Drag the *Analysis* web item into your web template and drop it inside the *Tab Pages* web item.

8.   Now, configure the *Tab Pages* web item. In the *Properties* work area, choose *TABSTRIP_CONTAINER_ITEM_1* and make the following settings:

Property/Parameter	Value
Internal Display > Tab Panel List (0) > * Tab Panel	`Select the button at the end of the row`.
Tab Panel > Subordinate Web Item	Select **ANALYSIS_ITEM_1**.
Tab Panel > Activation Action > Command	`Select the button at the end of the row`.
Edit Command screen	Select the **All Commands** tab.
Commands for Data Provider > Basic Data Provider Commands	Select **Set Data Provider Parameters**, then select **Next**.
Command Target > Data Provider Affected	Select **DP_1**, then select **Next**.
Edit Command screen	Choose **OK** twice to return to template.
Internal Display > Tab Panel List (1) > * Tab Panel	`Select the button at the end of the row`.
Tab Panel > Subordinate Web Item	Select **ANALYSIS_ITEM_2**.

Continued on next page

Tab Panel > Activation Action > Command	`Select the button at the end of the row.`
Edit Command screen	Select the **All Commands** tab.
Commands for Data Provider > Basic Data Provider Commands	Select **Set Data Provider Parameters**, then select **Next**.
Command Target > Data Provider Affected	Select **DP_1**, then select **Next**.
Edit Command screen	Choose **OK** twice to return to template.

a)   In the *Properties* work area for the *TABSTRIP_CONTAINER_ITEM_1* web item, set the properties for the fields as shown in the table above.

9.   Finally, you will add a reference to your *GR##FOOTER* web template into this web template. Choose the *Web Template* web item from the Advanced group of web items and drag it into web template as the last item in the template.

a)   In the Web Items area of the Web Application Designer, open the *Advanced* web item category.

b)   Drag the *Web Template* web item into your web template as the last item.

10.   Now, configure the *Web Template* web item. In the *Properties* work area, choose *TEMPLATE_INCLUDE_ITEM_2* and make the following settings:

Property/Parameter	Value
Internal Display > Web Template	Select `Select the button at the end of the row`.
Open screen	Select template **GR##FOOTER** and choose **Open**.

a)   In the *Properties* work area for the *TEMPLATE_INCLUDE_ITEM_2* web item, set the properties for the fields as shown in the table above.

11.   Before saving the template, delete data provider *DP_1* from the list of data providers. Ignore any error messages you may receive. Since this web template will later be used as the basis of other web templates, you will assign the appropriate source for data provider DP_1 then.

a)   Right click on data provider *DP_1* and from the context menu choose *Delete*. Respond to the pop-up message with **OK**.

12. Save your web template using the values below. Ignore any messages you may receive.

Field	Value
Description	`GR## Master`
Technical Name	`GR##MASTER`

a) In the Web Application Designer, click on the *Save* 💾 icon. Ignore any messages you may receive.

b) Enter the values shown above for the *Description* and *Technical Name* fields.

c) Select *OK* when finished.

Task 3: Create a Web Application from the Master Template

Now you will use your master template as the basis for your real web application. This will require creating a data provider, adding captions to the tabs and choosing the actual data providers for the tab results.

1. If it is not already open, then open your template *GR##MASTER* in the Web Application Designer.

a) If necessary, open the template *GR##MASTER* in the Web Application Designer.

2. From the menu bar, select *Web Template → Save as...* and save a copy of the master template using the values below.

Field	Value
Description	`GR## Modular Application`
Technical Name	`GR##MODULAR`

a) In the Web Application Designer, click on the *Save* 💾 icon. Ignore any messages you may receive.

b) Enter the values shown above for the *Description* and *Technical Name* fields.

c) Select *OK* when finished.

3. Now you can make some quick changes to create your new web application. First, create a data provider for the data you want to appear on the first tab. In this case, use query *T_CTYSALES*.

    a) Double click on *New Data Provider* in the design area.

    b) Select the data provider type *Query* using the radio button.

    c) Enter **T_CTYSALES** as the name of the query and select *OK*.

4. Configure the *Tab Pages* web item. In the *Properties* work area, choose *TABSTRIP_CONTAINER_ITEM_1* and make the following settings:

Property/Parameter	Value
Internal Display > Tab Panel List (2) > 1 Tab Panel	**Select the button at the end of the row.**
Tab Panel > Caption	Enter **Country Analysis**, then select **OK**.
Internal Display > Tab Panel List (2) > 2 Tab Panel	**Select the button at the end of the row.**
Tab Panel > Caption	Enter **Q4 Analysis**.
Tab Panel > Activation Action > Command	**Select the button at the end of the row.**
Edit Command screen	Select **Next**.
Data Binding > Query	**Select the button at the end of the row.**
Open screen	Select query **T_COQ4**, then select **Open**.
Edit Command screen	Choose **OK** twice to return to template.

    a) In the *Properties* work area for the *TABSTRIP_CONTAINER_ITEM_1* web item, set the properties for the fields as shown in the table above.

5. Finish by saving your new modular application again, then executing it to see the results.

 SAP

Click on the tabs to change the data that is displayed. Notice that the information fields change to correspond to the tab choice.

a)   In the Web Application Designer, click on the *Save* icon.

b)   Select the *Execute* icon to execute your web application.

c)   Logon to the NetWeaver portal using your assigned user ID and password.

d)   Check the displayed results. Navigate by selecting the tabs. Notice how the contents of the information fields change as you select a different tab.

Lesson Summary

You should now be able to:

- Efficiently develop web applications by reusing modules such as header and footer web templates
- Use the BI Pattern Wizard to modify the SAP-delivered patterns.

Lesson: Advanced XHTML Features

Lesson Overview

In addition to having a large number of web items to use in web application design, the user also can enhance the web application directly using the XHTML editor included as part of the Web Application Designer. This lesson will explore several uses of this editor.

Lesson Objectives

After completing this lesson, you will be able to:

- Use the features of the XHTML editor
- Generate parameterized URLs
- Use JavaScript functions to enhance web applications

Business Example

As you users become more comfortable using web applications, they may make requests for more sophisticated navigational options. Also, you may want to automate some web applications by embedding selected parameters in the URL used to call the web application.

XHTML Editor
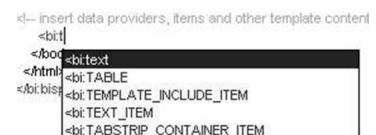
Figure 110: XHTML Editor Features - 1

In the Web Application Designer, the former 'HTML' tab has been replaced with the *XHTML (Extended HTML)* tab. This new view to the components of the web template provides many new enhancements for those designers who want to extend the functionality of the web template beyond the standard functions provided by the web items. HTML code that is entered into the template can be locally verified to catch syntax or tag usage mistakes. As HTML code is entered, the *Auto Complete* function will assist the programmer by suggesting available and appropriate tags based on the context of the entry. This can speed the entry of complex HTML code strings and help ensure accuracy.

Dynamic Display of Errors and Warnings

■ **Meaningful error messages**

■ **Line and Column indictors with hyperlinks: clicking on the 'line:col' will position the cursor on that coordinate. No scrolling through long templates to get to the error!**
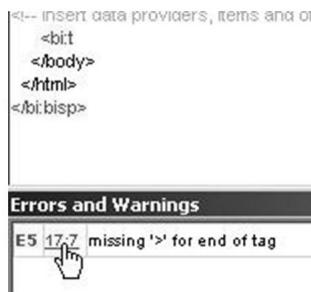
Figure 111: XHTML Editor Features - 2

As code is entered for a web template, the Editor dynamically checks the syntax of the code and displays the appropriate error messages in a window below the code. Highlighting the icon at the end of the message will provide a longer explanation of the error. Clicking on a link within the error message will position the cursor at that location within the code. This reduces the need to scroll and search through long code segments to find the errors.

■ **New, transparent XML based format**

■ **Direct navigation to web items via a dropdown box when in XHTML editor mode**

■ **Cursor positioned on the item and item is the first line in the editor**

Figure 112: XHTML Editor Features - 3

The code for each web item in the template can be easily found through the use of the dropdown box in the XHTML view. Choosing a web item immediately positions the cursor to the beginning of the code for that item in the editor.

Layout Mode

■ **Easier creation of HTML content via the 'Insert …' option in the context menu**

■ **Increased flexibility and control by providing access to the HTML tags as well as CSS elements, reducing the need for detailed HTML knowledge**
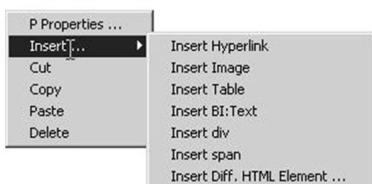
Figure 113: Using HTML from the Layout Tab

HTML code can now be easily inserted into the template while still in the *Layout* mode of the Web Application Designer. From the context menu in the Layout mode, simply choose *Insert* and choose from the available options. The appropriate HTML code or CSS element is automatically inserted at the cursor location into the template. This is a big help to those who do not have a detailed knowledge of HTML syntax.
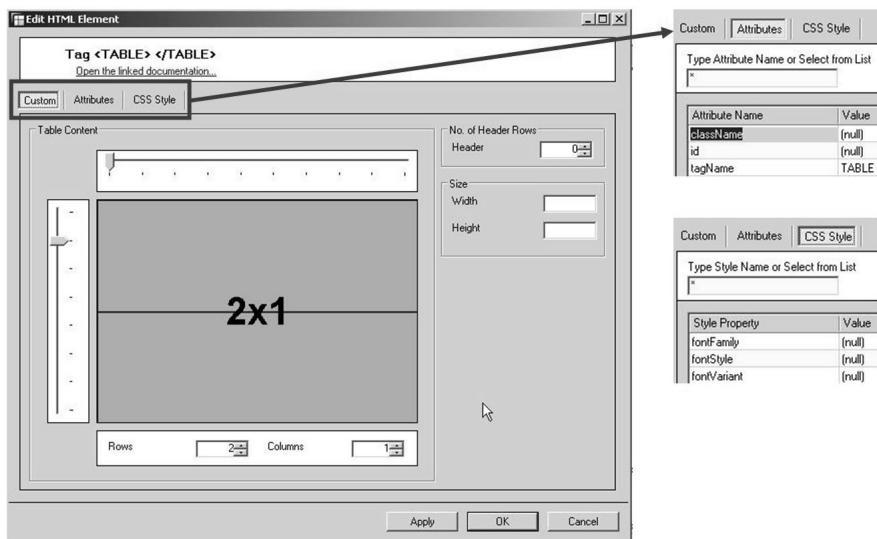
Figure 114: Example: Inserting an HTML Table

Shown above is an example of the dialog resulting from inserting an HTML table (*Insert → Table*). A dialog window opens in the Web Application Designer with three tabs – Custom, Attributes and CSS Style. On the *Custom* tab, the user selects the size of the HTML table grid expressed in rows and columns. Also, the number of header rows and the size of the table, expressed as either a percentage of the default size or in pixels, can be set on this tab. On the *Attributes* tab, the user sets values for all the attributes that can be assigned to the table. Some examples are height, cell spacing and title of the table. On the *CSS Style* tab, settings for the cascading style sheet attributes of the table can be set.

Parameters of both the web template and web items can be overridden at several different levels.
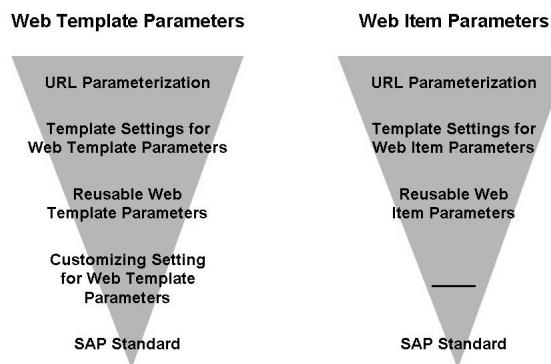
Figure 115: Web Application Parameter Hierarchy

The figure above shows the priority rules when overriding web template parameters and web item settings. You can override all parameters for web items and all web template parameters. At the time of execution, the relevant parameter value is specified for web items corresponding to the sequence: *URL → Web item in web template → Reusable web item → System default.* The priority for web templates is the same except for the addition of the web template parameters defined in the SAP Customizing Implementation Guide as shown above. This setting can be reached in the Implementation Guide with the path *SAP NetWeaver Business Intelligence → Reporting Relevant Settings → BEx Web → Set Settings for Web Templates.*

Parameterized URLs

- **URL Parameterization is used to convert command syntax into URL parameters**
- **Enables use of different URLs to change data providers, web item properties, filters, etc. for the same analysis**
- **More details can be found in <u>http://help.sap.com</u>.**

Example:
In the command wizard, you have set the command for setting Web item parameters for the Web item *ANALYSIS_ITEM_1*. The XHTML view of the Web Application Designer displays the following result (only the INSTRUCTION node is show in this example):
```
<bi:INSTRUCTION >
                <bi:SET_ITEM_PARAMETERS >
                  <bi:cmd_item_parameters type="ANALYSIS_ITEM" >
                    <bi:ALTERNATE_STYLES value="" />
                  </bi:cmd_item_parameters>
                  <bi:TARGET_ITEM_REF value="ANALYSIS_ITEM_1" />
                </bi:SET_ITEM_PARAMETERS>
</bi:INSTRUCTION>
This results in the following URL parameterization:
&BI_COMMAND_1-BI_COMMAND_TYPE=SET_ITEM_PARAMETERS
&BI_COMMAND_1-TARGET_ITEM_REF=ANALYSIS_ITEM_1
&BI_COMMAND_1-INIT_PARAMETERS-ALTERNATE_STYLES=
```

Figure 116: URL Parameterization

As seen from the prior graphic, *URL Parameterization* has the highest priority when changing web template parameters and web item settings. It is therefore a useful technique for embedding specific commands or parameters in the URL itself.

The process for determining the format of the URL parameters is as follows:

1. Insert a web item into your web application with which you can execute commands, such as the *Button Group* Web item.

2. Use the command wizard to create the required command. This enables you to identify the required parameters.

3. In the *XHTML* view of the Web Application Designer, find the command you created.

4. You can use now the parameters of this command for your URL parameterization.

There are specific formatting rules for the parameters.

*   Nested parameters must be converted to a flat notation: i.e.; parent parameters are separated by a hyphen (–), which is placed before the parameters.

*   Entries in lists are marked with the corresponding index as *_N*.

*   The command itself it passed using the command parameter *BI_COMMAND_TYPE=Name*.

*   Command sequences are formed by preceding the actual parameters with the text *bi_command_N*, where "N" is the index of the command in the command sequence.

The *SET_ITEM_PARAMETERS* and *SET_DATA_PROVIDER_PARAMETERS* commands have a special case rule. In both commands, the parent parameter is *INIT_PARAMETERS*. This can be seen in use in the graphic above.

■ You can parameterize a command dynamically at runtime using parameters from Web items or the states of data providers with path referencing.
■ The XML-DataProvider Information web item makes it easy to find the exact path (under BICS_VIEW) that will serve as the replacement path.

Example:
You want to create a Web template in which the user can filter according to an interval for the calendar month. The end of the interval is to be selected from a dropdown box.

Resulting HTML (partial):
```
<bi:CHARACTERISTIC value="0CALMONTH" text="" />
     <bi:RANGE_SELECTION_OPERATOR type="CHOICE"
       value="INTERVAL_SELECTION" >
          <bi:INTERVAL_SELECTION type="COMPOSITE" >
               <bi:INTERVAL_SELECTION_LOW_MEMBER type="CHOICE"
                  value="MEMBER_NAME" >
                      <bi:MEMBER_NAME value="200301" />
               </bi:INTERVAL_SELECTION_LOW_MEMBER>
               <bi:INTERVAL_SELECTION_HIGH_MEMBER type="CHOICE"
                  value="MEMBER_NAME" >
                      <bi:MEMBER_NAME
path="DATA_PROVIDER:DP_2?/BICS_VIEW/SELECTION_STATE/SELECTION/CHARACTERISTICS
/
CHARACTERISTIC[@name='0CALMONTH']/SELECTIONS/SELECTION/MEMBER/@name"/>
               </bi:INTERVAL_SELECTION_HIGH_MEMBER>
          </bi:INTERVAL_SELECTION>
```

Figure 117: Dynamic Parameterization

You can parameterize a command dynamically at runtime using parameters from web items or the states of data providers. You can use path referencing to do this.

Insert the *Data Provider - Information* web item into your web template to determine the path for the data provider replacement. In the source code for the executed web application, search for the XML tree starting with the tag *BICS_VIEW*. If required, insert the web item from which you want to determine the dynamic parameter value into your web template. Set the relevant parameters and then take a look at the tree structure in the *XHTML* view of the Web Application Designer.

In addition to the fixed parameter values in the Web Application Designer, you can also reference all parameters dynamically. The following elements are available as dynamic references:

- current Web item statuses and all parameters published in the Web Application Designer
- navigational states of the data providers
- data provider results

Dynamic parameters are specified using the *path* attribute (and not the *value* attribute). The first part of the attribute value depends on the type of dynamic replacement and references the data provider or the web item. The path is specified after a question mark (?) and follows the *XPath* notation.

> **Hint:** Examples of the proper syntax for the *XPath* notation can be found in the online help at *http://help.sap.com* under the *NetWeaver 2004s → SAP NetWeaver by Key Capability → Information Integration → Business Intelligence → BI Suite: Business Explorer → BEx Web → Web Application Design: BEx Web Application Designer → Web Design API → Dynamic Parameterization of Commands*.

- **The ending year/month for the Calendar Year/Month interval is chosen from the dropdown box.**
- **The Dropdown Box uses a command to set the 'To' value and pass it to the data provider of the Analysis item.**
- **In the XHTML view, the dynamic parameterization coding (*path* statement) inserted in the command sets the chosen value as the 'To' value of the interval.**

Calendar Year/Month:	08.2001	
Calendar Year/Month ⇕	Orders Sold ⇕	Open Orders ⇕
01.2001	1,150.000	2,492.000
02.2001	990.800	1,955.600
03.2001	1,184.600	2,280.900
04.2001	1,154.400	1,951.700
05.2001	1,199.500	2,363.000
06.2001	1,320.300	2,570.300
07.2001	1,411.000	2,409.500
08.2001	1,936.000	1,100.000
Overall Result	10,346.600	17,123.000

Figure 118: Dynamic Parameterization Example

The result of the dynamic parameterization example is shown in the graphic above. The *path* statement inserted on the *XHTML* tab dynamically intercepts the choice made from the dropdown box and uses it as the *To* value in the date range interval specified by the command.

JavaScript Functions

- **Integrates JavaScript into the web template**
- **Key properties:**
  - ◆ Internal Display → Script
    - − **Use the Command Wizard to generate JavaScript**
    - − **Enter or edit the JavaScript manually**
    - − **The resulting JavaScript can be referenced by other web items the same as commands.**

```
**/
function executeJS_SET_SELECTION_STATE_SIMPLE_R( currentState, defaultCommandSequence ){
        //Note: information can be extracted using the parameter 'currentState'
        //       and 'defaultCommandSequence'. In either case create your own object
        //       of type 'sapbi_CommandSequence' that will be sent to the server.
        //       To extract specific values of parameters refer to the following
        //       snippet:
                    var key = currentState.getParameter( 'key' ).getValue();
        //          alert( "Selected key: " + key );
        //
        //       ('PARAM_KEY' refers to any parameter's name)
        //Create a new object of type sapbi_CommandSequence
        var commandSequence = new sapbi_CommandSequence();
        //Create a new object of type sapbi_Command with the command named "SET_SELECTION_STATE_SIMPLE"
        var commandSET_SELECTION_STATE_SIMPLE_1 = new sapbi_Command( "SET_SELECTION_STATE_SIMPLE" );
        //Create parameter TARGET_DATA_PROVIDER_REF_LIST
```

Figure 119: Web Item - Script

To enhance web applications, a *JavaScript API* is now available with which you can also send commands within JavaScript functions to achieve a more flexible design of your Web Application.

JavaScript functions can be inserted into your web application by either directly referencing JavaScript include files stored in the MIME Repository or by using the *Script* web item.

You can store customer-specific scripts in the MIME Repository under the *SAP → BW → Customer → JavaScripts* folder. You may have to create this folder if it does not exist. Then, in the *XHTML* view in the web template you can reference the JavaScript include file using the following syntax: *bwmimerep:///sap/bw/mime/Customer/JavaScripts/myscript.js" type=*

For the *Script* web item, look in the *Properties* screen area and choose the *Script* property details button. On the resulting dialog window you can enter the JavaScript directly, or use the command wizard to automatically generate the JavaScript for executing the command.

Some Web items provide the option of using JavaScript functions. This can be useful, for example, if you want to assemble commands dynamically, or if you use portal eventing and want to link this action to a Web item.

Exercise 11: Advanced XHTML Features

Exercise Objectives

After completing this exercise, you will be able to:

- Create a parameterized URL for use in a web application
- Discuss the XHTML features of the Web Application Designer
- Insert JavaScript code into your web template to extend the functionality

Business Example

On occasion you want to use your HTML and JavaScript coding expertise to directly modify your web applications.

Task 1: Parameterized URLs

You will convert the HTML coding generated on the XHTML tab in the Web Application Designer into a URL with parameters to carry out the command coding.

1. Start the BEx Web Application Designer and create a new web template.

2. To begin, create a new data provider and use the query **T_COQ4** to supply the data.

3. Choose the *Analysis* web item from the Standard group of web items and drag it into your template.

4. Add a *Button Group* web item into your web template. This button will be used to change the query assigned to *DP_1*.

5. Now, configure the *Button Group*. The purpose of the button is to execute a command to change the query for *DP_1* to a different query. You will use the parameters for the command to interpret the parameterized URL that you will insert into your template later. In the *Properties* work area, choose the *BUTTON_GROUP_ITEM_1* web item and make the following settings:

Property/Parameter	Value
Internal Display > List of Buttons (1) > 1 ?? (Button)	`Select the button at the end of the row`
Button > Caption	`Change Query`
Button > Action > Command	`Select the button at the end of the row`
Edit Command window	Select the `All Commands` tab.

Commands for Data Provider > Basic Data Provider Commands	Expand the list and choose **Set Data Provider Parameters**. Then choose **Next**.
Command Target > Data Provider Affected	Select **DP_1**, then choose **Next**.
Data Binding > Query	**Select the button at the end of the row**
Open dialog window	Select query **Country Sales Analysis (T_CTYSALES)**. Then choose **Open**.
Finish the properties of this button	Select **OK** twice.

6. In the Web Application Designer, select the *XHTML* tab. Use the dropdown box at the top of the window to locate the XHTML code for *Button_Group_Item_1*. View the code inside the *bi:INSTRUCTION* tags to locate the important parameters of the command.

> **Hint:** The parameters will be those in all capital letters and with a font color of dark red.

7. Save your web template using the values below.

Field	Value
Description	**GR## XHTML 1**
Technical Name	**GR##XHTML1**

8. Return to the *Layout* tab. You will now construct a parameterized URL to call the same web application, but automatically use the *T_CTYSALES* query instead of the original query assigned to the data provider *DP_1*.

First, copy the URL for the web template into the Clipboard. You can use the *Web Template → Publish* menu path for this. The paste the URL into your web template below the other two web items using the *Edit → Paste* menu path.

Onto the end of the URL, add the following BI Command statements as one continuous line. Notice the use of the command parameters you located in the button command.

```
&BI_COMMAND_1-BI_COM-
MAND_TYPE=SET_DATA_PROVIDER_PA-
RAMETERS&BI_COMMAND_1-TAR-
GET_DATA_PROVIDER_REF=DP_1&BI_COMMAND_1-INIT_PA-
RAMETERS-INITIAL_STATE=QUERY&BI_COMMAND_1-INIT_PA-
RAMETERS-INITIAL_STATE-QUERY=T_CTYSALES
```

9. Save your web template again to preserve your changes. Then execute the web application and click on the URL to see it change the query used to populate the *Analysis* web item.

10. As a final test, return to the Web Application Designer and copy the entire URL from the web template into the Clipboard using **Ctrl+C**. Then open a separate web browser window from the *Start → Programs → Internet Explorer* path. Paste the URL into the *Address* window of the browser using **Ctrl+V** and then select *Enter*. After logging onto the portal, the web template will be executed, but it will start with the results from query *T_CTYSALES* because of the parameterized URL.

Task 2: JavaScript Functions

Using the *Script* web item, you will enhance your web template with JavaScript. You want to provide a dropdown box where the user will select the ending month of a time interval and have the results in the analysis area filtered accordingly.

1. Start the BEx Web Application Designer and create a new web template.

2. To begin, create two new data providers and use the query **T_CO2001** to supply the data for both.

3. Choose the *Dropdown Box* web item from the Standard group of web items and drag it into web template.

4. Choose the *Analysis* web item from the Standard group of web items and drag it into web template to the right of the *Dropdown Box*.

    Assign data provider *DP_1* to the *Analysis* web item.

5. Now, configure the *Dropdown Box*. You want the dropdown list to show all values of *0CALMONTH* that are used in the query. In the *Properties* work area, choose the *Dropdown Box* and make the following settings:

Property/Parameter	Value
Data Binding > Data Binding Type	Select **Char/Structure Memeber**.
Data Binding > %NM% (Selection of Characteristic)	**Select the button at the end of the row**

Continued on next page

 SAP

Selection of Characteristic > Data Provider	Select **DP_2**.
Selection of Characteristic > Characteristic	**Select the button at the end of the row**
Valuehelp Selector window	Select **0CALMONTH**. Then choose **OK**.
Selection of Characteristic > Additional Action	Select **Script Function**. Then choose **OK**. (You will return and select the generated JavaScript in a later step.)

6. Choose the *Script* web item from the Miscellaneous group of web items and drag it into web template to the right of the *Dropdown Box*.

7. Configure the *Script* web item. You will use the Command Wizard to help generate the JavaScript. The command used will be the *SET_SELECTION_STATE_SIMPLE* command for filtering data providers. Here you will generate a command for providing a value interval for the InfoObject *0CALMONTH*.

    In the *Properties* work area, choose the *Script* web item and make the following settings:

Property/Parameter	**Value**
Internal Display > Script	**Select the button at the end of the row**
Script Editing window	Select **Create with Wizard** button.
Edit Command window	Select the **All Commands** tab.
Commands for Data Provider > Data Provider Commands for Filter Values	Select **Set Filter Value for a Characteristic**. Then choose **Next**.
Command Target > Data Provider Affected	Select **DP_1**. Then choose **Next**.
Data Binding > Characteristic	**Select the button at the end of the row.**
Valuehelp Selector window	Choose **0CALMONTH**. Then choose **OK**.
Command Specific Parameters > Operator	Select **Interval Selection**.

Continued on next page

Command Specific Parameters > Interval Selection	**Select the button at the end of the row.**
Interval Selection > Low Member > Member Name	**Select the button at the end of the row.**
Select Values for Calendar Year/Month window	Select the value **January 2001** and then choose **OK**.
Interval Selection > High Member > Member Name	**Select the button at the end of the row.**
Select Values for Calendar Year/Month window	Select the value **December 2001** and then choose **OK**. (This value will be replaced in the JavaScript later.)
Finish the command parameters	Select **OK** twice to return to the JavaScript listing.

8.   Now you will edit your newly-generated JavaScript in order to allow the value chosen from the dropdown box to be used as the 'High' or 'To' month and year in the interval range.

To do this, carry out the following changes:

Property/Parameter	**Value**
Locate the *var key = currentState.....* statement in the JavaScript	Remove the *//* characters at the beginning of the statement.
Remain on the same *var key = currentState...* statement.	Replace *PARAM KEY* with **'key'** in the statement. (Include the single quotes.)
Locate the *INTERVAL_SELEC-TION_HIGH_MEMBER* section of your JavaScript.	Replace the *"200112"* value in that section with **key** (no quote marks).
Finish the command parameters.	Select **OK** to return to the web template.

9.   Return to the *Dropdown Box* to finish setting the parameters for using the modified JavaScript. In the *Properties* work area, choose the *Dropdown Box* and make the following settings:

Property/Parameter	Value
Data Binding > DP_2 (Selection of Characteristic)	`Select the button at the end of the row`
Selection of Characteristic > Script Function	`Select the button at the end of the row`
Choose Function window	Select your JavaScript function (JS_SET_SELECTION_STATE_SIMPLE_R) from the dropdown list and then select **OK**.
Edit Parameter window	Select **OK**.

10. Save your web template using the values below.

Field	Value
Description	`GR## XHTML 2`
Technical Name	`GR##XHTML2`

Then execute the web application and test the operation of the dropdown box.

Solution 11: Advanced XHTML Features

Task 1: Parameterized URLs

You will convert the HTML coding generated on the XHTML tab in the Web Application Designer into a URL with parameters to carry out the command coding.

1. Start the BEx Web Application Designer and create a new web template.

    a) Start the BEx Web Application Designer using the path *Start → Programs → Business Explorer → Web Application Designer*.

    b) Logon to the system with the user ID and password provided by your instructor.

    c)   Select *"Create new Web Template"*.

2. To begin, create a new data provider and use the query **T_COQ4** to supply the data.

    a) Double click on *New Data Provider* in the design area.

    b) Select the data provider type *Query* using the radio button.

    c) Enter **T_COQ4** as the name of the query and select *OK*.

3. Choose the *Analysis* web item from the Standard group of web items and drag it into your template.

    a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

    b) Drag the *Analysis* web item to your web template as the first web item.

4. Add a *Button Group* web item into your web template. This button will be used to change the query assigned to *DP_1*.

    a) In the Web Items area of the Web Application Designer, open the *Advanced* web item category.

    b) Drag a *Button Group* web item to your web template and drop it into your web template.

5. Now, configure the *Button Group*. The purpose of the button is to execute a command to change the query for *DP_1* to a different query. You will use the parameters for the command to interpret the parameterized URL that you will insert into your template later. In the *Properties* work area, choose the *BUTTON_GROUP_ITEM_1* web item and make the following settings:

Property/Parameter	Value
Internal Display > List of Buttons (1) > 1 ?? (Button)	**Select the button at the end of the row**
Button > Caption	**Change Query**
Button > Action > Command	**Select the button at the end of the row**
Edit Command window	Select the **All Commands** tab.
Commands for Data Provider > Basic Data Provider Commands	Expand the list and choose **Set Data Provider Parameters**. Then choose **Next**.
Command Target > Data Provider Affected	Select **DP_1**, then choose **Next**.
Data Binding > Query	**Select the button at the end of the row**
Open dialog window	Select query **Country Sales Analysis (T_CTYSALES)**. Then choose **Open**.
Finish the properties of this button	Select **OK** twice.

a)   In the *Properties* work area for the *BUTTON_GROUP_ITEM_1* web item, set the properties for the fields as shown in the table above.

b)   When finished, select *OK* twice to return to your template.

6. In the Web Application Designer, select the *XHTML* tab. Use the dropdown box at the top of the window to locate the XHTML code for *Button_Group_Item_1*. View the code inside the *bi:INSTRUCTION* tags to locate the important parameters of the command.

> **Hint:** The parameters will be those in all capital letters and with a font color of dark red.

a) In the *Template* screen area, select the *XHTML* tab.

b) From the dropdown box underneath the *XHTML* tab, select **BUTTON_GROUP_ITEM_1**. The cursor will be placed automatically on the beginning tag of the selected web item.

c) After locating the *bi:INSTRUCTION* tag, you will find the following parameters used to define the command within the button:

> SET_DATA_PROVIDER_PARAMETERS
>
> TARGET_DATA_PROVIDER_REF
>
> INITIAL_STATE
>
> QUERY

7. Save your web template using the values below.

Field	Value
Description	`GR## XHTML 1`
Technical Name	`GR##XHTML1`

a) In the Web Application Designer, click on the *Save* 💾 icon.

b) Enter the values shown above for the *Description* and *Technical Name* fields.

c) Select *OK* when finished.

8. Return to the *Layout* tab. You will now construct a parameterized URL to call the same web application, but automatically use the *T_CTYSALES* query instead of the original query assigned to the data provider *DP_1*.

First, copy the URL for the web template into the Clipboard. You can use the *Web Template → Publish* menu path for this. The paste the URL into your web template below the other two web items using the *Edit → Paste* menu path.

Onto the end of the URL, add the following BI Command statements as one continuous line. Notice the use of the command parameters you located in the button command.

&BI_COMMAND_1-BI_COM-
MAND_TYPE=SET_DATA_PROVIDER_PA-
RAMETERS&BI_COMMAND_1-TAR-
GET_DATA_PROVIDER_REF=DP_1&BI_COMMAND_1-INIT_PA-
RAMETERS-INITIAL_STATE=QUERY&BI_COMMAND_1-INIT_PA-
RAMETERS-INITIAL_STATE-QUERY=T_CTYSALES

    a)   Select the *Layout* tab in the *Template* screen area.

    b)   Copy the URL of the web template to the Clipboard using menu path
          Web Template → Publish → Copy URL to the Clipboard.

    c)   Paste the URL into your web template below the other two web items
          using **Ctrl+V** or the menu path *Edit → Paste*.

    d)   At the end of the URL that is now in your template, add the BI
          Command statements listed in the exercise above. Although there are
          four statements, enter them as one continuous text string.

9. Save your web template again to preserve your changes. Then execute the
    web application and click on the URL to see it change the query used to
    populate the *Analysis* web item.

    a)   In the Web Application Designer, click on the *Save* 💾 icon.

    b)   Select the *Execute* 🔷 icon to execute your web application.

    c)   The web application will first display the results of query *T_COQ4*.
          Click on the URL and the results from query *T_CTYSALES* will be
          displayed.

10. As a final test, return to the Web Application Designer and copy the entire
    URL from the web template into the Clipboard using **Ctrl+C**. Then open
    a separate web browser window from the *Start → Programs → Internet
    Explorer* path. Paste the URL into the *Address* window of the browser

Continued on next page

using **Ctrl+V** and then select *Enter*. After logging onto the portal, the web template will be executed, but it will start with the results from query *T_CTYSALES* because of the parameterized URL.

a) Return to the Web Application Designer. In the *Template* screen area, select the entire text of the URL by dragging over it with the mouse, then copy it to the Clipboard by using the key sequence **Ctrl+C**.

b) From the Citrix Start menu, use the following path: *Start → Programs → Internet Explorer*.

c) In the *Address* window of the browser, paste the URL into the browser using the key sequence **Ctrl+V**. Then select the *Enter* key.

d) Logon to the NetWeaver Portal using your user ID and password.

e) The web application will be displayed, but starting with the query *T_CTYSALES* as specified in the parameterized URL.

Task 2: JavaScript Functions

Using the *Script* web item, you will enhance your web template with JavaScript. You want to provide a dropdown box where the user will select the ending month of a time interval and have the results in the analysis area filtered accordingly.

1. Start the BEx Web Application Designer and create a new web template.

a) Start the BEx Web Application Designer using the path *Start → Programs → Business Explorer → Web Application Designer*.

b) Logon to the system with the user ID and password provided by your instructor.

c) Select *"Create New Web Template"*.

2. To begin, create two new data providers and use the query **T_CO2001** to supply the data for both.

a) Double click on *New Data Provider* in the design area.

b) Select the data provider type *Query* using the radio button.

c) Enter **T_CO2001** as the name of the query and select *OK*.

d) Create the second data provider in the same way as the first and use the same query to supply the data.

3. Choose the *Dropdown Box* web item from the Standard group of web items and drag it into web template.

a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

b) Drag the *Dropdown Box* web item into your web template.

4. Choose the *Analysis* web item from the Standard group of web items and drag it into web template to the right of the *Dropdown Box*.

   Assign data provider *DP_1* to the *Analysis* web item.

   a) In the Web Items area of the Web Application Designer, open the *Standard* web item category.

   b) Drag the *Analysis* web item into your web template and drop it to the right of the *Dropdown Box* web item.

   c) In the *Properties* screen area select *ANALYSIS_ITEM_1* and then select data provider **DP_1** on the *General* tab.

5. Now, configure the *Dropdown Box*. You want the dropdown list to show all values of *0CALMONTH* that are used in the query. In the *Properties* work area, choose the *Dropdown Box* and make the following settings:

Property/Parameter	Value
Data Binding > Data Binding Type	Select **Char/Structure Memeber**.
Data Binding > %NM% (Selection of Characteristic)	**Select the button at the end of the row**
Selection of Characteristic > Data Provider	Select **DP_2**.
Selection of Characteristic > Characteristic	**Select the button at the end of the row**
Valuehelp Selector window	Select **0CALMONTH**. Then choose **OK**.
Selection of Characteristic > Additional Action	Select **Script Function**. Then choose **OK**. (You will return and select the generated JavaScript in a later step.)

   a) In the *Properties* work area for the *DROPDOWN_ITEM_1* web item, set the properties for the fields as shown in the table above.

   b) When finished, select *OK*.

6. Choose the *Script* web item from the Miscellaneous group of web items and drag it into web template to the right of the *Dropdown Box*.

   a) In the Web Items area of the Web Application Designer, open the *Miscellaneous* web item category.

   b) Drag the *Script* web item into your web template and drop it to the right of the *Dropdown Box* web item.

   Continued on next page

7. Configure the *Script* web item. You will use the Command Wizard to help generate the JavaScript. The command used will be the *SET_SELECTION_STATE_SIMPLE* command for filtering data providers. Here you will generate a command for providing a value interval for the InfoObject *0CALMONTH*.

In the *Properties* work area, choose the *Script* web item and make the following settings:

Property/Parameter	Value
Internal Display > Script	**Select the button at the end of the row**
Script Editing window	Select **Create with Wizard** button.
Edit Command window	Select the **All Commands** tab.
Commands for Data Provider > Data Provider Commands for Filter Values	Select **Set Filter Value for a Characteristic**. Then choose **Next**.
Command Target > Data Provider Affected	Select **DP_1**. Then choose **Next**.
Data Binding > Characteristic	**Select the button at the end of the row.**
Valuehelp Selector window	Choose **0CALMONTH**. Then choose **OK**.
Command Specific Parameters > Operator	Select **Interval Selection**.
Command Specific Parameters > Interval Selection	**Select the button at the end of the row.**
Interval Selection > Low Member > Member Name	**Select the button at the end of the row.**
Select Values for Calendar Year/Month window	Select the value **January 2001** and then choose **OK**.

Interval Selection > High Member > Member Name	`Select the button at the end of the row.`
Select Values for Calendar Year/Month window	Select the value `December 2001` and then choose **OK**. (This value will be replaced in the JavaScript later.)
Finish the command parameters	Select **OK** twice to return to the JavaScript listing.

a) In the *Properties* work area for the *SCRIPT_ITEM_1* web item, set the properties for the fields as shown in the table above.

b) When finished, select *OK* twice to return to the JavaScript Editing window.

8. Now you will edit your newly-generated JavaScript in order to allow the value chosen from the dropdown box to be used as the 'High' or 'To' month and year in the interval range.

To do this, carry out the following changes:

Property/Parameter	**Value**
Locate the *var key = currentState.....* statement in the JavaScript	Remove the *//* characters at the beginning of the statement.
Remain on the same *var key = currentState...* statement.	Replace *PARAM KEY* with `'key'` in the statement. (Include the single quotes.)
Locate the *INTERVAL_SELEC-TION_HIGH_MEMBER* section of your JavaScript.	Replace the *"200112"* value in that section with **key** (no quote marks).
Finish the command parameters.	Select **OK** to return to the web template.

a) In the *Script Editing* work area for the generated JavaScript, make the changes to the coding as shown in the table above.

b) When finished, select *OK* to return to the web template.

9. Return to the *Dropdown Box* to finish setting the parameters for using the modified JavaScript. In the *Properties* work area, choose the *Dropdown Box* and make the following settings:

Property/Parameter	Value
Data Binding > DP_2 (Selection of Characteristic)	`Select the button at the end of the row`
Selection of Characteristic > Script Function	`Select the button at the end of the row`
Choose Function window	Select your JavaScript function (JS_SET_SELECTION_STATE_SIMPLE_R) from the dropdown list and then select **OK**.
Edit Parameter window	Select **OK**.

a) In the *Properties* work area for the *DROPDOWN_ITEM_1* web item, set the properties for the fields as shown in the table above.

b) When finished, select *OK*.

10. Save your web template using the values below.

Field	Value
Description	`GR## XHTML 2`
Technical Name	`GR##XHTML2`

Then execute the web application and test the operation of the dropdown box.

a) In the Web Application Designer, click on the *Save* 💾 icon.

b) Enter the values shown above for the *Description* and *Technical Name* fields.

c) Select *OK* when finished.

d) Select the *Execute* 🗗 icon to execute your web application.

e) The web application will first display the results of query *T_CO2001*. Select an ending month from the dropdown box to filter the interval displayed.

Lesson Summary

You should now be able to:

- Use the features of the XHTML editor
- Generate parameterized URLs
- Use JavaScript functions to enhance web applications

Unit Summary

You should now be able to:

- Access the Web Application Designer
- Identify the main components of the tool
- Create a simple web application
- Change the properties of web items
- Execute the web application and navigate in the results
- Align objects as desired in the web application
- Add texts, images and tabs to your web application
- List the impact of portal themes on your web application
- Add a range of content to your web application with the Report, Ticker and Information Field web items
- Enhance the navigation capabilities of your web applications with the Navigation Pane, Exceptions, Conditions, Checkbox, Radio Buttons and Context Menu web items
- Present data in chart format in your web applications
- Describe the process of geocoding for characteristics
- Present data in map format in your web applications
- Explain the basic operation of the Command Wizard
- Insert commands into the web application using various web items
- Use sequences of commands to accomplish multiple tasks
- Efficiently develop web applications by reusing modules such as header and footer web templates
- Use the BI Pattern Wizard to modify the SAP-delivered patterns.
- Use the features of the XHTML editor
- Generate parameterized URLs
- Use JavaScript functions to enhance web applications

Test Your Knowledge

Answers

Unit 5

BEx Broadcaster Advanced Features

Unit Overview

Broadcasting capabilities have expanded greatly in NetWeaver 2004s and in this unit you will explore all the capabilities available for your use with this important tool.

Unit Objectives

After completing this unit, you will be able to:

- Describe the basic features of the BEx Broadcaster including how to broadcast results via email or to the portal
- Extend your usage of the Broadcaster to take advantage of more advanced methods such as data bursting, value sets and pre-filling OLAP cache.
- Setup a broadcast that only distributes data if exceptions have been raised
- Define various scheduling settings to ensure broadcasts are executed and distributed at required times to meet business requirements

Unit Contents

Lesson: Recap of Key Features of Boadcaster

Lesson Overview

This short lesson provides a recap of the basic features of the Broadcaster as discussed in the BW305 class.

Lesson Objectives

After completing this lesson, you will be able to:

- Describe the basic features of the BEx Broadcaster including how to broadcast results via email or to the portal

Business Example

Before you start working on the advanced features of the Broadcaster you need to remind yourself of the basic capabilities

BEx Broadcaster Key Features

Information Broadcasting is a flexible function of the Business Explorer which provides to facility to distribute reporting results out to various audiences whether they are regular SAP BI users or not. Broadcasting is not restricted to one method. You can choose the source of the data, the format of the data, the target audience, the broadcast channel and of course the timing of the broadcast. You can even use a combination of these for a single broadcast. There is also a Wizard to help you work through the settings if you are an occasional user. Broadcasts can be defined as a one time distribution or they can be set to distribute result data on a regular basis at predetermined times. The data can be either precalculated or the broadcast can send a link to the online data. Users may decide if they want to subscribe to broadcasts, (or perhaps even unsubscribe).

The diagram below illustrates the positioning of the Information Broadcaster within NetWeaver BI.
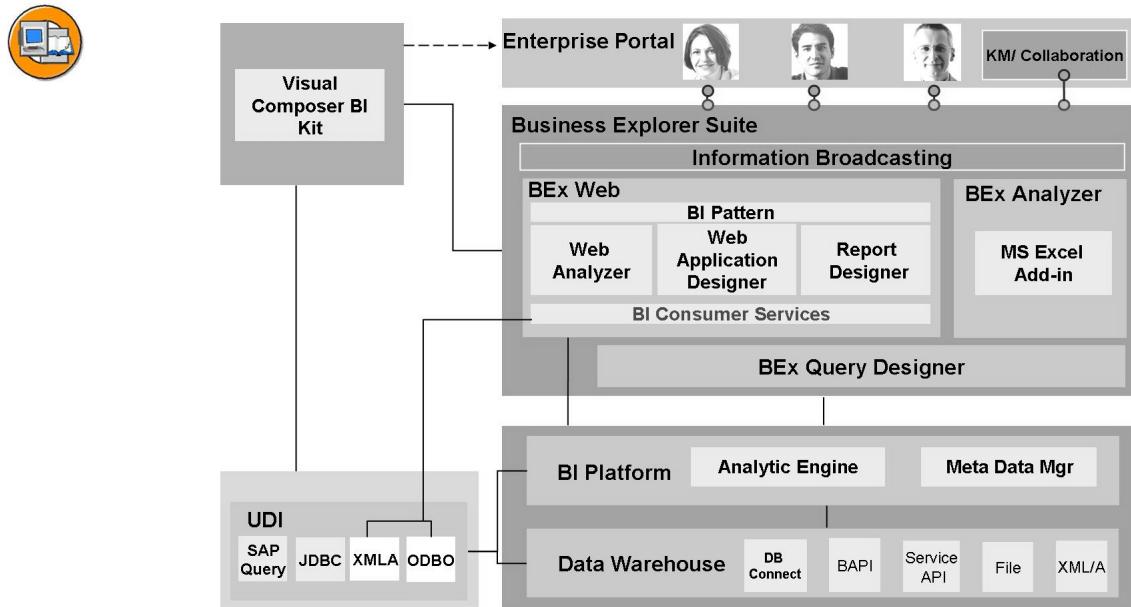
Figure 120: Broadcaster Architecture

There are many scenarios in which the Broadcaster can play a key role. Broadcasting can be executed as a one time event, regular scheduled event, or even used just to alert personnel of any critical situation which needs attention. The Broadcaster can also be used to improved the performance of the BEx environment by using the tools to precalculate reports, fill the query cache

Broadcasting runs across the entire BEx Suite and uses a consistent interface for the creation of broadcast settings regardless of the tool in which the broadcast is being defined. Broadcast data can come from any SAP BI data source, but also non-SAP data sources can be used in a broadcast. Source data can be read in real time at the time of the broadcast, the data does not need to reside in BI to be accessible by the Broadcaster.
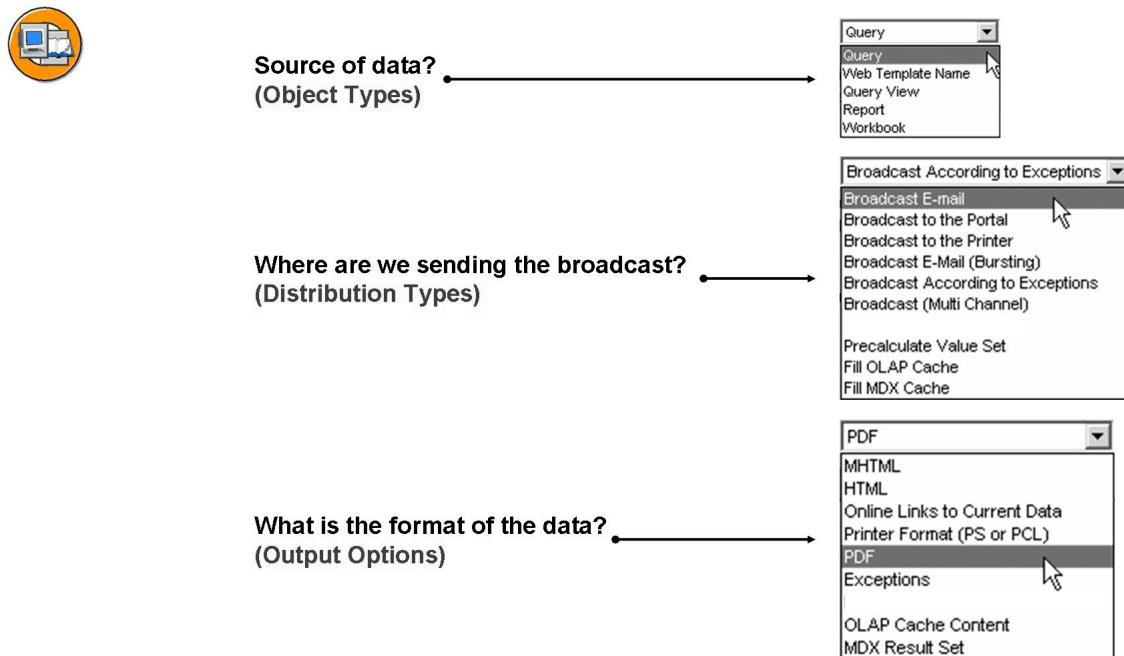
Figure 121: Flexible Broadcaster Choices

A key feature of the Broadcaster is the tight integration with the NetWeaver Portal. Broadcasts can be sent to the portal where users can take advantage of a range of Knowledge Management tools such as:

- Feedback
- Discussion
- Rating
- Subscription
- Search
- Subscription
- Download
- Collaboration
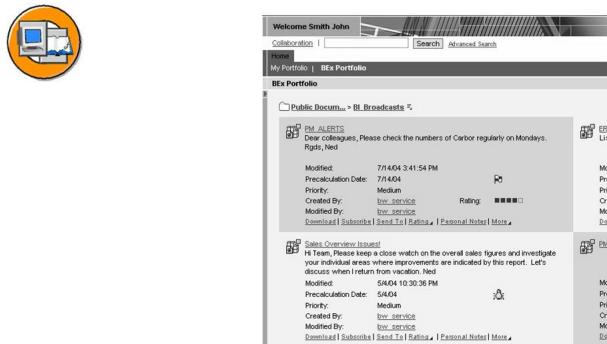
KM Services enabled on top of any broadcasted BI data as well as on any document attached to BI data
Use KM Services for:

- Subscription
- Feedback
- Discussion
- Collaboration
- Rating
- TREX search
- Notes
- Download

Figure 122: Portal Integration

A role is delivered with NetWeaver BI which provides easy access to Broadcasts via the Portal. Any portal user who is assigned to this role will be able to access their broadcasts easily via the Portal.

The BEx Information Broadcaster can be accessed from various BEx reporting tools. The broadcaster can also be accessed via the Portal through the delivered BI Role. Broadcast settings can be created and inserted into BI reports by the report developer during the development phase. The report user can also create broadcast settings during the run time as the need arises.

- BEx Query Designer
- BEx Analyzer (workbook)
- BEx Web Analyzer
- Portal
- BEx Report Designer
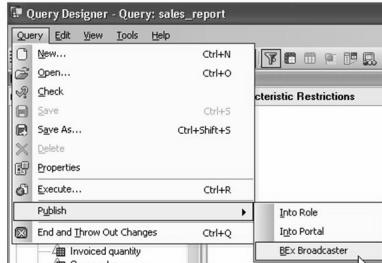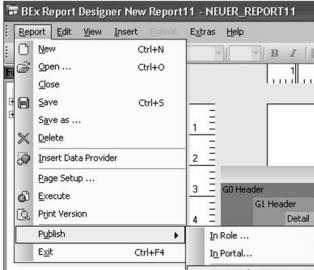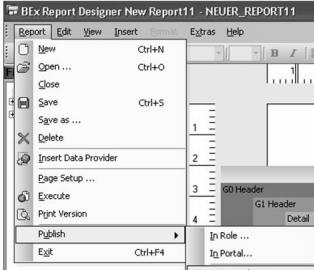
Figure 123: Accessing the Broadcaster

Lesson Summary

You should now be able to:

- Describe the basic features of the BEx Broadcaster including how to broadcast results via email or to the portal

Lesson: More Distribution Types and Output Formats

Lesson Overview

In this lesson you will thoroughly explore additional distribution types and output formats.

Lesson Objectives

After completing this lesson, you will be able to:

- Extend your usage of the Broadcaster to take advantage of more advanced methods such as data bursting, value sets and pre-filling OLAP cache.

Business Example

Your organization has been using the Broadcaster to distribute results via email to BI users. You now want to learn how to create settings for more advanced scenarios such as data bursting and value set creation

Broadcast to Printer

If you need hard copies of business reports and you would like to schedule these in the background then use this distribution type. A spool request is automatically generated for the report you require. There are two output format options to choose from for the distribution type **Broadcast to Printer**:

- Printer Format (Printer Control Language or PostScript Language)
- PDF

Note: Before PDF files can be created, the system administrator must configure and manage the Adobe document services.
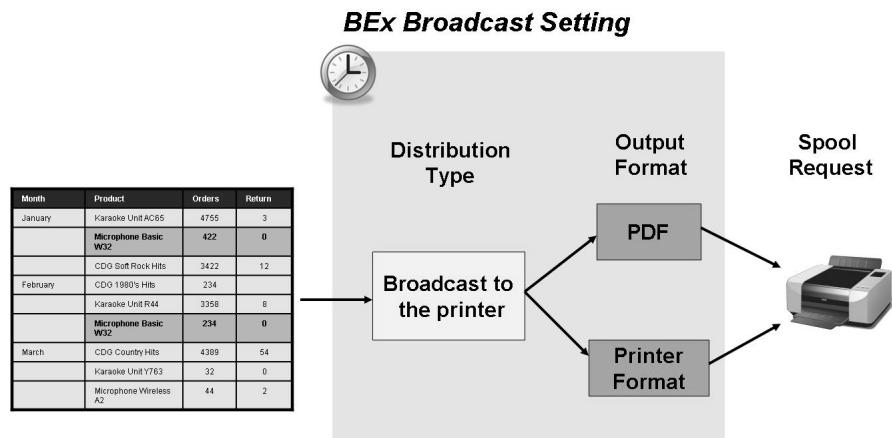
- Printer formats can be Printer Control Language (PCL) or Post Script (PS)
- Only printer devices which support the chosen format are available for selection
- Not all web items are compatible with pdf output format

Figure 124: Broadcast to Printer

When you are selecting your printer device you will only be allowed be prompted with the devices which supports the format you have chosen.

In addition to choosing your output format you can also choose various print parameters on the *Layout* tab:

Parameter	Purpose
Fit to Page Width	The BI application is fit to the width of the page in the print version.
Fit to Page	The BI application is fit to the size of the page
Poster	The BI application is printed in the form of a poster in which individual areas of the BI application are printed on different pages. You can repeat lead columns and column headings
Theme	Choose to use either the SAP default theme or Black and White
Paper Format	You can choose between DIN A4 format and letter format
Orientation	You can choose between portrait format or landscape format
Margins	You can specify margins for the Top, Bottom, Left, and Right in mm or in inches

On the tab *Header/Footer* you can choose to display various parameters such as :

Parameter	Purpose
Page 1	Show page number
Page 1 of ?	Show page number and total number of pages
Date	Show date of report execution
Date, Time	Show date and time of report execution
Date Time Page 1 of ?	Show date, time and page number
Free Text	Select this then add you own text in the text field OR select this and then select one of the system parameters by pressing the *Text Variables* icon.
Time	Show time of report execution

The remaining parameters within the broadcast setting are the same for many other distribution types.

Broadcast Value Sets

The distribution type Precalculated Value Set allows you to fill variables of the type precalculated value sets with values for characteristic values in the background. The precalculated value sets are then available as variable values in BEx queries. Typical examples where the precalculated value set is used include:

*   When a complex selection is required which could take some time to calculate
*   When a specific selection is required which will be frequently re-used as input filters to other queries

Precalculated value sets can be scheduled to run in the background, this can improve system performance by ensuring complex selection take place offline. When the broadcast is triggered the query specified in the broadcast is executed and depending on your choice of characteristic in the broadcast every value is captured and saved to a permanent storage table known as the value set. A values set contain exactly one characteristic but you can define a separate broadcast for each characteristic you wish to capture. Each time a broadcast is executed for a value set the characteristic values are simply overwritten in the value set each time, there is no recording of historical values. If you wanted to keep a value set you would simply make sure you do not re-execute the same one.

A value set is not tied to a single query. This means you can re-use the value set as the input selection in any number of queries, and of course, the query can be based on any InfoProvider. For example you could select the top 3 products by calculating the results from the DataStore object, and then you could use these result as the filters for a query based on the InfoCube.

2007/Q2   **339** SAP

In order to set up a scenario to generate and use a value set proceed as follows:

1. Create a query which generates the characteristic values you require.
2. Create a broadcast setting for distribution type *Precalculate Value Set* and assign the query from step 1, specify the characteristic you would like to capture and also specify the name of the value set.
3. Schedule to broadcast.
4. Create a second query which uses a variable to filter the incoming characteristic values.
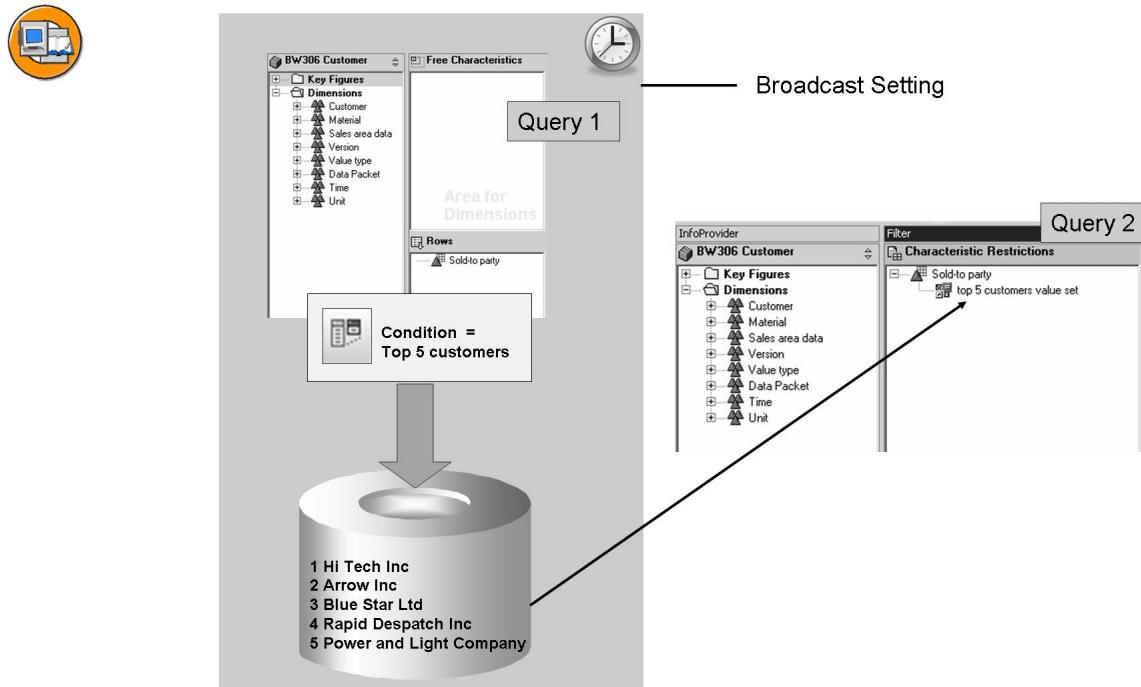
Figure 125: Broadcast Value Set

As a result of background processing, you can use the precalculated value set in a characteristic variable. You create a characteristic value variable in the Query Designer using the variable editor. Under Details you determine that the variable represents a precalculated value set. The value of this variable is the technical name of the value set specified in the broadcast setting. You can decide if you want to be prompted to enter the value set name (you may have multiple value sets holding different results from different time periods). If this is the case you would make the variable input ready so the user can select the required value set at query runtime.

Bursting

The distribution type Broadcast E-Mail (Bursting) allows you to broadcast
BI objects (queries, query views, Web templates, reports, or workbooks) as
precalculated documents or online links by email to recipients that you determine
on the basis of master data. If you have maintained the addresses of the recipients
in the master data, you can select a characteristic in whose attributes the e-mail
addresses or the BI user names are maintained, and the system can determine the
recipients from this information. This type of distribution is known as bursting.
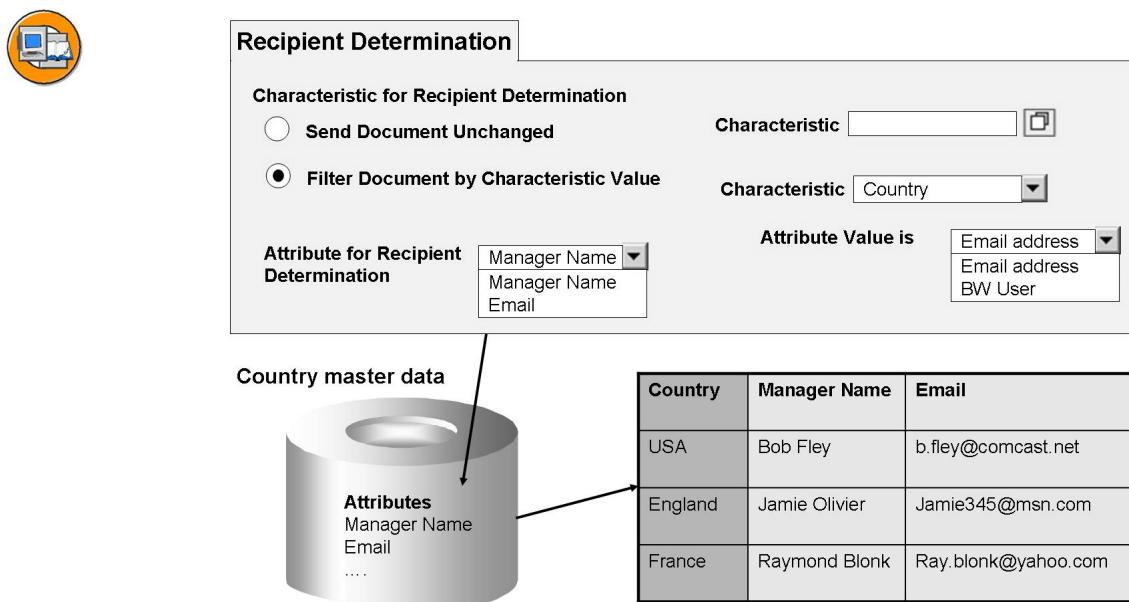
Figure 126: Bursting (email)

If required, you can also specify that the document is to be filtered for each
recipient by their relevant characteristic values. If you choose to do this, the
selected characteristic that you want to broadcast must be contained in the BI
object. Every recipient then receives the document with the values that correspond
to his or her area. For example, you can define the broadcast setting so that the
monthly sales report is sent to all the branch managers. Every branch manager
receives the sales report with the sales data from his or her branch. Bursting is
useful when you need to distribute business results to a large number of recipients.
It is also useful when the recipients of the results change often. Bursting is
basically the same as the distribution type *Broadcast Email* except with bursting
the recipient addresses are stored externally from the broadcast setting, as master
data attributes.

> **Hint:** Don't forget to keep the recipient addresses up to date in the master
> data attributes by scheduling regular updates of master data !

Recipient Determination

Parameter	Purpose
Characteristic for Recipient Determination	
Send Document Unchanged	The document is not adapted to suit the recipient. Enter the technical name of the required characteristic for which the recipient is maintained as an attribute either manually or by choosing the *Selection* icon. The characteristic can be any characteristic in your BI system and does not have to be contained in the BI object that you want to distribute. You must also make a selection of the characteristic values for the infoobject you have chosen. That is done under the parameter xxx
Filter Document by Characteristic Value	The document is filtered by the characteristic value with which the recipient is identified. Each recipient receives the document filtered by the relevant characteristic values. Choose the required characteristic for which the recipient is maintained as an attribute. You can only select from those characteristics that are included in the query or Data Provider of the Web Application or the report. With workbooks, you cannot filter the document by characteristic values.
Generate Document with Variable Value	The characteristic value used to determine the recipient is used as a variable entry when the document is generated. Select the variable you want to use. You can select from those characteristic value variables that are included in the query or Data Provider of the Web Application, the report, or the workbook. The system only displays this setting if the BI object (query, Web template, report, or workbook) contains variables for characteristic values.
Attribute for Recipient Determination	Choose the attribute for which the recipient is maintained. Under Attribute Value Is, specify whether the recipient is to be identified using the e-mail address or the user name.

Selecting Characteristic Values	
Using Following Selection	Under *Selection Create*, specify the characteristic values for each of which an e-mail recipient is to be determined.
Using a Control Query	Under Control Query, specify the query you want from which the characteristic values are selected, either manually or by choosing Selection. A document is calculated and broadcast for each characteristic value of the selected characteristic. Under Variant you can select an existing variant for a query. Variables of the query that are ready for input can then by filled with values

User/Language

Parameter	Purpose
Authorization User	If specified, this user will be used as the basis for the authorization check. Only result values this user is allowed to see will be presented in the broadcast.
Language	If specified the broadcast will be presented in this language, if left blank then the language of the user creating the broadcast will be used.

Text

Parameter	Purpose
Subject:	Enter a subject line for the e-mail manually or use a text variable by choosing with the quick info text Attach Text Variable. The text can contain up to 50 characters.
Importance	Select the importance level for the e-mail (low, medium, high).
Contents	Enter text for the contents of the e-mail manually or use a text variable by choosing with the quick info text Attach Text Variable. When online links are sent, the system automatically adds a link to the text of the e-mail if the text does not contain the link using the PR_ONLINE_LINK variable.

General Precalculation

Parameter	Purpose
Portal Theme	Choose the theme which will be used to format the results. Themes can be maintained in the Theme Editor within the portal.
Only Upon Data Change	If you check this box the broadcast will only be sent if the data has changed since the last time the broadcast was sent.
Exceptions	
No Include	Do not use the exceptions in the query or query view for checking whether to send the broadcast.
Only Precalculate at minimum alert level	If the broadcast is based on a query or query view which contains exceptions then you can specify that the broadcast should only be sent if the alert level (exception level) has been met. You specify the alert level here

Filter Navigation

Parameter	Purpose
No Filter	Broadcast the report unfiltered.
Filter by selected characteristic	Enter the characteristic from the query to use as the filter, when you select a characteristic you will then see a link to create the values for filtering. You will also have the option to create an unfiltered view for each recipient.
Filter by Control Query	Enter the name of a query which will be used to determine the characteristic values used for filtering the broadcast result. For example this query might select the worst performing sales regions on the basis of a condition in query a. These regions will then be sent a detailed breakdown of their results in the broadcast based on query b.

Fill OLAP Cache

This distribution type allows you to precalculate queries and fill the OLAP cache with the generated query data. If the users call BEx web applications, queries, reports or workbooks that are based on this data, the access time is significantly reduced and the workload on the application server is considerably less. This is because the data is retrieved from the application server memory (the cache) rather than from the database server, where disk reads would be required.
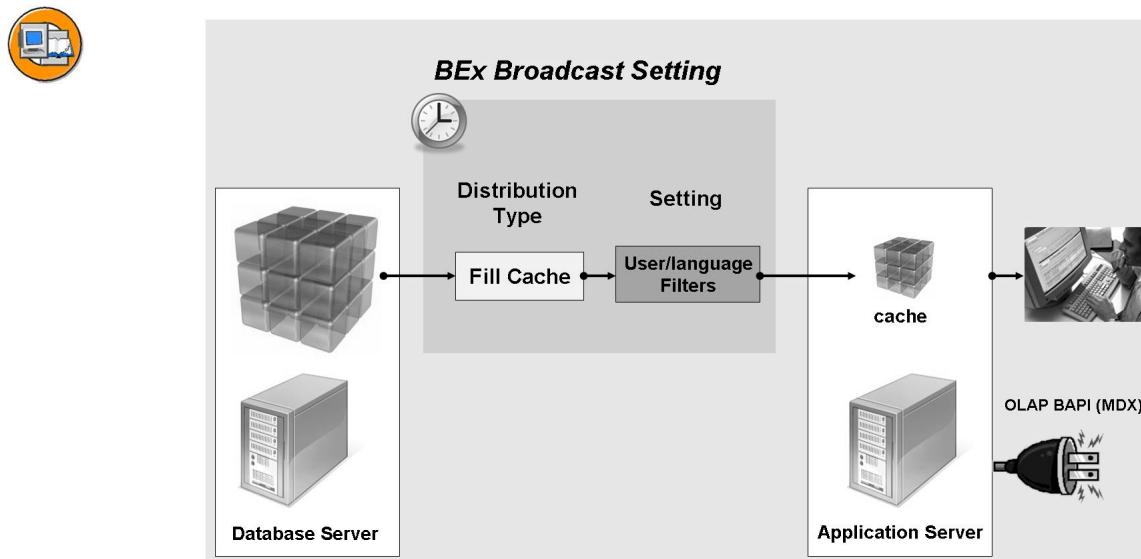
Figure 127: Fill OLAP / MDX cache

User/Language

Parameter	Purpose
Authorization User	If specified, this user will be used as the basis for the authorization check. Only result values this user is allowed to see will be presented in the broadcast.
Language	If specified the broadcast will be presented in this language, if left blank then the language of the user creating the broadcast will be used.

Filter Navigation

Parameter	Purpose
No Filter	Broadcast the report unfiltered.
Filter by selected characteristic	Enter the characteristic from the query to use as the filter, when you select a characteristic you will then see a link to create the values for filtering. You will also have the option to create an unfiltered view for each recipient.
Filter by Control Query	As an alternative to supplying fixed characteristic filters above you could also enter the name of a query which will be used to determine the characteristic filter values.

2007/Q2   **345**

Fill MDX Cache

The distribution type Fill MDX Cache allows you to precalculate BEx Queries in the background that can then be used as data providers for other applications. This function is used to improve system performance when executing these kind of applications. The BEx Broadcaster precalculates the queries as MDX result sets and fills the MDX cache. You can call up the precalculated MDX result sets using the OLAP BAPI.

User/Language

Parameter	Purpose
Authorization User	If specified, this user will be used as the basis for the authorization check. Only result values this user is allowed to see will be presented in the broadcast.
Language	If specified the broadcast will be presented in this language, if left blank then the language of the user creating the broadcast will be used.

Filter Navigation

Parameter	Purpose
No Filter	Broadcast the report unfiltered.
Filter by selected characteristic	Enter the characteristic from the query to use as the filter, when you select a characteristic you will then see a link to create the values for filtering. You will also have the option to create an unfiltered view for each recipient.
Filter by Control Query	Enter the name of a query which will be used to determine the characteristic values used for filtering the broadcast result. For example this query might select the worst performing sales regions on the basis of a condition in query a. These regions will then be sent a detailed breakdown of their results in the broadcast based on query b.

Multi Channel Broadcast

The distribution type Broadcast (Multiple Channel) allows you to distribute a BI object (query, query view, Web template, report or workbook) with one broadcast setting over multiple channels. Depending on the output format you choose, you can distribute the BI object by e-mail, export it to the portal, or print it. Once you have made you selection of the distribution type you will then be presented with the appropriate tabs in order to fill in the details of the broadcast.

BEx Broadcast Setting

Month	Product	Orders	Return
January	Karaoke Unit AC65	4755	3
	Microphone Basic W32	**422**	**0**
	CDG Soft Rock Hits	3422	12
February	CDG 1980's Hits	234	
	Karaoke Unit R44	3358	8
	Microphone Basic W32	**234**	**0**
March	CDG Country Hits	4389	54
	Karaoke Unit Y763	32	0
	Microphone Wireless A2	44	2
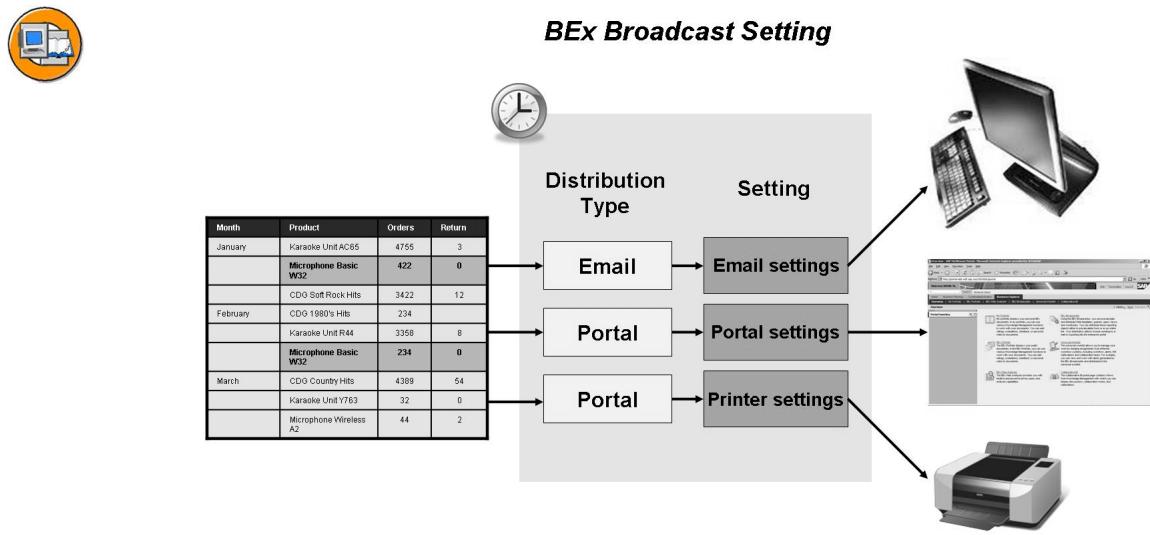
- One broadcast setting can distribute in multiple ways
- You only have to schedule and monitor one broadcast setting

Figure 128: Multi Channel Broadcast

Distributions

Parameter	Purpose
Distribution Type	Choose from printing, email, or portal output.

Exercise 12: Broadcast Value Set

Exercise Objectives

After completing this exercise, you will be able to:

- In this exercise you will learn how to create a broadcast to generate a value set

Business Example

Your organization would like to review the top selling product each month and for each of the product they would also like to analyze the customers who are buying these products. In the past this evaluation has been time consuming as the query simply took too long to run because it involved running two queries consecutively to get the results. You would like to investigate option of running the first query to calculate the top selling products offline once per month, storing these values on the BI server then using them as selections for the customer breakdown queries you run during the week.

Task 1:

Create a query, based on InfoCube *BW036 Customer* which will calculate the **top five** materials based on the key figure *Sales Volume EUR* for the period *Jan 2005*

1. Start the Query Designer and logon using your assigned user details.

2. Create a new query based on InfoCube *BW306 Customer (T_SDDEMO3)*. The result of this query should list the **top 5** materials sold for **January 2005** based on **Sales Volume EUR**.

3. Save your query to the role *BW306_Reporting → Unit 5 Broadcaster* using the query name **GR## Query Value Set 1** with the technical name **GR##QUERYVALUESET1**

4. Execute your query to make sure it returns the top 5 materials only for January 2005.

Task 2:

You will now create a broadcast setting which will generate a value set from your query results for the characteristic *Material*.

1. Log on to the portal using your assigned user details and access the Broadcaster screen.

2. Assign your query *GR## Query Value Set 1* to the broadcast setting.

3. Create a new setting using the description **GR## Value Set** for a distribution type *Precalculate Value Set* for characteristic *Material*. Leave the value set name as the default <SETTING_ID> but in the description field you should append the execution time of the broadcast using a text variable.

4. Save your broadcast setting. You already entered the description but now use the technical name **GR##VALUESET** to complete the parameters.

5. Execute the broadcast setting.

Task 3:

You will now create a second query that will use the value set as the selection for materials and breakdown the sales of each of the materials by customer.

1. Unless you already have the Query Designer open start it and logon using your assigned user details.

2. Create a new query based on InfoCube *BW306 Customer (T_SDDEMO3)*. The result of this query should list the **top 5** materials sold for **January 2005** broken down by customer with the key figure **Sales Volume EUR**. You must use your value set created in the previous step as a selection for materials.

3. Save your query to the role *BW306_Reporting → Unit 5 Broadcaster* using the query name **GR## Query Value Set 2** with the technical name **GR##QUERYVALUESET2**

4. Execute your query and when prompted enter **your** value set. Make sure it returns the top 5 materials broken down by customer and only for January 2005.

Solution 12: Broadcast Value Set

Task 1:

Create a query, based on InfoCube *BW036 Customer* which will calculate the **top five** materials based on the key figure *Sales Volume EUR* for the period *Jan 2005*

1. Start the Query Designer and logon using your assigned user details.

    a) Start the *Query Designer* using the path *Start → Programs → Business Explorer → Query Designer*. When prompted choose your assigned system and enter your assigned logon details.

2. Create a new query based on InfoCube *BW306 Customer (T_SDDEMO3)*. The result of this query should list the **top 5** materials sold for **January 2005** based on **Sales Volume EUR**.

    a) Click on the icon *New Query* then select the InfoCube *BW306 Customer (T_SDDEMO3)* from the InfoArea button. You can simply type in either the description or technical name of the InfoCube in the field *Name* to locate it.

    b) From the *Time* dimension drag the characteristic *Calendar Year/Month (0CALMONTH)* to the *Characteristic Restrictions* pane.

    c) Right-click on the characteristic *Calendar Year/Month* and from the context menu select the option *Restrict*.

    d) From the *Show* drop down list select *Single Values*.

    e) Scroll down until you reach **January 2005 (01.2005)** then drag this value into the *Selection* pane on the right. Press *OK*.

    f) From the *Material* dimension drag the characteristic *Material (0MATERIAL)* to the rows pane.

    g) From the *Key Figures* folder drag the key figure *Sales Volume EUR (T_INVCD_V)* to the columns pane.

    h) Click on the icon *Conditions* then right-click to select the menu option *New Condition*.

    i) Right-click on the new condition and select the menu option *Edit*, this will open up the dialog for making the settings.

    j) On the *General* tab click on the button *New*, then select from the *Key Figures* drop down list the key figure *Sales Volume EUR*.

    k) From the *Operator* drop down list select *Top N*.

    l) In the *Values* field enter the number **5**. Press the button *Transfer* then press the button *OK* to close the dialog box

3. Save your query to the role *BW306_Reporting → Unit 5 Broadcaster* using the query name **GR## Query Value Set 1** with the technical name **GR##QUERYVALUESET1**

    a) Press the icon *Save*, then enter the query name **GR## Query Value Set 1** in the field *Description*. Enter the name **GR##QUERYVALUESET1** in the field *Technical Name*.

4. Execute your query to make sure it returns the top 5 materials only for January 2005.

    a) Click on the icon *Execute* and when prompted enter your user details to log on to the portal. You should see only five materials listed. Correct any errors before moving onto the next step.

Task 2:

You will now create a broadcast setting which will generate a value set from your query results for the characteristic *Material*.

1. Log on to the portal using your assigned user details and access the Broadcaster screen.

    a) Use the shortcut *SAP Logon* from the Windows *Start* button to open the SAP Logon dialog.

    b) Double-click on your assigned BI system from the list to open the logon screen.

    c) Enter you logon detail assigned to you then double click on the menu option *My Portal*.

    d) When prompted enter the same logon details to access the portal.

    e) Click on the menu option *Business Explorer → BEx Broadcaster* to access the broadcaster dialog.

2. Assign your query *GR## Query Value Set 1* to the broadcast setting.

    a) From the dropdown list *Settings for Object Type* select *Query*. Then press the button *Open* to see the list of available queries. Select your query *GR## Query Value Set 1* which you should see at the top of the list. Click on this query and then press *Transfer*.

3. Create a new setting using the description **GR## Value Set** for a distribution type *Precalculate Value Set* for characteristic *Material*. Leave the value set name as the default <SETTING_ID> but in the description field you should append the execution time of the broadcast using a text variable.

    a) In the field *Description* enter the name **GR## Value Set**.

    b) From the dropdown list *Distribution Type* select the entry *Precalculate Value Set*.

    c) Click on the tab *Precalculation* and then click on the icon *Attach Text Variable* behind the description field. From the list select the entry *Execution of Setting : Time*.

Continued on next page

4. Save your broadcast setting. You already entered the description but now use the technical name **GR##VALUESET** to complete the parameters.

     a)   Press the button *Save As* then enter **GR##VALUESET** in the technical name field.

5. Execute the broadcast setting.

     a)   Execute the setting by pressing either the button *Execute* or you can right-click on the setting description (top of screen)

         and selecting *Execute* from the context menu.

     b)   Make sure you observe a message informing you that the setting was executed successfully.

         Make sure you correct any errors before moving on.

Task 3:

You will now create a second query that will use the value set as the selection for materials and breakdown the sales of each of the materials by customer.

1. Unless you already have the Query Designer open start it and logon using your assigned user details.

     a)   Start the *Query Designer* using the path *Start → Programs → Business Explorer → Query Designer*. When prompted choose your assigned system and enter your assigned logon details.

2. Create a new query based on InfoCube *BW306 Customer (T_SDDEMO3)*.
    The result of this query should list the **top 5** materials sold for **January 2005**
    broken down by customer with the key figure **Sales Volume EUR**. You must
    use your value set created in the previous step as a selection for materials.

    a) Click on the icon *New Query* then select the InfoCube *BW306
        Customer (T_SDDEMO3)* from the InfoArea button. You can simply
        type in either the description or technical name of the InfoCube in
        the field *Name* to locate it.

    b) From the *Time* dimension drag the characteristic *Calendar Year/Month
        (0CALMONTH)* to the *Characteristic Restrictions* pane.

    c) Right-click on the characteristic *Calendar Year/Month* and from the
        context menu select the option *Restrict*.

    d) From the *Show* drop down list select *Single Values*.

    e) Scroll down until you reach **January 2005 (01.2005)** then
        drag this value into the *Selection* pane on the right. Press *OK*.

    f) From the *Material* dimension drag the characteristic *Material
        (0MATERIAL)* to the *Characteristic Restrictions* pane.

    g) Right-click on the characteristic *Material* and from the context menu
        select *Restrict*.

    h) From the drop down list *Show* select the entry *Variables*.

    i) Alongside the drop down list *Type* you will see an icon *Create New
        Variable*, you should press this.

    j) In the field *Description* enter **GR## Value Set**.

    k) Click on the *Details* tab and from the drop down *Variable Represents*
        select the entry *Precalculated Value Set*. Press *OK* twice.

    l) Locate your new variable in the list on the left and then drag it to the
        Selection pane on the right.

        Press *OK*.

    m) From the *Materials* dimension drag the characteristic *Material
        (0MATERIAL)* to the rows.

    n) From the *Customer* dimension drag the characteristic *Sold-to Party
        (0SOLD_TO)* to the rows pane, and make sure this characteristic
        appears beneath the *Material* characteristic.

    o) From the *Key Figures* folder drag the key figure *Sales Volume EUR
        (T_INVCD_V)* to the columns pane.

3.   Save your query to the role *BW306_Reporting → Unit 5 Broadcaster* using the query name **GR## Query Value Set 2** with the technical name **GR##QUERYVALUESET2**

    a)   Press the icon *Save* then enter the query name **GR## Query Value Set 2** in the field *Description*. Enter the name **GR##QUERYVALUESET2** in the field *Technical Name*.

4.   Execute your query and when prompted enter **your** value set. Make sure it returns the top 5 materials broken down by customer and only for January 2005.

    a)   Click on the icon *Execute* and when prompted enter your user details to log on to the portal. When you are prompted for a variable value select **your** value set (you may see others). You should see only five materials listed, each broken down by customers.

Exercise 13: Bursting Broadcast

Exercise Objectives

After completing this exercise, you will be able to:

*   •   You will be able to create and execute a broadcast which uses the bursting distribution type.

Business Example

Your organization frequently distributes the customer master list so that the sales team can ensure the correct sales person is assigned to the account. It is very important to ensure the sales person's email address is also correctly specified for the account. The list should be emailed to all sales people but each person should only receive a list of the accounts which belong to them.

Task 1:

First you need to add two records to the customer master data table for your infoobject GR##CUST.

1.   Log on to the assigned BI system using the SAPGUI.

2.   For your assigned infoobject **GR##CUST** enter four records using the details in the table below for the field values.

GR## Customer	Email/Country Manager	Country	Description
1000	BW306-##	DE	Hi Tech
2000	BW306-##	US	Arrow Consulting
3000	BW306-(your neighbor's assigned number	FR	Blue Star
4000	BW306-(your neighbor's assigned number	UK	Powertech

Task 2:

You will now create a query to produce a list of customers for your sales people.

1.   Start the Query Designer and logon using your assigned user details.

2.   Create a new query based on master data infoprovider *GR## Customer (GR##CUST)*. There are no filters for this query. The rows should contain the characteristic *Customer* and the attributes *Email/Country Manager* and *Country*. The columns should contain the key figure *Number of Records*.

3.   Save your query to the role *BW306_Reporting → Unit 5 Broadcaster Advanced Features* using the query name **GR## Query Bursting** with the technical name **GR##QUERYBURSTING**

4.   Execute your query to make sure it presents your customers correctly, there should be five records. (where did the extra record come from ?)

Task 3:

You now need to setup the mechanism to actually broadcast this query result to your sales team, you should ensure that each team member only receives the details of customers they are currently assigned to.

1.   Log on to the portal and access the BEx Broadcaster.

2.   Your broadcast setting will be based on your query *GR## Query Bursting*, so select this.

3.   Create a new broadcast setting for bursting (Email), with the output as MHTML. The name of the setting should be **GR## Bursting Setting 1**.

4.   The recipients for the broadcast should be determined from the attribute *Email / Country Manager*. This attribute belongs to the characteristic *GR## Customer (GR##CUST)*. Make sure the sales person only receives his customer records.

5.   Add a subject to the broadcast, make sure you include your group number in the text and also the description of the master data query using a text variable. Also add some text for the contents, use text variables to provide useful broadcast information to the sales person.

6.   Save your setting with the technical name **GR##BURSTINGSETTING1**.

7.   Execute the broadcast setting immediately and check your results.

Solution 13: Bursting Broadcast

Task 1:

First you need to add two records to the customer master data table for your infoobject GR##CUST.

1. Log on to the assigned BI system using the SAPGUI.

   a) Select the Windows shortcut *SAP Logon* which you will find by pressing the *Start* button.

   b) When the logon dialog appears double-click on the assigned system then enter your assigned user details.

2. For your assigned infoobject `GR##CUST` enter four records using the details in the table below for the field values.

GR## Customer	Email/Country Manager	Country	Description
1000	BW306-##	DE	Hi Tech
2000	BW306-##	US	Arrow Consulting
3000	BW306-(your neighbor's assigned number	FR	Blue Star
4000	BW306-(your neighbor's assigned number	UK	Powertech

   a) Open the menu folder *Unit 5 BEx Broadcaster Advanced Features* and select the transaction *Master Data Maintenance*.

   b) Enter the technical name of your assigned infoobject `GR##CUST` then press the *Continue* button.

   c) When you reach the selection screen simply press the execute icon to continue.

   d) Press the *Create* icon in the toolbar, then enter the first record using the field values given in the table above. Repeat this step for the other records.

   e) Press the *Save* icon, then press the *Back* arrow twice then press the *Cancel* icon on the *Choose Characteristic* dialog to return to the main menu.

Task 2:

You will now create a query to produce a list of customers for your sales people.

1. Start the Query Designer and logon using your assigned user details.

    a) Start the *Query Designer* using the path *Start → Programs → Business Explorer → Query Designer*. When prompted choose your assigned system and enter your assigned logon details.

2. Create a new query based on master data infoprovider *GR## Customer (GR##CUST)*. There are no filters for this query. The rows should contain the characteristic *Customer* and the attributes *Email/Country Manager* and *Country*. The columns should contain the key figure *Number of Records*.

    a) Click on the icon *New Query* then select the master data infoprovider *GR## Customer (GR##CUST)* by pressing the *InfoArea* button and then click down the path *BW Training → BW Customer Training → BW306 BI Enterprise Reporting* . Highlight your assigned infoprovider then press *Open*. You can also simply type in either the description or technical name of the InfoProvider in the field *Name* to locate it.

    b) Click on the *Rows/ Columns* tab, then from the *Key Part* dimension drag the characteristic *GR## Customer (GR##CUST)* to the rows pane.

    c) Within the *Key Part* dimension expand the node *GR## Customer*, then expand the node *Attributes* and drag the attribute *Email/Country Manager* to the rows. Make sure this attribute appears beneath the *GR## Customer* characteristic.

    d) From the *Attributes* dimension drag the attribute *Country (0COUNTRY)* to the rows pane, make sure this appears beneath *Email/Country Manager* attribute.

    e) From the *Key Figures* folder drag the key figure *Number of Records (1ROWCOUNT)* to the columns pane.

3. Save your query to the role *BW306_Reporting → Unit 5 Broadcaster Advanced Features* using the query name **GR## Query Bursting** with the technical name **GR##QUERYBURSTING**

    a) Press the icon *Save*. Press the *Roles* button, then open the folder *Unit 5 BEx Broadcaster Advanced Features*. Enter the query name **GR## Query Bursting** in the field *Description*. Enter the name **GR##QUERYBURSTING** in the field *Technical Name*.

4. Execute your query to make sure it presents your customers correctly, there should be five records. (where did the extra record come from ?)

    a) Click on the icon *Execute* and when prompted enter your user details to log on to the portal. You should see five customer records, the fifth record is the *unassigned (#)* record.

    Continued on next page

Task 3:

You now need to setup the mechanism to actually broadcast this query result to your sales team, you should ensure that each team member only receives the details of customers they are currently assigned to.

1. Log on to the portal and access the BEx Broadcaster.

    a)   From the main BI menu double click on the transaction *My Portal*.

    b)   Enter your logon details when prompted.

    c)   Click on the link *Business Explorer* then click on the link *BEx Broadcaster*.

2. Your broadcast setting will be based on your query *GR## Query Bursting*, so select this.

    a)   From the drop down list *Setting for Object Type* select the entry *Query* and click the *Open* button and select the query *GR## Query Bursting* then press the button *Transfer*.

3. Create a new broadcast setting for bursting (Email), with the output as MHTML. The name of the setting should be **GR## Bursting Setting 1**.

    a)   Click on the button *New Setting* and enter then name **GR## Bursting Setting 1** in the *Description* field.

    b)   From the dropdown list *Distribution Type* select the entry *Broadcast Email (Bursting)*

    c)   From the dropdownn list *Output Format* select the entry *MHTML*.

4. The recipients for the broadcast should be determined from the attribute *Email / Country Manager*. This attribute belongs to the characteristic *GR## Customer (GR##CUST)*. Make sure the sales person only receives his customer records.

    a)   Select the radio button *Filter Document by Characteristic Value* and in the characteristic drop down list select *GR## Customer*.

    b)   From the drop down list *Attribute for Recipient Determination* select the entry *Email / Country Manager*.

    c)   From the drop down list *Attribute Value is* select the entry *BW User*.

    d)   In the section *Selection of the characteristic values* make sure the radio button *By Following Selection* is selected and click on the link *Create* to access the selection dialog.

    e)   Select all customer records except the unassigned (#) record by checking all the boxes except #. (there should be four to select). Click on the button *Transfer*.

5.   Add a subject to the broadcast, make sure you include your group number
     in the text and also the description of the master data query using a text
     variable. Also add some text for the contents, use text variables to provide
     useful broadcast information to the sales person.

     a)   Click on the *Text* tab and in the *Subject* field enter some text. Click
          on the *Text Variables* button and from the list choose the entry with
          key *Object Description*. In the *Contents* field enter some text but also
          remember to click on the text variables button to select some variables.

6.   Save your setting with the technical name **GR##BURSTINGSETTING1**.

     a)   Click on the button *Save As* and enter the technical name
          GR##BURSTINGSETTING1 and press the button *Transfer*.

7.   Execute the broadcast setting immediately and check your results.

     a)   Click on the button *Execute* and check the status box to make sure you
          do not receive any error messages before you move on.

     b)   Open the folder *Unit 5 BEx Broadcaster Advanced Features* and
          double click on the transaction *SAP Office Inbox*. Double click on the
          message to open it, in the report you should see the customers you
          assigned to your own user id. Check with you neighbor - they should
          have also received an email in their SAP Office inbox from you with
          only the customers you assigned to them.

Lesson Summary

You should now be able to:

- Extend your usage of the Broadcaster to take advantage of more advanced methods such as data bursting, value sets and pre-filling OLAP cache.

 SAP

Lesson: Broadcasting Exceptions

Lesson Overview

In this lesson you will learn how a broadcast can be set up to distribute results only when there has been an exception raised. You will learn all the steps needed to set up this type of broadcast.

Lesson Objectives

After completing this lesson, you will be able to:

•   Setup a broadcast that only distributes data if exceptions have been raised

Business Example

You want the BI system to send you notification each week if any of the teams fall below the sales targets you set.

Basic Architecture

The distribution type Broadcast by Exception allows you to check queries in the background for exceptions. If a threshold value for an exception is exceeded or not reached, the BEx Broadcaster immediately generates a document in accordance with the criteria defined by you and distributes this to the recipients by email, to the portal, or as an alert (for example, into the central worklist). A vital prerequisite to broadcasting based on exceptions is that the query contains at least one exception. In this lesson we don't cover in detail exception definition (that is covered in BW305).

There are three scenarios within this distribution type :

1.   Broadcast Email
2.   Create Alert
3.   Export to the Enterprise Portal
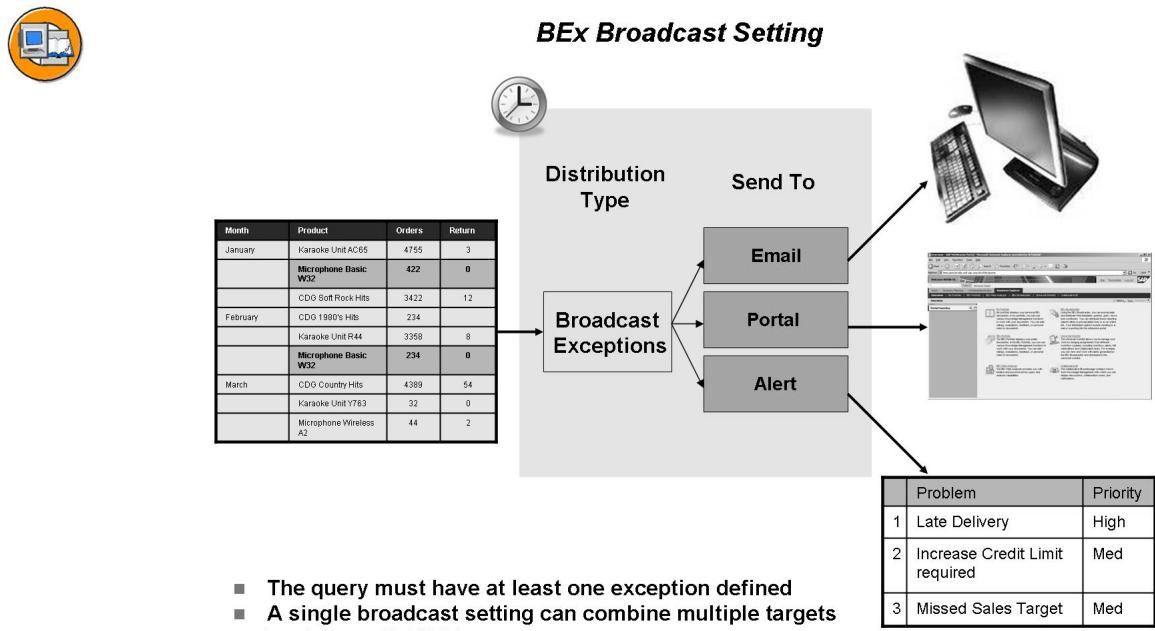
Figure 129: Broadcast Exceptions Overview

Broadcast Email

Select the distribution type *Broadcast Email* to send the details of the raised exception to the email inbox of the user. Once you select this option from the drop down list you will need to provide additional parameters as follows :

Distribution Type : Broadcast Email

Parameter	Purpose
From **Create** link	
Recipients (tab)	Same as the settings for a general email broadcast.
Texts (tab)	Same as the setting for a general email broadcast.
Exception Distribution tab	
Selection Criterion	Choose whether the broadcast should be triggered based on a check of a specific level within an exception or for any level within an exception.

Value	If you have chosen to check levels in the preceding parameter then here you specify the exact level to check within an exception. If you have chosen to check any level within an exception then here you specify the exception.
Contents	Decide if you would like to send an overview of the exceptions or a detailed report.
Format	Specify the output format of the broadcast, you can choose between HTML, XML, and text.

- ■ Common parameters for all three distribution types
- ■ Entries are dependant on distribution type

Figure 130: Broadcast Exceptions Common Parameters

Export to the Enterprise Portal

Select the distribution type *Export to the Enterprise Portal* to send the details of the raised exception to either the personal portfolio of a user or the shared portfolio (BEx Portfolio) within the portal. Once you select this option from the drop down list you will need to provide additional parameters as follows :

Distribution Type : Export to the Enterprise Portal

Parameter	Purpose
From **Create** link	
Target in Portal (tab)	
File Name	Enter the technical name by which the file will saved to the portal or use text variables to generate the name (or append to your text)
Export Document to Personal Portfolio	Enter the user ids or the role of the users who should receive this broadcast in their personal portfolio in the portal.
Export Document to Other Folder	Enter the name of the portal folder where this broadcast should be saved. Any user who has access to this folder will be able to open the report.
Texts (tab)	Same as the settings for a general email broadcast.
Exception Distribution tab	
Selection Criterion	Choose whether the broadcast should be triggered based on a check of a specific level within an exception or for any level within an exception.
Value	If you have chosen to check levels in the preceding parameter then here you specify the exact level to check within an exception. If you have chosen to check any level within an exception then here you specify the exception.
Contents	Decide if you would like to send an overview of the exceptions or a detailed report.
Format	Specify the output format of the broadcast, you can choose between HTML, XML, and text.

Create Alert

Select the distribution type *Create Alert* to send the details of the raised exception to the Universal Worklist in the portal. Once you select this option from the drop down list you will need to provide additional parameters as follows :

Distribution Type : Create Alert

Parameter	Purpose
From **Create** link	
Alert Generation (tab)	
Alert Category	Choose the alert category which you should have already defined. The alert category combines a number of parameters which determine how the alert will be managed in the Central Alert Framework (CAF)
Alert Recipients - Using Alert Category	The recipients will be determined from the alert category recipient list.
Alert Recipients - Export Additional Alert Recipient	In addition to the recipients determined from the alert category (if any) you can add more users or roles here.
Alert Parameter	
Parameters of the Alert Category	If the alert category has been defined with containers, you will see them appear here. You can then select them with the check box and then specify the text variable which will fill them.
Exception Distribution tab	
Selection Criterion	Choose whether the broadcast should be triggered based on a check of a specific level within an exception or for any level within an exception.
Value	If you have chosen to check levels in the preceding parameter then here you specify the exact level to check within an exception. If you have chosen to check any level within an exception then here you specify the exception.
Contents	This parameter is not required.
Format	This parameter is not required.

The following diagram illustrates the relationship between the Central Alert Framework and the BEx Broadcaster.
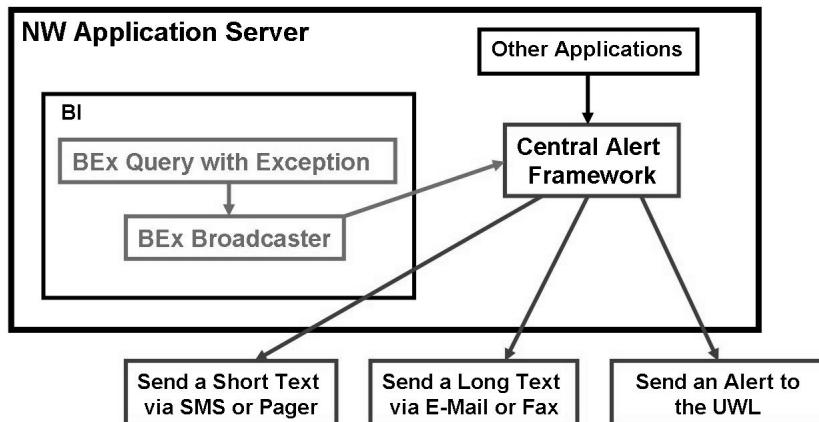
Figure 131: Broadcasting Alerts Architecture

An essential component in the architecture for alert broadcasting is the **alert category**. The alert category bring together a number of parameters which determine how an alert is managed through the NetWeaver Central Alert Framework (CAF). You access the alert category definition screen using the transaction **ALRTCATDEF**. The parameter of an alert category are parameters include :

*   Technical key (language-independent) for identification purposes
*   Description (language-dependent)
*   Classification
*   Priority
*   Expiry time (in minutes) after which the alert is deleted
*   Escalation recipient to whom the alert is sent if it is not confirmed by any of its recipients
*   Tolerance time before escalation
*   Short text, long text, and title The title is used as mail title, fax subject, and alert title in the inbox. The long text is used as mail/fax body and the long text view in the inbox. The short text is used for pager and SMS
*   Container for variable definition if variables are to be used in the texts, or for other application-specific attributes
*   Subsequent activities in the form of URLs
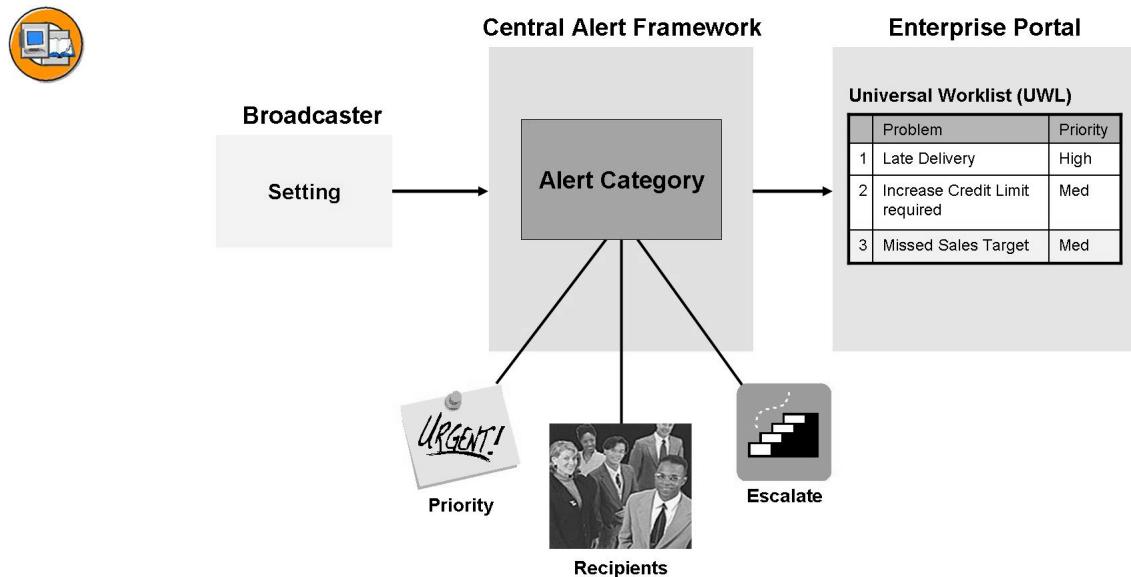
Figure 132: Alert Category

Exercise 14: Broadcast Exceptions

Exercise Objectives

After completing this exercise, you will be able to:

- • You will learn how to broadcast results which contain exceptions

Business Example

Your sales team have complained that they receive too many reports with too much data that doesn't concern them. They have asked that you send them only the sales results which are important to them, specifically the country sales figures where performance is low. You will send the reports to the personal portfolios of the sales team, but the manager has also insisted you send the alerts via the Universal Worklist in the portal (so that everyone has to confirm they have seen the alerts !)

Task 1:

First create a query which identifies the countries where the sales are low. The query should identify countries at two levels, one for countries generating sales less than 3,000,000 Euros and the other for countries generating sales between 3,000,000 and 5,000,000 Euros.

1.   Start the Query Designer and logon using your assigned user details.

2.   Create a new query based on the InfoCube *BW306 Customer (T_SDDEMO3)*. Filter the query to all months in 2005. Identify the customers where incoming orders are less than 3,000,000 Euros and also identify the customers where incoming orders are between 5,000,000 and 3,000,000 Euros.

3.   Create an exception with two levels. The first level *(Bad 3)* should trigger if the incoming orders are less than 3,000,000 Euros, the second level *(Critical 3)* trigger if the incoming orders are between 3,000,000 and 5,000,000 Euros.

4.   Save your query to the role *BW306_Reporting → Unit 5 Broadcaster Advanced Features* using the query name **GR## Query Exceptions** with the technical name **GR##QUERYEXCEPTIONS**.

5.   Execute your query to make sure it presents the results correctly, you should see at least two different colors for the rows and also some rows without color.

Task 2:

You now must create an **alert category** in order to process the alerts from the broadcast to the Universal Worklist. Often the alert category will already have been built, so you really should always check with your developers to find out if there is already a suitable alert category for your broadcast setting. In this exercise all students will build their own alert category.

1.   Access the alert category maintenance screen.

2.   Within the classification *BW306* create a new alert category using the name **GR##CAT** with the description **GR## Alert Category**.

3.   Create a new container so that you can pass the value of the query characteristic *Sold-toPartyCountry* to the alert.

4.   The subject text for the alert should be **Alert for X** where **X** is the country.

      Add your own text for the body.

5.   Assign your own user id to the alert category. Do not add any other user ids to your category, (this will only confuse everyone if they receive your alerts!)

Task 3:

You must now create a broadcast setting which will distribute the details of the poorly performing countries to the sales team. As well as sending the details to the Universal Worklist of the assigned recipients you should also use the same broadcast setting to send the report to each sales person's personal portfolio in the portal .

1.   Log on to the portal and access the BEx Broadcaster.

2.   Your broadcast setting will be based on your query *GR## Query Exceptions (GR##QUERYEXCEPTIONS)*, so select this.

3.   Create a new setting for broadcasting according to exceptions. The name of the setting should be **GR## Exceptions Setting 1**.

4.   There will be two distribution types for the exceptions, the first type should be to send the alert to the Universal Worklist using the alert category you created in the earlier step. Alerts should be sent only if there have been exceptions triggered for the very worst level. The second distribution type should be to send the alert to the sales person's personal portfolio in the portal. This alert should be triggered when there is an exception at any level.

5.   Save your setting with the technical name **GR##EXCEPTIONSETTING1**.

6.   Execute the broadcast immediately and check your results for both distribution types.

Solution 14: Broadcast Exceptions

Task 1:

First create a query which identifies the countries where the sales are low. The query should identify countries at two levels, one for countries generating sales less than 3,000,000 Euros and the other for countries generating sales between 3,000,000 and 5,000,000 Euros.

1. Start the Query Designer and logon using your assigned user details.

   a) Start the *Query Designer* using the path *Start → Programs → Business Explorer → Query Designer*. When prompted choose your assigned system and enter your assigned logon details.

2. Create a new query based on the InfoCube *BW306 Customer (T_SDDEMO3)*. Filter the query to all months in 2005. Identify the customers where incoming orders are less than 3,000,000 Euros and also identify the customers where incoming orders are between 5,000,000 and 3,000,000 Euros.

   a) Click on the icon *New Query*, then select the InfoCube *BW306 Customer (T_SDDEMO3)* by pressing the *InfoArea* button, and then click down the path *BW Training → BW Customer Training → BW306 BI Enterprise Reporting* . Highlight the InfoProvider then press *Open*. You can also simply type in either the description or technical name of the InfoProvider in the field *Name* to locate it.

   b) Expand the *Time* dimension and drag the characteristic *Calendar Year/Month* to the *Characteristic Restrictions* pane.

   c) Right-click on the characteristic *Calendar Year/Month* and from the context menu select the option *Restrict*.

   d) From the *Show* drop down list select *Value Range*, then from the next drop down list select the entry *Between* (if it is not already shown). Now press the icon *Select from List* and when the month selection dialog box appears simply enter **01.2005** in the *Direct Input* field, then press enter. Repeat this step using month **12.2005** in the next field to complete the month range.

   e) Drag the month range you just created into the *Selection* pane on the right. Press *OK*.

   f) Click on the *Rows/ Columns* tab then from the *Customer* dimension drag the characteristic *Sold-toPartyCountry (0SOLD_TO__0COUNTRY)* to the rows pane.

   g) From the Key Figures folder drag the key figure *Incoming Orders (T_INCORDV)* to the columns pane.

3. Create an exception with two levels. The first level *(Bad 3)* should trigger if the incoming orders are less than 3,000,000 Euros, the second level *(Critical 3)* trigger if the incoming orders are between 3,000,000 and 5,000,000 Euros.

   a) Click on the icon *Exceptions* in the Query Designer main toolbar.

   b) Right click anywhere in the exceptions pane and select the entry *New Exception*.

   c) Right click on the newly created entry at the top of the pane (it may be called *Exception1*) and select the entry *Edit*.

   d) In the *Description* field enter the name **Low Sales**.

   e) Click on the *New* button.

   f) From the drop down list *Alert level* select the entry *Bad 3*.

   g) From the drop down list *Operator* select the entry *Less than*.

   h) in the field *Value* enter the number **3,000,000**.

   i) Click on the button *Transfer*.

   j) Click on the *New* button.

   k) From the drop down list *Alert level* select the entry *Critical 3*.

   l) From the drop down list *Operator* select the entry *Between*.

   m) In the field *From Value* enter the number **3,000,000**.

   n) In the field *To Value* enter the number **5,000,000**.

   o) Click on the button *Transfer*, then click on the button *OK*.

4. Save your query to the role *BW306_Reporting → Unit 5 Broadcaster Advanced Features* using the query name **GR## Query Exceptions** with the technical name **GR##QUERYEXCEPTIONS**.

   a) Press the icon *Save*. Press the *Roles* button then open the folder *Unit 5 BEx Broadcaster Advanced Features*. Enter the query name **GR## Query Exceptions** in the field *Description*. Enter the name **GR##QUERYEXCEPTIONS** in the field *Technical Name*.

5. Execute your query to make sure it presents the results correctly, you should see at least two different colors for the rows and also some rows without color.

   a) Click on the icon *Execute* and when prompted enter your user details to log on to the portal. Do you see two exception colors and also rows without color ?

Task 2:

You now must create an **alert category** in order to process the alerts from the broadcast to the Universal Worklist. Often the alert category will already have been built, so you really should always check with your developers to find out if there is already a suitable alert category for your broadcast setting. In this exercise all students will build their own alert category.

1. Access the alert category maintenance screen.

    a) From the Windows *Start* button select the shortcut *SAP Logon* , then double click on your assigned system. .

    b) Logon to the BI system using the detail supplied to you, then expand the menu folder *Unit 5 BEx Broadcaster Advanced Features* and double click on the menu option *Editing Alert Categories*.

2. Within the classification *BW306* create a new alert category using the name **GR##CAT** with the description **GR## Alert Category**.

    a) Double-click on the classification folder *BW306*, then press the toolbar button *Display/Change* to switch to change mode. Now press the icon *Create Alert Category* and in the fields immediately beneath the icon enter the name of the category **GR##CAT** and description **GR## Alert Category**. You must now **press the enter button** so that you can continue to enter parameters (you should see the name of your alert category in the field *Description* within the *Properties* tab.

3. Create a new container so that you can pass the value of the query characteristic *Sold-toPartyCountry* to the alert.

    a) Click on the *Container* tab and press the *Create* icon.

    b) In the field *Element* enter the name **COUNTRY_ELEMENT**.

    c) In the field *Name* enter **Country**.

    d) In the field *Description* enter **Country**.

    e) Under the *D.Type* tab select the radio button *Object Type* then from the drop down list select the entry *Not Defined*. Click the check box to close the container dialog.

4. The subject text for the alert should be **Alert for X** where **X** is the country.

Continued on next page

Add your own text for the body.

a) Click on the tab *Long and Short Text* and in the *Message Title* field enter the text **Alert for** .

b) Keep the cursor at the end of the text and press the icon *Insert Expression*. Expand the folder *Container* and double click on your container *Country* to select it.

c) Click on the tab *Long Text (email, Fax)* and enter the text **This customer needs attention.**

5. Assign your own user id to the alert category. Do not add any other user ids to your category, (this will only confuse everyone if they receive your alerts!)

a) Press the button *Fixed Recipients* , then press the button *New Entries* and enter your own user id in the first field.

b) Press the *Save* icon, then press the *Exit* icon to return to main alert category screen, then press *Save* icon one more time to save your alert category.

Task 3:

You must now create a broadcast setting which will distribute the details of the poorly performing countries to the sales team. As well as sending the details to the Universal Worklist of the assigned recipients you should also use the same broadcast setting to send the report to each sales person's personal portfolio in the portal .

1. Log on to the portal and access the BEx Broadcaster.

a) From the main BI menu double click on the transaction *My Portal*.

b) Enter your logon details when prompted.

c) Click on the link *Business Explorer*, then click on the link *BEx Broadcaster*.

2. Your broadcast setting will be based on your query *GR## Query Exceptions (GR##QUERYEXCEPTIONS)*, so select this.

a) From the drop down list *Setting for Object Type* select the entry *Query* and click the *Open* button and select the query *GR## Query Exceptions (GR##QUERYEXCEPTIONS)*, then press the button *Transfer*.

3. Create a new setting for broadcasting according to exceptions. The name of the setting should be **GR## Exceptions Setting 1**.

    a)   Click on the button *New Setting* and enter then name **GR## Exceptions Setting 1** in the *Description* field.

    b)   From the dropdown list *Distribution Type* select the entry *Broadcast according to Exceptions*

4. There will be two distribution types for the exceptions, the first type should be to send the alert to the Universal Worklist using the alert category you created in the earlier step. Alerts should be sent only if there have been exceptions triggered for the very worst level. The second distribution type should be to send the alert to the sales person's personal portfolio in the portal. This alert should be triggered when there is an exception at any level.

    a)   Under the tab *Exception Distribution* use the drop down list *Distribution Type* to select the entry *Create Alert*.

    b)   Click on the link *Create* next to the label *Parameter* then you will see the tab *Alert Generation*. From the drop down list *Alert Category* select your alert category *GR## Alert Category*.

    c)   Click on the tab *Alert Parameter* and select the *Country* checkbox.

    d)   Click on the icon *Attach Text Variables* and from the list select the entry *Exception Report : Charact. Value for Sold-toPartyCountry*. **You will have to scroll over the page to see this entry**.

    e)   Press the *Transfer* button.

    f)   From the drop down list *Selection Criterion* select the entry *Exception/Alert Level*.

    g)   From the drop down list *Value* select the entry *Low Sales / Bad 3*.

    h)   We will now create the second distribution type so from the dropdown list *Distribution Type* select the entry *Export to the Enterprise Portal*.

    i)   From the drop down list *Selection Criterion* select the entry *Exception*. Leave the other settings as defaults on this screen.

    j)   Click on the link *Create* next to the label *Parameter* and open the dialog box to make the portal target settings.

    k)   Add your own text to the description field then press the button *Transfer*.

5. Save your setting with the technical name **GR##EXCEPTIONSETTING1**.

    a)   Click on the button *Save As* and enter the technical name **GR##EXCEPTIONSETTING1** and press the button *Transfer*.

 SAP

6. Execute the broadcast immediately and check your results for both distribution types.

   a) Follow the portal menu *Business Explorer → Universal Worklist*. You should see the alerts. Click on any of them to see the detailed report.

   b) Follow the portal menu *Business Explorer → My Portfolio* and you should see your exception broadcast. Click on the report to see the details.

Lesson Summary

You should now be able to:

- •   Setup a broadcast that only distributes data if exceptions have been raised

Lesson: Scheduling Broadcasts

Lesson Overview

In this lesson we cover the options available to you for scheduling the broadcasts.

Lesson Objectives

After completing this lesson, you will be able to:

*   Define various scheduling settings to ensure broadcasts are executed and distributed at required times to meet business requirements

Business Example

Sales managers would like to have their team reports sent to them each week at the same time just prior to the sales meeting. Some managers only want to see the report if there has been sales activity. Occasionally a manager may want to monitor carefully a product promotion and so they want know how to setup their own scheduling options to suit their own requirements

Introduction

Once a broadcast setting has been created the schuduling options can be considered. There are three basic methods for scheduling broadcast settings.

*   Predefined Timepoint
*   On change of data in InfoProvider
*   User defined time setting

Each one of these scheduling methods causes the broadcast setting to be triggered, which means the assigned query, query view, web template, report or workbook is executed in the background. The results are then processed according to the broadcast settings.

Of course broadcast settings can also be executed immediately if the results need to be sent right away. However, using the broadcasting scheduling options allows the BI system to be set up distribute business results according to the timing requirements of the users. Carefully scheduling broadcast settings can also help to avoid system performance problems y allowing users to choose timepoints that don't interfere with critical system activity or heavy workloads.

It is important to remember that scheduling broadcast setting may not something you would want every user to be able to do. To help control who is able to work with scheduling options you need to take a look at the authorization object **S_RS_BCS** With this object you can determine which user is allowed to register broadcasting settings for execution and in which way.

Execution with data change in the InfoProvider

With this method of scheduling you can ensure that broadcast settings are only triggered when there has been a change to the underlying InfoProvider to the assigned query, query view, report, web template or workbook. The InfoProvider can be a physical object such as a Basic InfoCube, Data Store Object or the InfoProvider can be a virtual object such as an InfoSet or MultiProvider. If the InfoProvider is a virtual object then the check of data change is made on the underlying physical InfoProviders. For example if the broadcast setting was based on a query which receives it data from a MultiProvider and the MultiProvider is assigned to three InfoCubes then it would only be necessary for a change to be made to one of the InfoCubes for the broadcast to be triggered.

An integral component in this scheduling method is the use of a **process chain**. Within the process chain function a **process type** with the name *Trigger Event Data Change (for Broadcaster)* must be used to determine which InfoProviders should be checked for data changes. You can specify multiple InfoProviders within this process type either directly (by simply entering the names of the InfoProviders) or indirectly by specifying the name of the variant based on one of the following process types:

- Execute InfoPackage (LOADING)
- Activate DataStore Object Data
- Update DataStore Object Data (Further Updating)
- Read PSA and Update Data Target

The diagram below illustrates the relationship between the various components in this scenario.
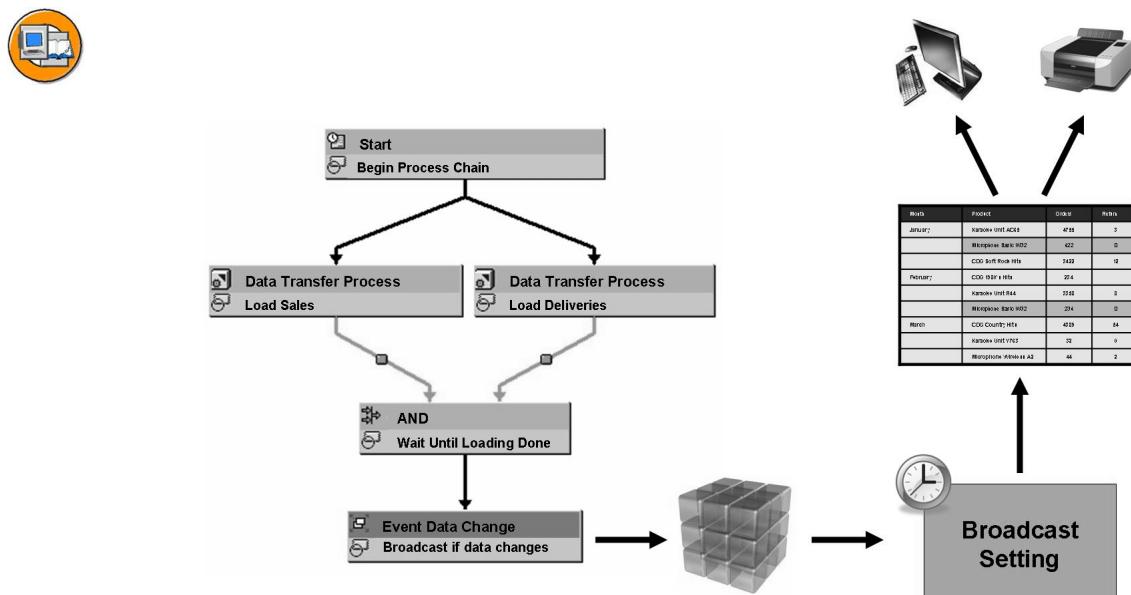
Figure 133: Data Change in InfoProvider

Predefined Time Point

By setting up predefined time points users are able to select a suitable time point when their broadcast should be triggered. This is a great way to ensure users are only able to select time points that have been setup specifically with system performance and job scheduling in mind.

A time point is basically a background job which has been pre-defined by the system administrator. The job name must be prefixed with the following : **TP_BROADCAST_**. You can add your own text at the end of this string to differentiate the jobs. If you forget to use the fixed prefix then the job will not be recognized by the broadcaster and will be ignored for scheduling. Jobs may be set to run once, but it is more likely that the job will be set to repeat on a daily, weekly or monthly basis.

Bear in mind that each time point will be seen by everyone who is scheduling a broadcast setting. You cannot setup a time point specifically for one InfoProvider, or a reporting object or a broadcast setting, time points are shared across BI.

It is possible to choose more than one time point for a broadcast, for example a user may want to trigger the broadcast daily for a quick update on sales figures, then again weekly to get the full sales results.

BI Job Schedule

Time	Frequency	
8.00 am	Daily	☑
7.00 pm	Daily	
3.30 pm (Saturdays)	Weekly	☑
4.00 am (30th)	Monthly	

Broadcast Setting Orders

Broadcast Setting Deliveries
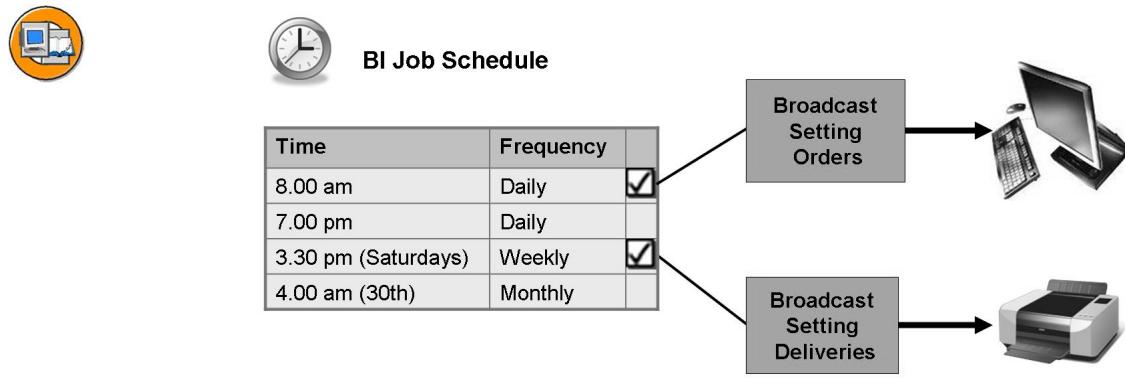
- Job must begin **TP_BROADCAST_**
- Job step is ABAP program **RSRD_BROADCAST_FOR TIMEPOINT**

Figure 134: Time Point Scheduling

The basic procedure for setting up a time point is as follows :

1. Choose Define Job in the SAP menu, or call transaction SM36
2. Use the following naming convention when entering the job name:
    TP_BROADCASTING_
3. Choose Start Condition. The Start Time dialog box appears
4. Choose Date/Time and enter the required values and select Execute Job
    Periodically.
5. Choose Period Values then select the period unit you require.(Daily, Weekly,
    Monthly, Other)
6. Choose Step, the Create Step 1 dialog box appears
7. Choose ABAP and in the Name field, you MUST enter the ABAP program
    (report) RSRD_BROADCAST_FOR_TIMEPOINT,
8.   Save your entries. The job is now scheduled and ready to use with a
    broadcast setting.

User Defined Time Setting

It is possible to allow a user to schedule their own broadcast setting. This is useful
in the following situations :

•   When there is no suitable predefined time point available.
•   When the broadcast setting should only be sent a limited number of times.

In order to define your own scheduling settings you must open the scheduling
dialog for the broadcast setting and check the box *Create New Scheduling* Specify
the required start date and time. When you put the cursor in the date field you
will see a calendar pop up to help you with the date selection. If scheduling is to
occur periodically, select Periodic and specify the interval (for example Every 2
Weeks). The system distributes the document according to your selection. If you
have already defined schedules in SAP background processing for this broadcast
setting, they are listed here with helpful scheduling information so you can see
what is already scheduled. You can also delete them here as required.
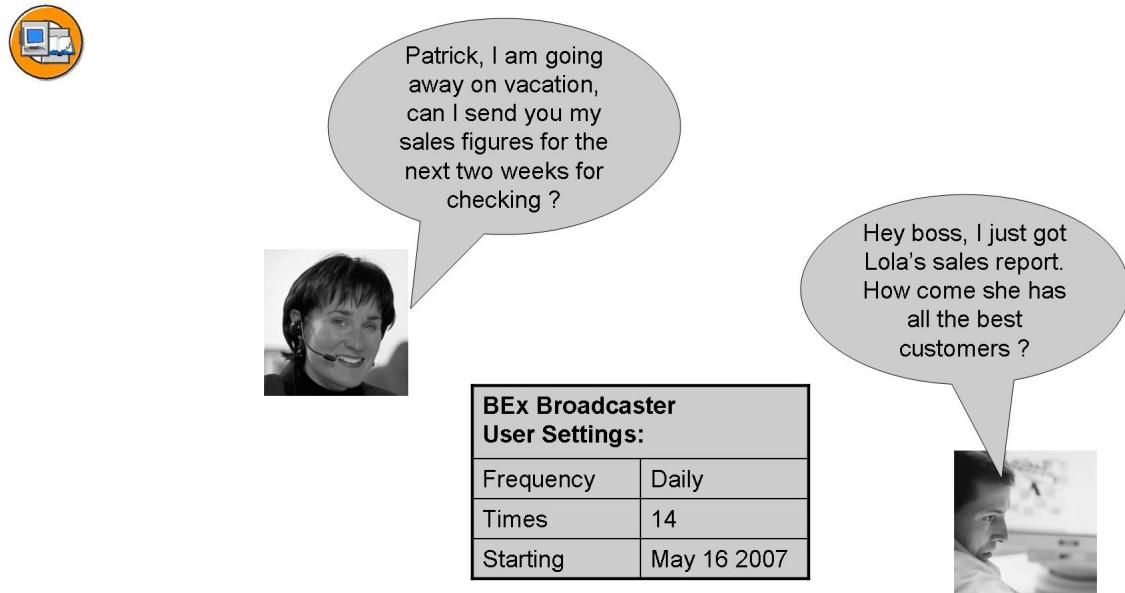
Figure 135: User Defined Settings

Exercise 15: Scheduling a broadcast using a timepoint

Exercise Objectives

After completing this exercise, you will be able to:

•   You will be able to schedule a broadcast based on a specific time point

Business Example

Your organization is concerned that the broadcasts will start to cause performance problems if they interfere with other critical jobs on the system during the daytime. You need to learn how to ensure that users can only schedule their broadcast at the specific times which you allow.

Task 1:

You will begin by defining a new background job, the job will be a weekly recurring job.

1.   Logon to the BI system using the SAPGUI then using transaction SM36 setup a background job using the parameters below:

Parameters	Value
Job Name	TP_BROADCASTING_GR##
ABAP Program	RSRD_BROADCAST_FOR_TIMEPOINT
Date / Time	Date is today. Choose a time 10 minutes from now.
Periodic Job	Weekly

Task 2:

You will now create a simple broadcast for distribution of a report via email but the trigger for the broadcast will be the weekly background job which you have just defined.

1.   Log on to the portal using your assigned user details and access the Broadcaster screen.

2.   Assign the existing query *GR## Query Value Set 2* to the broadcast setting.

3.   Create a new setting using the description **GR## Time Point** for a distribution type *Broadcast Email*, for the subject enter **GR## Weekly Sales**, and also enter some text in the content field.

4.   Save your broadcast setting. You already entered the description but now use the technical name **GR##TIMEPOINT** to complete the parameters.

5.   Schedule your setting using **your** background job. (How do you know which job is yours ?)

6.   Wait until the schedule time has past then check your results.

7.   Make sure your broadcast timepoint job has the correct status and is scheduled to run again next week. Use the transaction SM37

Solution 15: Scheduling a broadcast using a timepoint

Task 1:

You will begin by defining a new background job, the job will be a weekly recurring job.

1. Logon to the BI system using the SAPGUI then using transaction SM36 setup a background job using the parameters below:

Parameters	Value
Job Name	TP_BROADCASTING_GR##
ABAP Program	RSRD_BROADCAST_FOR_TIMEPOINT
Date / Time	Date is today. Choose a time 10 minutes from now.
Periodic Job	Weekly

a)   From the Windows *Start* button select the shortcut *SAP Logon*.

b)   Double click on the assigned BI training system then enter your assigned user logon details.

c)   In the command field enter the transaction **SM36** then press enter.

d)   In the field *Job Name* enter **TP_BROADCASTING_GR##**.

e)   Press the button *Start Condition*.

f)   Press the button *Date / Time*.

g)   In the field *Scheduled Start Data* enter today's date.

h)   In the field *Scheduled Start Time* enter the current time +10 minutes (remember the BI system time may not be the same as your local time. Check this using the menu option *System → Status*). You now have 10 minutes to complete the rest of this exercise so you better get a move on ! Make a note of the scheduled time so you can monitor this.

i)   Check the field *Periodic Value*.

j)   Press the button *Period Values* then press the button *Weekly* and close the dialog box by pressing *Save*. Now press *Save* one more time to close the Start Condition dialog.

k)   Press the button *Step*.

l)   Press the button *ABAP Program* then enter the program name **RSRD_BROADCAST_FOR_TIMEPOINT**. Press *Save* to close the Step dialog box.

m)   Press the *Back* arrow, then press *Save* to save your job.

Task 2:

You will now create a simple broadcast for distribution of a report via email but the trigger for the broadcast will be the weekly background job which you have just defined.

1. Log on to the portal using your assigned user details and access the Broadcaster screen.

    a)   From the main BI menu select the menu option *My Portal*.

    b)   When prompted enter your assigned logon details to access the portal.

    c)   Click on the menu option *Business Explorer → BEx Broadcaster* to access the broadcaster dialog.

2. Assign the existing query *GR## Query Value Set 2* to the broadcast setting.

    a)   From the dropdown list *Settings for Object Type* select *Query*. Then press the button *Open* to see the list of available queries. Select your query *GR## Query Value Set 2* which you should find under the role *BW306 Reporting - Unit 5 Broadcaster Advanced Features*. Click on this query and then press *Transfer*.

3. Create a new setting using the description **GR## Time Point** for a distribution type *Broadcast Email*, for the subject enter **GR## Weekly Sales**, and also enter some text in the content field.

    a)   In the field *Description* enter the name **GR## Time Point**.

    b)   Click on the tab *Text* and in the field *Subject* enter the text **GR## Weekly Sales**.

    c)   In the field *Content* enter some text.

4. Save your broadcast setting. You already entered the description but now use the technical name **GR##TIMEPOINT** to complete the parameters.

    a)   Press the button *Save As*, then enter **GR##TIMEPOINT** in the technical name field.

5. Schedule your setting using **your** background job. (How do you know which job is yours ?)

    a)   Press the button *Schedule*, then hover the mouse pointer over the background jobs until you see the hover text displaying your group number. Select the check box next to your job then press *Transfer*. The broadcast is now scheduled. You can close the broadcaster settings dialog.

6.   Wait until the schedule time has past then check your results.

a)   From the main BI menu, open the folder *Unit 5 BEx Broadcaster Advanced Features* and double click on the transaction *SAP Office Inbox*. Double click on the message to open it (look for the title *GR## Weekly Sales*).

7.   Make sure your broadcast timepoint job has the correct status and is scheduled to run again next week. Use the transaction SM37

a)   In the command field enter the transaction **SM37**, then press enter.

b)   Remove all the checks from the *Job Status* fields except *Released*. Enter today date + 1 week in the field *Job Start Condition - To*. Leave all the other defaults as they are and press the button *Execute*.

c)   You should see only one job (unless you created more), double click on this job and observe the date and time in the *Planned Start* area, it should show next weeks date.

Lesson Summary

You should now be able to:
- Define various scheduling settings to ensure broadcasts are executed and distributed at required times to meet business requirements

Unit Summary

You should now be able to:

- Describe the basic features of the BEx Broadcaster including how to broadcast results via email or to the portal
- Extend your usage of the Broadcaster to take advantage of more advanced methods such as data bursting, value sets and pre-filling OLAP cache.
- Setup a broadcast that only distributes data if exceptions have been raised
- Define various scheduling settings to ensure broadcasts are executed and distributed at required times to meet business requirements

Unit 6

Portal Integration

Unit Overview

The NetWeaver Portal is very tightly integrated with BI in the NetWeaver 2004s release. In this unit you will explore the nature of this integration and how you can use it to easily publish information to your users.

Unit Objectives

After completing this unit, you will be able to:

- Publish BI content in the portal to iViews and organize the iViews in the Portal Content Directory using folders and pages.

Unit Contents

Lesson: Publishing and Organizing BI Content in the Portal

Lesson Overview

In the BW305 class we introduced the NetWeaver Portal. In this lesson we will look at how content from BI can be published directly to the Portal and how to organize the BI content using folders, pages and roles.

Lesson Objectives

After completing this lesson, you will be able to:

- Publish BI content in the portal to iViews and organize the iViews in the Portal Content Directory using folders and pages.

Business Example

You want to make the BI report results available to users via the portal. In order to do this you need to learn how to work with the main portal objects iViews, pages, folders and roles.

Key Concepts

You can integrate business-relevant content from *SAP Business Intelligence* seamlessly into a *SAP Enterprise Portal*. The *SAP Enterprise Portal* enables you to access applications from other systems and sources, such as the Internet or Intranet. Using one entry point, you can reach both structured and unstructured information. In addition to content from *Knowledge Management*, business data from data analysis is available to you from the Internet and from the Intranet.
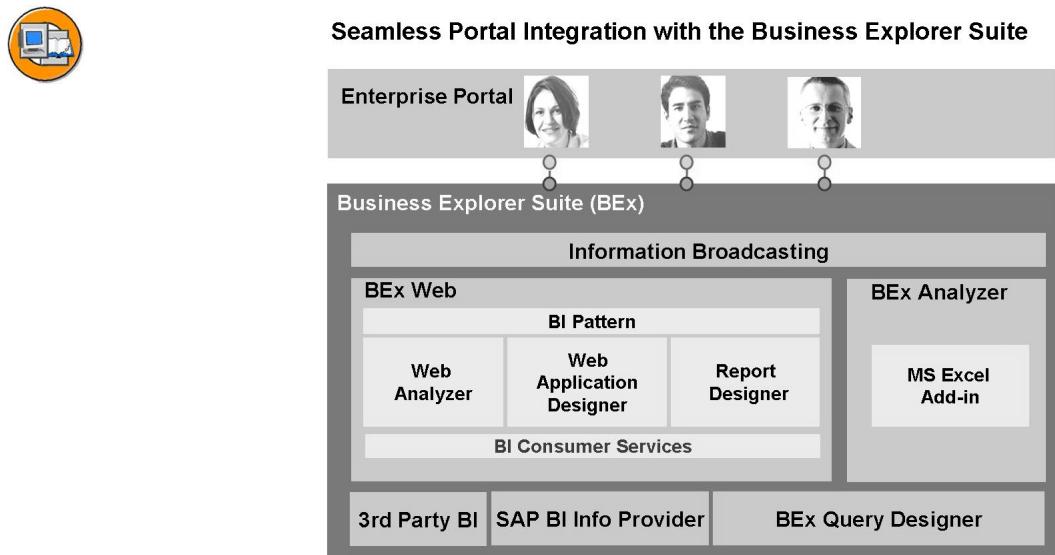
Figure 136: SAP BI and Portal Integration

The preparation of information is based on user roles in the company. Since *SAP Business Intelligence* uses the role concept, integrating *SAP BI* contents into the *SAP Enterprise Portal* is easy. *SAP BI* users see their *SAP BI* role in the *SAP Enterprise Portal* with the same content .

Furthermore, you can also use the *iView* concept to integrate *SAP BI* applications and bring individual Web applications from *SAP BI* as an *iView* into the *SAP Enterprise Portal*. You can then display and use them from a page in the portal, together with other *iViews* from the *SAP BI* system or from other systems.

The unification concept for the *SAP Enterprise Portal* enables you to directly set contents from the *SAP BI* system with contents from other systems or from the Internet in relation to one another. With the function *Drag & Relate*, you can link data together across system boundaries in order to obtain additional information.

Publishing BI Content to the Portal
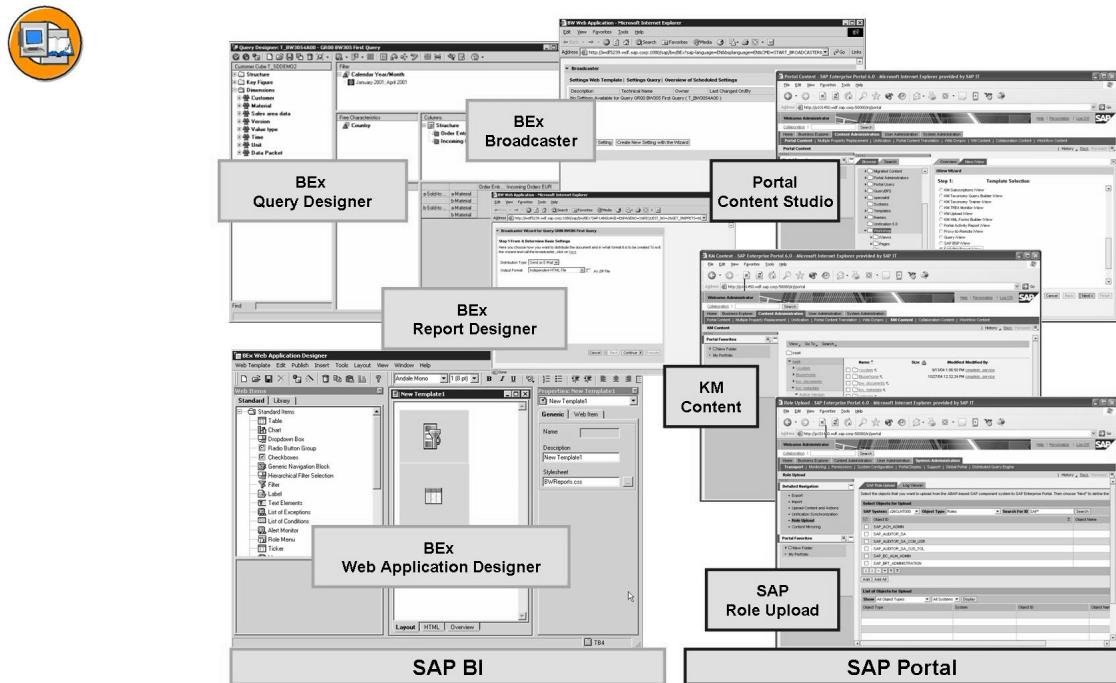
Figure 137: Tools for Integrating BI Content into the Portal

To distribute or integrate content from *SAP BI* to the *SAP Enterprise Portal* you can use the following tools:

Tools for Integrating BI Objects into the Enterprise Portal

BEx Broadcasting Wizard	An assistant that supports you in precalculating and distributing queries, Web templates or workbooks with step-by-step instructions.
BEx Broadcaster	Tool for precalculation and distribution of queries, Web templates and workbooks. The BEx Broadcaster offers more extensive options than the BEx Broadcasting Wizard.
BEx Web Application Designer	A desktop application for creating Web pages using BI content.
BEx Query Designer	Tool for defining queries that are based on a selection of characteristics and key figures (InfoObjects) or on reusable structures of an InfoProvider.

Portal Content Studio	Central environment for the creation and management of Portal Content in the Enterprise Portal.
KM Content	Central environment for the creation and management of documents and links in Knowledge Management.
SAP Role Upload	Tool for uploading role definitions and the associated objects from backend systems (for example a BI system) into the Enterprise Portal.

> **Hint: Portal Content Studio**
>
> The Portal Content Studio is the central environment in the Enterprise Portal for creating and managing portal content. You can use it to edit the following types of portal content: iViews, Portal Pages, layouts of Portal Pages, folders, Portal Roles, Worksets.

When you want to publish your BI Content from the BEx tools you use the menu path *Publish → To Portal*. You will then need to choose the name that will be given to the portal iView, the name of the BEx reporting object is usually defaulted, but you can certainly pick a new name if you prefer. You will also be asked to choose the target Portal folder where the new iViews will be stored. You can press the selection icon next to the folder field to see the full folder hierarchy in the Portal. The folder hierarchy you see is called the **Portal Content Directory** and is where all folders, pages, roles and iViews are stored in the Portal.

You can define your own folder structure in the Portal Content Directory, when you select the *Publish → To Portal* menu option from the BEx tools you will see this structure. You expand the structure and click on the target folder. The diagram below provides an example structure.
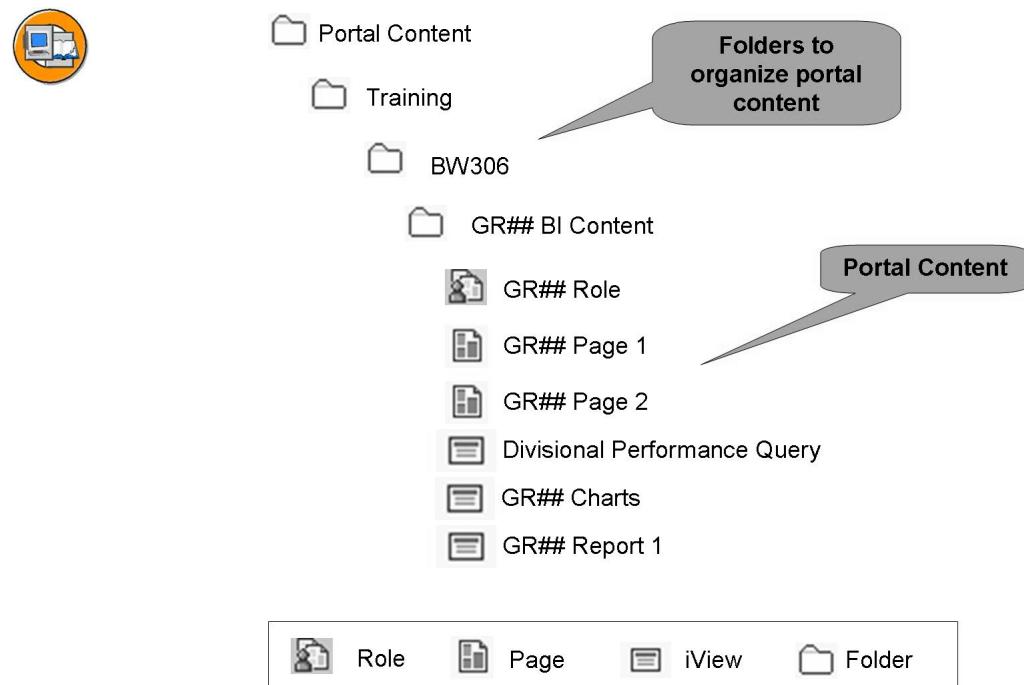
Figure 138: Portal Content Directory

Page Development in the Portal

Once you have published the iViews to the Portal Content Directory folders you need to create portal pages. Portal pages are used to present the iViews to the user and can appear in a variety of layout options, such as one or two column widths. Each page is stored in a folder in the Portal Content Directory. Once you have created a page and chosen a layout option for the page you can then add the iViews to this page, the iViews may have been developed from BI content or other external services such as SAP ECC transactions, URL's etc.

For each page you can set properties which determine how the page will behave. Also for each iView you add to a page you can also set unique properties, for example you could set a fixed height for an iView, set the initial state to be collapsed and many other settings. For each page you define you can also specify alternate layouts. This allows the user to personalize the page by choosing their own preferred view.
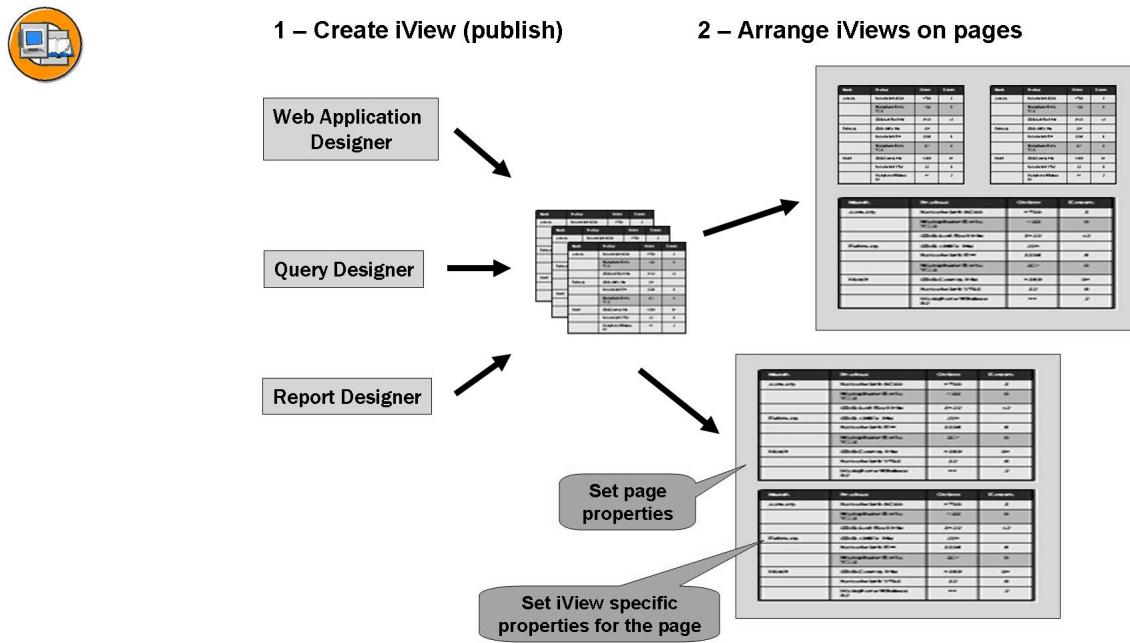
Figure 139: Building Portal Pages

Role Development in the Portal

Once you have organized your iViews on portal pages, you can then organize the pages using a portal role. Pages are assigned to the role using a hierarchical structure of folders. It is important that you understand the relationship between the folders and the layout of the portal interface. Basically there are three key layers for navigation in the portal folders. These are clearly identified in the diagram below.

- First level of top level navigation
- Second level of top level navigation
- Detailed navigation

The first two layers relate to the horizontal menus you see at the top of your screen. The third layer relates to the navigation panel you see on the right of the screen. When you define folders in a role it is important to be aware of these layers as their hierarchical position determines where they will appear on your final screen.
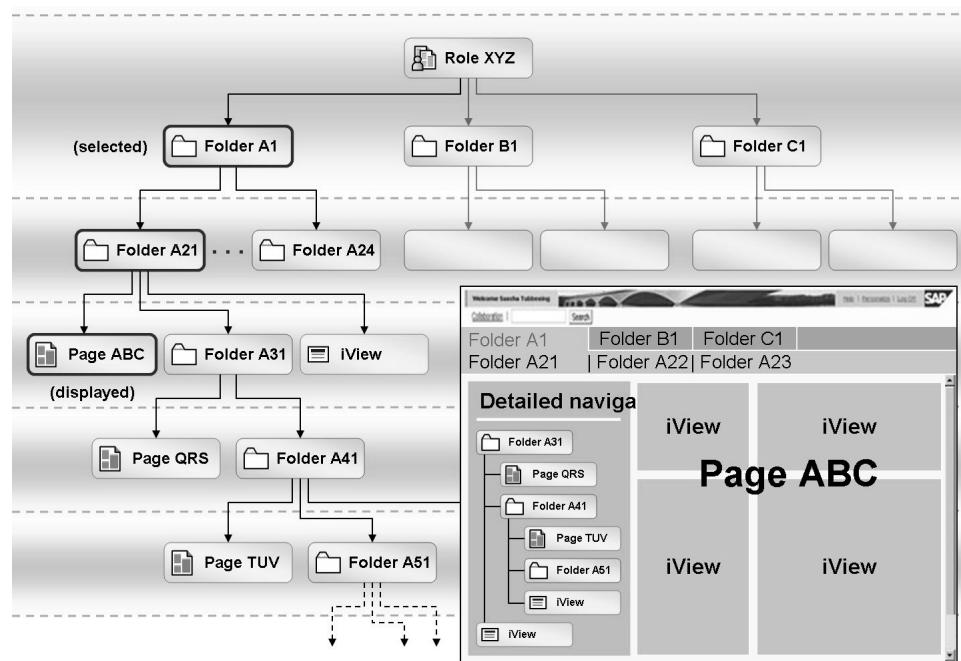
Figure 140: Organizing pages in role folders

When you are defining folders within a role you will have the opportunity to mark them as **entry points**. A folder marked as an entry point will be used as the default, i.e. when you log on, the portal needs to know which folders it should open. Typically you would make sure the folders containing the key pages are marked as entry points so the users have a useful initial view in their portal.

The diagram below is used to illustrate the folder hierarchy you will build in your role.
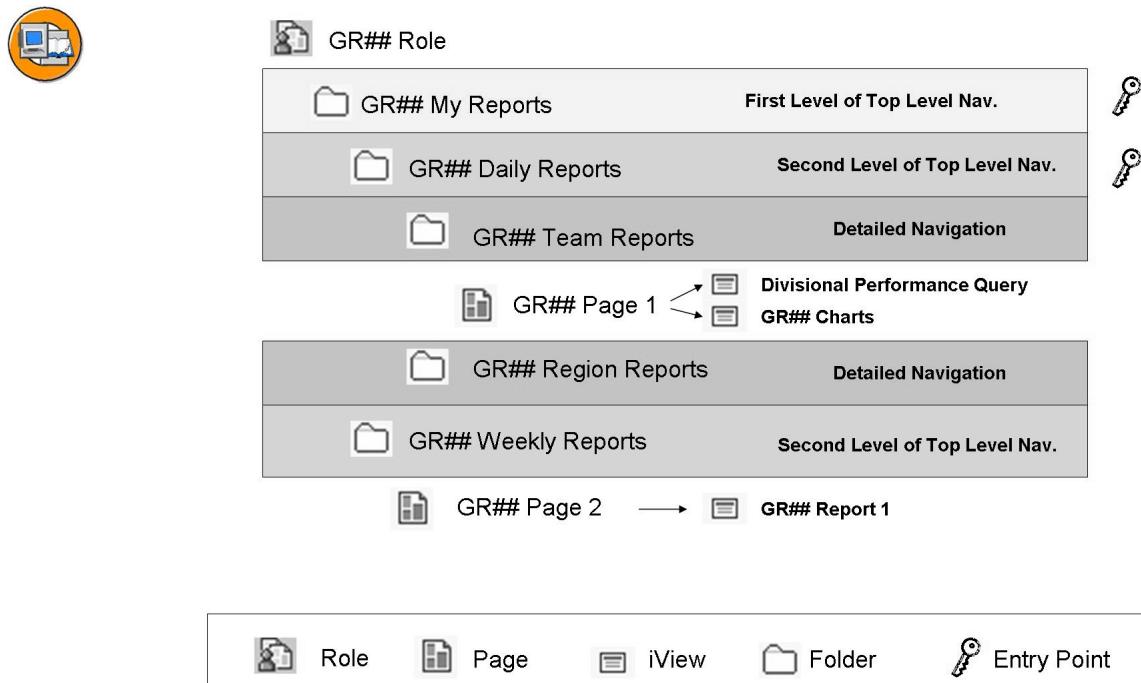
Figure 141: Exercise Folder Hierarchy for the Role

Assigning Roles to Users

Once the role have been built it is then possible to assign them to the users. A user would typically expect to be assigned many roles which represent aspects of their own job. Roles can be used to store job function pages which might contain iViews for managing sales orders and deliveries etc. A user would also be assigned to roles which may contain generic business functions such as time sheet entry, leave and absence management, and IT support.

Publishing Workbooks to the Portal

Publishing workbooks to the portal is a little different from the steps you take to publish other BI content to the portal. The workbook cannot be displayed within the portal (for technical reasons), it has to be displayed in a separate window.

> **Hint:** SAP recommends you use the BEx tools which can publish content completely in the portal rather than launching workbooks form the portal. You can use the function *Export as Excel 2000 File* from the BEx Web application context menu as an alternative to launching the workbook from the portal if you need to use the functions of Excel

So unlike other BEx tools where you have the option to *Publish → To Portal*
directly from the tool the Analyzer does not offer this feature. You need to log on
to the portal, create a new iView and supply the parameters which direct the iView
to your workbook. The iView is very similar to the type you would use to call an
ECC transaction, the iView type is an *SAP Transaction iView*. When you choose
this type you need to supply the transaction code. In order to launch the Analyzer
we call the transaction code *RRMXP*. You may recognize this transaction, but you
may wonder why there is a 'P' at the end. This version of the familiar transaction
not only calls the BEx Analyzer from the SAPGUI but also offers a prompt to
allow you to supply the workbook or query name to be displayed. What we do
in the portal is to call *RRMXP* and supply the **workbook id** in the iView. The
workbook id is easily found by launching the workbook from the Analyzer then
click on the icon *Workbook Settings* to see the workbook id at the top of the dialog.
Another important setting in the portal is to make sure the iVIew process the first
screen automatically. If you do not make this setting the transaction will stop at
the prompt for *RRMXP*, the user would then have to press enter to continue the
display the workbook.

- Create SAP Transaction iView
- Specify SAP GUI for Windows
- Launch RRMXP
- Specify Workbook ID (WBID=)
- Process First Screen = TRUE
- Avoid system messages
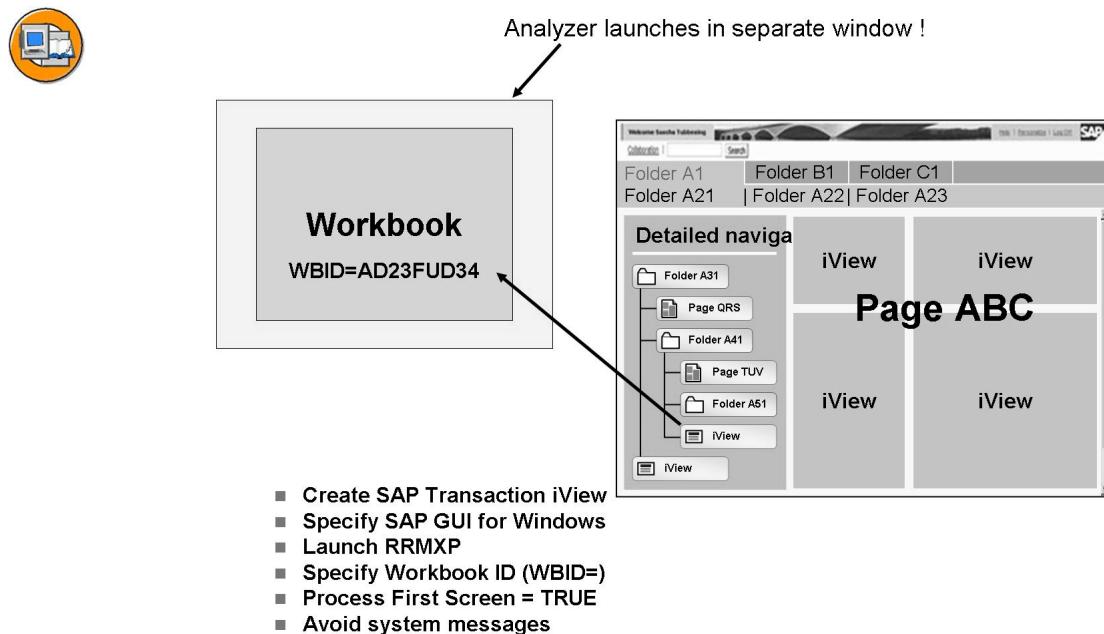
Figure 142: Publish Workbooks to Portal

Once you have created you iView to call the Analyzer with your workbook id you must add the iView to a role so that is can be accessed by the user. The procedure is that same for any iView, as described above.

Hint: If you have any system messages (as we do in SAP Training) you will see the pop up when the Analyzer starts from the portal. This is normal and in fact you may even prefer this. If you want a more seamless flow from portal to workbook display simply you need to remove the system messages. Use transaction *SM02* for this.

Caution: If your Analyzer starts but the workbook does not appear you may need to ask yourself which version of the Analyzer is being called. A BI workbook cannot be opened in the BW 3.x Analyzer and so if your settings determine that you always call the older Analyzer you will see no results. You can determine which verson of the Analyzer RRMX (or RRMXP) calls by making a setting in the IMG. Also the choice can be made per user by using a user parameter id (PID), and the setting also can be made in the Analyzer. Take a look at this note to learn more : https://service.sap.com/sap/support/notes/970002

Caution: Make sure you have a SAPGUI for Windows installed with the BI Front End, otherwise you cannot launch the workbook from the portal.

Exercise 16: Publishing BI Content to the Portal

Exercise Objectives

After completing this exercise, you will be able to:

- You will learn how to publish content from the BEx tools to the portal
- You will learn how to organize your BI content using the tools in the Portal Content Studio

Business Example

Your organization has recently implemented the SAP Portal and would now like to learn more about integrating the content from BI into the Portal.

Task 1:

Using the Portal Content Studio create a folder that will be used to store your own BI content.

1.   Logon to the Portal Content Studio using your assigned user details.

2.   Within the Portal Content Directory create a new folder under the path *Portal Content → Training → BW306*. The name of the folder should be **GR## BI Content** with the Folder ID name as **GR##BICONTENT** and the prefix as **training.bw306**.

Task 2:

Publish existing BI content from various BEx tools to the portal, use your own portal folder to store the content.

1.   Open the existing query *Divisional Performance Query (DIVISIONAL_PERFORMANCE_QUERY)* under role *BW306 Reporting → Unit 2 BEx Analyzer for Business Experts*. Publish your query to the portal folder you created in the first task then close the Query Designer.

2.   Open the existing web template *GR## Charts(GR##CHARTS)* which you should find under your *Favorites* or *History*. Publish the web application to the portal folder you created in the first task then close the Web Application Designer.

3.   Start the Report Designer and open the existing report *GR## Report 1 (GR##REPORT1)*. Publish this report to the portal folder you created in the first task then close the Report Designer.

Task 3:

You should now organize the BI content you just published in the portal.

1. Logon to the portal and from the Content Administration folder expand the Portal Content Directory until you reach the folder you created in task 1. (If your portal session is already open you can simply right-click on your folder *GR## BI Content* and use the option *Refresh*)

2. In your own folder create a new page. The page **GR## Page 1 (GR##PAGE1)** uses a single column layout as the default with a two column layout as a user personalization option. The page should present the iViews *Divisional Performance Query* at the top with a fixed height of 200 pixels and also the iView *GR## Charts* with the initial state as closed.

3. Create a second page **GR## Page 2 (GR##PAGE2)** to present the iView *GR## Report 1 (GR##REPORT1)* set to automatic height.

4. Within your folder create a new role *GR## Role (GR##ROLE)*. The role should contain five folders in total. One folder at the top level navigation marked as the entry point with the name *GR## My Reports (GR##MYREPORTS)*. Within that folder create two more folders for the second level navigation, one is named *GR## Daily Reports (GR##DAILYREPORTS)* and should be marked as the entry point. Within this folder created two more folders for the detailed navigation level. One folder will be named *GR## Team Reports (GR##TEAMREPORTS)* and should be assigned the page *GR## Page 1*. The other folder at for the detailed navigation should be named *GR## Region Reports (GR##REGIONREPORTS)* and has no pages assigned. The other folder at the second level navigation is named *GR## Weekly Reports (GR##WEEKLYREPORTS)* doesn't have any lower level folders, and should simply be assigned page *GR## Page 2*.

5. Finally, assign the new role to your user profile and test you can access your new page from the portal.

Solution 16: Publishing BI Content to the Portal

Task 1:

Using the Portal Content Studio create a folder that will be used to store your own BI content.

1.   Logon to the Portal Content Studio using your assigned user details.

   a)   From the Windows *Start* button select the shortcut *SAP Logon* and double click on your assigned training system.

   b)   Enter your assigned user details to log on.

   c)   Select the menu option *Access Portal* and enter your assigned user details to log on to the portal.

2.   Within the Portal Content Directory create a new folder under the path *Portal Content → Training → BW306.* The name of the folder should be **GR## BI Content** with the Folder ID name as **GR##BICONTENT** and the prefix as **training.bw306**.

   a)   Choose *Content Administration* on the first level navigation.

   b)   Expand the folder path *Portal Content → Training → BW306*.

   c)   Right Click on the folder *BW306* and select the menu option*New → Folder*.

   d)   In the field *Folder Name* enter **GR## BI Content**

   e)   In the field *Folder ID* enter the name **GR##BICONTENT**

   f)   In the field *Folder ID Prefix* enter **training.bw306**.

   g)   Press the button *Finish* and when prompted ensure the radio button is set to *Close the Wizard* and press *OK*.

Task 2:

Publish existing BI content from various BEx tools to the portal, use your own portal folder to store the content.

1. Open the existing query *Divisional Performance Query (DIVISIONAL_PERFORMANCE_QUERY)* under role *BW306 Reporting → Unit 2 BEx Analyzer for Business Experts*. Publish your query to the portal folder you created in the first task then close the Query Designer.

    a) Start the *Query Designer* using the path *Start → Programs → Business Explorer → Query Designer*. When prompted choose your assigned system and enter your assigned logon details.

    b) Click on the icon *Open Query* then select the *Role* button. Expand the folder path *BW306 Reporting → Unit 2 BEx Analyzer for Business Experts* . Highlight the query *Divisional Performance Query(DIVISIONAL_PERFORMANCE_QUERY)* and press the *Open* button.

    c) Select the menu option *Query → Publish → To Portal*

    d) Press the icon *Selection* which is to the right of the field *Folder*. and follow the path *PCD Root Folder → Training → BW306* . Click on the folder name*GR## BI Content* and press *Transfer* then press *Execute*. Close the confirmation window

    e) Select the menu option *Query → Exit and Discard Changes*.

2. Open the existing web template *GR## Charts(GR##CHARTS)* which you should find under your *Favorites* or *History*. Publish the web application to the portal folder you created in the first task then close the Web Application Designer.

    a) Start the Web Application Designer using the path *Start → Programs → Business Explorer → Web Application Designer*.

    b) Click on the toolbar icon *Open* and you should the template *GR## Charts(GR##CHARTS)* in the list. Highlight this template then press *OK* to select it.

    c) Select the menu option *Web Template → Publish → To Portal*

    d) Press the icon *Selection* which is to the right of the field *Folder*. and follow the path *PCD Root Folder → Training → BW306* . Click on the folder name*GR## BI Content* then press *Transfer* then press *Execute*. Close the confirmation window.

    e) Select the menu option *Web Template → Exit* to close the application.

3. Start the Report Designer and open the existing report *GR## Report 1 (GR##REPORT1)*. Publish this report to the portal folder you created in the first task then close the Report Designer.

   a) Start the Report Designer using the path *Start → Business Explorer → Report Designer* and enter you assigned user detail in the logon prompt.

   b) Press the toolbar icon *Open* then from the *History* highlight your report *GR## Report 1 (GR##REPORT1)* and press *Open*.

   c) Select the menu option *Report → Publish → To Portal*.

   d) Logon to the portal using your assigned user details.

   e) Click on the *Selection* icon to the right of the field *Folder* and expand the path *PCD Root Folder → Training → BW306* . Highlight your folder *GR## BI Content* then press *Transfer* and finally press *Execute*.

   f) Close the confirmation window and then close the Report Designer using the menu option *Report → Exit*.

Task 3:

You should now organize the BI content you just published in the portal.

1. Logon to the portal and from the Content Administration folder expand the Portal Content Directory until you reach the folder you created in task 1. (If your portal session is already open you can simply right-click on your folder *GR## BI Content* and use the option *Refresh*)

   a) Select *Start → SAP Logon* and double-click on the assigned training system. Enter your assigned user details.

   b) Logon to the portal using the menu option *Access Portal* and then click on the header folder *Content Administration*.

   c) Expand the Portal Content Directory folder structure *Portal Content → Training → BW306 → GR BI Content*.

2. In your own folder create a new page. The page **GR## Page 1 (GR##PAGE1)** uses a single column layout as the default with a two column layout as a user personalization option. The page should present the iViews *Divisional Performance Query* at the top with a fixed height of 200 pixels and also the iView *GR## Charts* with the initial state as closed.

    a)   Right click on your folder *GR## BI Content* and select the menu option *New → Page*.

    b)   In the field *Page Name* enter **GR## Page 1**.

    c)   In the field *Page ID* enter **GR##PAGE1**.

    d)   In the field *Prefix* enter **training.bw306** then press *Next*.

        Select the radio button *Default Page Template* and press *Next*.

    e)   From the list of layouts highlight *1 Column (Full Width)* and press the button *Add*. Also add the layout *2 Columns (Equal Width)*. You should now have two layout in the *Selected Layouts* list. Press *Next* .

    f)   Press *Finish* then press *OK* to continue editing the object.

    g)   Right click on the iView *Divisional Performance Query* and select the menu option *Add iView to Page → Delta Link*. Repeat this step for the iView *GR## Charts*.

    h)   Select the radio button *Page Layout* then using the drop down list *Show Layout* select the layout *2 Columns (Equal Width)*.

    i)   Drag the iView *GR## Charts* from column 1 to column 2.

    j)   Press the button *Save*.

    k)   Highlight the iView *Divisional Performance Query* then press the button *Properties* at the bottom of the screen.

    l)   From the drop down list *Property Category* select the entry *Appearance - Size* then for the property *Fixed Height (Pixels)* enter **200** and from the drop down list *Height Type* select *FIXED*. Press *Save*.

    m)   Highlight the iView *GR## Charts* then press the button *Properties* at the bottom of the screen.

    n)   From the drop down list *Property Category* select the entry *Appearance - Tray* then for the property *Initial State* select *Close*. Press *Save*.

    o)   Click the *Preview* button to ensure your page settings are working.

    p)   Click the button *Close* to close the page editor.

3. Create a second page **GR## Page 2 (GR##PAGE2)** to present the iView *GR## Report 1 (GR##REPORT1)* set to automatic height.

   a)   Right click on your folder *GR## BI Content* and select the menu option *New → Page*.

   b)   In the field *Page Name* enter **GR## Page 2**.

   c)   In the field *Page ID* enter **GR##PAGE2** .

   d)   In the field *Prefix* enter **training.bw306** then press *Next*.

   e)   Select the radio button *Default Page Template* and press *Next*.

   f)   From the list of layouts highlight *1 Column (Full Width)* and press the button *Add*. Press *Next* .

   g)   Press *Finish* then press *OK* to continue editing the object.

   h)   Right click on the iView *GR## Report 1* and select the menu option *Add iView to Page → Delta Link*.

   i)   Press the button *Save* then click the button *Close* to close the page editor.

4. Within your folder create a new role *GR## Role (GR##ROLE)*. The role should contain five folders in total. One folder at the top level navigation marked as the entry point with the name *GR## My Reports (GR##MYREPORTS)*. Within that folder create two more folders for the second level navigation, one is named *GR## Daily Reports (GR##DAILYREPORTS)* and should be marked as the entry point. Within this folder created two more folders for the detailed navigation level. One folder will be named *GR## Team Reports (GR##TEAMREPORTS)* and should be assigned the page *GR## Page 1*. The other folder at for the detailed navigation should be named *GR## Region Reports (GR##REGIONREPORTS)* and has no pages assigned. The other folder at the second level navigation is named *GR## Weekly Reports (GR##WEEKLYREPORTS)* doesn't have any lower level folders, and should simply be assigned page *GR## Page 2*.

   a)   Right click on your folder *GR## BI Content* and select the menu option *New → Role*.

   b)   In the field *Role Name* enter **GR## Role**.

   c)   In the field *Role ID* enter **GR##ROLE**.

   Press the button *Next* then press the button *Finish*.

   d)   Leave the radio button selected with *Open the Object for Editing* and press *OK*.

e)   Right click on the role icon at the top of the tree and select the menu option *New Folder*. Enter the folder name **GR## My Reports** with the technical name **GR##MYREPORTS**.

f)   Highlight your new folder then press the button *Properties*. On the far right of the screen you should see the property *Entry Point*. Set this to *YES*.

g)   Right click on your new folder and select the menu option *New Folder*. Enter the folder name **GR## Daily Reports** with the technical name **GR##DAILYREPORTS**.

h)   Highlight folder *GR## Daily Reports* then press the button *Properties*. On the far right of the screen you should see the property *Entry Point*. Set this to *YES*.

i)   Right click on the folder *GR## Daily Reports* and select the menu option *New Folder*. Enter the folder name **GR## Team Reports** with the technical name **GR##TEAMREPORTS**.

j)   Ensure folder *GR## Team Reports* is highlighted by clicking on it once, then over in the navigation panel (far left) right click on the page *GR## Page 1* and select the menu option *Add Page to Role → Delta Link*. The page should appear under the folder.

k)   Right click on the folder *GR## Daily Reports* and select the menu option *New Folder*. Enter the folder name **GR## Region Reports** with the technical name **GR##REGIONREPORTS**.

l)   Right click on folder *GR## My Reports* and select the menu option *New Folder*. Enter the folder name **GR## Weekly Reports** with the technical name **GR##WEEKLYREPORTS**.

m)   Ensure folder *GR## Weekly Reports* is highlighted by clicking on it once, then over in the navigation panel (far left) right click on the page *GR## Page 2* and select the menu option *Add Page to Role → Delta Link*. The page should appear under the folder.

n)   Click on *Save*.

    5.   Finally, assign the new role to your user profile and test you can access your new page from the portal.

        a)   Click on the folder *Delegated User Administration* then in the *Search Criteria* text field enter you user id**BW306-##** and press *Go*.

        b)   Your user profile should appear in the list, now press the push button to the left of your user id to open up the *Details* section.

        c)   Press the button *Modify* then in the *Search Criteria* text field on the left side of the screen enter **GR##***. The role you build earlier should appear.

        d)   Press the button *Add* at the bottom of the screen. The role should appear under your user on the right side of the screen. (yo may have to scroll to see the full detail).

        e)   Press the button *Save* towards the top of the screen.

        f)   Refresh the browser to reload the portal. You should see you new folder in the header, click on this and navigate around the pages you created.

Lesson Summary

You should now be able to:
- Publish BI content in the portal to iViews and organize the iViews in the Portal Content Directory using folders and pages.

Unit Summary

You should now be able to:

- Publish BI content in the portal to iViews and organize the iViews in the Portal Content Directory using folders and pages.

Unit 7

Building Sophisticated BI Reports

Unit Overview

This unit provides an opportunity for you to use what you've learned to this point to solve a realistic business analysis problem. You will be able to choose your approach to the solution as well as the particular aspects of the solution that interest you most.

Unit Objectives

After completing this unit, you will be able to:

- Translate business requirements into BI solutions using the BEx Suite
- List the strengths and weaknesses of your solution, and gain more ideas of using alternative methods for developing the solution

Unit Contents

Lesson: Developing the BI Solution

Lesson Overview

This lesson will provide you with an opportunity to exercise your newly learned skills in BI reporting. You will be given data and a base query from which to develop your own workbooks, formatted reports and web applications. You will then present your results through the portal. The task you are being asked to perform are very loosely described so that you can enjoy the freedom of making your own design choices.

Lesson Objectives

After completing this lesson, you will be able to:

- Translate business requirements into BI solutions using the BEx Suite

Business Example

You would like to practice building end to end BI solutions based on business requirements.

Case Study

The following diagrams will be presented by the instructor.

- You are hoping to join a formula one racing team as Racing Data Analyst
- As part of your job interview you have been asked to produce a number of reports based on racing data using a state of the art BI application. The management team have been well trained at SAP and are familiar with the capabilities of the BEx Suite. You need to impress them with your BEx knowledge and report creativity. You will be awarded points according to your efforts in three areas :
  - Use of advanced BEx functionality
  - Presentation
  - Completeness of the task

InfoCube

Attributes

Racing Land (Nav)
Degree of Difficulty (Nav)
Lap Record
Number of Laps
Lap Distance
Number of Curves

Characteristics

Racing Course
Racing Driver
Start Position
Placing
Calendar Day

Attributes

Marital Status

Key Figures

Number of Accidents
Racing Score
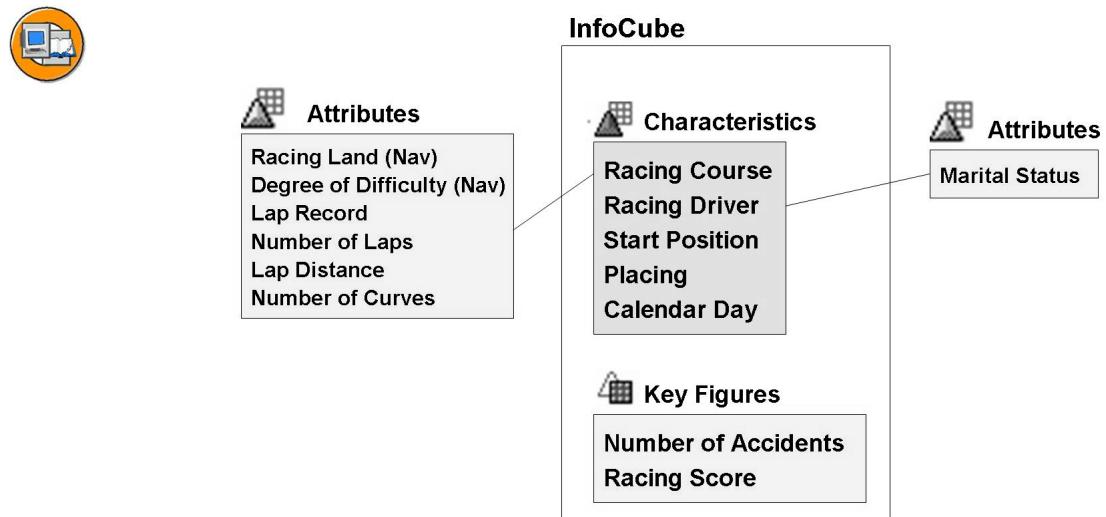
Figure 143: The Race Data

- Based on the query General Race Query develop a professional and flexible workbook for 'Race Accident Analysis' which should include :
    - The ability to select any combination of driver, course, level of difficulty and race date using only navigation control items (switch off context menus). Make sure the report navigation activity does not interfere with any other part of the report.
    - Include a summary table in the workbook that highlights the driver, course and date where the most accidents occurred. Use the formula mode to select and position the data.
    - Add logos, images, titles and a clean and well laid out report.
    - Make room for a header area of the workbook so you can present a list of query filter values and include date of data refresh and query refresh and any other useful pieces of information.

- Based on the query Driver Static Query develop a well formatted report to present the driver final scores. The report should include :
    - A well organized layout making use of the ability to reposition the cells freely
    - Addition of logos, images and titles – feel free to gather more images (top drivers)
    - Conditional formatting for the best/worst drivers

- Based on the query General Race Query develop a web application to allow analysis of driver performance. The application should include :
    - At least one analysis item which uses only navigation controls for limited analysis
    - A chart and also the ability for the user to modify the chart properties
    - The formatted report you created earlier
    - All three items should appear on individual tab pages
    - A well designed application including images, titles and formatting

- Present your three reports to the users via the portal
- Develop iViews and pages as required
- Create a new role add the pages to you role then assign the role to your own user ready for presentation

420   2007/Q2

- **Case study role is** BW306 Reporting > Unit 6 Building Sophisticated BI Reports
- **All predefined objects are available in the case study role**
- **Save all** *your* **work to the case study role using your group number in the name**
- **Make a copy of the shared query** General Race Query**, please do not modify original query**
- **Plenty of graphics available in workbook** BW306 Case Study Graphics **or use your own**
- **Two hours to develop solution – then we have a presentation session for one hour**
- **Allocate tasks to your team or work together - be ready to present (as a team or individual)**
- **Use the time guide below (unless you split up tasks) to help gauge your progress:**

Task	Minutes
Workbook	30
Report	30
Web Template	40
Portal Development	20

Figure 144: Case Study Guidelines

Lesson Summary

You should now be able to:
- Translate business requirements into BI solutions using the BEx Suite

Lesson: Presenting the Solution

Lesson Overview

It is now time to present your development to the rest of the class !

Lesson Objectives

After completing this lesson, you will be able to:

*   List the strengths and weaknesses of your solution, and gain more ideas of using alternative methods for developing the solution

Business Example

You want to share your BI development experiences with other team members and also want to learn from their ideas. To do this, you need to make a list of the strengths and weaknesses of your solution.

Presenting your solutions

When it is your turn be ready to go up to the front of the class and use the instructor's pc to show everyone your solutions. You may want to do this alone or with other members of your team. It is up to you how you present your solutions. Not all solutions will be the same, try to use this session to pick up ideas and tips for developing BI content.

*   **It is now time to present your solutions to the rest of the class**

*   **Each team will be asked to come up to the front**

*   **Try to pick out the unique aspects of your design**

Figure 145: Presenting the Solutions

Based on the number of teams in the class the instructor will allocate a time slot for each group. Try to keep to this so everyone has an equal chance to present their solutions.

Lesson Summary

You should now be able to:
- List the strengths and weaknesses of your solution, and gain more ideas of using alternative methods for developing the solution

Unit Summary

You should now be able to:

- • Translate business requirements into BI solutions using the BEx Suite
- • List the strengths and weaknesses of your solution, and gain more ideas of using alternative methods for developing the solution

2007/Q2   **425**

Unit 8
xApps Visual Composer

Unit Overview

NetWeaver 2004s provides a new integration toolkit for BI and the Visual Composer. This makes it even easier than before to bring BI data into highly interactive Visual Composer analytical applications.

Unit Objectives

After completing this unit, you will be able to:

- Explain the concept of xApps Analytics and how the Visual Composer is used to build analytical applications across system boundaries using services based architecture
- Explain how to integrate BI Content into the Visual Composer Model
- Describe how to use the BI Integration Wizard to build queries from templates and freeform mode

Unit Contents

Lesson: Overview of Visual Composer

Lesson Overview

This lesson will provide an introduction to the SAP NetWeaver Visual Composer. We will begin by explaining xApps - Analytics and then move on to describe the Visual Composer.

Lesson Objectives

After completing this lesson, you will be able to:

- Explain the concept of xApps Analytics and how the Visual Composer is used to build analytical applications across system boundaries using services based architecture

Business Example

The management team of your organization would like you to present a high level overview of the Visual Composer and explain how it fits with NetWeaver 2004s BI.

What are Analytical Applications ?

Before we introduce the Visual Composer it is important we understand the basic concept of analytical applications. Analytical Applications (or Analytics for short) provide business users with the ability to execute processes in context alongside tactical and strategic information **presented within a single application**. This is also known as a **Composite Application**. Using very plain language this could also be described as combining the functions from a transactional system together with the relevant reporting information right on the same screen. Now you could say, in order to achieve this why don't we just allow users access to the required transaction system functions and in addition for the analysis requirements we should give all users access to SAP NetWeaver BI reporting tools. Ever heard of the expression 'using a hammer to crack a walnut' ? Quite often the analysis does not need to be complex, and so the power of BI and accompanying OLAP tools are not always required. If I was put in charge of ordering stock and I only need some basic demand history information to hand, I certainly do not need a complex, state of the art BI tool to provide this to me. What I would really want is the ability to check the recent stock consumption and current stock level and forecast together with the ability to place orders **all one one screen**. I certainly do not want to have to switch multiple business systems in order to obtain all the information I need to execute a single business process. I need an analytical application.
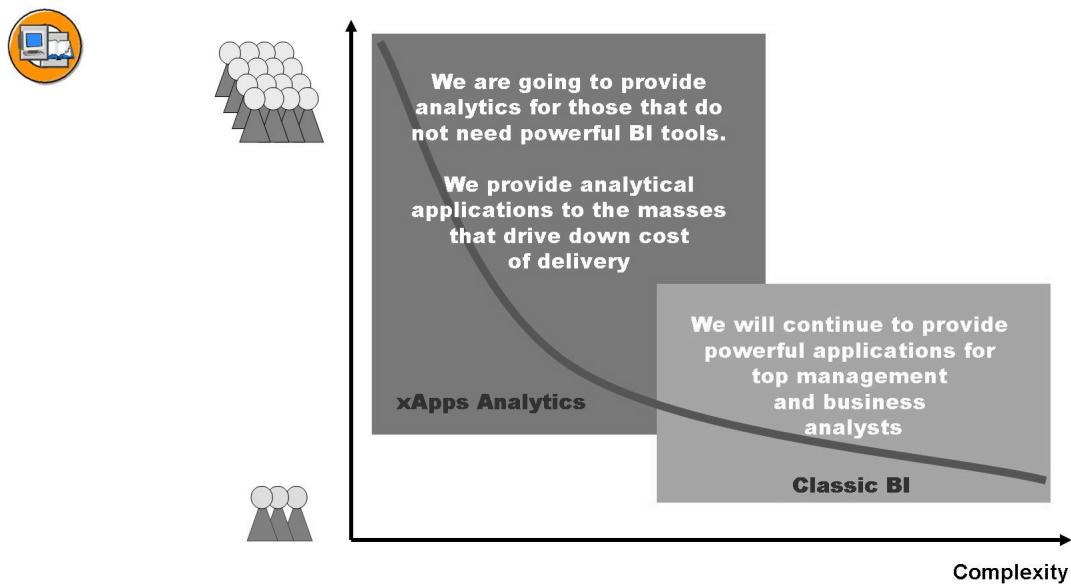
Figure 146: Analytics and classic BI

In the diagram below we see the transactional world separated from the analytical world. The CRM application is quite separate from the BI application, this means that any decision we make based on our analysis needs to be carried out in a second step. Imagine the scenario, you want to reclassify your customer's rebate category based on their recent spending behavior. You first need to study the historical sales data from your analytical systems, you then need to log on to the CRM system to access the customer's master data in order to change the rebate category. You can see this is not efficient and emphasizes the separation between the transaction based systems (CRM) and analytical systems (BI). What we need is a single presentation of both analytical and transactional systems so that we improve the speed and ease of action taking following analysis.
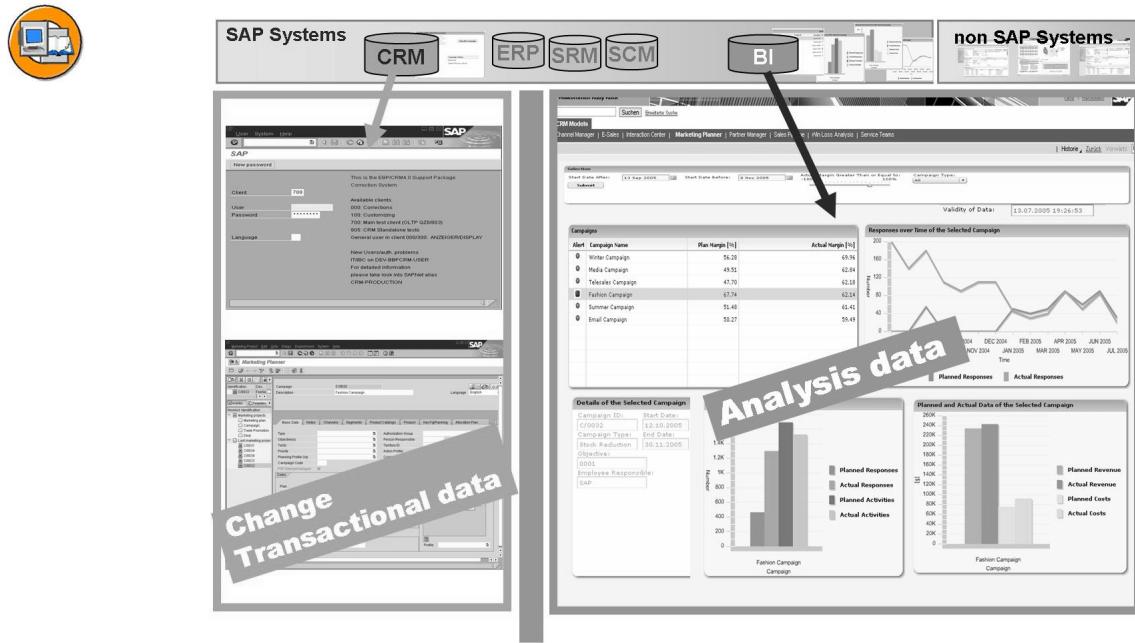
Figure 147: Composite application without ability to take action

The diagram below now shows what a composite application looks like, notice the separation of the CRM system has disappeared. We now have the ability to change the customer master data directly in the analytical application.
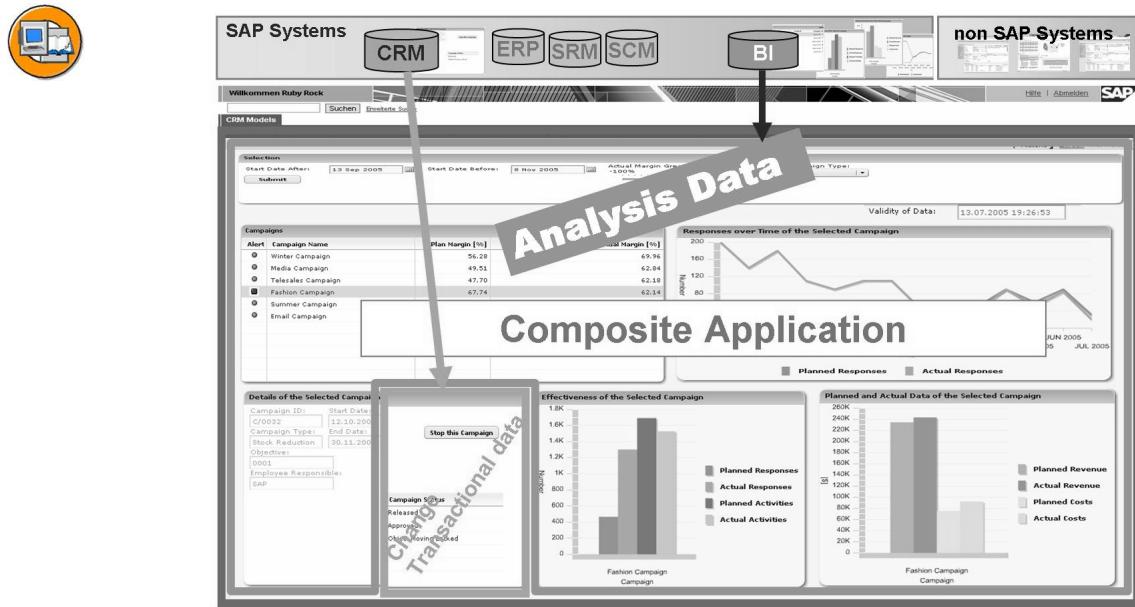

Figure 148: Composite application with ability to take action

Overview of Visual Composer

SAP NetWeaver Visual Composer is a component of xApps Analytics and allows you to compose model-based business applications in a flexible way, without manual coding. This includes **analytical applications**. With Visual Composer, you create pattern-based or freestyle user interfaces using simple drag-and-drop services and then define the data flow between them. Because SAP NetWeaver Visual Composer is model-driven, you can compose an application without having to write a single line of code. When a model is deployed, SAP NetWeaver Visual Composer translates the model into the required code for the application.

SAP NetWeaver Visual Composer is a Web-based visual modeling tool that enables rapid, code-free development of highly customizable portal applications by a business analyst

Designtime

Deploy

Runtime

Figure 149: What is Visual Composer ?

When it is time to build an analytical application with the Visual Composer there are a series of steps that must be worked through. The diagram below illustrates the work flow.
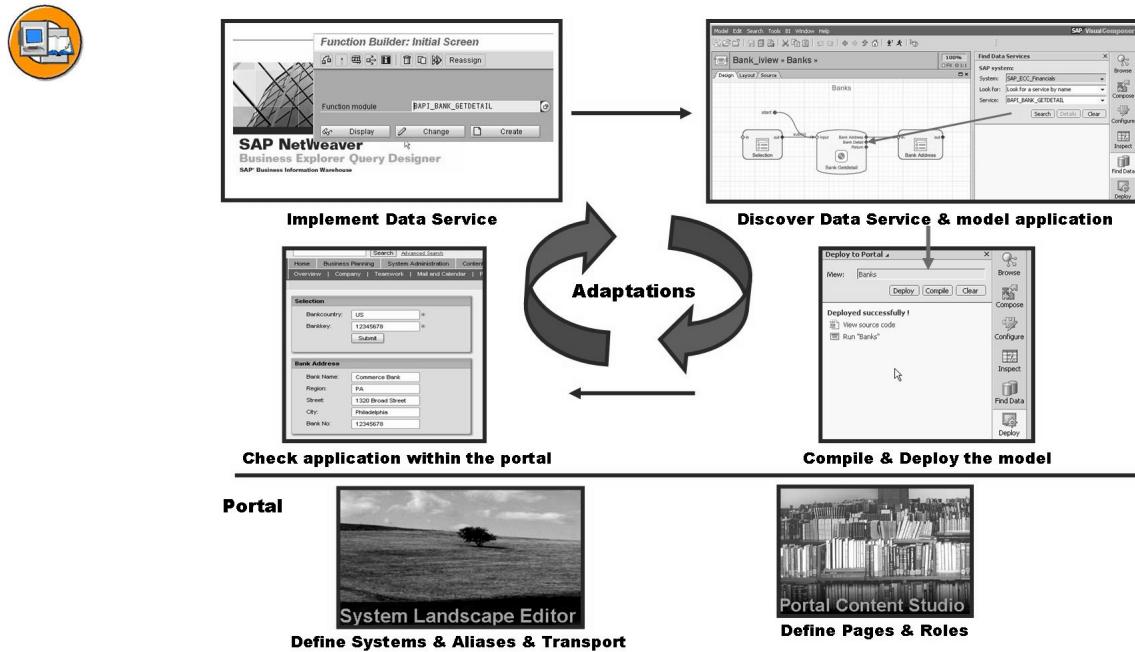
Figure 150: Model creation workflow

First, it must be distinguished between design-time (working environment for SAP NetWeaver Visual composer for 2004s to create models and design model-based applications), and the runtime (working environment for end users to execute the applications, e.g. accessing the NetWeaver Portal and execute iViews according to their roles).

The design-time is mainly driven by the Visual Composer Storyboard: This is the working area of the Visual Composer and is executed in a Web-Browser. In addition, the storyboard uses the MS XML Parser, Adobe SVG, and the Adobe Flash client as additional software units.

The Adobe SVG Plugin is used to manipulate, scale and manage graphics in the Visual Composer storyboard. The XML Parser is used to manage the models in the tool properly. All models are entirely described in XML as an open markup language. The Flash client is used at the design-time to check the results in the preview (e.g. see how a previously modelled iView looks like) The runtime requires a Web Browser according to SAP's Product Availability Matrix (PAM).

432   2007/Q2
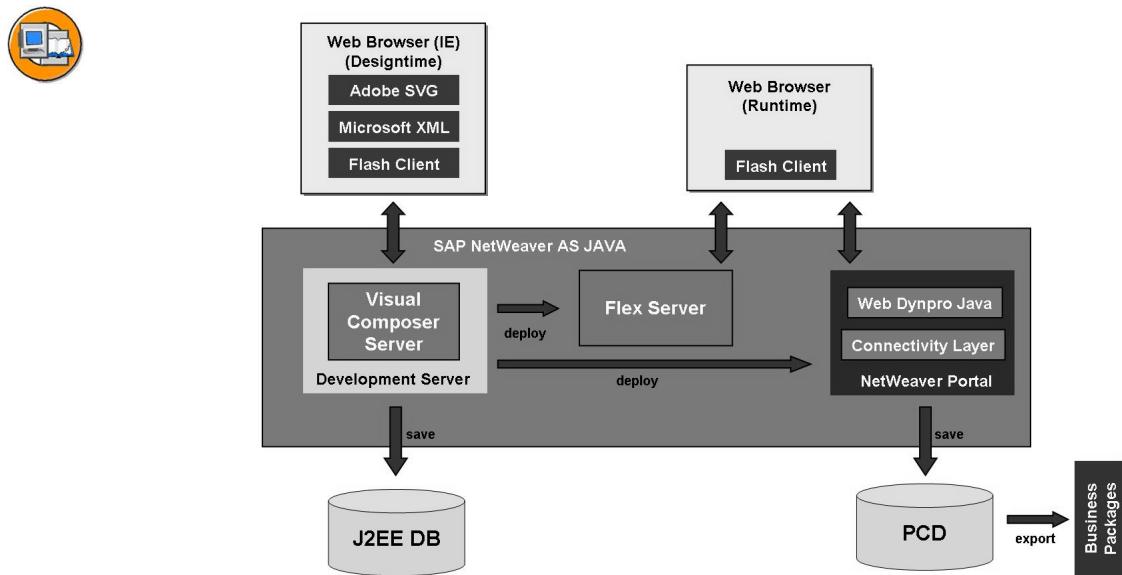
Figure 151: Visual Composer Architecture

At server-side, SAP NetWeaver Visual Composer is executed in the runtime environment of the SAP NetWeaver Application Server based upon the Java stack. The Visual Composer Server is here the central component. When a model is created, the result is stored in the J2EE database and at file level as XML description. The content developer can then decide, which runtime will be used. The model remains without changes, but the iView will be created for Flash or for WebDynpro for Java as runtimes.

After the model is deployed, the iView will be transported to the SAP NetWeaver portal, where it can be treated like any other portal content. For example, a particular iView can be used as a template, translated, or transported.

Lesson Summary

You should now be able to:

- Explain the concept of xApps Analytics and how the Visual Composer is used to build analytical applications across system boundaries using services based architecture

Related Information

- The class ANA10 SAP xApps Analytics covers the Visual Composer in more detail. The class is currently 3 days.

Lesson: Visual Composer BI Integration

Lesson Overview

The lesson describes the tight integration between the Visual Composer and
SAP BI.

Lesson Objectives

After completing this lesson, you will be able to:

- Explain how to integrate BI Content into the Visual Composer Model
- Describe how to use the BI Integration Wizard to build queries from
  templates and freeform mode

Business Example

Your organization would like to understand more about integration between Visual
Composer and BI

BI Kit

Let's start with the complete name for the BI Kit: BI extension Kit for the SAP
NetWeaver Visual Composer. The BI Kit is a plug in and it enables users access
to a variety of data sources. The core Visual Composer allows only the use
of Remote Function Calls (RFC) and Web Services (both SAP and non SAP
systems). With the BI Kit, additional types of data sources include JDBC sources,
ODBO, XMLA and SAP BI. To increase efficiency, wizards are included in the BI
Kit. This way, data can be retrieved quite easily and in an integrated application.
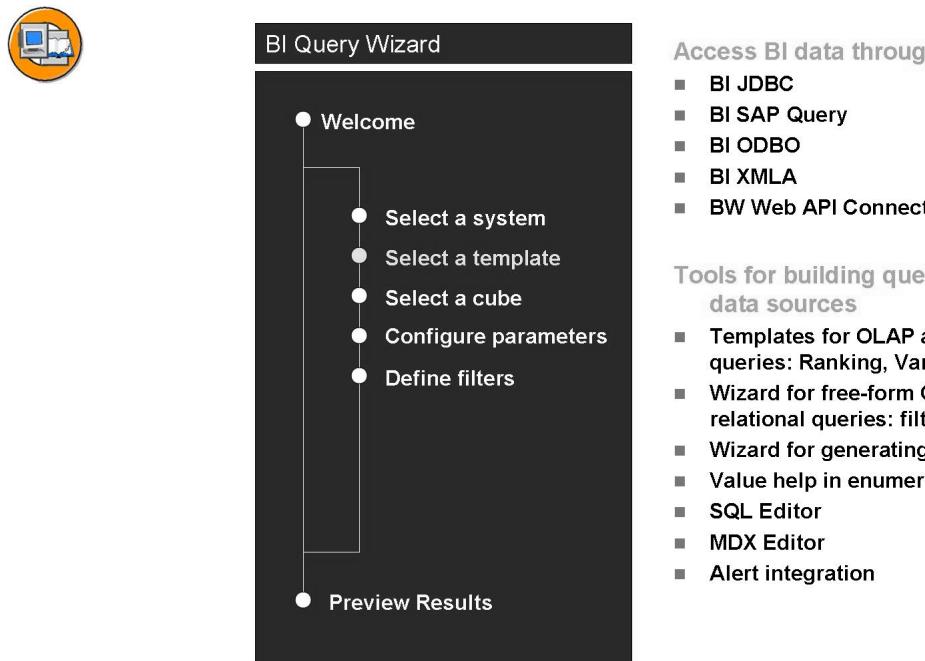
Figure 152: BI Kit for Visual Composer

The diagram below provides a list of the connectors that are supplied with the BI
Kit. Notice the split between connector used to extract relational data sources
(such as SAP ECC tables) and multi dimensional data sources (such as BI
InfoCubes). Also notice how some connectors are platform independent.

	Connector	Technology Based on	Provides Access To
Relational (OLTP)	BI JDBC	Sun's JDBC – the standard API for RDBMS	Connectivity to over 200 JDBC drivers (examples including: DB2, Microsoft SQL, Server, Oracle, Microsoft Access)
	BI SAP Query	SAP Query	SAP operational applications (Infosets from transaction SQ02)
Multidimensional (OLAP)	BI ODBO	Microsoft's OLE DB for OLAP – industry standard OLAP API for Windows	OLE DB for OLAP compliant data sources (examples including: Microsoft Analysis Services, SAS, Microsoft PivotTable, Services and SAP BW)
	BI XMLA	Microsoft's XMLA – Web Services based access to OLAP providers	Platform-independent access to OLAP data sources (examples including: SAP BW 3.x, MS Analysis Services, Hyperion, MicroStrategy)
	BI Web API	SAP RFC and SAP BW Web API	SAP BW queries, query views and InfoObjects

Figure 153: Connectors provided with BI Kit

Below is an overview over the BI System requirements. In order to setup the various types of data connections, some connectors are necessary however not all systems have to be connected. You can choose whether you would like to connect a JDBC system, a XMLA system or other systems.

- Visual Composer Installation instruction contains required BI Components
    - BI UDI (contains BI Java Connectors)
    - VC BI Kit Portal add-ons
- BI JDBC system
    - Requires JDBC driver (normally comes with RDBMS, not included with VC installation)
    - Supports any JDBC compliant DB
- BI SAPQ system
    - Requires SAP R/3 4.6C or greater
- BI XMLA system
    - Requires XMLA-compliant OLAP data sources, such as SAP BW 3.0A or greater
- BI ODBO system
    - Requires Microsoft ODBO-compliant data sources, such as SAP BW 1.2B or greater
    - Requires Windows platform on J2EE server
    - Requires Microsoft Data Access Components (MDAC) 2.6 or greater
- BW Web API system
    - Requires BW 3.5 SP14 or greater

 SAP

Lesson Summary

You should now be able to:

- Explain how to integrate BI Content into the Visual Composer Model
- Describe how to use the BI Integration Wizard to build queries from templates and freeform mode

Unit Summary

You should now be able to:

- Explain the concept of xApps Analytics and how the Visual Composer is used to build analytical applications across system boundaries using services based architecture
- Explain how to integrate BI Content into the Visual Composer Model
- Describe how to use the BI Integration Wizard to build queries from templates and freeform mode

Migration of Reporting Objects

Unit Overview

Customers upgrading from SAP BW 3.x to NetWeaver 2004s BI must plan for the migration of any BW 3.x reporting objects to the corresponding NetWeaver 2004s-level objects. This unit focuses on the important points to know before formulating your plans.

Unit Objectives

After completing this unit, you will be able to:

- Describe the migration process for BW 3.x queries.
- Describe the migration process for BW 3.x web templates
- Describe the migration process for BW 3.x workbooks
- Describe the migration process for BW 3.x broadcast settings

Unit Contents

Lesson: Migration of Reporting Objects

Lesson Overview

The migration of reporting objects such as queries, web templates, workbooks and broadcaster settings is a large task facing any customer upgrading from SAP BW 3.x to SAP NetWeaver 2004s BI. This lesson covers the important facets of migrating those objects to take advantage of the many new features offered by the BI environment.

Lesson Objectives

After completing this lesson, you will be able to:

- Describe the migration process for BW 3.x queries.
- Describe the migration process for BW 3.x web templates
- Describe the migration process for BW 3.x workbooks
- Describe the migration process for BW 3.x broadcast settings

Business Example

You are an existing SAP BW 3.x customer and have implemented a collection of queries, workbooks, web templates and broadcast settings in your productive environment. You would like to reap the benefits offered in the Reporting, Query and Analysis scenario for NetWeaver 2004s BI and so must plan the migration of your reporting objects to the new format.

Migration of BEx Reporting Objects

Customers who have implemented versions of BW prior to NetWeaver 2004s BI must consider how they will choose to implement the many new functions available as well as how existing reporting objects will be migrated, where necessary, as they roll out their BI implementation. This is necessary because the definition, structure and capabilities of reporting objects has, in many cases,

changed with the new BI release. Reporting objects include queries, workbooks, web templates, broadcast settings, and global objects like structures, restricted key figures and calculated key figures.

- Goals
    - Minimize complications
    - Avoid user confusion. Do not ask them to use multiple tools at the same time
    - Reduce worries about backwards compatibilities or accidental migrations
- Strategy
    - Simplified migration path
    - Small, highly defined phases
    - Focus on short term projects that can add immediate benefit to the organization

Advanced planning of this process is key to a successful implementation. The best plans seek, as the graphic above states, to minimize any complications for the users by simplifying the tools they must use while guarding against potential mistakes during the migration process.

In light of this, it is wise to develop a migration plan that is both simple and well defined so as to avoid potential communication problems. If you are able to limit the scope of the rollout of the new BI tools to a well-defined audience, then the project will be easier to control.

2007/Q2 **443** SAP

With SAP NetWeaver 2004s, new tools and a new runtime are available in the Business Explorer (BEx). At the same time, all of the previous tools and the web runtime from SAP BW 3.5 are delivered in order to guarantee a step-by-step, demand-oriented transition.

- New BEx Tools in SAP NetWeaver 2004s

  - New front end features ONLY available with new NetWeaver 2004s BEX tools
  - Objects created with new BEx tools can no longer be edited with 3.x BEx tools

- SAP BW 3.x Tools on top of SAP NetWeaver 2004s BI

  - The 3.x BEx tools are delivered to support editing of existing scenarios
  - The 3.x BEx tools are usable on the new SAP NetWeaver 2004s BI server

- Transition from SAP BW 3.x to SAP NetWeaver 2004s BEx Tools

  - Conversion of 3.x objects is done on an 'as needed' basis. No mass conversion
  - Converted objects are stored as new objects. Old objects are never automatically deleted
  - The recommendation is to make the change step by step, because further new features will only be made for the new tools

NetWeaver 2004s BI is shipped complete with both the BEx toolset for NetWeaver 2004s BI and the BEx toolset for BW 3.5. To support these two toolsets, two distinct runtimes are delivered within the NetWeaver 2004s BI system. The new BI toolset offers many new and important features, and these tools are the only ones that can use the new frontend functionality in BI. These tools are all members of the Business Explorer (BEx) family and include the BEx Query Designer, BEx Analyzer, BEx Web Application Designer and BEx Report Designer.

Objects that have been created with the new toolset cannot be further maintained with the BW 3.5 toolset. However, the BW 3.5 toolset is provided within the NetWeaver 2004s BI frontend to allow customers to continue to use and maintain their existing Prix reporting objects. It is possible to work with both toolsets at the same time, although you can also block access to one toolset or the other if you would like to simplify your migration process and reduce the risk of accidental migration of reporting objects. This will be discussed in more detail later in this lesson.

The migration process for a reporting object takes place when a BW 3.x reporting object is opened using the relevant new BI tool and then saved. The details of the migration process will be discussed throughout this lesson and are specific to the type of reporting object being migrated.

- All existing SAP BW 3.x queries can be edited with the SAP NetWeaver 2004s BEx Query Designer without further manual adaptation
- After editing with the new tool queries cannot be edited with the SAP BW 3.x BEx Query Designer anymore
- Queries created or adapted with the SAP NetWeaver 2004s BEx Query Designer will still appear in the Open Dialog of the SAP BW 3.x tool but they cannot be opened anymore by that version of the tool
- For those scenarios were customers do not want to use the SAP NetWeaver 2004s BEx Query Designer, SAP ships an SAP BW 3.x version of the BEx Query Designer additionally to the SAP NetWeaver 2004s version
- New capabilities are only implemented in the SAP NetWeaver 2004s BEx Query Designer
- Query Views that were created before will still run after a query has been changed with the new BEx Query Designer
- In general, query views do not need to be migrated

The BEx Query Designer is the only tool in NetWeaver 2004s BI that allows the creation and maintenance of query objects. The BI version of the BEx Query Designer has been totally rewritten as a .NET Visual Basic application, complete with a redesigned user interface and many new features and capabilities. New query capabilities offered with NetWeaver 2004s BI are only available through the use of the new NetWeaver 2004s BEx Query Designer.

Opening a BW 3.x query with the new BEx Query Designer and then saving that query will migrate that query from the BW 3.x version to a new BI version of that query. Once this is done, it is no longer accessible by the BW 3.x BEx Query Designer, although it will continue to be visible in the 'Open' dialog of that tool.

Query views created in BW 3.x do not need to be migrated and will continue to perform as before whether accessed from new or old versions of the BEx tools.

SAP NetWeaver 2004s – Maintenance and Migration of Query Objects

Query Designer 3.x

Query Designer 2004s

*

upgrade

open   save   open   save
(Migrated queries stored with same name but new version to existing database table)

Queries | Queries
3.x | 2004s

* Visible but can not be opened
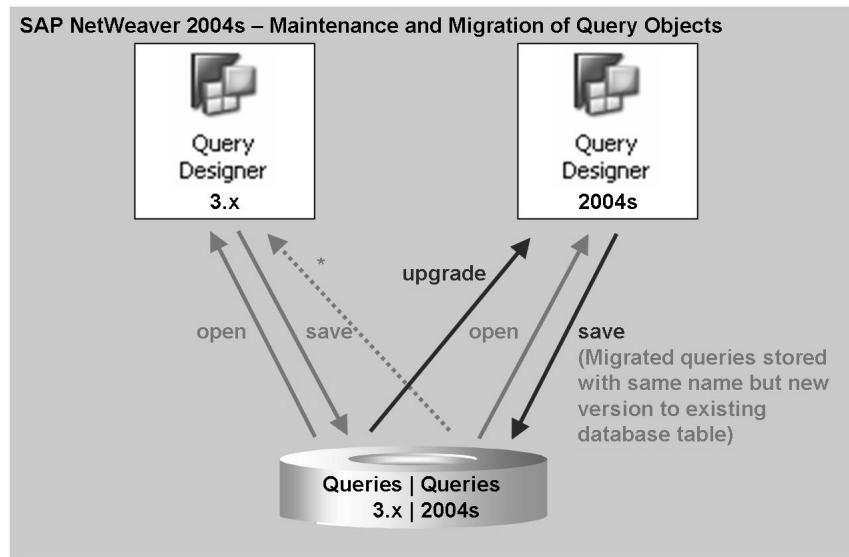
Figure 154: BEx Query Designer – Maintenance and Migration

The graphic above shows the possible scenarios with both BW 3.x and BI versions of queries. Each version of a query can be opened, maintained and saved using the respective version of the BEx Query Designer. No migration of BW 3.x queries will occur as long as they are only opened with the BW 3.x BEx Query Designer. This allows existing query scenarios to continue to perform unchanged.

Migration of BW 3.x queries is initiated by opening those queries with the NetWeaver 2004s BEx Query Designer and then saving that query. The query will be saved with the same internal name but with a different version. Table RSZCOMPDIR contains a listing of query objects, their internal names, and their

content release level. A value of 100 or greater in the *VERSION* field for a query indicates that the query has been migrated to NetWeaver 2004s. This table can be used to track which queries have been migrated to date.

- Upgrade of Queries
    - Optional, but necessary to use any of the new functionality
    - Migration occurs when BW 3.x query is opened and saved with new BEx Query Designer
    - See note 962530 for how to restrict access to the Query Designer
    - See note 949330 regarding query backup and restore
- Caution
    - Once a query is upgraded, it can not be opened anymore with the old tool
    - If a reusable component (variable, restricted or calculated key figure) is changed in the new tool, then all queries that use this component can only be maintained with the new BEx Query Designer

In summary, the migration of queries from BW 3.x to NetWeaver 2004s enables the use of the enhanced functionality of the product, but can be done in a planned progression of steps in your company.

As mentioned in the graphic above, note 962530 describes a technique for restricting access to the NetWeaver 2004s BEx Query Designer. This will be useful in the early phases of the migration process before the new tools are rolled out to the users, and is an effective way to prevent the accidental migration of queries.

2007/Q2   **447** SAP

Additionally, note 946330 describes the process of backing up BW 3.x queries before migration. This provides a way to return to the old version of the query if necessary.

- SAP ensures that existing customer scenarios can continue to be edited with the BW 3.x Web AD shipped in NetWeaver 2004s along with the new Web AD
- In addition to many new items in the SAP NetWeaver 2004s Web AD, some BW 3.x items are no longer available
- Examples

  – The Role Menu item is replaced by Portal functionality

  – The Alert Monitor is replaced by the Universal Worklist (Portal feature)

- A migration is started using the SAP NetWeaver 2004s Web AD and then calling the migration tool using the *Tools –> Migration Tool* path
- BEx Web applications in which certain customer-specific enhancements (e.g., Table Interface, custom Java-Script) were made cannot be automatically converted. Manual adaptation might be necessary
- After saving with the NetWeaver 2004s Web AD, the migrated version of a web application **can no longer be used** in the BW 3.x BEx Web AD. Migrated versions will not appear in the open dialog of the old tool
- After migration, the old version of a web application is still available in the BW 3.x BEx Web AD. The old version will **still appea**r in the open dialog of the old tool. Migration can be repeated since old web applications are not deleted
- New capabilities are only implemented in the SAP NetWeaver 2004s Web AD
- Web templates for the new BEx Web runtime can only be created with the new BEx Web AD

The BEx Web Application Designer (Web AD) for NetWeaver 2004s has been redesigned as a .NET Visual Basic application and is delivered with many new features and functions to enhance web applications in your BI environment. Much like the BEx Query Designer, SAP delivers a BW 3.5 version of the Web AD to support the use and maintenance of existing BW 3.x web applications.

The new version of the Web AD adds several new web items, but some of the BW 3.x web items are no longer available. This change means that some rework of existing web templates may be necessary when they are migrated to NetWeaver 2004s. Likewise, JavaScript functions and enhancements made with the Table Interface in BW 3.x will need to be manually adapted .

A BW 3.x web template is migrated by opening the NetWeaver 2004s Web AD and using the *Tools → Migration Tool* path to start the *SAP BW 3.x to SAP NetWeaver Migration Tool*. This tool carries the migration process through four steps:

1. Select the web template to migrate.
2. Make the desired choices for conversion settings and start the migration.
3. Watch the progress of the migration.
4. Close the migration tool and return to the Web Application Designer to edit and adapt the results.

If the BW 3.x web template contains items that are not supported in the new BI runtime, then they are removed in the migrated version with comments explaining the reason for omitting the old item shown in the XHTML view of the new template.

When the new template is saved, it is stored with a new internal name and status in a separate database table. This leaves the original version of the web template intact so that it can be migrated again, if necessary. The new version of the web template will not be accessible from the SAP BW 3.x BEx Web AD.
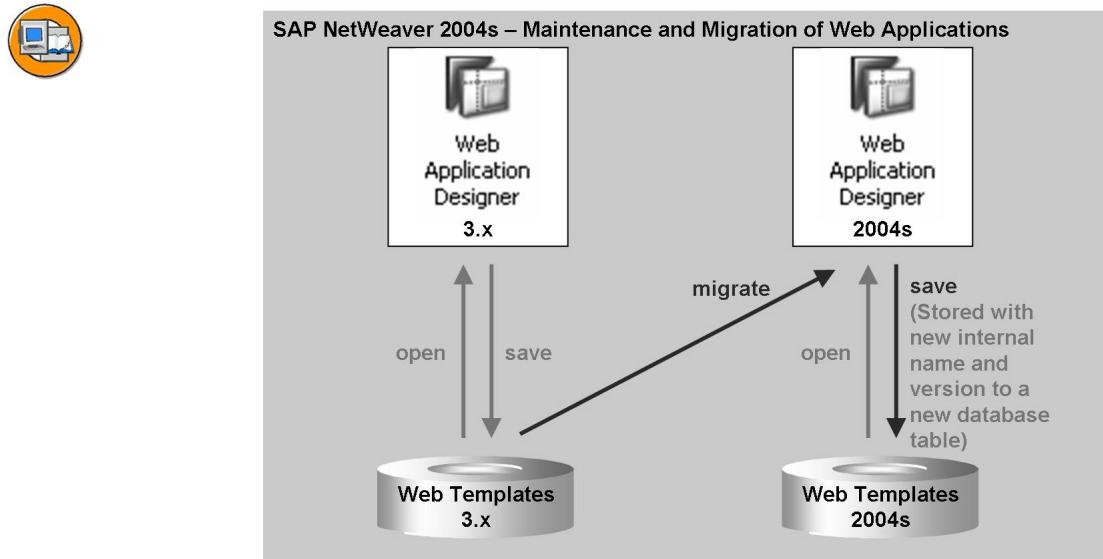

Figure 155: BEx Web Application Designer – Maintenance and Migration

Possible scenarios with both BW 3.x and NetWeaver 2004s versions of web templates are shown above. Each version of a web template can be opened, maintained and saved using the respective version of the BEx Web Application Designer. BW 3.x web templates will not be migrated as long as they are only opened with the BW 3.x BEx Web Application Designer. Existing web templates can thus continue to be used and will run in the BW 3.x runtime.

To migrate a BW 3.x web template, the template is opened in the NetWeaver 2004s BEx Web Application Designer using the Migration Tool and then processed using that tool. The web template will be saved with the a different internal name and version in a new database table.

Process of Migration

- ■ **Rebuild or migrate with Tool in SAP NetWeaver 2004s BEx Web Application Designer**
- ■ **Web Templates with…**

  → Standard Web Items and HTML can be migrated
  → JavaScript used for Commands can be migrated
  → JavaScript used for Layout (Tab Pages, etc.) should be rebuilt
  → Charts can be migrated, manual adjustments may be required

Rebuilding Web Templates

- ■ **Content (Queries) are already existing**
- ■ **Required functionality is already known**
- ■ **Command Wizard accelerates creation of commands**
- ■ **New Layout Web Items (Tab Pages, Container, Buttons, …) minimize need for JavaScript**
- ■ **New Web Items (Navigation Panel, Properties Pane)**

  → Required time for rebuilding a Web Template is less than creating it initially in SAP BW 3.x

Figure 156: BEx Web Application Designer – Migration Summary

The Migration Tool in the NetWeaver 2004s BEx Web Application Designer is a convenient way to migrate existing BW 3.x web templates. However, some elements such as JavaScript used to create tab pages or the BEx Web Design API for Tables will not be converted by the migration tool. These elements will need to be rebuilt using the supported web items and techniques.

Because of this, in certain instances you may choose to rebuild the web templates directly rather than rely on the migration tool. The graphic above lists several of the features in the NetWeaver 2004s Web AD that reduce the time required for building sophisticated web templates. The Command Wizard, the new layout-oriented web items as well as the other new web items make tasks that were very time consuming in the BW 3.x environment much faster now.

Official statement in note 931395

... The BEx Web Design API for Tables (table interface) and
related customized ABAP coding that was available within
the ABAP Web runtime is not available in the Java Web
runtime.
...
Therefore, in the next major SAP NetWeaver Release after
SAP NetWeaver 2004s, SAP will provide a Java-based
interface that can offer additional flexibility for
customized coding. ...

Situation in SAP BW 3.x

- HTML: Unrestricted usage of HTML
- JavaScript: Own coding without any guiding principles and limits
- Commands: Usable in links or in JavaScript but no tool support
- Table Interface: Change content and style of table in own ABAP class
- Help Service: Unrestricted generation of HTML in own ABAP coding

➔ BEx Web offers a platform for *freestyle* web design

Figure 157: BEx Web Design API for Tables - 1

The BEx Web Design API for Tables (table interface) is not supported in the
new Java runtime that the NetWeaver 2004s BEx Web AD uses. This interface
was useful in the BW 3.x environment for enhancing the display of data in the
'Table' web item with additional coding to perform complex transformations
and presentations.

The next release of NetWeaver will provide a similar facility for creating
customer-developed coding.

Situation in SAP NetWeaver 2004s

- Standard functionality replaces customer-specific coding (printing, exceptions, sorting, …)
- BEx Web Application Designer with Command Wizard and new Web Items (Tab Pages, Button Group), BEx Report Designer, …
- XHTML and Unified Rendering is used for Web Items

➔BEx Web offers a platform and a wide range of tools for *well-structured* web design

Use cases of table interface need to be categorized by:

- Use cases replaceable by standard functionality
- Use cases with reasonable extension of standard functionality (add to planned feature list)
- Use case with too custom-specific requirements; for these gaps an extension interface is planned.

Figure 158: BEx Web Design API for Tables - 2

If your BW 3.x web templates make use of the BEx Web Design API for tables (table interface), you can continue to execute those templates in the BW 3.x runtime (ABAP) if the table interface coding is essential for your users.

However in many cases, the custom coding placed in the table interface can be replaced with standard functionality in the new Web AD, lessening the need for the table interface. And, as the new Web AD is enhanced, other uses of the table interface may also be replaced.

- Existing BW 3.x queries and workbooks can be opened with the SAP NetWeaver 2004s BEx Analyzer
- Standard workbooks are automatically upgraded, workbooks with customer coding will be upgraded with a best guess strategy
- Manual adaptation may be necessary for the workbook. (e.g., customer Visual Basic has to be manually adapted)
- After saving in the new tool, workbooks **can no longer be used** in the BW 3.x BEx Analyzer and will not appear in the open dialog anymore
- For those scenarios were customers do not want to use the new BEx Analyzer, SAP additionally ships a BW 3.x version
- After migration, the old version of a workbook is still available in the old BW 3.x BEx Analyzer. Migration can be repeated. Old workbooks are not deleted
- New capabilities are **only** implemented in the NetWeaver 2004s BEx Analyzer
- GIS functionality is **only** available via BEx Web (BEx Web AD)
- A few features of conditions/exceptions are only available in the BEx Web Analyzer and not in the new BEx Analyzer

The graphic above lists the major points regarding the compatibility and migration of workbooks between the BW 3.x BEx Analyzer and the NetWeaver 2004s version of the tool. The migration of workbooks takes place in the NetWeaver 2004s BEx Analyzer and is done by opening and saving a BW 3.x workbook. You will be prompted that the migration process is commencing.
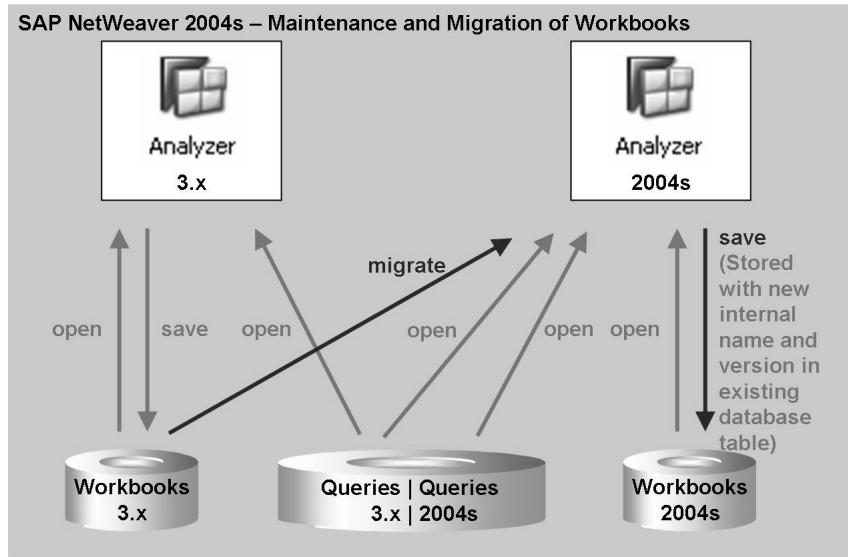
Figure 159: BEx Analyzer – Maintenance and Migration

The graphic above shows the possible scenarios with both BW 3.x and BI versions of workbooks. Each version of a workbook can be opened, maintained and saved using the respective version of the BEx Analyzer. No migration of BW 3.x workbooks will occur as long as they are only opened with the BW 3.x BEx Analyzer. This allows existing query scenarios to continue to perform unchanged.

Migration of BW 3.x workbooks is initiated by opening those workbooks with the NetWeaver 2004s BEx Analyzer and then saving the workbook. The workbook will be saved with a new internal name and version in the existing database table. Since the BW 3.x workbook is not deleted, the migration can take place as many times as necessary.

- Migration of Workbooks
  - Optional but necessary to use any of the new functionality
  - If customer-specific enhancements have been made (for example Visual Basic), then manual adjustments might be necessary
  - Migrated workbooks are automatically saved under a new internal name and version
- Caution
  - GIS functions are only available with the BEx Web
  - BEx Analyzer does not support some of the functionality that is available in BEx Web (for example, specific settings of exceptions)

BW 3.x workbooks with custom Visual Basic coding must be manually adjusted after migration.

It is also important to remember that the GIS functionality of BW 3.x (BEx Map) is only available on the web in NetWeaver 2004s. Likewise, the BEx Analyzer does offer every navigation option that is available in BEx Web tools. One example occurs with exceptions where there are some additional settings available in BEx Web.

- In SAP NetWeaver 2004s, the BEx Broadcaster is shipped in an BW 3.x version and in a NetWeaver 2004s version
- The BW 3.x version is started from the BW 3.x BEx tools and/or the context menu of web applications running on the old runtime
- The SAP NetWeaver 2004s version is started from NetWeaver 2004s BEx tools and/or context menu of web applications running on the new runtime
- Existing BW 3.x broadcasting settings can still be maintained and processed with the BW 3.x version of the BEx Broadcaster
- If you use the new BEx Broadcaster, you can build settings on all queries in the system but you have to build them from scratch. You cannot use already existing BW 3.x broadcasting settings for queries
- You can build settings on all web applications that were built or converted with the SAP NetWeaver 2004s Web AD. You cannot build settings on old Web applications
- BW 3.x broadcasting settings for workbooks can be used as before. There are no changes

As with the other BEx development tools, the BEx Broadcaster is available in both a BW 3.x and NetWeaver 2004s version. The version which is called at runtime is dependent on the tool and runtime used to access the Broadcaster.

The BW 3.x version of the Broadcaster is used to maintain BW 3.x broadcast settings. This allows existing broadcast scenarios to continue during the migration process.

NetWeaver 2004s broadcast settings for both queries and web applications must be created from scratch using the NetWeaver 2004s BEx Broadcaster. However, broadcast settings for workbooks do not have to be recreated.

* Only for workbook settings
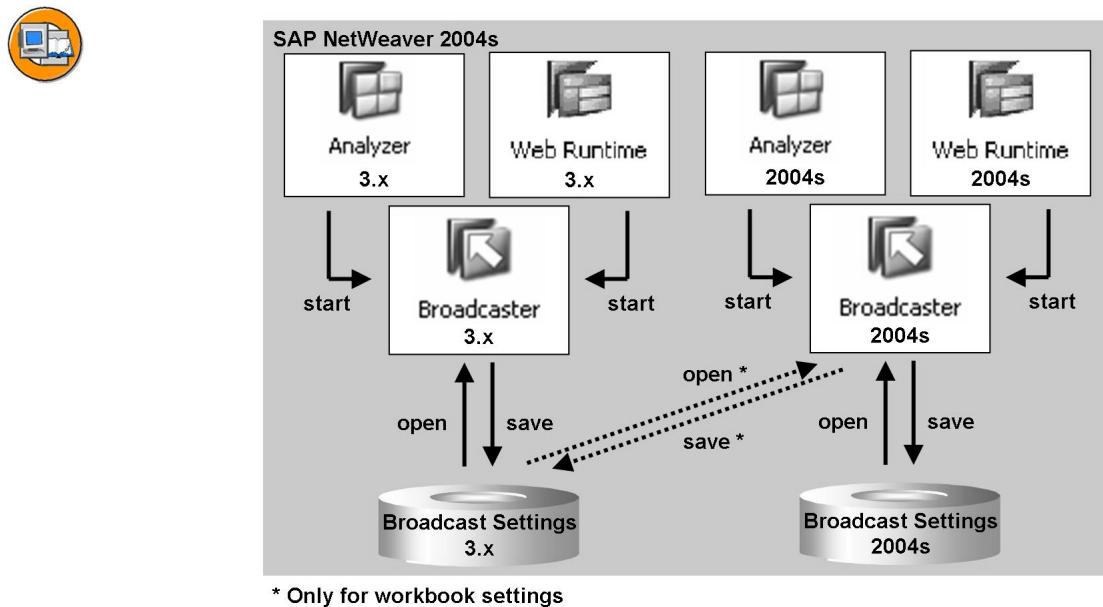
Figure 160: BEx Broadcaster – Maintenance and Migration

The graphic above shows the clear separation of maintenance activities for broadcast settings. BW 3.x broadcast settings for queries and web applications are maintained by the BW 3.x BEx Broadcaster. NetWeaver 2004s broadcast settings for queries and web applications are maintained by the NetWeaver 2004s BEx Broadcaster.

Settings for workbooks, whether they are BW 3.x settings or NetWeaver 2004s settings can be maintained by the NetWeaver 2004s BEx Broadcaster.

• Optional, but necessary to use any of the new functionality
• Queries
  – New Broadcaster can be used for ALL queries (3.x and 2004s)
  – Settings have to be created from scratch
• Web Templates
  – New Broadcaster can be used only for new web templates (2004s)
  – Settings have to be created from scratch
• Workbooks
  – New Broadcaster can be used for ALL workbooks (3.x and 2004s)
  – Old settings continue to work

2007/Q2   **455**

In summary, broadcast settings are generally not subject to migration such as would be done with queries, workbooks and web applications. In general, broadcast settings for NetWeaver 2004s will be created from scratch by the NetWeaver 2004s BEx Broadcaster. The only exception to this are the existing settings for broadcasting workbooks which can be used without change.

Lesson Summary

You should now be able to:
- Describe the migration process for BW 3.x queries.
- Describe the migration process for BW 3.x web templates
- Describe the migration process for BW 3.x workbooks
- Describe the migration process for BW 3.x broadcast settings

Related Information

- To view a weblog regarding migration of reporting objects on the SDN forum, go to https://www.sdn.sap.com/irj/sdn/weblogs?blog=/pub/wlg/4495 .

Unit Summary

You should now be able to:
- Describe the migration process for BW 3.x queries.
- Describe the migration process for BW 3.x web templates
- Describe the migration process for BW 3.x workbooks
- Describe the migration process for BW 3.x broadcast settings

Course Summary

You should now be able to:

- Describe the purpose and outline the key features of each component of the BEx Suite and also xApps Visual Composer.
- Design and build customized workbooks using the Design Mode of the BEx Analyzer and also use native Excel formulas to integrate Excel functionality with BI functions.
- Develop highly formatted reports for displaying on the Web and also for printing using the new Report Designer.
- Develop web applications that use the full range of features of the Web Application Designer.
- Expand your usage of the BEx Broadcaster to distribute NetWeaver BI reports.
- Publish a variety of BI reports to the NetWeaver Portal and organize them efficiently using iViews, pages and folders with portal roles.
- Put into practice the skills developed in this class to develop a comprehensive reporting solution based on detailed business requirements.
- Explain the possibilities presented with the tight integration of the xApps Visual Composer with NetWeaver BI.
- Discuss the issues with migrating reporting objects between BW 3.x and NetWeaver 2004s BI.

2007/Q2   **459** SAP

Feedback

SAP AG has made every effort in the preparation of this course to ensure the accuracy and completeness of the materials. If you have any corrections or suggestions for improvement, please record them in the appropriate place in the course evaluation.

 SAP