

Unit 1 Introduction to IBM Cognos Active Reports

IBM Training



Introduction to IBM Cognos Active Reports

IBM Cognos Analytics (v11.0)

© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM.

Objectives

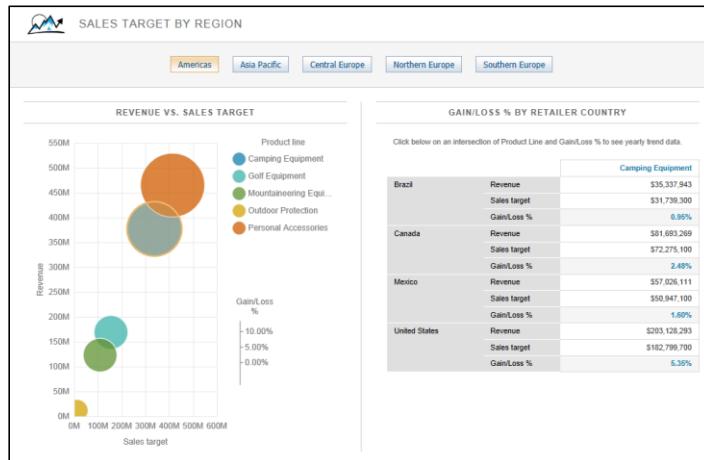
- Describe Active Reports, and their value
- Save Active Reports
- Use Active Reports as prompt pages
- Convert existing reports to Active Reports
- Explain security considerations in Active Reports
- Debug Active Report behavior

Business drivers for Active Reports

- Interactivity
 - interact with a rich user interface, and easily filter data in reports with prompts
- Performance
 - access and work with information without wait times
- Disconnected functionality
 - access and interact with report content while disconnected from the network
- Distribution
 - access reports as attachments in email rather than through a website
- Authoring
 - pixel-perfect formatting for reports

What is an Active Report?

- Active Report is a report type whose output is an MHT file
- Report consumers can view and interact with an Active Report data without hitting the server and the database again



Introduction to IBM Cognos Active Reports

© Copyright IBM Corporation 2018

What is an Active Report?

An Active Report can be run in the view mode of IBM Cognos Analytics - Reporting like any other report, or it can be downloaded locally as an MHT file.

MHT files are archived HTML files (MHT is the file extension used for MIME HTML file formats). MHT files include all necessary data and images to allow report consumers to interact with Active Report data without connecting to the server and the database again.

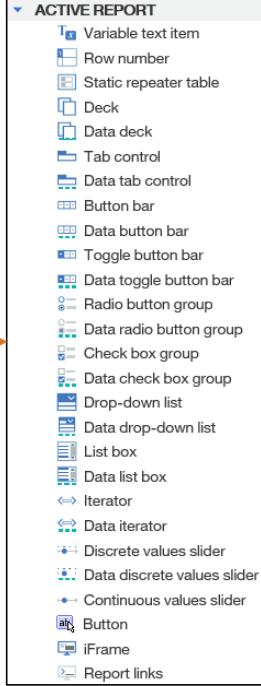
Consumers can explore the report by changing different prompt values and rendering new output in the client. Because all the data is embedded in the report at runtime, when user selections change the data is rendered in the client without going back to the database server, nor the IBM Cognos Analytics server.

IBM Training 

Examine Active Report controls

- When you create an Active Report, there are additional controls available for use.
- Controls can be of two types:
 - static
 - data driven

Active report controls on the Toolbox tab 



```

ACTIVE REPORT
  Variable text item
  Row number
  Static repeater table
  Deck
  Data deck
  Tab control
  Data tab control
  Button bar
  Data button bar
  Toggle button bar
  Data toggle button bar
  Radio button group
  Data radio button group
  Check box group
  Data check box group
  Drop-down list
  Data drop-down list
  List box
  Data list box
  Iterator
  Data iterator
  Discrete values slider
  Data discrete values slider
  Continuous values slider
  Button
  iFrame
  Report links

```

Introduction to IBM Cognos Active Reports  © Copyright IBM Corporation 2018

Examine Active Report controls

There are two versions of most Active Report controls: Static and Data-driven.

Static controls allow you to manually define values.

Data-driven controls have values that are determined by the associated data item.

For example, Drop-down list, Radio button group, and Check box group are all controls whose values are created manually by the report author.

The equivalent data-driven controls are Data drop-down list, Data radio button group, and Data check box group, where each one of these would be populated with the values in a chosen data item.

For more information on available controls for use in Active Reports, please refer to [Active Report Controls](#)

(https://www.ibm.com/support/knowledgecenter/SSEP7J_11.0.0/com.ibm.swg.ba.cognos.ug_cr_rptstd.doc/c_rpt_app_cntrls.html).

IBM Training IBM

Describe Active Report variables

- Active Report variables work with controls to add interactivity and control behavior in reports.

variable

Introduction to IBM Cognos Active Reports © Copyright IBM Corporation 2018

Describe Active Report variables

You can use variables to define the behavior of controls and other report objects. Variables allow you to specify a control's selection behavior or reaction behavior.

For the control selected in the menu (region_Data_BB in the slide example):

- Selection Behavior: upon selection of the control, the value of the Active Report variable is set to the control selection value
- Reaction Behavior: specifies how the control reacts to events on variables, when another control, or an object in the control, is selected.
- There are three types of reaction behavior:
- select an item in a control
- filter an item in a control
- enable a control

Not all reaction behaviors are available for every combination of source and target control, but possible reactions depend on the source and target controls.

Language support in controls

- Create variables that support multiple languages.
- When the report is run in that language, all text will appear as translated.

Label	DB Value
Email	E-mail
Fax	Fax
Post	Mail

Language support in controls

If you want the Active Report to be viewed in different languages, you will have to supply the translated value for any text you create in static controls.

Proven practice: naming conventions

- Demonstrations and exercises in this course explicitly name all Active Report controls, Active Report variables, and queries.

Rename	Default name	Preferred Name
Static Control	Radio Button Group1	Country_RBG
Data Control	Data Drop-Down List1	Region_Data_DDL
Variable	Product line	vProductLine
Parameter	N/A	?pCountry?

Proven practice: naming conventions

Throughout this course, all Active Report controls, Active Report variables, and queries follow a naming standard, to make it easier to identify a control, variable, or query in the report based on its name.

- Control names have a prefix that indicates the primary measure or data item, and a suffix whose abbreviation indicates the data type (i.e. Country_RBG).
- Variables are prefixed with a "v" to denote a variable.
- Parameters are prefixed with a "p" to denote a parameter.

Proven practice: report object sizing

- Plan for scalable reports across different resolutions.
- Percentage-based sizes allow objects to scale to the available resolution.
- Fixed size objects may appear too small on larger resolutions, or may push objects outside the report on smaller resolutions.

Proven practice: report object sizing

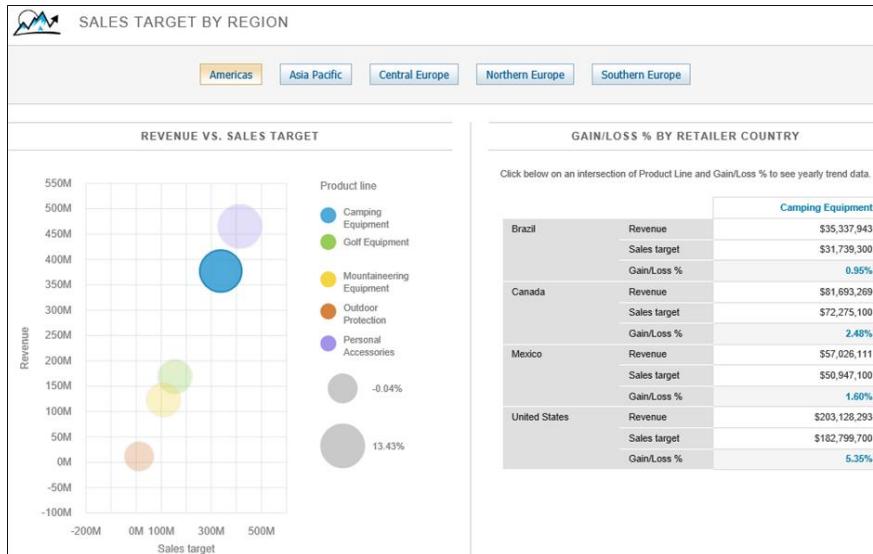
When designing the layout of an Active Report, it is important to consider the different ways the report may be used. For example, reports that are run on a mobile device may have less available space than a report that is run in a browser.

To create a flexible report that can be scaled according to the available resolution, try to size objects as a percentage of the available space, rather than as a fixed size.

As resolutions change, controls that use fixed sizing may have their content clipped (if the resolution is too low), or may leave large unused sections on a larger resolution, which may not be visually appealing to the report consumer.

Demonstration 1

Run an Active Report



Introduction to IBM Cognos Active Reports

© Copyright IBM Corporation 2018

Demonstration 1: Run an Active Report

Demonstration 1: Run an Active Report

Purpose:

You will open an Active Report to examine some of the interactive behavior that is possible in an Active Report.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

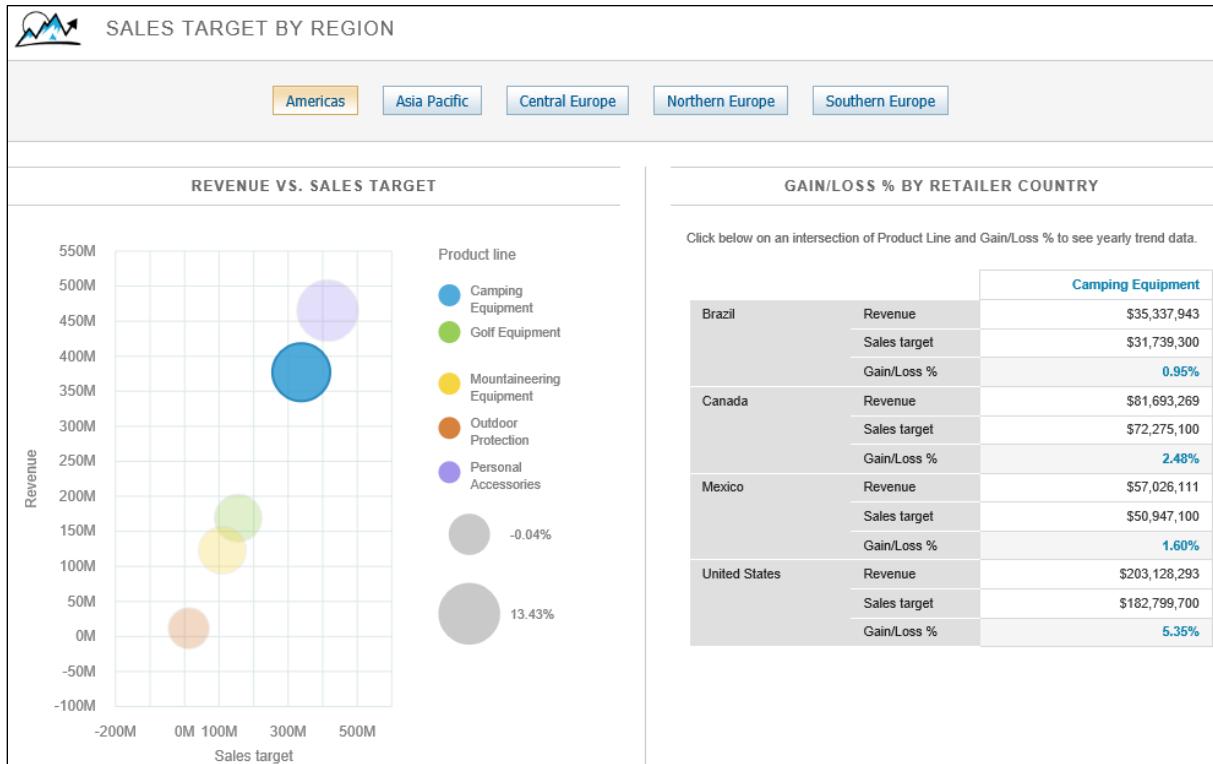
Package: GO sales (query)

Report: Samples > Reports > Active Reports > Sales target by region

Task 1. Open a report in IBM Cognos Analytics

1. On the side panel, click Team content .
2. Click Samples > Reports > Active reports.
3. Click Sales target by region.

The results appear as follows:

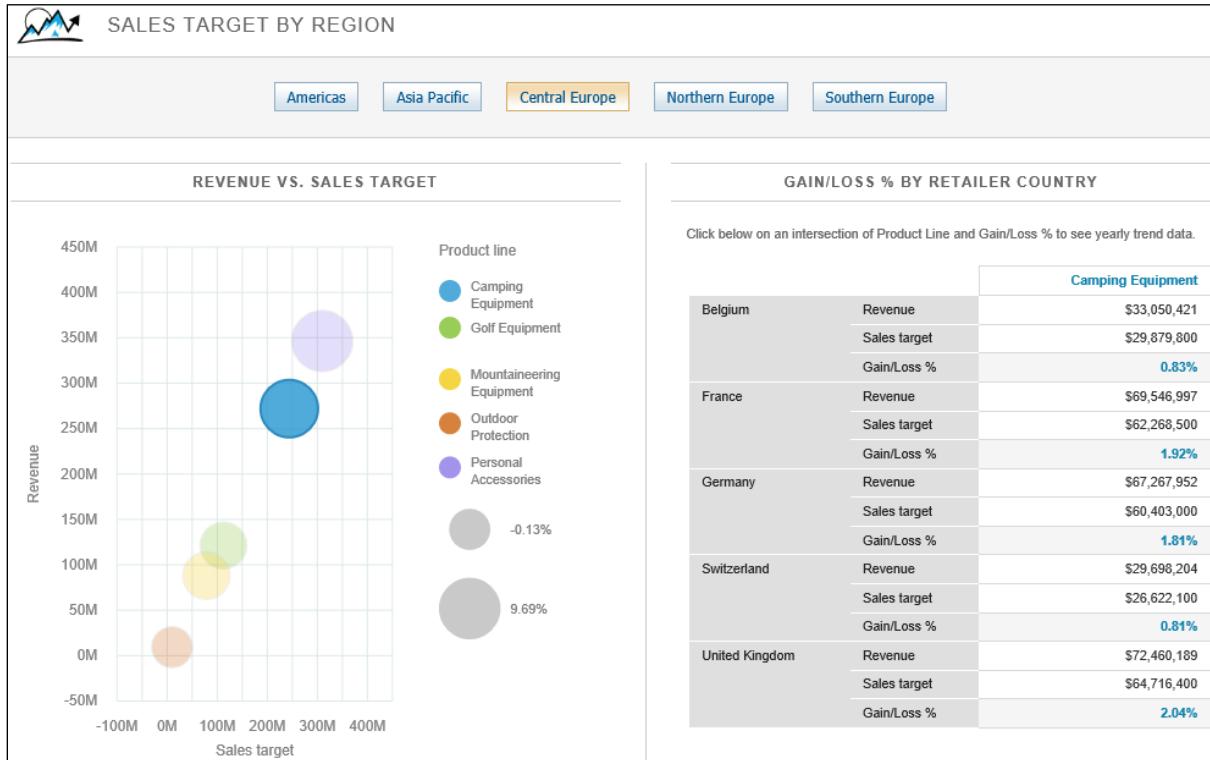


The Active Report opens in a new browser tab.

Task 2. Test the interactive features

1. Click Central Europe.

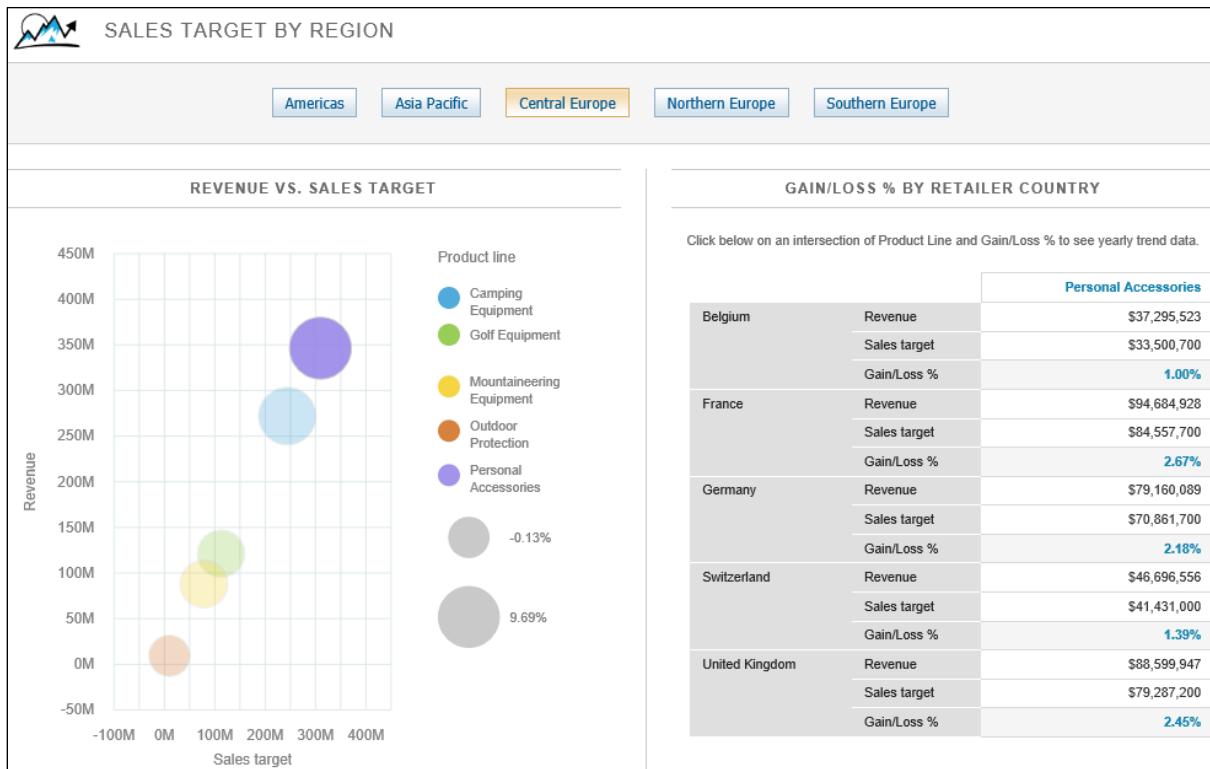
The results appear as follows:



The bubble visualization and Crosstab are updated to display data for Central Europe.

2. In the bubble visualization, click the largest bubble to select **Personal Accessories**.

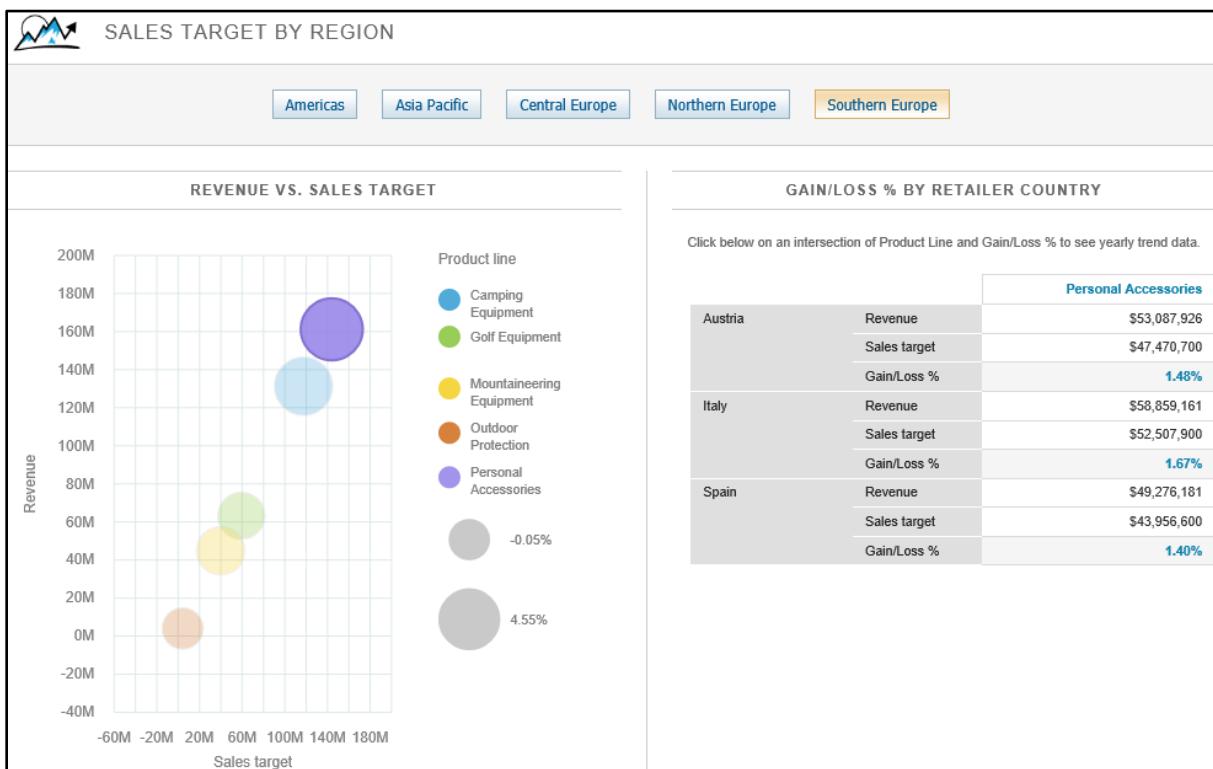
The results appear as follows:



The Crosstab is updated to show data for Personal Accessories.

3. Click Southern Europe.

The results appear as follows:



The visualization and Crosstab are updated to display data for Southern Europe, but the Crosstab continues to display data for Personal Accessories.

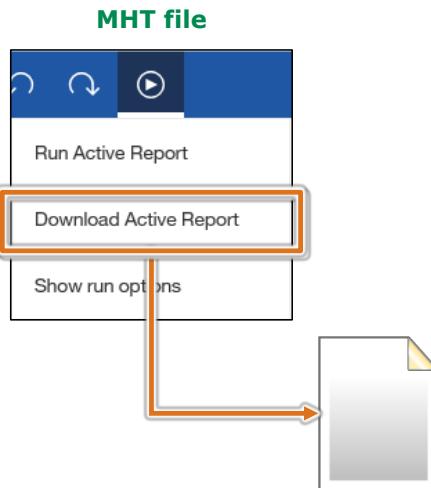
- Close the rendered report tab and leave IBM Cognos Analytics open for the next demonstration.

Results:

You opened and ran an Active Report. And tested some of the interactive behaviors available with this type of report.

Save an Active Report for distribution

- Download the report as MHT output.



Introduction to IBM Cognos Active Reports

© Copyright IBM Corporation 2018

Save an Active Report for distribution

To save an Active Report as an MHT file for distribution, select Download Active Report from the Run options menu. You will be prompted to open or save the report.

Note that if you simply run the report under the IBM Cognos Analytics portal, the MHT file will be unpacked to an XHTML format. To get the actual MHT file, you must save the report and download the saved output.

All data required to execute the Active Report is retrieved at the time the MHT file is generated, which means that the MHT file does not require a connection to the IBM Cognos servers once it has been created.

Because the MHT file generates its data on the initial run, opening the MHT file after the data has changed on the IBM Cognos servers will not update the data in the Active Report. To reflect the latest data, you must run and download the Active Report again.

Demonstration 2

Create a simple Active Report using Static and Data-driven controls

Americas	<input checked="" type="checkbox"/> Email		
Asia Pacific	<input checked="" type="checkbox"/> Fax		
Central Europe	<input type="checkbox"/> Post		
Northern Europe	<input type="checkbox"/> Sales rep		
Southern Europe	<input type="checkbox"/> Promo		
	<input type="checkbox"/> Phone		
	<input type="checkbox"/> Web site		
Retailer country	Retailer site	Order method type	Revenue
Belgium	Brussel	Fax	\$279,876.39
	Liège	Fax	\$2,286,557.32
France	Calais	Fax	\$585,814.01
	Lyon	E-mail	\$65,940.08
	Lyon	Fax	\$2,344,759.22
	Paris	E-mail	\$299,899.71
	Paris	Fax	\$4,430,206.52

Introduction to IBM Cognos Active Reports

© Copyright IBM Corporation 2018

Demonstration 2: Create a simple Active Report using Static and Data-driven controls

Demonstration 2: Create a simple Active Report using Static and Data-driven controls

Purpose:

You want to create a report that allows the consumer to filter a list by one of the available regions. You also want to create labels for order methods whose names are more meaningful to report consumers.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

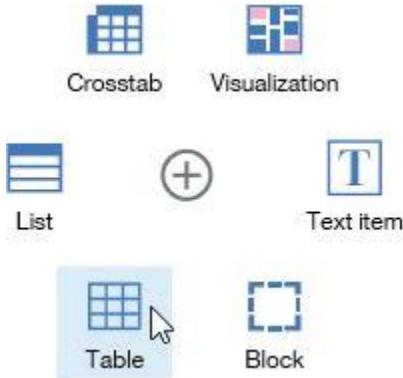
Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create an Active Report with a List

1. From the **Welcome** page, navigate to **Team content** > **Samples** > **Models**.
2. Right-click the **GO sales (query)** package, then click **Create report** .
3. In the Templates and themes dialog, double-click the **Blank active report** template to use it.
4. On the Application bar, click **More** , and then click **Options**.
5. Click the **Report** tab, then uncheck **Automatic group and summary behavior for lists**.
6. Click **OK**.
7. In the work area, click **Add** .

The results appear as follows:



8. Click **Table**, set **Number of rows** to **2**, and then click **OK**.
9. Click the bottom left table cell, Ctrl-click the bottom right table cell, and then on the Toolbar, click **Merge Cells** .
10. On the Toolbar, click **Horizontal alignment options** > **Center** .
11. In the merged table cell, click **Add** , then click **List** .

12. In the **Object and query name** dialog, set **Name** to **Sales_List**, set **Query Name** to **Sales_List_Query**, and then click **OK**.
13. From the **Data**  tab, drag the following query items to the **List**:
 - Retailers: **Retailer country**, and **Retailer site**
 - Sales: **Revenue**
- Click the **<Revenue>** List column body, then on the Toolbar, click **Summarize**  then click **Total**.

The results appear as follows:



Retailer country	Retailer site	Revenue
<Retailer country>	<Retailer site>	<Revenue>
<Retailer country>	<Retailer site>	<Revenue>
<Retailer country>	<Retailer site>	<Revenue>
Overall - Total		<Total(Revenue)>

Task 2. Add a control to filter on regions

You will add a Data button bar to filter the List on the Region data item.

1. On the side bar, click **Toolbox** , and then expand ACTIVE REPORT.
2. Drag a **Data button bar** object into the top left table cell.
3. In the **Object and query name** dialog, for **Name**, type **Region_Data_BB**, and for **Query**, type **Region_Data_BB_Query**, and then click **OK**.
4. On the side bar, click **Data** .
5. From the **Source** tab, under **Retailers**, drag **Region** to the **Region_Data_BB** control's drop zone .

You want to create a connection between the **Region_Data_BB** control and the **Sales_List** control so that as the user clicks a region button, the List will filter the data on one sales region only.

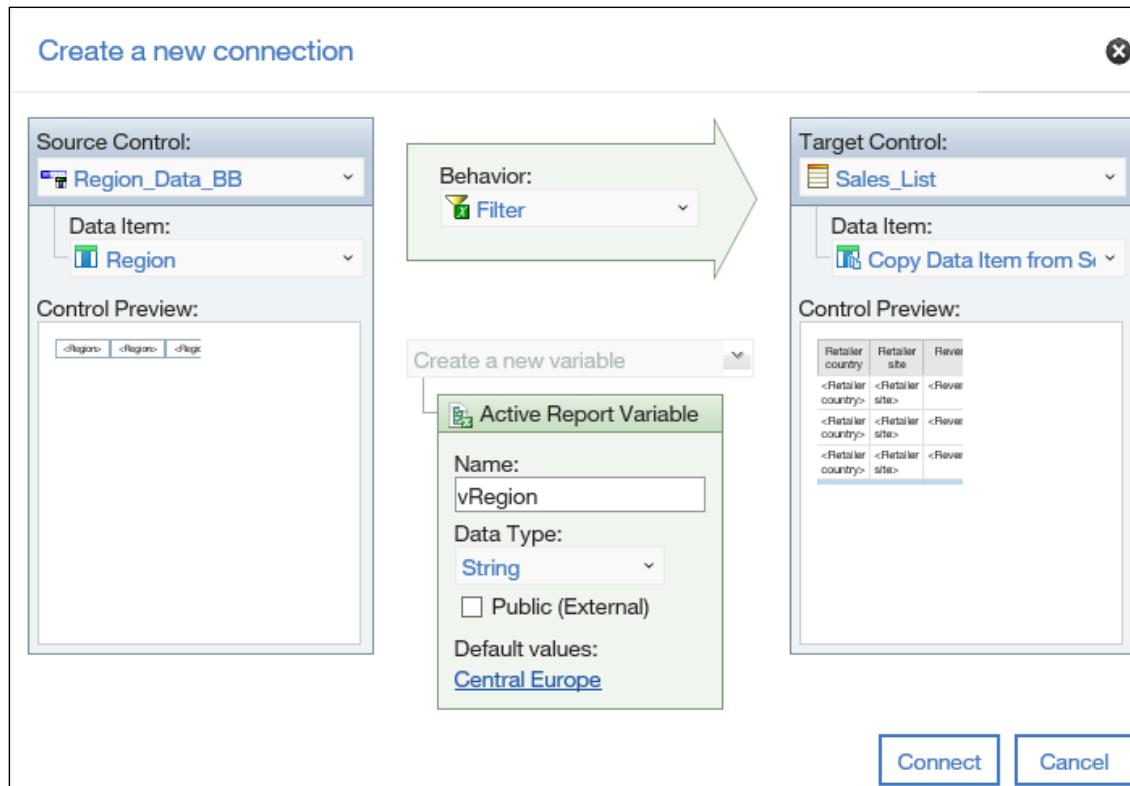
6. In the Toolbar, click **Create a new connection** .

Because there are only two objects in the report, this connection is pre-populated with the correct values to make the connection. On the left is the **Region_Data_BB** control. In the center is the connection method between the two objects, which is a selection by default. On the right is the **Sales_List** control. In the bottom right, **Copy Data Item from Source** will add the data item column to the List as hidden. Once this connection is made, the data button bar

selection will filter the Sales_List control through the use of a variable. You will rename the variable.

7. Under Active Report Variable, rename the new variable as **vRegion**.
8. Under Default values, click **Set default values**.
9. Under Values, double-click **Central Europe** to add it to Selected values.
10. Click **OK** to close the dialog.
11. Under **Behavior**, select **Filter**.

The result appears as follows:



The vRegion variable will be set to the value selected in the Region_Data_BB control. By default, vRegion will be set to Central Europe.

12. Click **Connect**.

Notice that Region is now added to the List because of the connection being created, however it is hidden. Region must be included in the List query so that the filter will work.

13. On the Application bar, click **Run options** , then click **Run Active Report**.
14. Click **Asia Pacific** in the Data button bar.

A section of the results appears as follows:

Americas	Asia Pacific	Central Europe	Northern Europe	Southern Europe
	Retailer country	Retailer site	Revenue	
	China	Beijing	\$15,038,255.58	
	Australia	Brisbane	\$14,990,079.66	
	Australia	Cairns	\$23,445,233.27	
	China	Chaiwan	\$6,105,593.12	
	China	Chengdu	\$8,218,397.39	
	Japan	Chiyoda-ku	\$7,701,778.39	
	Japan	Chuo-ku	\$9,104,505.46	
	Japan	Fukuoka-shi	\$16,328,626.03	
	Australia	Geelong	\$2,155,234.84	
	China	Guangzhou	\$9,952,473.87	
	China	Hangzhou	\$3,235,889.96	
	China	Harbin	\$12,402,588.45	
	Japan	Hiroshima-shi	\$28,518,306.27	
	China	Huhehaote	\$5,515,268.93	
	Korea	Incheon	\$3,872,981.29	
	China	Jinan	\$3,548,805.89	
	Korea	Kwangju	\$2,739,633.83	
	Australia	Melbourne	\$19,238,576.57	
	Japan	Minato-ku	\$5,413,555.03	

Revenue data is displayed for the Asia Pacific region.

15. Close the browser tab.

Task 3. Add a control to filter on order method type

You will add a Button bar control to filter on order method. You will create button labels that are different than the data item values found in the database. The new labels will be helpful for the report consumers who use terminology that is different from the data values in the underlying data source.

1. On the side bar, click **Toolbox**.
2. Drag a **Button bar** object into the top right table cell
3. On the Toolbar, click **Buttons Definition** .

You will be changing the button label names. First you need to create a column to hold the data item values that will be passed when the user clicks the button.

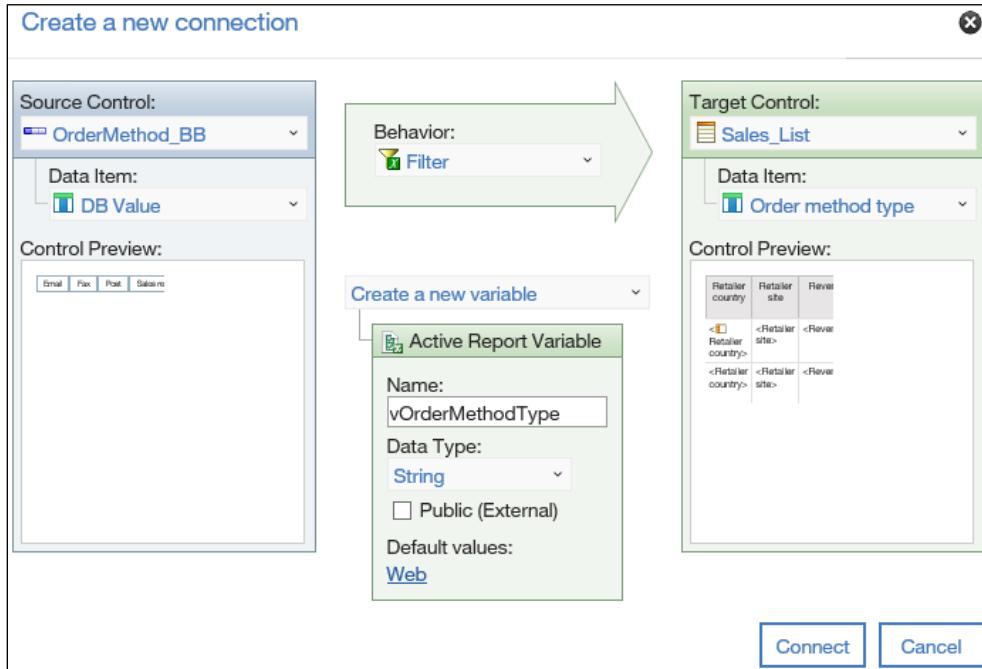
4. Click **New** , type **DB Value**, and then click **OK**.
5. Enter the labels and values as follows:

Label	DB Value
Email	E-mail
Fax	Fax
Post	Mail
Sales rep	Sales visit
Promo	Special
Phone	Telephone
Web site	Web

Tip: Use New  to create new rows.

6. Click **OK**.
7. On the Application bar, click **Show properties**  to open the Properties pane.
8. In the Properties panel, under MISCELLANEOUS, set the **Name** property to **OrderMethod_BB**.
You will create the filter connection between the OrderMethod_BB control and the Sales_List control. First you must add Order method type to the Sales_List control so it will be available when you specify the connection.
9. On the side bar, click **Data**.
10. From the **Source** tab, expand **Order method**, and then drag **Order method type** after the **Retailer site** column.
11. Click the **OrderMethod_BB** control to select it.
12. From the Toolbar, click **Create a New Connection**.
13. Select the following:
 - For **Target Control**, select **Sales_List**.
 - Under **Behavior**, select **Filter**.
 - Create a new variable > Name: **vOrderMethodType**
 - Data Item (for the Source Control): **DB Value**
 - Data Item (for the Target Control): **Order method type**
14. Under **Default values**, click **Set default values**.
15. Under **Values**, double-click **Web**, and then click **OK**.

The result appears as follows:



16. Click **Connect**.
17. On the Application bar, click **Run options** , then click **Run Active Report**.
A section of the results appears as follows:

Americas	Asia Pacific	Central Europe	Northern Europe	Southern Europe	Email	Fax	Post	Sales rep	Promo	Phone	Web site
		Retailer country	Retailer site	Order method type	Revenue						
		Belgium	Antwerpen	Web	\$22,882,658.97						
			Arlon	Web	\$8,160,492.71						
			Brugge	Web	\$7,945,692.93						
			Brussel	Web	\$2,390,137.83						
			Bruxelles	Web	\$21,071,312.35						
			Charleroi	Web	\$884,794.13						
			Leuven	Web	\$2,674,352.37						
			Liège	Web	\$1,492,696.16						
			Louvain-La-Neuve	Web	\$6,422,044.57						
			Sint-Niklaas	Web	\$5,134,408.50						
			Waterloo	Web	\$13,912,554.90						
			Zaventem	Web	\$6,960,811.87						

18. Close the browser tab.

Task 4. Enhance controls and the report

You will convert the controls to enhance the report appearance. You will further enhance the report by grouping the Country column and hiding the Order method type column.

1. Select the **Region_Data_BB** control, and then ensure that the **Properties** pane is showing **Data button bar**.
2. In the Properties pane, under GENERAL, set **Orientation** to **Vertical**.

The Region_Data_BB control now runs vertically, eliminating the need to scroll right to access the order method buttons.

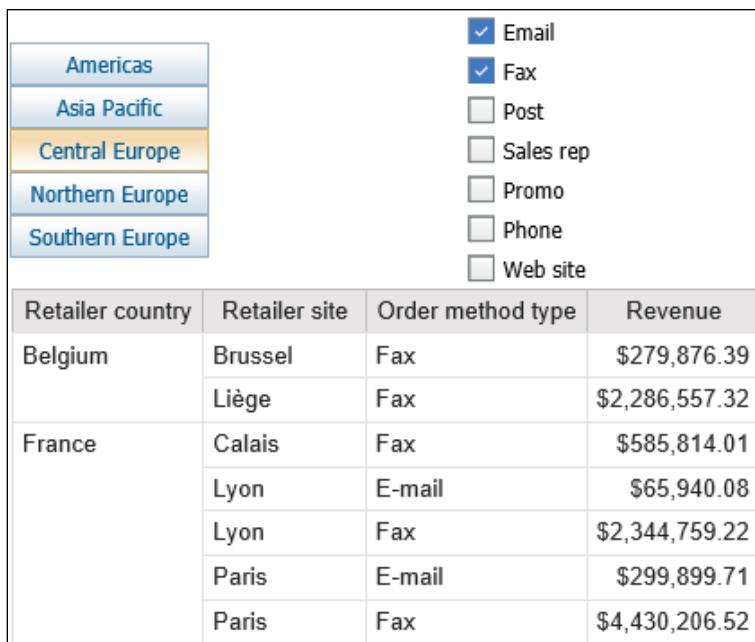
Next, you will convert the OrderMthd_BB control to a Check box group.

3. Select the **OrderMethod_BB** control, then on the Toolbar, click **More**  . Hint: You may need to click the report background to clear the selection first.
 4. Point to **Convert Control**, and then click **Check box group** .
- Note that because you have changed the type of this control, you will also need to change the control's name to reflect that it is now a Check box group. Because this is not a Data-driven control, you do not need to update any query names.
5. Select the **OrderMethod_BB** control, and then in the Properties pane, under MISCELLANEOUS, change the **Name** property to **OrderMethod_CBG**.
 6. Repeat step 2 for the **OrderMethod_CBG** control so that it appears vertically. By converting the control to a Check box group, the user can make multiple selections to filter the List.

To enhance the report, you will group by the Retailer country and Retailer site columns.

7. Click <Retailer country>, then Ctrl-click <Retailer site>, and then on the Toolbar, click **Group / Ungroup** .
8. From the Application bar, click **Run options** , then click **Run Active Report**.
9. Select **Northern Europe** from the **Region_Data_BB** control, and then select **Email** and **Fax** from the **OrderMethod_CBG** control.

A section of the results appears as follows:

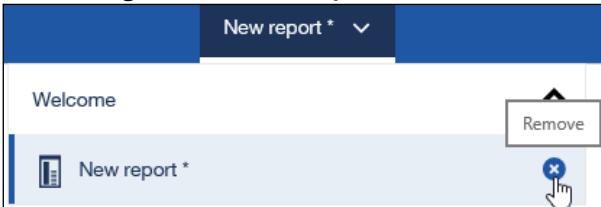


The screenshot shows a report interface. On the left, there is a vertical navigation bar with categories: Americas, Asia Pacific, Central Europe (which is highlighted in orange), Northern Europe, and Southern Europe. To the right of this is a list of order method types with checkboxes: Email (checked), Fax (checked), Post (unchecked), Sales rep (unchecked), Promo (unchecked), Phone (unchecked), and Web site (unchecked). Below this is a table with four columns: Retailer country, Retailer site, Order method type, and Revenue. The data is grouped by Retailer country. For Belgium, there are two rows: one for Brussel (Fax, \$279,876.39) and one for Liège (Fax, \$2,286,557.32). For France, there are five rows: Calais (Fax, \$585,814.01), Lyon (E-mail, \$65,940.08), Lyon (Fax, \$2,344,759.22), Paris (E-mail, \$299,899.71), and Paris (Fax, \$4,430,206.52).

Retailer country	Retailer site	Order method type	Revenue
Belgium	Brussel	Fax	\$279,876.39
	Liège	Fax	\$2,286,557.32
France	Calais	Fax	\$585,814.01
	Lyon	E-mail	\$65,940.08
	Lyon	Fax	\$2,344,759.22
	Paris	E-mail	\$299,899.71
	Paris	Fax	\$4,430,206.52

10. Close the rendered report tab.

11. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



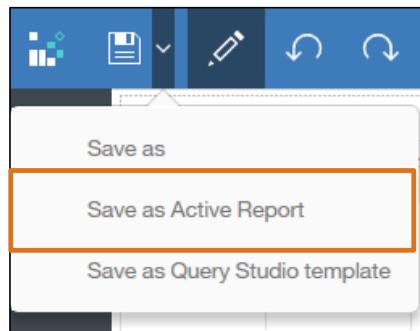
12. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You created a simple Active Report containing Static and Data-driven controls. The Static control allows you to use business terms, rather than the data values, so that consumers will be better able to interact with the report.

Convert an existing report to an Active Report

- Save as Active Report
 - Active Report menu items, objects, and properties become available in Cognos Analytics.
 - objects that are not supported in Active Reports are removed.



Convert an existing report to an Active Report

When you convert an existing report to an Active Report, the report author is still required to modify the report to deliver the full value of an Active Report to consumers. For example, a report author may add a Data Check Box Group to be able to filter on the Product data item.

Objects not supported in IBM Cognos Active report are:

- prompt controls in report pages
- table of contents
- table of contents entries
- bookmarks

Active report security

- Role-based security features supported
 - IBM Cognos Analytics portal
 - scheduled and burst reports
 - mobile distribution (initial access only)
- MHT files do not have IBM Cognos security
- MHT file security can be achieved through:
 - email
 - network-accessible location
 - report-by-report access code
 - (contents not encrypted)



Active report security

IBM Cognos Analytics supports role-based security features whenever the user is connecting to the IBM Cognos servers. This includes reports run in the IBM Cognos Analytics portal, scheduled and burst reports, and reports accessed from a mobile device.

Note that in the case of mobile devices, only the first access to the report undergoes the security check. Once the report has been downloaded, it is viewed on the device without further security verification.

When an Active Report is saved as an MHT file, the file itself should be treated as any other file in the file system. By default, there is no IBM Cognos security protecting the file, although a password can be added to an Active Report on a report-by-report basis.

Active Report MHT files can be secured by sending them using secure e-mail, by storing them in a secured network-accessible location, or by any other precautions that can be taken to secure files containing potentially sensitive data.

IBM Training 

Debug an Active Report

- There are various debugging options available to help you troubleshoot an Active Report in development
- Run Options
 - Enable right-click debug menu

Name	Value
vRegion	Central Europe
vOrderMethodType	Web

Introduction to IBM Cognos Active Reports  © Copyright IBM Corporation 2018

Debug an Active Report

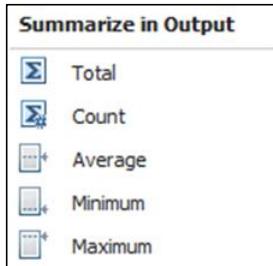
You use debugging options to help you troubleshoot active report behavior. For example, you can track the value of variables while you interact with the report, or reset the report to the default values.

The debug feature is turned on by default. It can be opened from the Application bar, under Run options.

When running the report from IBM Cognos Analytics - Reporting, from the IBM Cognos Analytics portal, or from an MHT file, the debug menu is activated when you right-click the report.

Describe interactive behavior properties

- Several properties control interactive behavior:
 - No Values List
 - Sort
 - Column Visibility
- Summarize measures to aggregate on what is displayed.
 - Summarize in Output options



Describe interactive behavior properties

No Values List - specifies a value to show when no value is selected, for example: <All> can appear at the top of a list. Applies to Drop-Down List and List Box only

Sort: specify whether users can sort data in a List.

Column Visibility: show or hide a column in a List, or a row/column in a Crosstab. It can be based on a condition on an Active Report variable.

Summarize in Output: show the summarized value for what is displayed in an Active Report.

Demonstration 3

Convert an existing report to an Active Report

Falcon Outfitters		<All>	
Region	Retailer country	Order method type	Revenue
Americas	United States	Sales visit	\$1,630,869.35
		Telephone	\$2,489,882.05
		Web	\$9,176,464.06
	Canada	E-mail	\$3,352,349.69
		Mail	\$720,845.68
		Telephone	\$2,441,618.79
		Web	\$12,225,876.72

Demonstration 3: Convert an existing report to an Active Report

Demonstration 3: Convert an existing report to an Active Report

Purpose:

You will create a report that was requested by a client. You will then convert the report to an Active Report, enhance its appearance, and add password protection, in order to exceed the client's expectations.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create a List report

1. From the side panel, navigate to **Team content** > **Samples** > **Models**, then right-click the **GO sales (query)** package, and then click **Create report** .
2. In the Templates and themes dialog, double-click the **Blank** template.
3. In the page body, click **Add** , and then click **List** .
4. In the **Object and query name** dialog, under **Name**, type **Retailer_List**, under **Query**, type **Retailer_List_Query**, and then click **OK**.
5. From the **Data** tab, drag the following query items to the **Retailer_List**:
 - Retailers > **Region**
 - Retailers > **Retailer country**
 - Retailers > **Retailer**
 - Order method > **Order method type**
 - Sales > **Revenue**
6. Click **Region** and Ctrl-click **Retailer country**, then from the Toolbar, click **Group / Ungroup** .

Task 2. Convert the report to an Active Report and add controls

1. Click the down arrow beside **Save** , then click **Convert to Active Report**. Notice that interactive behavior and connection icons appear to the right of the **Retailer_List** if you select the List. The List is considered an interactive control. Other menu items, properties and objects are now available to you for working with an Active Report.

You will create a Data drop-down list control for Retailer and Order method so that a report consumer can filter the Retailer_List control based on their selection.

2. From the **Toolbox**, drag a **Table** below the List, and set **Number of rows** to **2**.
3. Click **OK**.
4. Click the bottom left table cell, Ctrl+click the bottom right table cell, then on the Toolbar, click **Merge Cells** .
5. On the Toolbar, click **Horizontal alignment options** > **Center** .
6. Click then drag the **Retailer_List** object into the bottom table cell.
7. From the **Toolbox**, under **ACTIVE REPORT**, drag a **Data drop-down list** to each of the left and right table cells:
 - Left table cell:
 - Name: **Retailer_Data_DDL**
 - Query Name: **Retailer_Data_DDL_Query**
 - Right table cell:
 - Name: **OrderMethodType_Data_DDL**
 - Query Name: **OrderMethodType_Data_DDL_Query**
8. From the **Data \ Source** tab, drag **Retailers > Retailer** to the **Retailer_Data_DDL** control's drop zone.
9. From the **Source** tab, drag **Order method > Order method type** to the **OrderMethodType_Data_DDL** control's drop zone.

Task 3. Connect the controls to the List

1. Select the **Retailer_Data_DDL**, then from the Toolbar, click **Create a new connection** .
2. Specify the following connections:
 - **Retailer_Data_DDL > Filter > Retailer_List**
 - Create a new variable > Name: **vRetailer**
3. Click **Connect**.
4. Select the **OrderMethodType_Data_DDL**, then from the Toolbar, click **Create a New Connection** .
5. Specify the following connection:
 - **OrderMethodType_Data_DDL > Filter > Retailer_List**
 - Create a new variable > Name: **vOrderMethodType**

6. Click **Connect**.
7. Run the report and select **4 Golf only** and **Web** from the controls.

The result appears as follows:

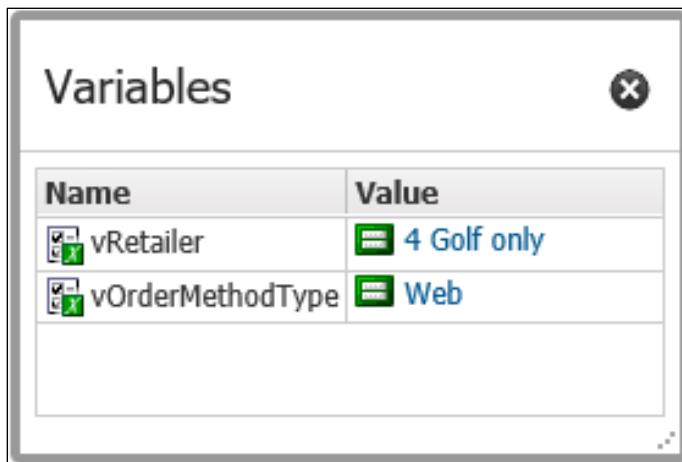
4 Golf only		Web		
Region	Retailer country	Retailer	Order method type	Revenue
Asia Pacific	Australia	4 Golf only	Web	\$2,673,033.19

8. Right-click the report to access the debugging menu.

You can use debug features to help you troubleshoot your report. For example, you can view variable values and reset the default values of controls.

9. Click **Show Variables** to see the current variable values.

The result appears as follows:



10. Close the Variables dialog.
11. Right-click the report, and then click **Reset Variables**.
The variables are reset to their default value, which is no value. And the List is unfiltered again.
12. Close the browser tab.

Task 4. Download the Active Report as an MHT file and open it

1. On the Application bar, click **Run options**, and then click **Download Active Report**.
2. At the prompt to save **New Report.mht**, click the down arrow beside **Save**, and click **Save as**.
3. Save the report as **C:\Training\B6098\ActiveReport.mht**.
Hint: Create the folders in Windows Explorer if they do not exist.
4. Click **Open folder**, and then double-click **ActiveReport.mht**.
The report runs as expected.

5. Close the browser tab.
6. Close **Windows Explorer**.

Next, you will update the report to enhance the content and add some interactivity using Interactive Behavior properties.

Task 5. Define interactive properties

You want to hide the Retailer column, add the ability to sort on Region and Retailer Country columns, and add the default option of <All> to the Order method type drop-down list.

1. In the **Retailer_List**, click <**Retailer**>.
2. On the Application bar, click **Show properties**  to open the Properties pane.
3. In the Properties pane, under INTERACTIVE BEHAVIOR, set **Column Visibility** to **Hidden**.

Next, you will set the Region and Retailer country columns to allow sorting.

4. Click <**Region**>, Ctrl-click <**Retailer country**>, and then in the **Properties** pane, under **INTERACTIVE BEHAVIOR**, set **Allow Sorting** to **Yes**.

Next, you will define a default value of <All> for the **Retailer_List** control.

5. Select the **OrderMethodType_Data_DDL** control.
6. In the Properties pane, under GENERAL, set the **No Value List Item** property to **Show**.

The properties for No Value List Item appear.

7. In the Properties pane, double-click the **Label** property.
8. In the **Default** text box, type <**All**>, and then click **OK**.
9. From the **Run Options** menu, click **Run Active Report**, and then select **Falcon Outfitters** from the **Retailer_Data_DDL** control.

10. Click the **Retailer country** column header twice to sort in descending order.
 The results appear as follows:

Falcon Outfitters		<All>	
Region	Retailer country	Order method type	Revenue
Americas	United States	Sales visit	\$1,630,869.35
		Telephone	\$2,489,882.05
		Web	\$9,176,464.06
	Canada	E-mail	\$3,352,349.69
		Mail	\$720,845.68
		Telephone	\$2,441,618.79
Overall - Summary			\$4,686,775,768.85

The list is sorted by Retailer country, and the Retailer column is no longer visible.

Note that the results may not be identical because the Order method type column is not sorted.

1. Close the browser tab.

Task 6. Add an access code to the Active Report

1. In the side bar, click **Pages** , and then click **Report** .
2. In the Properties pane, under REPORT, double-click **Active reports**.
3. In the **Active Report Properties** dialog, under **Access Code**, type **Education1**.
4. Click **OK**.
5. On the Application bar, click **Run options** , then click **Run Active Report**. You are prompted to enter an access code for the report.
6. Type **Education1**, and then click **OK**.
 The report runs as expected.
7. Close the rendered report tab.
8. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.
9. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

After creating a report that was requested by a client, you converted it to an Active Report, added controls, and saved it as a standalone (MHT) file. You then defined interactive properties to enhance the report, and added a password in order to prevent unauthorized access.

Demonstration 4

Summarize on a filtered list in an Active Report

Mountaineering Equipment		
Product line	Revenue	Quantity
Mountaineering Equipment	\$409,660,132.90	9,900,091
Overall - Total	\$4,686,775,768.85	89,237,091

Mountaineering Equipment		
Product line	Revenue	Quantity
Mountaineering Equipment	\$409,660,132.90	9,900,091
Overall - Total	409,660,132.9	9,900,091

Demonstration 4: Summarize on a filtered list in an Active Report

Demonstration 4: Summarize on a filtered list in an Active Report

Purpose:

You want to create a report with summarized values at the bottom, and ensure that the summarized data accurately reflects what is selected in the active controls. You will use summary options that are specific to Active Reports.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create a List and add an Active Report control

1. Using the **GO sales (query)** package, create a new **Blank active report**.
2. In the work area, click **Add** , and then click **Table** .
3. Set **Number of columns** to 1, **Number of rows** to 2, and then click **OK**.
4. In the bottom table cell, click **Add** , then click **List** .
5. In the **Object and Query Name** dialog, under **Name**, type **Revenue_List**, under **Query Name**, type **Revenue_List_Query**, and then click **OK**.
6. From the **Data**  tab, drag the following query items to the **Revenue_List**:
 - Products > **Product line**
 - Sales > **Revenue, Quantity**
7. Click the **Toolbox**, then expand **ACTIVE REPORT**, and then drag a **Data dropdown list** to the top table cell.
8. On the **Object and Query Name** dialog, under **Name**, type **ProductLine_Data_DDL**, and under **Query Name**, type **ProductLine_Data_DDL_Query**, and then click **OK**.
9. From the **Data** tab, Add **Products > Product line** to the drop zone of **ProductLine_Data_DDL**.
10. In the Toolbar, click **Create a new connection**.
11. Specify the following connection:
 - **ProductLine_Data_DDL > Filter > Revenue_List**
 - New variable: **vProductLine**
 - Data Item (left): **Product line**
 - Data Item (right): **Product line**
12. Click **Connect**.

The results appear as follows:

<Product line>		
Product line	Revenue	Quantity
<Product line>	<Revenue>	<Quantity>
<Product line>	<Revenue>	<Quantity>
<Product line>	<Revenue>	<Quantity>

Task 2. Add a summary row to display aggregated measure values

1. Click <Revenue>, then Ctrl-click <Quantity>.
2. In the Toolbar, click **Summarize** Σ , then under **Default summary**, click **Total**.
3. Run the report.

The results appear as follows:

Product line	Revenue	Quantity
Camping Equipment	\$1,589,036,664.03	27,301,149
Golf Equipment	\$726,411,367.89	5,113,701
Mountaineering Equipment	\$409,660,132.90	9,900,091
Outdoor Protection	\$75,994,296.25	12,014,445
Personal Accessories	\$1,885,673,307.78	34,907,705
Overall - Total	4,686,775,768.85	89,237,091

Note that the total for Revenue is approximately \$4.6B, and for Quantity is approximately 89K.

4. From the **ProductLine_Data_DDL** control, select **Mountaineering Equipment**.

The results appear as follows:

Mountaineering Equipment		
Product line	Revenue	Quantity
Mountaineering Equipment	\$409,660,132.90	9,900,091
Overall - Total	\$4,686,775,768.85	89,237,091

Notice that the totals did not change. The aggregate values being displayed were calculated at run time at the server don't reflecting what is being selected. To achieve the expected results, you will specify an aggregate that is specific to Active Reports, one that is calculated at the client.

5. Close the rendered report tab.
6. Click the summary cell **Overall - Total**, then press **Delete** in the keyboard.
7. Click <Revenue>, then Ctrl-click <Quantity>.

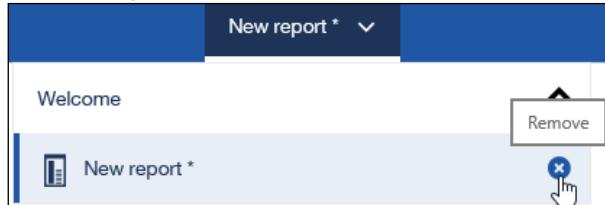
8. In the Toolbar, click **Summarize**, then under **Summarize in Output**, click **Total**.
9. On the Application bar, click **Run options** , then click **Run Active Report**.
10. From the **ProductLine_Data_DDL** control, select **Mountaineering Equipment**.

The results appear as follows:

Mountaineering Equipment		
Product line	Revenue	Quantity
Mountaineering Equipment	\$409,660,132.90	9,900,091
Overall - Total	409,660,132.9	9,900,091

The expected summary values are now displayed for Mountaineering Equipment.

11. Close the rendered report tab.
12. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



13. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You created a report with summarized values that accurately reflect the user's selection in Active Report controls.

Use Active Reports as prompt pages

- Active Reports can be used as a prompt for existing reports.
 - author the target report with parameterized filters.
 - create an Active Report
 - add the Report links object to the Active Report
- Reduce prompt startup time
 - schedule the Active Report to run overnight
- Target report can be an iFrame in the Active Report, or run in a separate viewer.

Use Active Reports as prompt pages

Active Reports can be used as a prompt page to filter an existing report.

To create a report with an Active Report prompt page, first author the target report, using parameterized filters.

Next, create the Active Report, and add a Report links object to the Active Report. This object will allow the Active Report to set parameters for the target report.

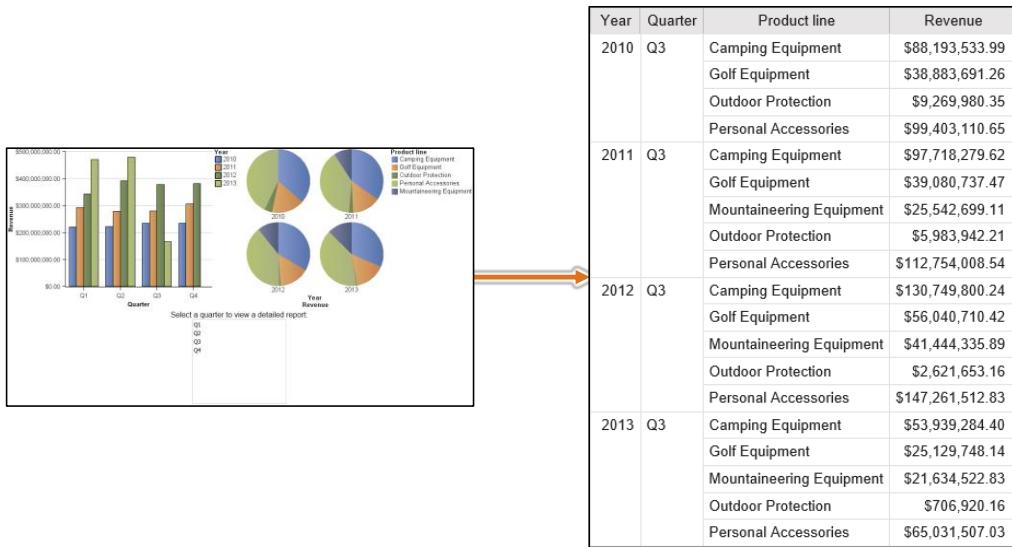
Because Active Reports query their data at the time they are first created, having an Active Report as a prompt page can improve startup performance, because the required data for the prompts has already been queried.

Active Reports can also make use of more advanced features, such as visualizations and charts, to make the prompt page more interactive and visually appealing.

You can configure the Active Report to show the target content in an iFrame within the report, or in a separate viewer.

Demonstration 5

Create an Active Report prompt page



Introduction to IBM Cognos Active Reports

© Copyright IBM Corporation 2018

Demonstration 5: Create an Active Report prompt page

Demonstration 5: Create an Active Report prompt page

Purpose:

Your company has requested a filtered report to compare quarterly income for specific quarters across all years. They would like to be able to select the quarter based on regional income

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create the target report

1. Using the **GO sales (query)** package, create a new **Blank active report**.
2. On the report page, click **Add** , then click **List** .
3. Click **OK** to accept the defaults and close the dialog.
4. From the **Data**  tab, Ctrl-click then drag the following query items to the List:
 - Time: **Year, Quarter**
 - Products: **Product line**
 - Sales: **Revenue**
5. Click the **<Year>** List column body, Ctrl-click **<Quarter>**, then on the Toolbar, click **Group / Ungroup** .

The results appear as follows:

Year	Quarter	Product line	Revenue
<Year>	<Quarter>	<Product line>	<Revenue>
	<Quarter>	<Product line>	<Revenue>
<Year>	<Quarter>	<Product line>	<Revenue>
	<Quarter>	<Product line>	<Revenue>

Task 2. Create a parameterized filter for the report

1. Click any cell in the List, then in the Toolbar, click **Filters**  > **Edit Filters**.
2. In the **Filters** dialog, click **Add** , select **Advanced**, then click **OK**.
3. In the **Detail filter expression** dialog, enter the following expression:
[Quarter]=?pQuarter?
 Hint: Drag Quarter from the Data items tab.
4. Click **Validate** , and then under **Provide a value**, select **Q2**, and click **OK**.
5. Click **OK** to close the **Detail filter expression** dialog.

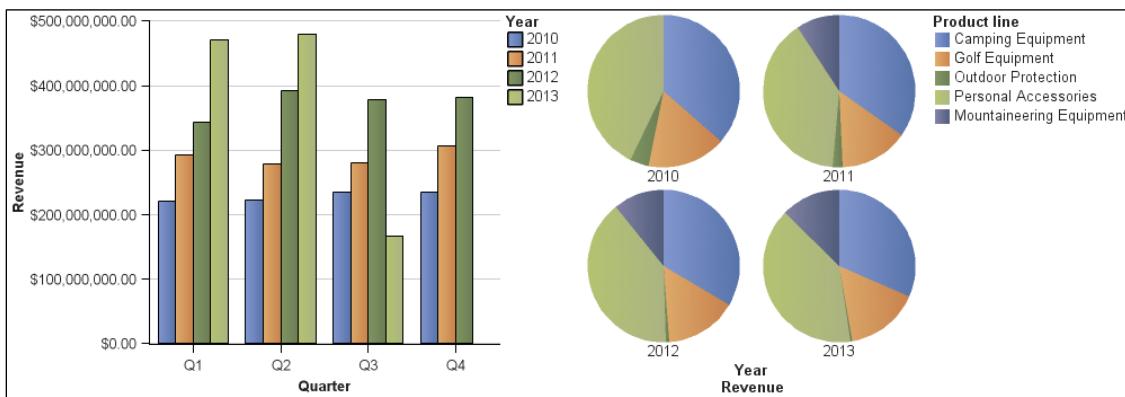
6. Click **OK** to close the **Filters** dialog.
7. On the Application bar, click **Save**, and then save the report under **My content** as **Prompt Target Report**.

Task 3. Create the prompt page report

You will create a prompt report that displays two charts to give a high-level view of the data for report consumers.

1. From the side bar, click **New** , and then click **Report** .
2. Double-click the **Blank active report** template.
3. In the **Data**  tab, click **Add report data** .
4. Navigate to **Team content**  > **Samples** > **Models**, click **GO sales (query)**, and then click **Open**.
5. In the page body, click **Add** , then click **Table** .
6. Set **Number of columns** to **2**, **Number of rows** to **2**, then click **OK**.
7. Click the bottom left table cell, Ctrl+click the bottom right table cell, and then on the Toolbar, click **Merge Cells** .
8. On the Toolbar, click **Horizontal alignment options** > **Center** .
9. In the top-left table cell, click **Add** , then click **Visualization** .
10. From the top right corner, click the visualization filter, then click **Charts**.
11. From the categories on the left, click **Column**, then double-click **Clustered Column**.
12. On the Application bar, click **Show properties**  to open the Properties pane.
13. In the Properties pane, under MISCELLANEOUS, set the **Name** property to **Quarterly_Revenue_Chart**.
14. Populate the Chart with the following data items from the Source tab:
 - Default measure (y-axis): Sales: **Revenue**
 - Series (primary axis): Time: **Year**
 - Categories (x-axis): Time: **Quarter**
15. In the top-right table cell, click **Add** , then click **Visualization** .
16. From the categories on the left, click **Pie, Donut**, then double-click **Pie**.
17. In the Properties pane, under MISCELLANEOUS, set the **Name** property to **Revenue by Product line**.
18. Populate the Chart with the following data items from the Source panel:
 - Default measure: Sales: **Revenue**
 - Series (pie slices): Products: **Product line**

- Categories (pies): Time: Year
19. On the Application bar, click **Run options** , then click **Run Active Report**.
The result appears as follows:



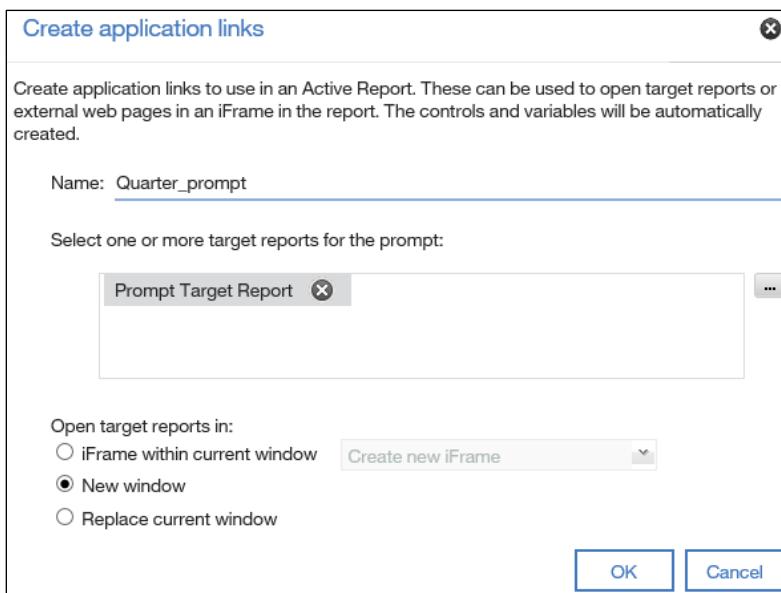
20. Close the rendered report tab.

Next, you will add a Report links object to connect this report to the target report.

Task 4. Add Report links to the report page

- From the **Toolbox**, expand ACTIVE REPORT, then drag a **Report links** object to the bottom table cell.
- In the **Create application links** dialog, beside **Name**, type **Quarter_prompt**.
- Under **Select one or more target reports for the prompt**, click the ellipsis .
- Navigate to **My content**, click **Prompt Target Report**, and then click **Open**.
- Under **Open target reports in**, select **New Window**.

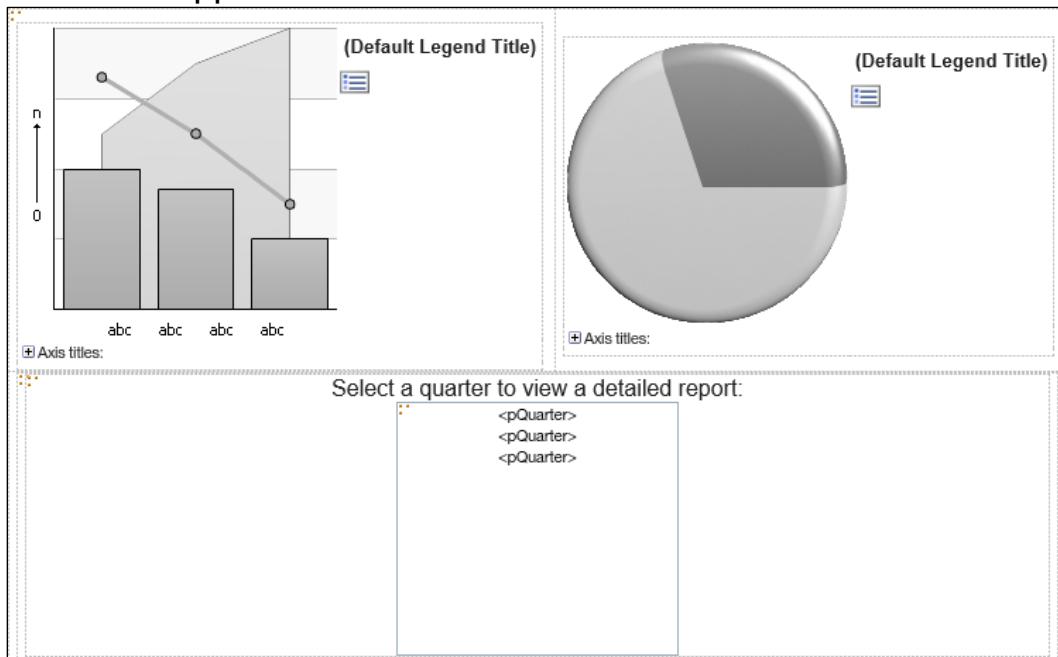
The results appear as follows:



- Click **OK**.
- Click the Data list box that was added.

8. In the Properties pane, under INTERACTIVE BEHAVIOR, set **Auto Select** to **None**.
9. Double-click the **pQuarter** Text item above the Data list box.
10. Type **Select a quarter to view a detailed report:**, type a space, and then click **OK**.

The result appears as follows:



11. From the Application bar, click **Run options** , then click **Run Active Report**.
12. In the Data list box, click **Q3**.

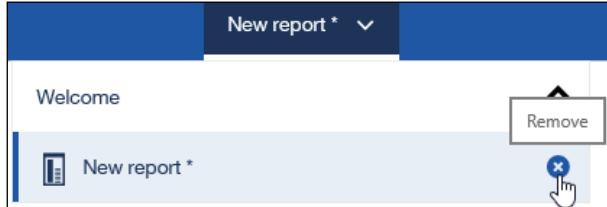
A new browser tab opens with the detailed report.

The results appear as follows:

Year	Quarter	Product line	Revenue
2010	Q3	Camping Equipment	\$88,193,533.99
		Golf Equipment	\$38,883,691.26
		Outdoor Protection	\$9,269,980.35
		Personal Accessories	\$99,403,110.65
2011	Q3	Camping Equipment	\$97,718,279.62
		Golf Equipment	\$39,080,737.47
		Mountaineering Equipment	\$25,542,699.11
		Outdoor Protection	\$5,983,942.21
		Personal Accessories	\$112,754,008.54
2012	Q3	Camping Equipment	\$130,749,800.24
		Golf Equipment	\$56,040,710.42
		Mountaineering Equipment	\$41,444,335.89
		Outdoor Protection	\$2,621,653.16
		Personal Accessories	\$147,261,512.83
2013	Q3	Camping Equipment	\$53,939,284.40
		Golf Equipment	\$25,129,748.14
		Mountaineering Equipment	\$21,634,522.83
		Outdoor Protection	\$706,920.16
		Personal Accessories	\$65,031,507.03

The List displays only data for Q3.

13. Close the rendered report's tab.
14. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



15. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You created an Active Report prompt page with charts and a prompt to select a specific quarter to view in a detailed report. The report links link your selection to the parameter in the target report and filters out data from other quarters.

Summary

- Describe Active Reports, and their value
- Save Active Reports
- Use Active Reports as prompt pages
- Convert existing reports to Active Reports
- Explain security considerations in Active Reports
- Debug Active Report behavior

Unit 2 Use Active Report connections

IBM Training



Use Active Report connections

IBM Cognos Analytics (v11.0)

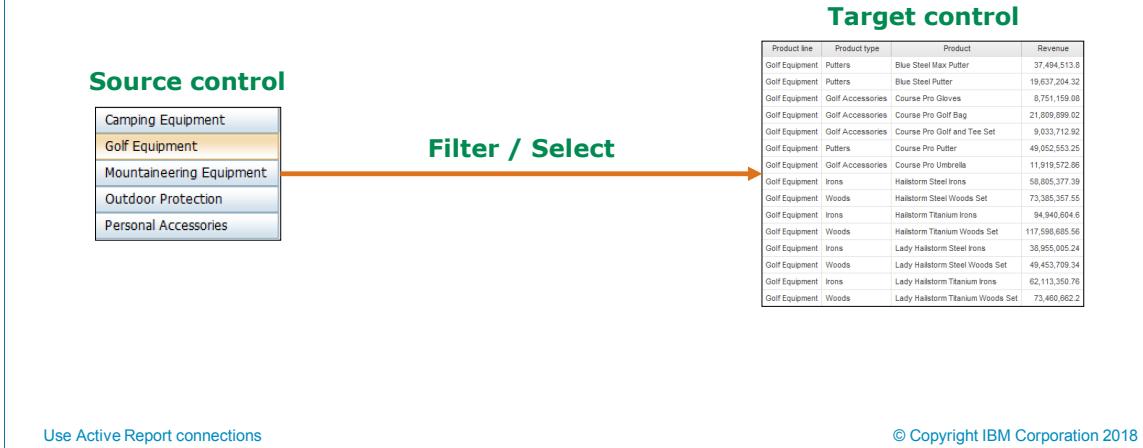
© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM.

Unit objectives

- Describe Active Report connections
- Filter and select Active Report controls
- Modify the interactive behavior of report controls
- Identify Active Report controls and variables
- Use variables to control multiple controls independently
- Control multiple controls from a single variable
- Author and optimize Active Reports for mobile consumption

Examine Active Report connections

- When you create a connection, the following occurs:
 - selection behavior is specified for the source control
 - variable is specified (created or reused)
 - reaction behavior is specified for the target control



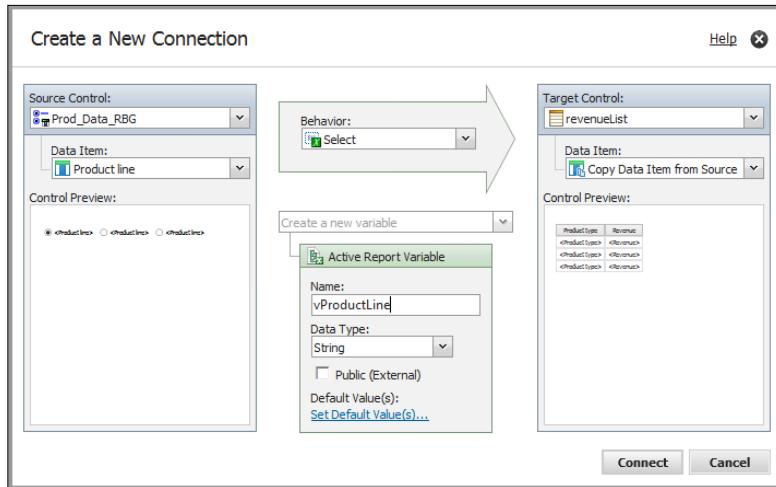
Examine Active Report connections

In this example, the author has created a connection between a Data button bar control and a List control, and has specified the relationship/behavior as Filter. The Data button bar uses a data item that will drive the behavior of the target control. In this case it is Product line. The List uses a data item that will determine what appears in the List. In this case it is Product line. In the report, when the end user clicks a Product line in the Data button bar, the List will be filtered based on what was selected. In this case, the user has clicked Golf Equipment, so the List displays rows for Golf Equipment.

IBM Cognos Analytics automatically sets the reaction behavior when you define a connection between controls using the Create a new connection wizard. However, you may want to change the reaction behavior or specify additional interactive behavior. This can be performed using the Interactive Behavior dialog box.

Create a New Connection wizard

- Use the Create a New Connection wizard to create a connection between two controls



Use Active Report connections

© Copyright IBM Corporation 2018

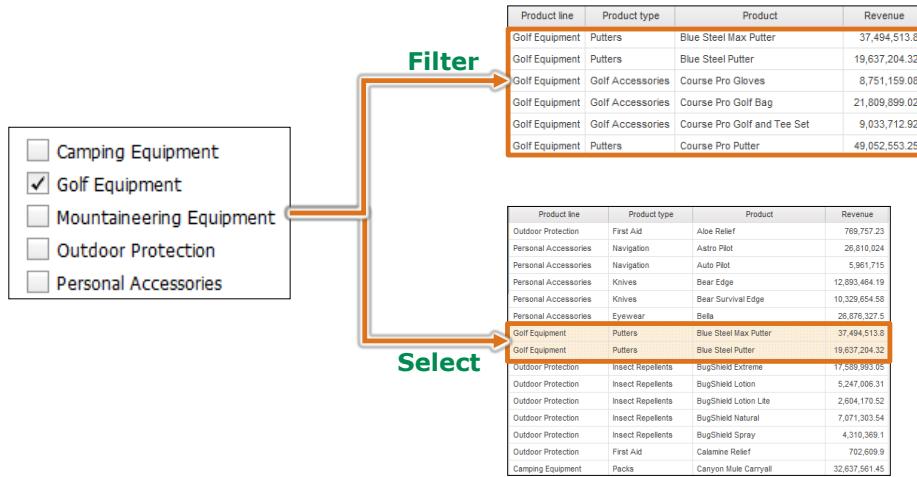
Create a new connection wizard

1. In the Source Control drop-down list, select the source control for which you want to define a connection.
2. Click the Data Item drop-down list under the source control and select the data item that will drive the behavior of the target control.
3. In the Target Control drop-down list, select the target control for which you want to define a connection.
4. Click the Data Item drop-down list under the target control and select the data item that will determine what appears in the target control.
5. Click the Behavior drop-down list between the source and target controls and specify the type of relationship that you want to define between the two controls:
 - Select
 - Filter

6. Below the Behavior area, click the Active Report Variable drop-down list and choose to create a new variable for the connection or use an existing variable.
Tip: If there are no variables defined in the report, you cannot click the Active Report Variable drop-down list.
 - If you create a new variable, type a name for the variable and specify its data type. You can also specify its default value(s)
 - If you choose to use an existing variable, choose the variable that you want. You can also specify its default value(s)
7. Click Connect.

Filter and select behavior

- In the Create a New Connection wizard, specify the behavior between the controls.



Use Active Report connections

© Copyright IBM Corporation 2018

Filter and select behavior

Specify Filter to filter data in the target control based on what will be selected in the source control.

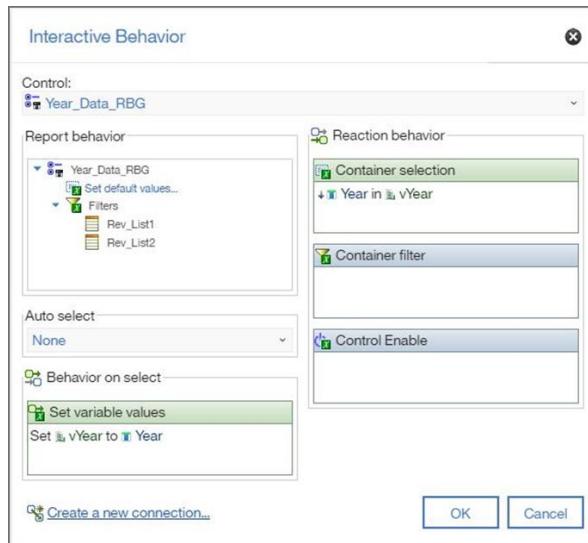
Specify Select to select data in the target control based on what will be selected in the source control.

Select behavior is different depending on the type of target control you are using. If the target is a control that displays data, such as a List or Crosstab, then the data values will be highlighted. If the target control is a container for data, such as a Radio Button Group or Deck, then the active button or card is selected.

After creating a connection, filter and select behavior between controls can be edited in the Interactive Behavior dialog.

Interactive Behavior dialog

- Use the Interactive Behavior dialog to edit behavior between controls.



Use Active Report connections

© Copyright IBM Corporation 2018

Interactive Behavior dialog

This is the Interactive Behavior dialog for an Active Report control:

Report behavior

- Shows the relationship between the control and other controls in the report, if you defined connections between it and other controls. You can specify default values for the variable associated with the control by clicking Set Default Value(s)

Behavior on select

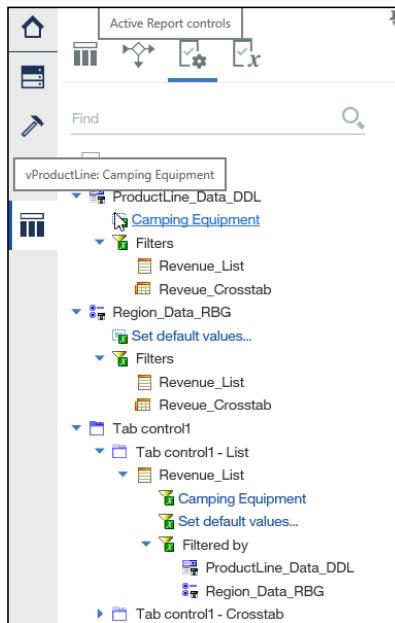
- Shows the effect of existing variables on the control
- Can edit behavior of existing variables for the control
- Create new variables and set behavior for the control

Reaction behavior

- The behavior of the target control is specified through a conditional statement that is applied to the value of the variable that is passed to the target control. Again, the value of the variable is set based on what is clicked or selected in the source control. Here you can:
 - Shows existing conditional statements
 - Can edit existing conditional statements
 - Create new conditional statements
- Container Select: data in the target control will be selected when the condition is met.
- Container filter: data in the target control will be filtered when the condition is met
- Control Enable: the target control will become interactive when the condition is met.

Identify Active Report controls in a report

- Identify all the controls in a report using the Active Report Controls tab



Use Active Report connections

© Copyright IBM Corporation 2018

Identify Active Report controls in a report

To identify all the Active Report controls and their relationships to one another you can use the Active Report controls tab (inside the Queries tab). There, you can also:

- click a control in this tab to quickly locate the control in the report.
- set default values for variables affecting the control.

This is how you interpret the example in the slide:

- Data drop down list named ProductLine_Data_DDL filters two controls: Revenue_List and Revenue_Crosstab, using the variable vProductLine, that has a default value of Camping Equipment, you can click the default value to change it.
- Data radio button group named Region_Data_RB filters two controls: Revenue_List and Revenue_Crosstab, using a variable that does not have a default value. You can click the link to set the default value.
- Tab control1 has two cards (one is called List, and the other is called Crosstab)
- Revenue_List is affected by two variables, one of them has a default value of Camping Equipment (it is vProductLine), and the other has no default value. It is filtered by two other controls: ProductLine_Data_DDL and Region_Data_RB

IBM Training IBM

Identify Active Report variables in a report

- Identify all the variables in a report using the Active Report variables tab.

Use Active Report connections © Copyright IBM Corporation 2018

Identify Active Report variables in a report

Active report variables work with active report controls to add interactivity to the report. If you want to identify all the variables and their behavior you can use the Active Report Variables tab. There you can also:

- create new variables
- set default values for variables
- manage range variables

This is how you interpret the variables in the slide:

- The vProductLine variable:
 - has a default value of Camping Equipment
 - is set by and selects the Data drop down list called ProductLine_Data_DDL
 - filters two controls: a List called Revenue_List and a Crosstab called Revenue_Crosstab
- The vRegion variable:
 - It does not have a default value

The “Manage ranges” link can be used to set the minimum and maximum values for range variables. Range variables are used in association with slider type controls to set the minimum and maximum values for the control. The variables must have the number data type. Two variables are used to define the minimum and maximum values of the

range variable. The two variables are used to define the minimum and maximum values of the slider.

Demonstration 1

Change filtering and selection behavior in a report

		2010	2011	2012	2013		2010	2011	2012	2013		2010	2011	2012	2013			
Year	Product line	Product type	Revenue	Year	Product line	Product type	Revenue	Year	Product line	Product type	Revenue	Year	Product line	Product type	Revenue	Year	Product line	
2010	Camping Equipment	Cooking Gear	\$59,761,536.50	2010	Camping Equipment	Cooking Gear	\$59,761,536.50	2010	Camping Equipment	Cooking Gear	\$59,761,536.50	2010	Camping Equipment	Cooking Gear	\$59,761,536.50	2010	Camping Equipment	
		Lanterns	\$28,662,904.19			Lanterns	\$28,662,904.19				Lanterns	\$28,662,904.19					Lanterns	
		Packs	\$70,296,289.17			Packs	\$70,296,289.17				Packs	\$70,296,289.17					Packs	
		Sleeping Bags	\$65,239,462.96			Sleeping Bags	\$65,239,462.96				Sleeping Bags	\$65,239,462.96					Sleeping Bags	
		Tents	\$109,026,145.24			Tents	\$109,026,145.24				Tents	\$109,026,145.24					Tents	
	Camping Equipment - Total		332,984,338.86		Camping Equipment - Total		332,984,338.86		Camping Equipment - Total		332,984,338.86		Camping Equipment - Total		332,984,338.86		Camping Equipment - Total	
	Golf Equipment	Golf Accessories	\$10,655,401.10		Golf Equipment	Golf Accessories	\$10,655,401.10		Golf Equipment	Golf Accessories	\$10,655,401.10		Golf Equipment	Golf Accessories	\$10,655,401.10		Golf Equipment	Golf Accessories
		Irons	\$54,093,311.24			Irons	\$54,093,311.24				Irons	\$54,093,311.24					Irons	
		Putters	\$29,419,377.82			Putters	\$29,419,377.82				Putters	\$29,419,377.82					Putters	
		Woods	\$59,385,760.82			Woods	\$59,385,760.82				Woods	\$59,385,760.82					Woods	
	Golf Equipment - Total		151,553,850.86		Golf Equipment - Total		151,553,850.86		Golf Equipment - Total		151,553,850.86		Golf Equipment - Total		151,553,850.86		Golf Equipment - Total	
	Outdoor Protection	First Aid	\$6,902,750.07		Outdoor Protection	First Aid	\$6,902,750.07		Outdoor Protection	First Aid	\$6,902,750.07		Outdoor Protection	First Aid	\$6,902,750.07		Outdoor Protection	First Aid
		Insect Repellents	\$17,964,327.13			Insect Repellents	\$17,964,327.13				Insect Repellents	\$17,964,327.13					Insect Repellents	
		Sunscreen	\$11,298,443.87			Sunscreen	\$11,298,443.87				Sunscreen	\$11,298,443.87					Sunscreen	
	Outdoor Protection - Total		36,163,521.87		Outdoor Protection - Total		36,163,521.87		Outdoor Protection - Total		36,163,521.87		Outdoor Protection - Total		36,163,521.87		Outdoor Protection - Total	
	Personal Accessories	Binoculars	\$29,246,444.08		Personal Accessories	Binoculars	\$29,246,444.08		Personal Accessories	Binoculars	\$29,246,444.08		Personal Accessories	Binoculars	\$29,246,444.08		Personal Accessories	Binoculars
		Eyewear	\$154,310,479.02			Eyewear	\$154,310,479.02				Eyewear	\$154,310,479.02					Eyewear	
		Knives	\$36,374,634.09			Knives	\$36,374,634.09				Knives	\$36,374,634.09					Knives	
		Navigation	\$51,598,510.99			Navigation	\$51,598,510.99				Navigation	\$51,598,510.99					Navigation	
		Watches	\$120,117,025.43			Watches	\$120,117,025.43				Watches	\$120,117,025.43					Watches	
	Personal Accessories - Total		391,647,093.61		Personal Accessories - Total		391,647,093.61		Personal Accessories - Total		391,647,093.61		Personal Accessories - Total		391,647,093.61		Personal Accessories - Total	
	2010 - Total		914,352,803.72		2010 - Total		914,352,803.72		2010 - Total		914,352,803.72		2010 - Total		914,352,803.72		Overall - Total	

Use Active Report connections

© Copyright IBM Corporation 2018

Demonstration 1: Change filtering and selection behavior in a report

Demonstration 1: Change filtering and selection behavior in a report

Purpose:

You need to modify the selection behavior within an Active Report so that when the value in a control is selected, data in another control is filtered instead of selected.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create the layout

1. Using the **GO sales (query)** package, create a new **Blank active report**.
2. Click the **Toolbox**  tab, then expand the **ACTIVE REPORT** section, then drag a **Data radio button group**  to the work area.
3. In the **Object and query name** dialog, under **Name**, type **Year_Data_RBG**, under **Query**, type **Year_Data_RBG_Query**, and then click **OK**.
4. From the **Data**  tab, drag **Time > Year**, to the **Year_Data_RBG** drop zone.
5. Click the **Toolbox** tab, then drag a **Table**  to the work area under **Year_Data_RBG**.
6. In the **Insert table** dialog, click **OK**.
7. On the Application bar, click **Show properties**  to open the Properties pane.
8. In the Properties pane, under **BOX**, double-click the **Margin** property to open it.
9. Set **Top margin** to **20**, then click **OK**.
10. Click the left table cell, then Ctrl-click the right cell.
11. From the **Toolbar**, click the small arrow **Vertical alignment options**  , then click **Top** .
12. In the left table cell, click **Add**  , then click **List** .
13. In the **Object and query name** dialog, set **Name** to **Revenue_List1**, and **Query Name** to **Revenue_Query**, then click **OK**.
14. From the **Data**  tab, drag the following data items to **Revenue_List1**:
 - **Time > Year**
 - **Products > Product line, Product type**
 - **Sales > Revenue**
15. Click **<Year>**, then Ctrl-click **<Product line>**, then on the **Toolbar**, click **Group / Ungroup** .

16. Click <Revenue>, then on the Toolbar, click **Summarize** Σ > **Summarize in Output** > **Total** Σ .

17. With **Revenue_List1** still selected, press **Ctrl-C** to copy it.

18. Click the right table cell, then press **Ctrl-V** to paste.

The result appears as follows:

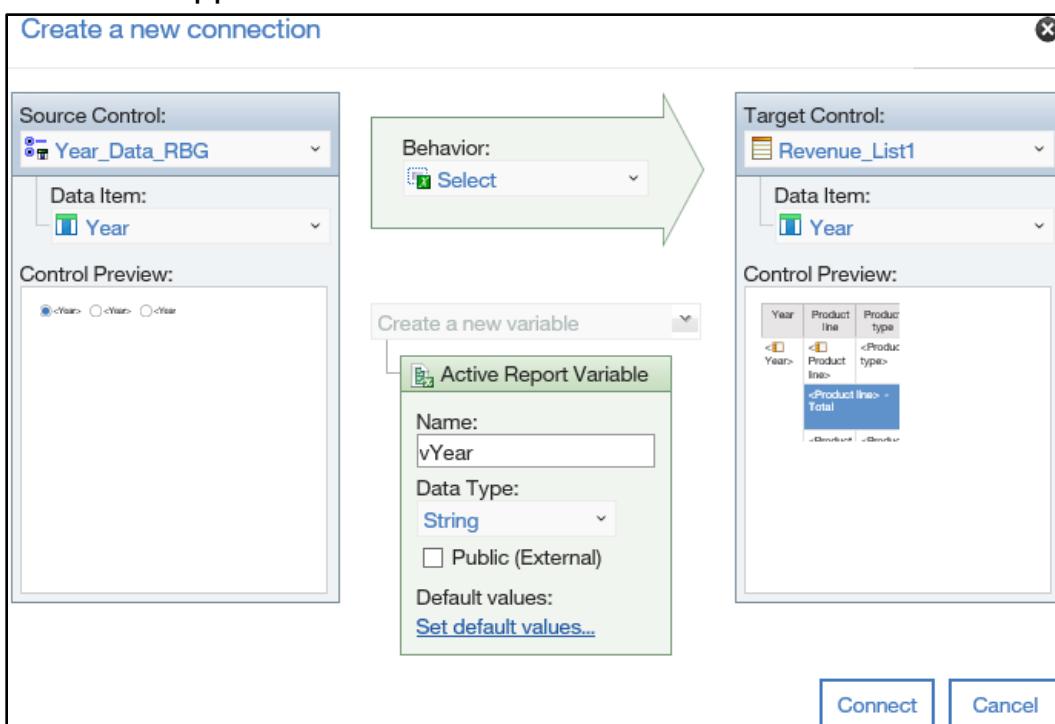
Year	Product line	Product type	Revenue
<Year>	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
<Year> - Total			<Total(Revenue)>
<Year>	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
<Year> - Total			<Total(Revenue)>
Overall - Total			<Total(Revenue)>

Year	Product line	Product type	Revenue
<Year>	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
<Year> - Total			<Total(Revenue)>
<Year>	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
<Year> - Total			<Total(Revenue)>
Overall - Total			<Total(Revenue)>

Task 2. Create two connections to the List objects

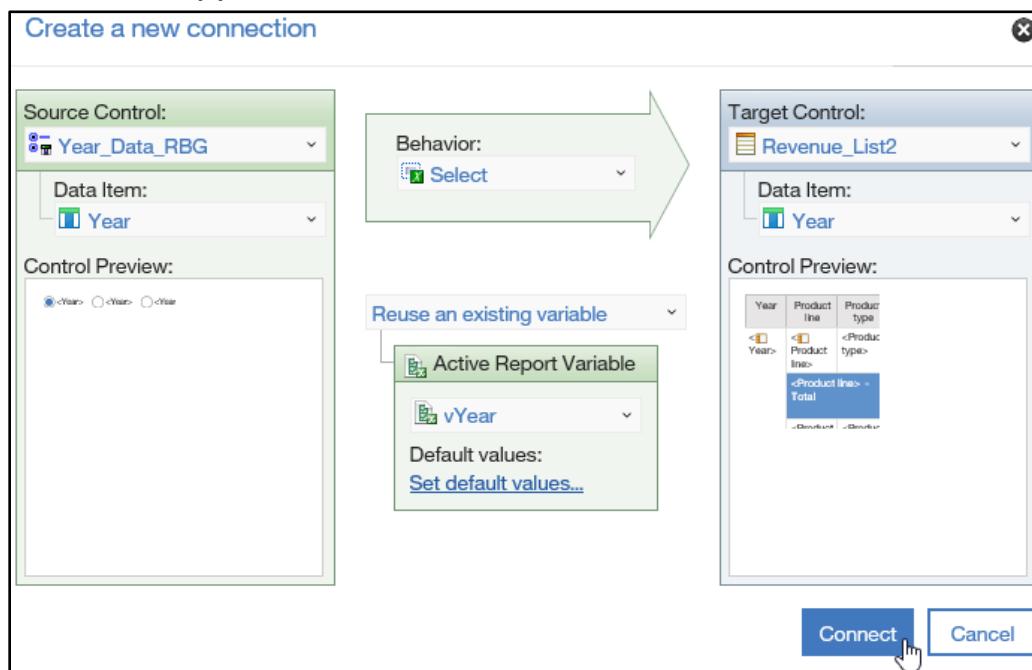
1. Click the **Year_Data_RBG** control, then from the Toolbar, click **Create a new connection** and specify the following connection information:
- **Year_Data_RBG (Year) > Select > Revenue_List1 (Year)**
 - New variable: **vYear**

The result appears as follows:



2. Click **Connect**.
3. In the Toolbar, click **Create a new connection** and specify the following connection information:
 - Year_Data_RBG (Year) > Select > Revenue_List2 (Year)
 - Reuse an existing variable: vYear

The result appears as follows:



4. Click **Connect**.
5. On the Application bar, click **Run options** (play icon), then click **Run Active Report**.
6. In the **Year_Data_RBG** control select **2010**.

The result appears as follows:

2010 2011 2012 2013				
Year	Product line	Product type	Revenue	
2010	Camping Equipment	Cooking Gear	\$59,761,536.50	
		Lanterns	\$28,662,904.19	
		Packs	\$70,296,289.17	
		Sleeping Bags	\$65,239,462.96	
		Tents	\$109,026,145.24	
Camping Equipment - Total		332,986,338.06		
2010	Golf Equipment	Golf Accessories	\$10,655,401.10	
		Irons	\$54,093,311.24	
		Putters	\$29,419,377.82	
		Woods	\$59,385,760.82	
Golf Equipment - Total		153,553,850.98		
2010	Outdoor Protection	First Aid	\$6,902,750.07	
		Insect Repellents	\$17,964,327.13	
		Sunscreen	\$11,298,443.87	
Outdoor Protection - Total		36,165,521.07		
2010	Personal Accessories	Binoculars	\$29,246,444.08	
		Eyewear	\$154,310,479.02	
		Knives	\$36,374,634.09	
		Navigation	\$51,598,510.99	
		Watches	\$120,117,025.43	
Personal Accessories - Total		391,647,093.61		
2010 - Total		914,352,803.72	914,352,803.72	

Both Lists are highlighted based on what is selected in the Data radio button group control.

7. Close the rendered report tab.

You will modify the reaction behavior so that when a value is selected in the Year_Data_RBG control, both lists are filtered instead of selected.

Task 3. Modify the reaction behavior of the List controls

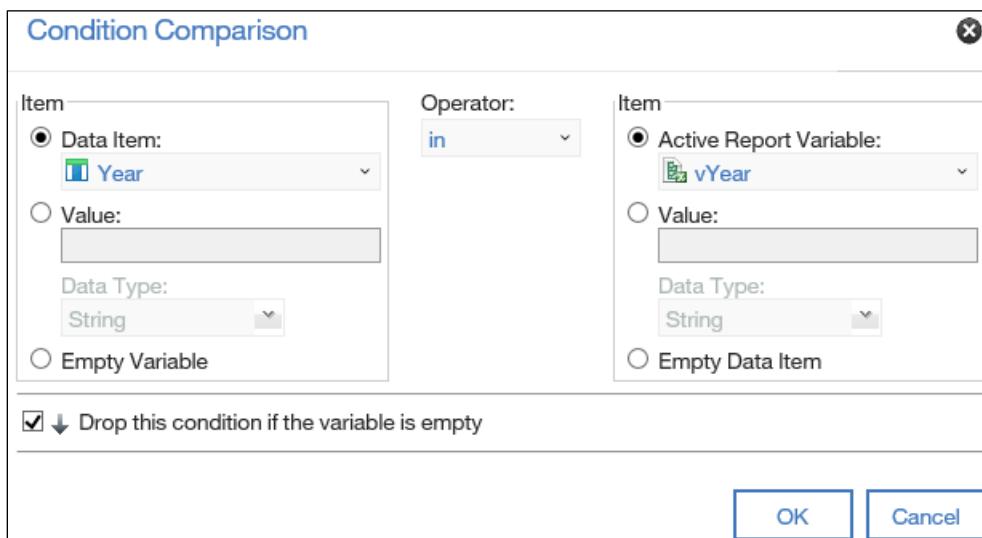
1. Click Revenue_List1, then in the Toolbar, click Interactive Behavior .

Under Report behavior, you can see that Revenue_List1 is currently selected by the Data radio button group control named Year_Data_RBG. You would like to change the reaction behavior of this control so that when a selection is made in Year_Data_RBG, Revenue_List1 is filtered instead of selected.

2. Under Reaction behavior, hover the cursor over Container selection, and then click Delete .

3. Hover the cursor over Container filter, and then click Edit .

4. Ensure that the Condition Comparison box appears as shown below:



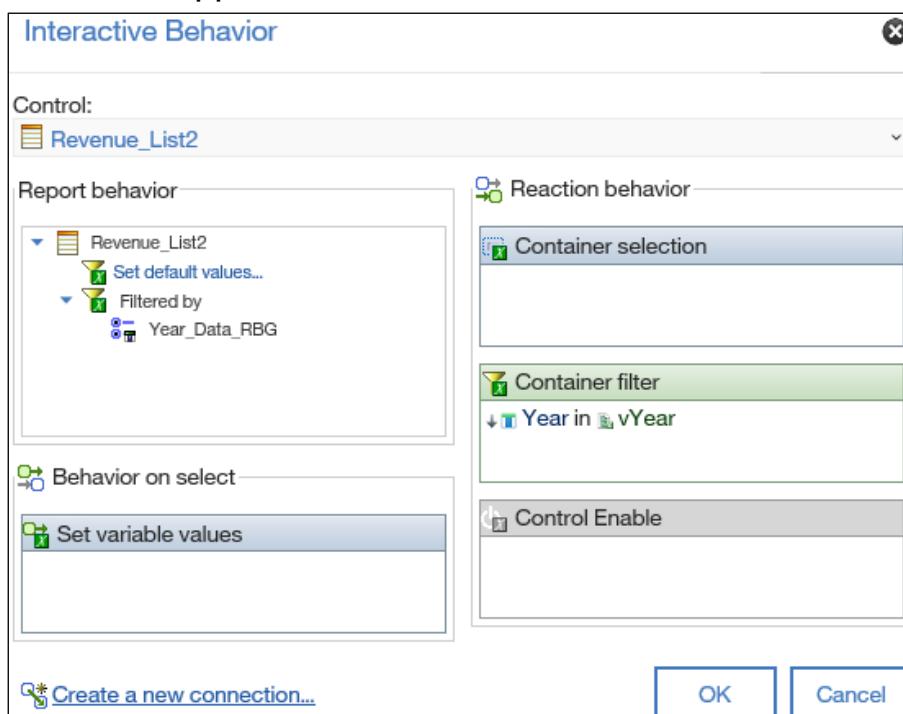
5. Click **OK**, and then click **OK** again.

The condition for Container filter appears as Year in vYear.

Now you would like to change the behavior on Revenue_List2 so that it is the same as Revenue_List1, with respect to how it will react to selections in Year_Data_RBG.

6. From the **Control** combo box, change the selected control from **Revenue_List1** to **Revenue_List2** to see its interactive behavior.
 7. Repeat steps **2 to 5**.

The result appears as follows:



8. Click **OK**.

9. On the Application bar, click **Run options** , then click **Run Active Report**.
10. In the **Year_Data_RB** control, select **2010**.

The result appears as follows:



2010 2011 2012 2013				
Year	Product line	Product type	Revenue	
2010	Camping Equipment	Cooking Gear	\$59,761,536.50	
		Lanterns	\$28,662,904.19	
		Packs	\$70,296,289.17	
		Sleeping Bags	\$65,239,462.96	
		Tents	\$109,026,145.24	
	Camping Equipment - Total		332,986,338.06	
2010	Golf Equipment	Golf Accessories	\$10,655,401.10	
		Irons	\$54,093,311.24	
		Putters	\$29,419,377.82	
		Woods	\$59,385,760.82	
Golf Equipment - Total			153,553,850.98	
2010	Outdoor Protection	First Aid	\$6,902,750.07	
		Insect Repellents	\$17,964,327.13	
		Sunscreen	\$11,298,443.87	
Outdoor Protection - Total			36,165,521.07	
2010	Personal Accessories	Binoculars	\$29,246,444.08	
		Eyewear	\$154,310,479.02	
		Knives	\$36,374,634.09	
		Navigation	\$51,598,510.99	
		Watches	\$120,117,025.43	
Personal Accessories - Total			391,647,093.61	
2010 - Total			914,352,803.72	
Overall - Total			914,352,803.72	

Year	Product line	Product type	Revenue	
2010	Camping Equipment	Cooking Gear	\$59,761,536.50	
		Lanterns	\$28,662,904.19	
		Packs	\$70,296,289.17	
		Sleeping Bags	\$65,239,462.96	
		Tents	\$109,026,145.24	
	Camping Equipment - Total		332,986,338.06	
2010	Golf Equipment	Golf Accessories	\$10,655,401.10	
		Irons	\$54,093,311.24	
		Putters	\$29,419,377.82	
		Woods	\$59,385,760.82	
Golf Equipment - Total			153,553,850.98	
2010	Outdoor Protection	First Aid	\$6,902,750.07	
		Insect Repellents	\$17,964,327.13	
		Sunscreen	\$11,298,443.87	
Outdoor Protection - Total			36,165,521.07	
2010	Personal Accessories	Binoculars	\$29,246,444.08	
		Eyewear	\$154,310,479.02	
		Knives	\$36,374,634.09	
		Navigation	\$51,598,510.99	
		Watches	\$120,117,025.43	
Personal Accessories - Total			391,647,093.61	
2010 - Total			914,352,803.72	
Overall - Total			914,352,803.72	

The behavior is changed, and now when a value is selected in the **Year_Data_RB** control, both Lists are filtered instead of selected.

11. Close the rendered report tab.

You will use this report as a starting point for a later demonstration, so you will save this report to My content.

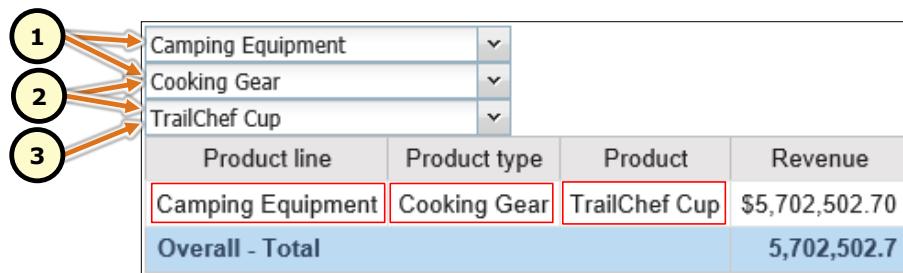
12. Save the report as **Unit 2 Demo 1** in **My content**.
13. Leave **IBM Cognos Analytics - Reporting** open for the next demonstration.

Results:

You modified the selection behavior within an existing Active Report so that when the value in a control is selected, data in another control is filtered instead of selected.

Interactive behavior example - cascading

- Set the interactive behavior between controls to achieve a cascading effect.



Use Active Report connections

© Copyright IBM Corporation 2018

Interactive behavior example - cascading

You can set/clear Active Report variables to achieve cascading.

1. A selection is made in the first Data drop-down list control. Variable values are set, and the second Data drop-down list control is populated based on selection. List is filtered.
2. A selection is made in second Data drop-down list control. Variable values are set, and the third Data drop-down list control is populated based on selection. List is filtered.
3. A selection is made in third Data drop-down list control. Variable values are set, and the List control is filtered based on selections in all Data drop-down list controls.

Demonstration 2

Create an Active Report with cascading controls

Camping Equipment	▼		
Cooking Gear	▼		
TrailChef Cup	▼		
Product line	Product type	Product	Revenue
Camping Equipment	Cooking Gear	TrailChef Cup	\$5,702,502.70
Overall - Total			5,702,502.7

Use Active Report connections

© Copyright IBM Corporation 2018

Demonstration 2: Create an Active Report with cascading controls

Demonstration 2: Create an Active Report with cascading controls

Purpose:

Add interactive behavior to Active Report controls to create a cascading effect between controls.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

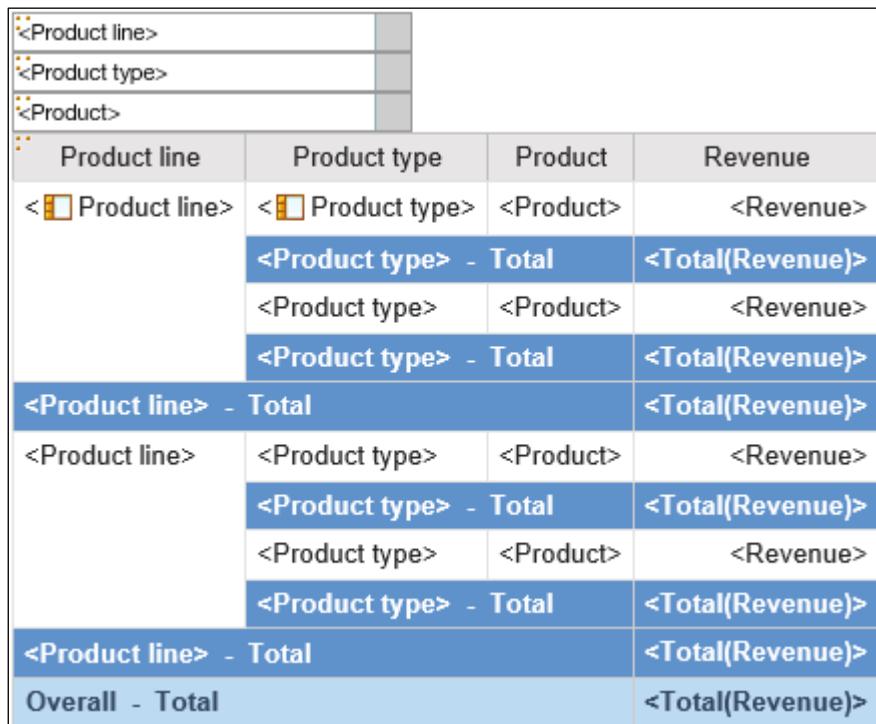
Namespace: Sales (query)

Task 1. Add controls and data to the Active Report

1. Using the **GO sales (query)** package, create a new **Blank active report**.
2. Click the **Toolbox**  tab, then expand the **ACTIVE REPORT** section.
3. Drag a **Data drop-down list**  to the work area. In the **Object and query name** dialog, specify the following properties:
 - Name: **ProductLine_Data_DDL**
 - Query: **Products_Query**
 - Click **OK**.
 You will use the same query (**Products_Query**) for the other Data drop-down lists.
4. From the Toolbox, drag a **Data drop-down list**  to the work area. In the **Object and query name** dialog:
 - Name: **ProductType_Data_DDL**
 - Query: **Products_Query**
 - Click **OK**.
5. From the Toolbox, drag a **Data drop-down list**  to the work area. In the **Object and query name** dialog:
 - Name: **Product_Data_DDL**
 - Query: **Products_Query**
 - Click **OK**.
6. From the **Data**  tab, drag Products > **Product line**, to the **ProductLine_Data_DDL** drop zone.
7. From the **Data**  tab, drag Products > **Product type**, to the **ProductType_Data_DDL** drop zone.
8. From the **Data**  tab, drag Products > **Product**, to the **Product_Data_DDL** drop zone.

9. From the **Data items** tab, Ctrl-click all data items, then drag them to the page under the last Data drop-down list.
This will create List1, containing the three data items.
10. From the **Source** tab, drag Sales > **Revenue** to the end of the List.
11. Click **<Product line>**, then Ctrl-click **<Product type>** then from the Toolbar, click **Group / Ungroup** .
12. Click **<Revenue>**, then from the Toolbar, click **Summarize**  > **Summarize in Output > Total** .

The result appears as follows:



<Product line>			
<Product type>			
<Product>			
Product line	Product type	Product	Revenue
<Product line>	<Product type>	<Product>	<Revenue>
		<Product type> - Total	<Total(Revenue)>
		<Product type>	<Revenue>
		<Product type> - Total	<Total(Revenue)>
<Product line> - Total			<Total(Revenue)>
<Product line>	<Product type>	<Product>	<Revenue>
		<Product type> - Total	<Total(Revenue)>
		<Product type>	<Revenue>
		<Product type> - Total	<Total(Revenue)>
<Product line> - Total			<Total(Revenue)>
Overall - Total			<Total(Revenue)>

Task 2. Connect the Data drop-down lists to the List

1. Click **ProductLine_Data_DDL**, then on the Toolbar, click **Create a new connection**, and specify the following connection properties:
 - **ProductLine_Data_DDL (Product line) > Filter > List1 (Product line)**
 - New variable: **vProductLine**
 - Default values: **Camping Equipment**
2. Click **ProductType_Data_DDL**, then in the Toolbar, click **Create a new connection**, and specify the following connection properties:
 - **ProductType_Data_DDL (Product type) > Filter > List1 (Product type)**
 - New variable: **vProductType**
 - Default values: **Camping Equipment**

3. Click **Product_Data_DDL**, then in the Toolbar, click **Create a new connection**, and specify the following connection properties:
 - **Product_Data_DDL (Product type)** > Filter > **List1 (Product)**
 - New variable: vProduct

Task 3. Add interactive behaviors on the Product type drop-down list

Upon selection of a product line, you want the product type and product drop-down lists to default to empty.

1. Click **ProductLine_Data_DDL**, then in the Toolbar, click **Interactive Behavior**.
2. Under **Behavior on select/Set variable values**, click **Edit**.
3. Beside **vProdType**, select **Set to Empty**.
4. Beside **vProduct**, select **Set to Empty**.
5. Click **OK** twice.

Task 4. Add interactive behaviors on the Product type drop-down list

Upon selection in the product line drop-down list, you want the product type drop-down list to be filtered on the values that belong to the product line selected. You will add product line to the query so the filter can be defined.

1. Click **ProductType_Data_DDL** to select it.
2. From the Data/Data Items tab, drag **Product line** to the **ProductType_Data_DDL/Extra Items** drop zone.
3. Click **ProductType_Data_DDL**, then in the Toolbar, click **Interactive Behavior**.
4. Under **Reaction behavior/Container filter**, click **Edit**.
5. Under **Data Item**, select **Product line**, under **Active Report Variable**, select **vProductLine**, and then click **OK** twice.

In addition, you want the product drop-down list to default to empty.

6. Under **Behavior on select/Set variable values**, click **Edit**.
7. Beside **vProduct**, select **Set to Empty**, and then click **OK** twice.

Task 5. Add interactive behaviors on the Product drop-down list

Upon selection in the product type drop-down list, you want the product drop-down list to be filtered on the values that belong to the product type selected. You will add product type to the query so the filter can be defined.

In addition, you want the product drop-down list to only be enabled once a user selects a product line.

1. Click the **Product_Data_DDL** object to select it.
2. From the Data/Data Items tab, drag **Product type** to the **Product_Data_DDL/Extra Items** drop zone.
3. Click **Product_Data_DDL**, then from the Toolbar, click **Interactive Behavior**.
4. Under **Reaction behavior/Container filter**, click **Edit**.
5. Under **Data Item**, select **Product type**, and then under **Active Report Variable**, select **vProductType**.
6. Click **OK** twice.
7. Under **Reaction behavior/Control Enable**, click **Edit**.
8. Under **Item** (on the left), select **Empty Variable**.
9. Under **Item** (on the right), under **Active Report Variable**, select **vProductType**.
10. Click **OK**, and then click **NOT**.
The expression should now read: NOT (vProductType is empty).
11. Click **OK** twice.

Task 6. Create default values for data drop-down lists

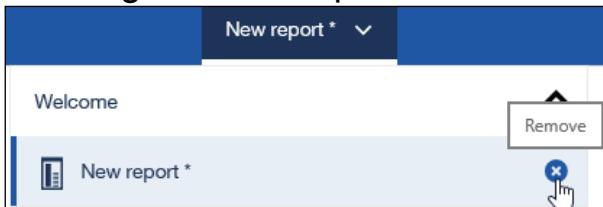
1. Select the **ProductType_Data_DDL** control, and then on the **Application bar**, click **Show properties**.
2. In the Properties pane, under **GENERAL**, set **No Value List Item** to **Show**.
The No Value List Item Properties appear.
3. Double-click **Label**, in the **Default** text box, type **All**, and then click **OK**.
4. Repeat steps 1 and 2 for the **Product_Data_DDL** control.
Next, you want to specify a product line default value.
5. On the side bar, click **Queries**, then click the **Active Report variables**  tab.
6. Expand **vProductLine**, and then click **Set default values**.
7. On the Application bar, click **Run options** , then click **Run Active Report**.
8. Select **Cooking Gear** as product type and **TrailChef Cup** as product.

The result appears as follows:

Product line	Product type	Product	Revenue
Camping Equipment	Cooking Gear	TrailChef Cup	\$5,702,502.70
Overall - Total			5,702,502.7

9. Close the rendered report tab.

10. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



11. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You have added interactive behavior to Active Report controls to create a cascading effect between controls.

Single vs multiple variables (1 of 2)

- Use a single variable to achieve the same data results in multiple controls.

vYear=2013

Year	Product line	Product type	Revenue
2013	Camping Equipment	Cooking Gear	\$58,313,800.35
		Lanterns	\$28,034,475.54
		Packs	\$83,157,796.99
		Sleeping Bags	\$68,730,008.17
		Tents	\$114,674,248.92
Golf Equipment	Golf Accessories	\$11,422,333.75	
	Irons	\$63,606,666.75	
	Putters	\$18,145,999.48	
	Woods	\$81,565,819.31	
Mountaineering Equipment	Climbing Accessories	\$29,483,205.87	

Year	Product line	Product type	Revenue
2013	Camping Equipment	Cooking Gear	\$58,313,800.35
		Lanterns	\$28,034,475.54
		Packs	\$83,157,796.99
		Sleeping Bags	\$68,730,008.17
		Tents	\$114,674,248.92
Golf Equipment	Golf Accessories	\$11,422,333.75	
	Irons	\$63,606,666.75	
	Putters	\$18,145,999.48	
	Woods	\$81,565,819.31	
Mountaineering Equipment	Climbing Accessories	\$29,483,205.87	

Use Active Report connections

© Copyright IBM Corporation 2018

Single vs multiple variables

In this example, two connections have been created on the Data radio button group control. For each of the connections specified, the same variable, vYear, is used to control the display of data in the two List controls. The result is that both Lists are filtered in the same manner, based on what is selected in the Data radio button group control. In this case, 2013 has been selected.

Single vs multiple variables (2 of 2)

- Use multiple variables to achieve different data results in multiple controls.

vPL1=Golf Equipment

2013

Year	Product line	Product type	Revenue
2013	Golf Equipment	Golf Accessories	\$11,422,333.75
		Irons	\$63,606,666.75
		Putters	\$18,145,999.48
		Woods	\$81,565,819.31

vPL2=Mountaineering Equipment, Outdoor Protection

2013

Year	Product line	Product type	Revenue
2013	Mountaineering Equipment	Climbing Accessories	\$29,483,205.87
		Rope	\$39,444,903.89
		Safety	\$28,014,081.20
		Tools	\$44,578,458.74
	Outdoor Protection	First Aid	\$846,984.93
		Insect Repellents	\$2,062,062.11
		Sunscreen	\$1,561,978.22

Use Active Report connections

© Copyright IBM Corporation 2018

In this example, a connection has been created for each of the Data check box group controls to connect to each List control. Each connection uses a different variable to control the display of data in the List controls. The Data check box group control on the left uses the variable PL1, while the Data check box group control on the right uses PL2. Both variables represent the same data item, and in this case it is Product line. The result is that each List is filtered based on which Product line is selected in each of the Data check box group controls. Golf Equipment has been selected on the left, which filters the List on the left. Mountaineering Equipment and Outdoor Protection have been selected on the right, which filters the List on the right.

Demonstration 3

Use single vs. multiple variables

○ 2010 ○ 2011 ● 2012 ○ 2013

Camping Equipment
 Golf Equipment
 Mountaineering Equipment
 Outdoor Protection
 Personal Accessories

Camping Equipment
 Golf Equipment
 Mountaineering Equipment
 Outdoor Protection
 Personal Accessories

Year	Product line	Product type	Revenue	
2011	Golf Equipment	Golf Accessories	\$13,251,774.09	
		Irons	\$55,116,575.97	
		Putters	\$28,923,250.88	
		Woods	\$70,714,826.13	
Golf Equipment - Total		168,006,427.07		
Overall - Total		168,006,427.07		

Year	Product line	Product type	Revenue	
2011	Mountaineering Equipment	Climbing Accessories	\$21,876,490.73	
		Rope	\$28,655,271.69	
		Safety	\$22,505,865.68	
		Tools	\$34,062,031.84	
Mountaineering Equipment - Total		107,099,659.94		
Personal Accessories		Binoculars		
		\$30,310,573.76		
		Eyewear		
		\$208,648,605.39		
		Knives		
		\$33,164,183.25		
		Navigation		
		\$43,724,569.80		
		Watches		
		\$140,475,423.70		
Personal Accessories - Total		456,323,355.9		
Overall - Total		563,423,015.84		

Use Active Report connections

© Copyright IBM Corporation 2018

Demonstration 3: Use single vs. multiple variables

Demonstration 3: Use single vs. multiple variables

Purpose:

Create a report to show how Active Report controls can be configured to use a single variable or multiple variables. You will begin by examining a report with a control that uses the same variable to control the display of the same data in different controls. You will then modify the report to include additional controls that use multiple variables to control the display of the same data in different controls.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Examine Active Report controls that are configured to use a single variable

1. On the side panel, click **My content**, then right-click **Unit 2 Demo 1**, then click **Edit report**.
2. Click the **Revenue_List1**, then in the Toolbar, click **Interactive Behavior** .
3. Hover the cursor over **Container filter**, and then click **Edit**.
According to the container filter expression, the behavior for this List control is controlled by a variable named vYear.
4. Click **Cancel** twice.
5. Repeat steps 2 to 4 for the **Revenue_List2** control.
Both List controls are controlled by the vYear variable.
You will now modify the report to include controls that use multiple variables to control the display of data in different controls.

Task 2. Add additional controls that use multiple variables

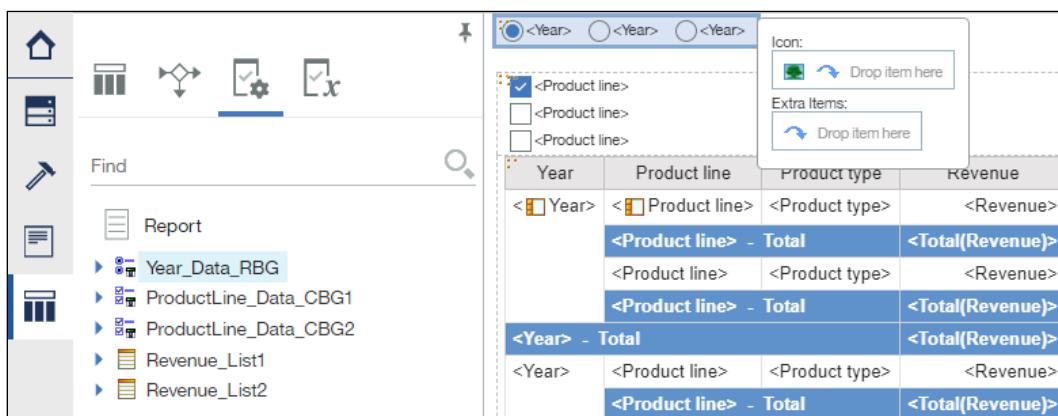
1. Click anywhere in **Revenue_List1**, then from the Toolbar, click **More > Insert > Rows above**.
This will insert a table row before the row containing the two List objects.
2. On the side bar, click **Toolbox**.
3. Drag a **Data check box group** to the left table cell of the new row.
 - Name: **ProductLine_Data_CBG1**
 - Query: **ProductLine_Query**
4. Drag a **Data check box group** to the right table cell of the new row.

- Name: **ProductLine_Data_CBG2**
 - Query: **ProductLine_Query**
5. Click the **ProductLine_Data_CBG1** control to select it, and then in the **Properties** pane, under **GENERAL**, double-click **Orientation** to change from **Horizontal** to **Vertical**.
 6. Repeat step 5 to make the **ProductLine_Data_CBG2** control display vertically.
 7. On the side bar, click **Data**.
 8. On the **Source** tab, from **Sales (query) > Products**, drag **Product line** to the drop zone of the **ProductLine_Data_CBG1** control.
 9. In the Toolbar, click **Create a new connection**, and specify the following connection information:
 - **ProductLine_Data_CBG1 > Filter > Revenue_List1**
 - New variable: **vProductLine1**
 - Click **Connect**
 10. On the **Data items** tab, from **ProductLine_Query**, drag **Product line** to the drop zone of the **ProductLine_Data_CBG2** control.
 11. In the Toolbar, click **Create a new connection**, and specify the following connection information:
 - **ProductLine_Data_CBG2 > Filter > Revenue_List2**
 - New variable: **vProductLine2**
 - Click **Connect**

Task 3. Review controls and variables, and run the report

1. On the side bar, click **Queries**, then click the **Active Report controls**  tab, then click **Year_Data_RBG**.

The result appears as follows:



The screenshot shows the SAP BusinessObjects Web Intelligence interface. On the left, the navigation pane displays several reports: Year_Data_RBG (highlighted in blue), ProductLine_Data_CBG1, ProductLine_Data_CBG2, Revenue_List1, and Revenue_List2. The main work area contains a table with the following data:

Icon	Extra Items		
<input checked="" type="checkbox"/> <Product line>	<input type="button" value="Drop item here"/>		
<input type="checkbox"/> <Product line>	<input type="button" value="Drop item here"/>		
<input type="checkbox"/> <Product line>	<input type="button" value="Drop item here"/>		
<Year>	Product line	Product type	Revenue
< <input type="checkbox"/> Year>	< <input type="checkbox"/> Product line>	<Product type>	<Revenue>
< <input checked="" type="checkbox"/> Product line>	- Total		<Total(Revenue)>
<Product line>	<Product type>		<Revenue>
< <input checked="" type="checkbox"/> Product line>	- Total		<Total(Revenue)>
<Year> - Total			<Total(Revenue)>
<Year>	<Product line>	<Product type>	<Revenue>
< <input checked="" type="checkbox"/> Product line>	- Total		<Total(Revenue)>

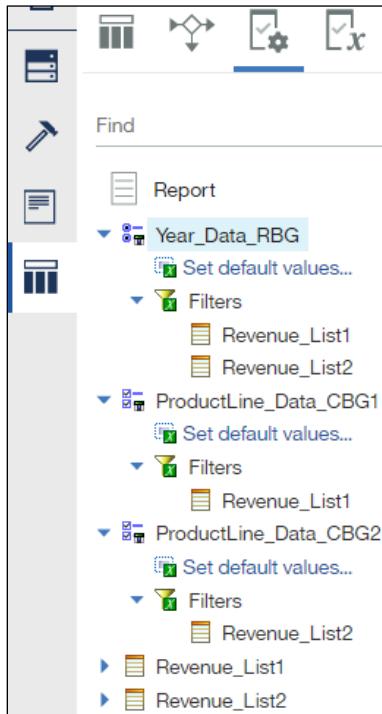
The **Year_Data_RBG** control is highlighted in the work area.

You can use this method to identify all the controls in the report, and select controls in the work area.

2. Expand the following:

- **Year_Data_RBG > Filters**
- **ProductLine_Data_CBG1 > Filters**
- **ProductLine_Data_CBG2 > Filters**

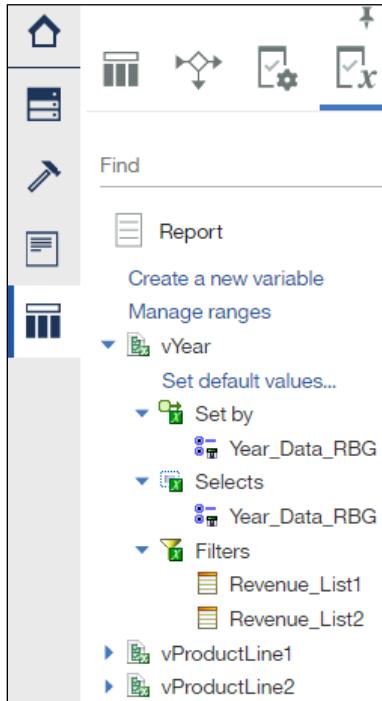
The result appears as follows:



You can see all the controls and their relationship to one another. For example, Year_Data_RBG filters both Revenue_List1 and Revenue_List2. While ProductLine_Data_CBG1 filters only Revenue_List1, and ProductLine_Data_CBG2 filters only Revenue_List2.

3. Click the **Active Report variables** tab, then expand **vYear**, **Set By**, **Selects**, and **Filters**.

The result appears as follows:



You can see all the variables in this report and how they are used. For example, the vYear variable, is set by the Year_Data_RBG control, is used to select a value in that control, and filters both Revenue_List1 and Revenue_List2.

4. Run the report, and then select **2011** in the **Year_Data_RBG** control.
Both lists are filtered based on what is selected in this control.

5. In the **ProductLine_Data_CBG1** control (on the left) select **Golf Equipment**, and then in the **ProductLine_Data_CBG2** control (on the right) select **Mountaineering Equipment** and **Personal Accessories**.

The result appears as follows:

Year	Product line	Product type	Revenue	
2011	Golf Equipment	Golf Accessories	\$13,251,774.09	
		Irons	\$55,116,575.97	
		Putters	\$28,923,250.88	
		Woods	\$70,714,826.13	
Golf Equipment - Total		168,006,427.07		
Overall - Total		168,006,427.07		

Year	Product line	Product type	Revenue	
2011	Mountaineering Equipment	Climbing Accessories	\$21,876,490.73	
		Rope	\$28,655,271.69	
		Safety	\$22,505,865.68	
		Tools	\$34,062,031.84	
Mountaineering Equipment - Total		107,099,659.94		
Personal Accessories - Total		456,323,355.9		
Overall - Total		563,423,015.84		

For each of the connections created for the Data check box group controls, different variables are used to control the display of data in the List controls. The result is that each List is filtered based on what is selected in each of the Data check box group controls.

6. Close the browser tab.
7. Leave **IBM Cognos Analytics** open for the next demonstration.

Results:

You have created a report that shows how Active Report controls can be configured to use a single variable or multiple variables.

Optimize Active Reports for mobile devices

- Consumers can use the IBM Cognos Mobile on iOS and Android to access Active Reports.
- Controls are automatically optimized to use the multi-touch gestures and style of the application.
- Use the optimal screen sizing for the device to ensure that the entire report is visible without scrolling
- For objects whose data can be difficult to select on a touch interface (i.e. maps), provide an alternate control (i.e. drop-down list).
- Use the iterator control with a deck control to enable the swipe gesture.

Optimize Active Reports for mobile devices

For both iOS and Android, consumers can use either the native or Web versions of IBM Cognos Analytics. You can search your application store for “IBM Cognos Mobile”.

Using the optimal screen sizing for the device improves the end user experience by ensuring that the entire report is visible without scrolling.

Ensure that all objects and containers are sized properly. If an object is larger than its container, the container will be pushed out and the entire report layout will be affected.

Large objects (such as lists) in a smaller block can result in scrollable data. To remain consistent with iOS standards, scrollable areas do not have visible scroll bars. End users should be made aware of this functionality.

Page dots have default colors (Dot Color and Active Dot Color) that you can change if the background color, or image, is too similar.

Demonstration 4

Create an Active Report using slider controls

2010 2011 2012 2013

Regions selected:
Americas, Central Europe, Northern Europe

Americas Asia Pacific Central Europe Northern Europe Southern Europe

Row number	Year	Region	Quantity	Row number
1	2010	Americas	6,439,569	1
2	2010	Central Europe	4,489,368	3
3	2010	Northern Europe	2,195,300	4
4	2011	Americas	7,091,968	6
5	2011	Central Europe	5,421,128	8
6	2011	Northern Europe	2,643,372	9
7	2012	Americas	7,866,166	11
8	2012	Central Europe	5,662,099	13
9	2012	Northern Europe	2,926,050	14
10	2013	Americas	5,814,848	16
11	2013	Central Europe	4,366,972	18
12	2013	Northern Europe	2,270,477	19
Overall - Total			57,187,317	

Use Active Report connections

© Copyright IBM Corporation 2018

Demonstration 4: Create an Active Report using slider controls

Demonstration 4: Create an Active Report using slider controls

Purpose:

You will create an Active Report that allows users to display data based on a range of values for Sales Quantity and Year, as well as the ability to filter data by Region.

Portal: <http://vclassbase/ibmcognos>

User/Password: **brettonf/Education1**

Template: **Blank active report**

Package: Go sales (query)

Namespace: **Sales (query)**

Task 1. Create an Active Report

1. From the **Welcome** page, navigate to **Team content** > **Samples** > **Models**.
 2. Right-click the **GO Sales (query)** package, then click **Create report** .
 3. In the **Templates and themes** dialog, double-click **Blank active report**.
 4. In the page body, click **Add** , then click **Table** .
 5. Set **Number of rows** to **3**, then click **OK**.
 6. From the **Toolbox** tab, drag a **List** to the bottom right cell of the table:
 - Name: **Quantity_List**
 - Query Name: **Quantity_List_Query**
 7. On the side bar, click **Data**, then expand **Sales (query)**.
 8. From the **Source** tab, drag the following query items to the List:
 - **Time > Year**
 - **Retailers > Region**
 - **Sales > Quantity**
 9. Click **<Quantity>**, then from the Toolbar, click **Summarize** > Summarize in Output > **Total**.

The result appears as follows:

Year	Region	Quantity
<Year>	<Region>	<Quantity>
<Year>	<Region>	<Quantity>
<Year>	<Region>	<Quantity>
Overall - Total		<Total(Quantity)>

Task 2. Create a Year filter using a discrete values slider

1. In the **Toolbox** tab, expand **ACTIVE REPORT**, and then drag a **Data discrete values slider** control to the top right table cell:
 - Name: **Year_Data_DVS**
 - Query Name: **Year_Data_DVS_Query**
2. On the side bar, click **Data**.
3. On the **Source** tab, under **Time**, drag **Year** to the **Labels** drop zone of the **Year_Data_DVS** control.
4. In the Toolbar, click **Create a new connection**, and specify the following:
 - **Year_Data_DVS (Year) > Select > Quantity_List (Year)**
 - New variable: **vYear**
 - Click **Connect**.
5. On the **Application bar**, click **Show properties**.
6. In the Properties pane, under **GENERAL**, set the **Range Type** property to **Range**.
The control will now display two thumb controls that let the user set both a minimum and maximum range of values.
7. Run the report, and then drag the right thumb to **2012** and the left thumb to **2011**.
The results appear as follows:

Year	Region	Quantity
2010	Americas	6,439,569
2010	Asia Pacific	5,072,815
2010	Central Europe	4,489,368
2010	Northern Europe	2,195,300
2010	Southern Europe	1,977,678
2011	Americas	7,091,968
2011	Asia Pacific	5,907,433
2011	Central Europe	5,421,128
2011	Northern Europe	2,643,372
2011	Southern Europe	2,460,784
2012	Americas	7,866,166
2012	Asia Pacific	6,617,098
2012	Central Europe	5,662,099
2012	Northern Europe	2,926,050
2012	Southern Europe	2,870,377
2013	Americas	5,814,848
2013	Asia Pacific	4,992,194
2013	Central Europe	4,366,972
2013	Northern Europe	2,270,477
2013	Southern Europe	2,151,395
Overall - Total		89,237,091

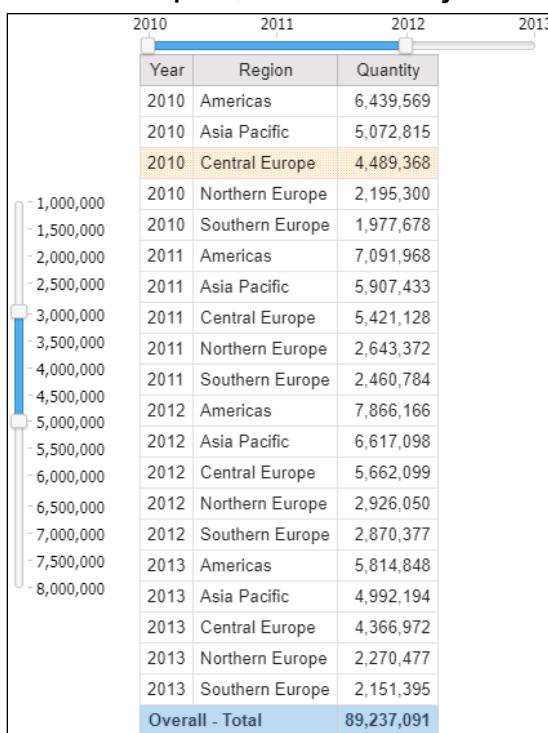
When the Data discrete values slider thumbs are set to 2011 and 2012, only the data for those years is highlighted.

8. Close the browser tab.

Task 3. Create a Quantity filter using a Continuous values

slider

1. From the Toolbox tab, drag a **Continuous values slider** to the bottom left table cell.
 2. In the Properties pane, set the following properties:
 - GENERAL > Orientation: **Vertical**
 - GENERAL > Range Type: **Range**
 - CONTINUOUS SLIDER > Step: **500000** (five hundred thousand)
 - CONTINUOUS SLIDER > Minimum Value: **1000000** (one million)
 - CONTINUOUS SLIDER > Maximum Value: **8000000** (eight million)
 - MISCELLANEOUS > Name: **Quantity_CVS**
- Note: If you do not enter the Step value before you enter the Minimum and Maximum values, an error message displays. You can simply click OK and set the Step value at this point.
3. In the Toolbar, click **Create a new connection**, and specify the following:
 - Quantity_CVS (Continuous Value) > **Select** > **Quantity_List** (Quantity)
 - **Minimum** tab > Active Report Variable > **vQuantity (Minimum)**
 - **Maximum** tab > Active Report Variable > **vQuantity (Maximum)**
 - Click **Connect**.
 4. Click the table cell that contains the **Quantity_CVS** control.
 5. In the **Properties** pane, under **POSITIONING**, set the **Size & Overflow > Width** property to **100px**.
 6. Run the report, and then adjust the slider thumbs to match the following report:



The screenshot shows a report interface with a header containing the years 2010, 2011, 2012, and 2013. Below the header is a horizontal slider with a blue track and a white thumb, positioned between the 2010 and 2011 markers. To the left of the slider is a vertical scale from 1,000,000 to 8,000,000. Below the slider is a table with four columns: Year, Region, and two numerical columns. At the bottom of the table is a row labeled "Overall - Total".

Year	Region	Quantity	Quantity
2010	Americas	6,439,569	
2010	Asia Pacific	5,072,815	
2010	Central Europe	4,489,368	
2010	Northern Europe	2,195,300	
2010	Southern Europe	1,977,678	
2011	Americas	7,091,968	
2011	Asia Pacific	5,907,433	
2011	Central Europe	5,421,128	
2011	Northern Europe	2,643,372	
2011	Southern Europe	2,460,784	
2012	Americas	7,866,166	
2012	Asia Pacific	6,617,098	
2012	Central Europe	5,662,099	
2012	Northern Europe	2,926,050	
2012	Southern Europe	2,870,377	
2013	Americas	5,814,848	
2013	Asia Pacific	4,992,194	
2013	Central Europe	4,366,972	
2013	Northern Europe	2,270,477	
2013	Southern Europe	2,151,395	
Overall - Total		89,237,091	

The highlighted data reflects only the results that match the values of both slider controls.

7. Close the browser tab.

Task 4. Add a Data check box group to filter data by region

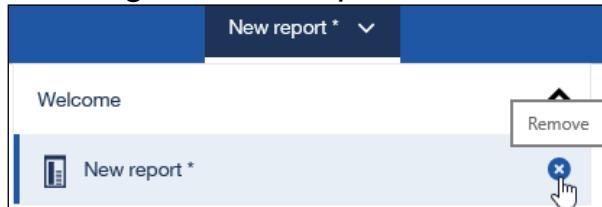
1. From the **Toolbox**, drag a **Data check box group** to the middle-right cell, above the **Quantity_List** control:
 - Name: **Region_Data_CBG**
 - Query Name: **Region_Data_CBG_Query**
2. On the side bar, click **Data**.
3. From the **Source** tab, from **Retailers**, drag **Region** to the **Region_Data_CBG** drop zone.
4. In the Toolbar, click **Create a new connection**, and specify the following:
 - Region_Data_CBG (Region) > **Filter** > **Quantity_List (Region)**
 - New variable: **vRegion**
 - Click **Connect**.
5. On the side bar, click **Toolbox**.
6. From the **Toolbox**, drag a **Variable text item** object to the left of the **Region_Data_CBG** control.
7. Set the **Active Report Variable** to **vRegion**, then click **OK**.
8. From the **Toolbox** tab, drag a **Text item** to the left of the new variable text item, type **Regions selected:**, type a space, and then click **OK**.
9. In the **Toolbox** tab, from the **TEXTUAL** section, drag a **Row number** object to the List control as the last column.
10. In the **Toolbox** tab, from the **ACTIVE REPORT** section, drag a **Row number** object to the List control as the first column.
11. On the Application bar, click **Run options** , then click **Run Active Report**.
12. Adjust the controls as follows:
 - Year_Data_CSV: **2010** to **2012**
 - Quantity_CSV: **2000000** to **5000000** (2 million to 5 million)
 - Region_Data_CBG: **Americas**, **Central Europe**, and **Northern Europe**

The results appear as follows:

	2010	2011	2012	2013
Regions selected:	Americas, Central Europe, Northern Europe	<input checked="" type="checkbox"/> Americas <input type="checkbox"/> Asia Pacific <input checked="" type="checkbox"/> Central Europe <input checked="" type="checkbox"/> Northern Europe <input type="checkbox"/> Southern Europe		
- 1,000,000				
- 1,500,000				
- 2,000,000				
- 2,500,000				
- 3,000,000				
- 3,500,000				
- 4,000,000				
- 4,500,000				
- 5,000,000				
- 5,500,000				
- 6,000,000				
- 6,500,000				
- 7,000,000				
- 7,500,000				
- 8,000,000				
Overall - Total				57,187,317

Note: The values in the ACTIVE REPORT Row Number column reflect the number of visible rows in the report. The values in the TEXTUAL Row Number column reflect the number of rows that the server returned for the query.

13. Close the rendered report tab.
14. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



15. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You created an **Active Report** that allows users to display data based on a range of values for Sales Quantity and Year, as well as the ability to filter data by Region.

Unit summary

- Describe Active Report connections
- Filter and select Active Report controls
- Modify the interactive behavior of report controls
- Identify Active Report controls and variables
- Use variables to control multiple controls independently
- Control multiple controls from a single variable
- Author and optimize Active Reports for mobile consumption

Unit 3 Active Report charts, visualizations, and decks

IBM Training

IBM

Active Report charts, visualizations, and decks

IBM Cognos Analytics (v11.0)

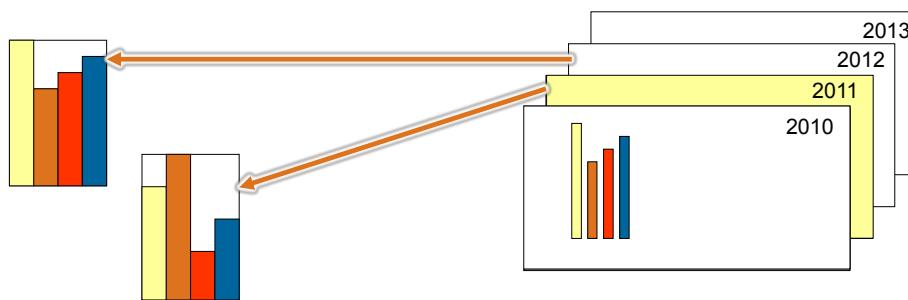
© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM.

Objectives

- Describe characteristics of traditional charts in Active Reports
- Control data display using decks and data decks
- Use decks and data decks to display traditional charts
- Optimize decks for performance
- Describe characteristics of RAVE Visualizations

Traditional charts in Active Reports

- No client-side charting engine
 - chart images generated at runtime
 - all images stored in MHT file
- Use a deck or query to filter content



Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Traditional charts in Active Reports

When using traditional charts in Active Reports, there are some differences to be aware of compared to charts used in other IBM Cognos Analytics reports.

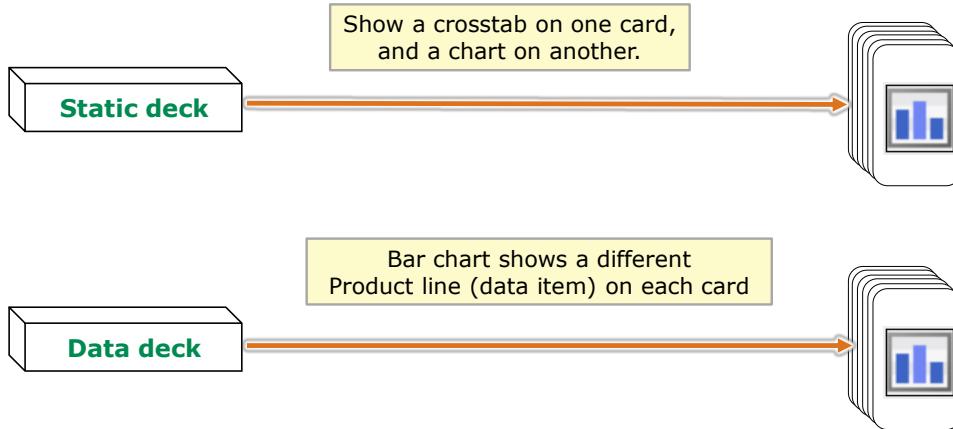
First, the generated MHT file for an Active Report does not contain a charting engine to dynamically generate charts. Instead, IBM Cognos Analytics generates all possible renditions of charts in an Active Report when it is first run, and then stores images of each of those charts in the MHT for later retrieval.

Because the charts are generated at the time the MHT file is first generated, filters cannot be applied to a chart to allow it to update its content based on the user's selections in the Active Report.

In order to create charts that update to reflect the user's selections, you need to either assign each unique chart to its own query, or put the chart inside a Deck to show each chart image based on the card selection in the Deck.

Examine decks

- Decks allow you to show cards containing different objects or data based on user selection in a control.



Examine decks

Decks in Active Reports can be compared to a deck of playing cards. At any time, the top card of the deck is visible to the user, and the user can cycle through each card, one at a time.

You can include objects such as Lists, Crosstabs, or Charts in a Deck.

Data items are added to a Data deck. The data items added to the Deck will define the cards that are created.

In static Decks, you define the number of cards you want, and define the content of each card.

Deck or Tab control?

- Tab control has one selection method (tabs).
- Static decks can link to other selection methods:
 - radio button groups
 - button bars
 - custom image buttons (static repeater table)
 - slider bars

Deck or Tab control?

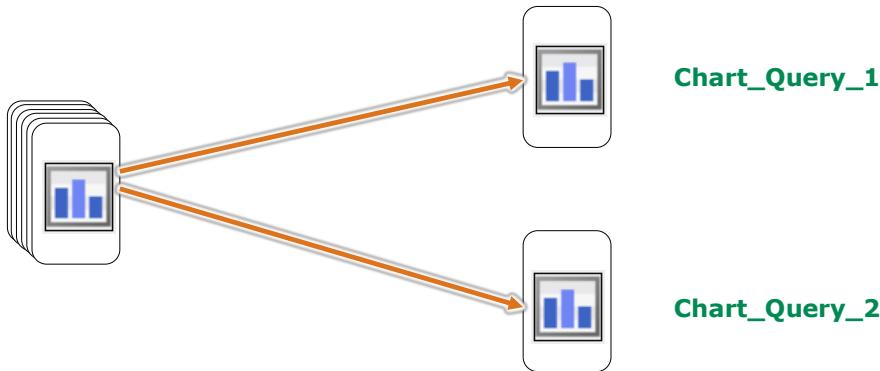
When choosing a control to display different sets of data, you can also use a Tab control. A Tab control can display different content for each tab.

A Deck or Data deck behaves in much the same way, but because Decks do not have a default method for the user to select a card by default, a wide variety of controls can be used to determine which card should be displayed in the report.

For basic report purposes, a Tab control may be sufficient, but for more complex reports and report layouts, a Deck can give more flexibility to the report author.

Traditional charts and static decks

- Deck controls the number of charts generated.
- Use a different query to generate each chart image.



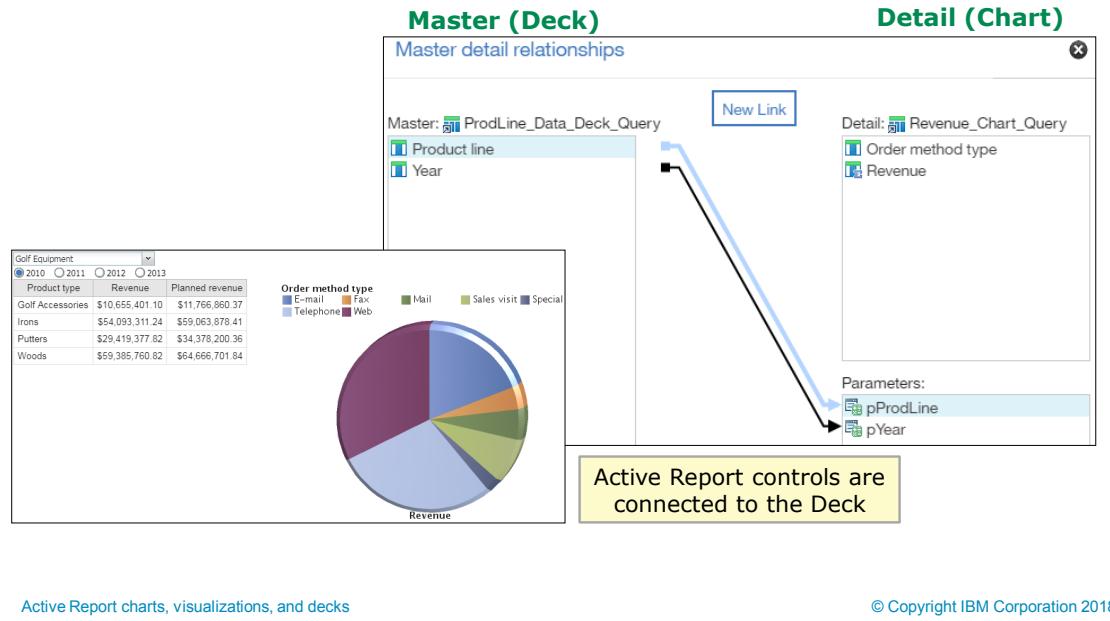
Traditional charts and Static decks

With a static Deck, the number of cards in the Deck is determined ahead of time by the report author. Any Deck card can contain a traditional chart, but in order for traditional charts on different cards to show differently-filtered data, each chart needs to be based on its own unique query with a filter applied to the query.

Charts based on the same query with the same data items will appear the same in the Active Report, regardless of the selected card.

Master detail relationships and decks

- Master detail relationships are used to section a data report object within a data deck.



Master detail relationships and decks

Master detail relationships can be used to define a relationship between a Deck and an object contained within the Deck. The purpose of the relationship is to define what data will be shown on the specific card within the Deck. The same technique can be used in a Tab control.

When working with traditional charts, the master detail relationship allows you to connect an Active Report variable to a parameter that is used in a filter on the chart for a particular card in the Deck.

Creating a Data deck with traditional charts

- Add the chart to the Data deck.
- Filter the chart based on a parameter.
 - e.g. [Year]=?pYear?
- Link the parent query to the chart using a master detail relationship.

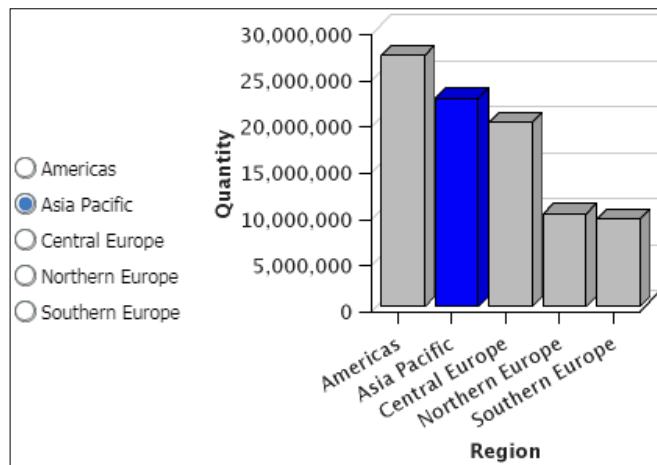
Creating a Data deck with traditional charts

To create a Data deck that contains different traditional charts for each selection, first add the chart to the Data deck, and then filter the chart using a parameter in the filter expression.

Once a parameter has been defined in the chart filter, define a master detail relationship between the parent query and the chart to link the Data deck's control data item to the parameter.

Demonstration 1

Create an Active Report using a Data deck



Demonstration 1: Create an Active Report using a Data deck

Demonstration 1: Create an Active Report using a Data deck

Purpose:

You have been asked to create a column chart showing quantity by region. The chart should be filtered based on a user's region selection. The report may or may not be used in an environment where the IBM Cognos servers are available, so an Active Report is needed. You will use a Data deck to create the various charts that can be displayed.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create an Active Report with a data radio button group control

1. Using the **GO sales (query)** package, create a new **Blank active report**.
2. In the work area, click **Add** , then click **Table** .
3. Click **OK** to accept the defaults (2 columns, 1 row).
4. Click the **Toolbox**  tab, then expand the **ACTIVE REPORT** section.
5. Drag a **Data radio button group**  to the left table cell:
 - Name: **Region_Data_RBG**
 - Query Name: **Region_Data_RBG_Query**
 - Click **OK**
6. From the **Data**  tab, drag Retailers > **Region** to the drop zone of **Region_Data_RBG**.
7. On the Application bar, click **Show properties**  to open the Properties pane.
8. In the Properties pane, under GENERAL, set **Orientation** to **Vertical**.

Task 2. Create a Data deck with a column Chart

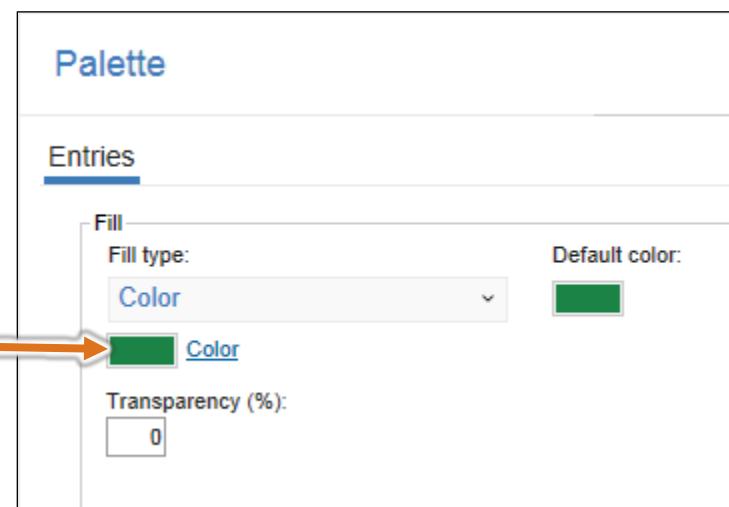
1. Click the **Toolbox**, then drag a **Data deck** to the right table cell.
 - Name: **Region_Data_Deck**
 - Query Name: **Region_Data_Deck_Query**
2. Click the **Data** tab, then drag Retailers > **Region** to the **Region_Data_Deck** object's **Deck Items** drop zone.
3. From the **Toolbox**, drag a **Visualization** to the **Region_Data_Deck** drop zone.
4. From the top right corner, click the visualization filter, then click **Charts**.

5. In the categories, click **Column**, then double-click **Clustered Column with 3D-Effects** .
6. In the Properties pane, change **Name** to **Region_Chart**.
7. Click the **Data** tab, then drag **Retailers > Region** to **Categories (x-axis)**.
8. Drag **Sales > Quantity** to **Default measure (y-axis)**.
9. In the Toolbar, click **More > Go to query**.
10. In the Properties pane, change **Name** to **Region_Chart_Query**.

Task 3. Add a conditional palette to the chart

You will highlight the bar of the selected region and gray out all the other region bars.

1. While the Chart is selected, in the Toolbar, click **More > Conditional Palette**.
2. Click **New Conditional Palette Entry** , and then click **New Advanced Condition**.
3. Click **New advanced condition** , and then in the **Expression Definition** pane, create the following expression:
 - `[Region_Data_Deck_Query].[Region] <> [Region_Chart_Query].[Region]`
4. Click **OK**.
5. Under **Palette Entry**, click **Edit Palette Entry** .
6. Under **Fill**, click the color box beside **Color**:



7. Click the **Basic colors** tab, click **Silver**, and then click **OK**.
8. Click **OK**.
9. Repeat steps 3 to 8, using the following:

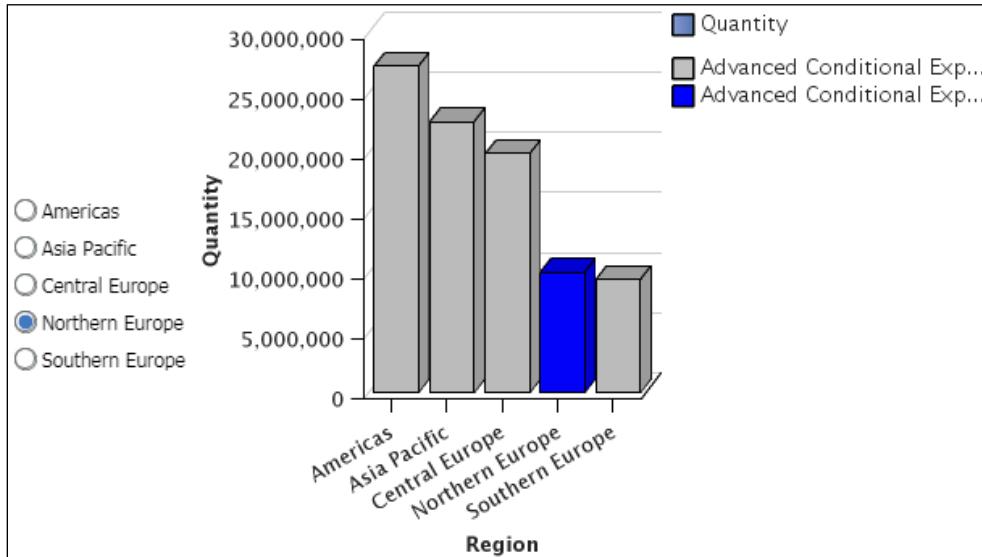
- Expression: **[Region_Data_Deck_Query].[Region] = [RegionChart_Query].[Region]**
 - Color: **Blue**
10. Click **OK**, and then click **OK** again to close the **Conditional palette** dialog.

Task 4. Create a connection between the Radio button group control and the Data deck

By creating this connection, you will be specifying which card to display when the user selects a region.

1. Click the **Region_Data_RBG** control, then in the Toolbar, click **Create a new connection**, and specify the following:
 - Region_Data_RBG (Region) > Select > Region_Data_Deck (Region)
 - New variable: **vRegion** (default value: **Americas**)
 - Click **Connect**.
2. Click the Chart, then in the Toolbar, click **Interactive Behavior**.
You want to specify that when a user clicks a column in the chart, the corresponding region will be selected in the radio button group.
3. Under **Behavior on select**, hover over **Set variable values**, and then click **Edit**.
4. Set **vRegion** to **Region** in the **Value** column, and then click **OK** twice.
5. On the Application bar, click **Run options** , then click **Run Active Report**.
6. Select **Northern Europe** from the **Region_Data_RBG** control.

The results appear as follows:



The report functions correctly. Hover the cursor over a grey bar and observe how it is highlighted.

7. Close the rendered report tab.

Task 5. Enhance the Active Report

The legend shows the conditional palette you created. This does not add value, so you will delete the legend.

1. Select the Chart.
2. In the Properties panel, under CHART ANNOTATIONS, double-click **Legend**.
3. Unselect **Show legend** and then click **OK**.

You will also add an animation effect to the report by specifying the slide direction for the card. To ensure the animation is rendered correctly, you will specify that the chart and the Deck are the same size.

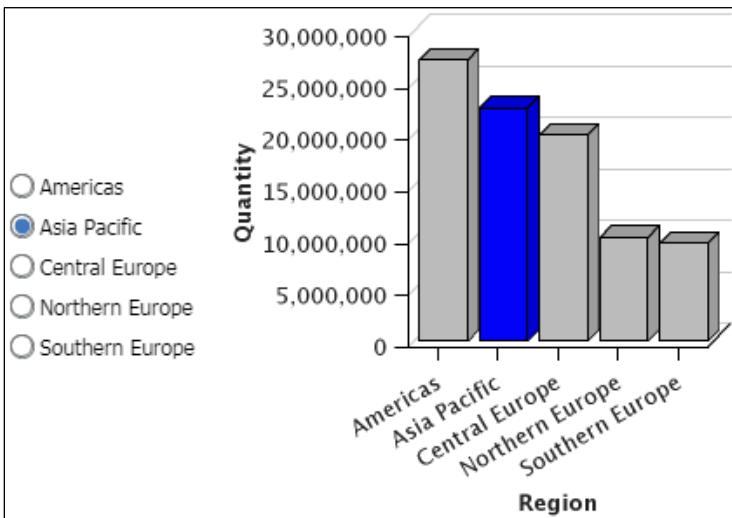
4. Select the **Region_Data_Deck** control, and then in the Properties pane, under General, Set **Slide Animation Direction** to **Auto Vertical**.
5. Select the left table cell, and then in the Properties pane, under POSITIONING, double-click **Size & overflow**, and set **Width** to **25px**.
6. Select the right table cell, and then in the Properties pane, under POSITIONING, double-click **Size & overflow**, and set **Height** and **Width** to **300px** each.

Hint: You may need to select the Data deck, and then in the Toolbar, click **Select Ancestor** to locate the table cell.

7. Select the chart, then in the Properties pane, under POSITIONING, double-click **Size & overflow** and set **Height** and **Width** to **300px** each.
8. On the Application bar, click **Run options** , then click **Run Active Report**.

9. Select **Asia Pacific** from the **Region_Data_RBG** control.

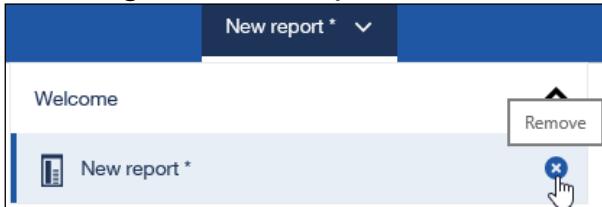
The results appear as follows:



The chart reflects the selection you made using the slide animation direction you specified.

10. Close the rendered reports' tabs.

11. In the Application bar, click the report dropdown menu, then click **Remove** to the right of New report to close it. Then click **OK** to confirm your action.



12. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You created an Active Report using a Data deck. You observed the hover property and defined a selection behavior on the chart.

Examine cards in a Data deck

- Cards are driven by the data values in the Data deck.
- The card displayed depends on the data item selection.

<input checked="" type="checkbox"/>	Camping Equipment	<input checked="" type="checkbox"/>	2010
<input checked="" type="checkbox"/>	Golf Equipment	<input checked="" type="checkbox"/>	2011
<input checked="" type="checkbox"/>	Mountaineering Equipment	<input checked="" type="checkbox"/>	2012
<input checked="" type="checkbox"/>	Outdoor Equipment	<input checked="" type="checkbox"/>	2013
<input checked="" type="checkbox"/>	Personal Accessories		

Product lines (5) * Years (4) = 20 cards

Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Examine cards in a Data deck

The number of cards in a Data deck is determined by the number of data values in the Deck's data items.

Care must be taken when designing Data decks, because the number of cards can increase exponentially as more potential data combinations are added.

Choose proper controls for use in Data decks

- Use single select controls.
- Impact of multi-select controls (e.g. Data check box group):
 - Number of combinations = 2^n
 - Complex logic is required to display the card

Values on multi-select control (n)	Number of cards generated on a deck (2^n)
1	2
2	4
3	8
4	16
5	32
6	64

Choose proper controls for use in Data decks

Single select controls, such as Drop-down lists, Radio button groups, and Button bars will result in a limited number of cards being generated on a Deck that is manageable to display.

With multi-select controls, such as a Check box group, the different combinations of data can result in a deck of 2 to the power of n of cards, where n is the number of values in the data item used in the data control.

The logic required to display the different combinations of card data is very difficult to create with the AND, OR, NOT operators.

Reduce report size with static decks

- Limit selections to a subset of available data.
- Unnecessary charts are not generated / stored.



Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Reduce report size with Static decks

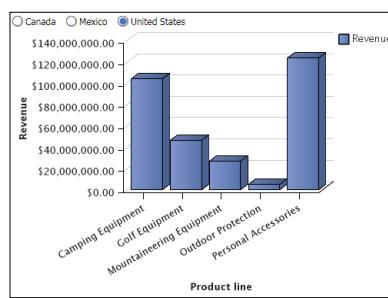
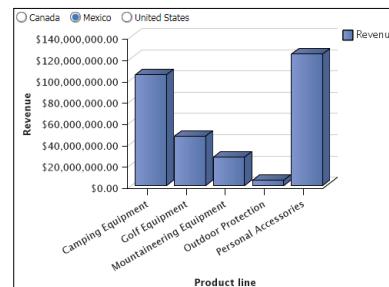
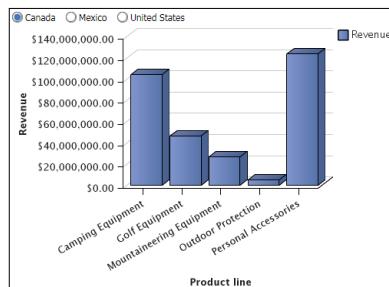
When dealing with a large number of data items, it may be worthwhile to use a Static deck instead of a Data deck, if there is a subset of the data that will not likely be required by the report consumers.

By controlling the number of cards using a static Deck, the report author has more control over how quickly the size of an Active Report grows, as only those data items needed by the users will generate cards for the report.

For example, a report intended only for consumers in a particular geographical region may not need to include all global regions in the Deck. A static Deck using only the regions required by the report consumers can dramatically reduce the overall size of the generated MHT file.

Demonstration 2

Create a Static deck to conserve report size



Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Demonstration 2: Create a Static deck to conserve report size

Demonstration 2:

Create a Static deck to conserve report size

Purpose:

You will create an Active Report for users located in Canada, Mexico, and the United States. You will include charts displaying data for each of these countries, but avoid bringing in other countries' data to optimize the size of the report.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

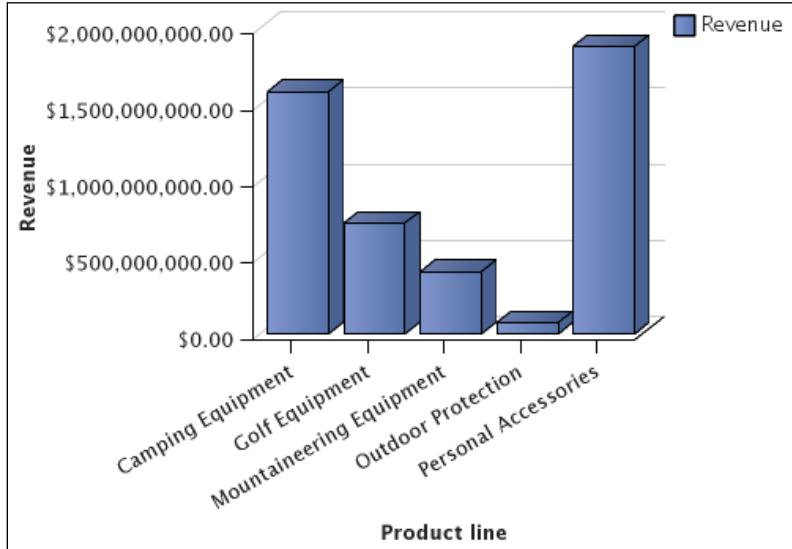
Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create a chart to show revenue by product line

1. From the **Welcome** page, navigate to **Team content** > **Samples** > **Models**.
2. Right-click the **GO Sales (query)** package, then click **Create report** .
3. In the Templates and themes dialog, double-click the **Blank active report** template to use it.
4. In the work area, click **Add** , then click **Table** .
5. In the **Insert table** dialog, set **Number of columns** to 1, **Number of rows** to 2, and then click **OK**.
6. In the bottom table cell, click **Add** , then click **Visualization** .
7. From the top right corner, click the visualization filter, then click **Charts**.
8. In the categories, click **Column**, then double-click **Clustered Column with 3D-Effects** .
9. On the Application bar, click **Show properties** to open the Properties pane.
10. In the Properties pane, under MISCELLANEOUS, set **Name** to **Revenue_Chart**.
11. In the side bar, click **Queries** , then right-click **Query1**, then click **Rename query**, then enter **Revenue_Chart_Query**.
12. Click the **Data** tab, expand **Sales (query) > Products**, and then drag **Product line to Categories (x-axis)**.
13. Expand **Sales**, and then drag **Revenue** to **Default measure (y-axis)**.
14. On the Application bar, click **Run options** , then click **Run Active Report**.

The results appear as follows:



The desired data is displayed, but we want to have separate charts to show data for Canada, Mexico, and the United States.

15. Close the browser tab.

Task 2. Add a Radio button group and Deck to select the country

1. From the **Toolbox** tab, the ACTIVE REPORT section, drag a **Radio button group** to the top table cell.
 2. In the Properties pane, under MISCELLANEOUS section, set **Name** to **Country_RBG**.
 3. It the Toolbar, click **Radio Buttons Definition** . And create these labels:
 - Canada
 - Mexico
 - United States
 4. Click **OK**.
- We are unable to filter a chart object, so we will need a Deck to hold a chart for each country.
5. From the Toolbox, drag a **Deck** to the right of the **Revenue_Chart** control.
 6. In the Properties pane, under MISCELLANEOUS, specify the **Name** property as **Country_Deck**.
 7. Click **Deck Cards Definition**.
 8. It the Toolbar, create three cards as follows:
 - Canada

- Mexico
 - United States
9. Click **OK**.
 10. Click the **Country_RBG**, then in the Toolbar, click **Create a new connection**.
 11. Under **Active Report Variable**, set the **Name** to **vCountry**.
 12. Click **Set Default Value(s)**, and set the default value to **Canada**.
 13. Click **OK**.
 14. Click **Connect**.

Task 3. Add the chart to the Deck, and create copies for each country

1. Drag **Revenue_Chart** into the drop zone of **Country_Deck**.
2. Select **Revenue_Chart**, then in the Properties pane, under MISCELLANEOUS, change **Name** to **Canada_Chart**.
Next, you will put a copy of the chart on each card in the Deck.
3. On the Toolbar, click **More** , and then click **Copy**.
4. From the drop-down list of **Country_Deck**, select **Mexico**.
5. Click the **Country_Deck** control's drop zone to select it, then press **Ctrl+V**.
6. In the Properties pane, under miscellaneoUS section, set **Name** to **Mexico_Chart**.
7. Repeat steps **4** through **6** to paste the Chart into the **United States** card.
 - Name: **UnitedStates_Chart**
8. On the Application bar, click **Run options** , then click **Run Active Report**.
9. Select each of the countries in the **Country_RBG** control.
Changing the selected country does not affect the appearance of the chart. This is because the charts are all based on the same query, and you are unable to filter charts in Active Reports.
You will define queries to filter out data that does not relate to the selected country.

9. Close the rendered report tab.

Task 4. Filter the query to collect data for only one country

1. On the side bar, click **Queries** , and then click **Revenue_Chart_Query**.
2. On the side bar, click **Data**.
3. Drag Retailers > **Retailer country** into the **Detail Filters** pane.
4. In the **Expression Definition** pane, edit the expression to be:

- [Sales (query)].[Retailers].[Retailer country]='Canada'
5. Click **Validate**, and then click **OK**.
 6. On the side panel, click **Queries**, then right-click **Revenue_Chart_Query**.
 7. Click **Rename query**.
 8. Type **Canada_Query**, and then click **OK**.

Task 5. Create separate queries for each country

1. In the Queries pane, click the **Queries**  folder.
2. Right-click **Canada_Query**, and select **Copy**.
3. Right-click the background of the Query explorer, then click **Paste**.
4. Right-click **Canada_Query1** and click **Rename query**.
5. Type **Mexico_Query**, and then click **OK**.
6. Repeat steps 2 to 5 to create a query named **UnitedStates_Query**.
7. Double-click **Mexico_Query**.
8. Double-click the expression in the **Detail Filters** pane.
9. In the **Expression Definition** pane, replace **Canada** with **Mexico**. The expression should now be:
 - [Sales (query)].[Retailers].[Retailer country]='Mexico'
10. Click **Validate**, and then click **OK**.
11. In the **Queries** tab, click **UnitedStates_Query**.
12. Double-click the expression in the **Detail Filters** pane.
13. In the **Expression Definition** pane, replace **Canada** with **United States**. The expression should now be:
 - [Sales (query)].[Retailers].[Retailer country]='United States'
14. Click **Validate**, and then click **OK**.

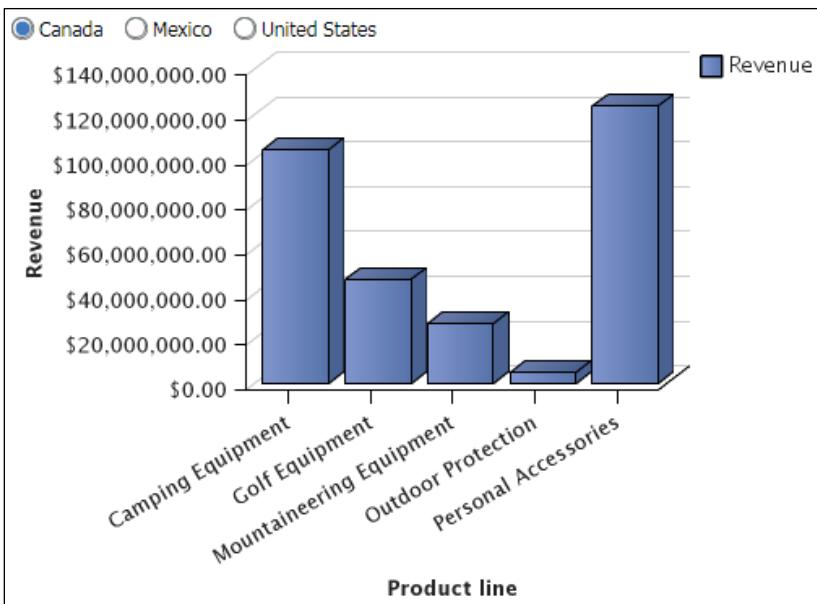
Task 6. Assign the appropriate query to each of the charts

1. In the side bar, click **Pages** , then under **Report pages**, click **Page1**.
2. From the **Country_Deck** control's drop-down list, select **Canada**.
3. Click the **Canada_Chart** control, and then in the Properties pane, under DATA, verify that the **Query** property is set to **Canada_Query**.
4. From the **Country_Deck** control's drop-down list, select **Mexico**.
5. Click the **Mexico_Chart** control, and then in the Properties pane, under DATA, set the **Query** property to **Mexico_Query**.
6. From the **Country_Deck** control's drop-down list, select **United States**.

7. Click the **UnitedStates_Chart** control, and then in the Properties pane, under DATA, set the **Query** property to **UnitedStates_Query**.

Task 7. Run the report and verify that the country-specific charts display correctly

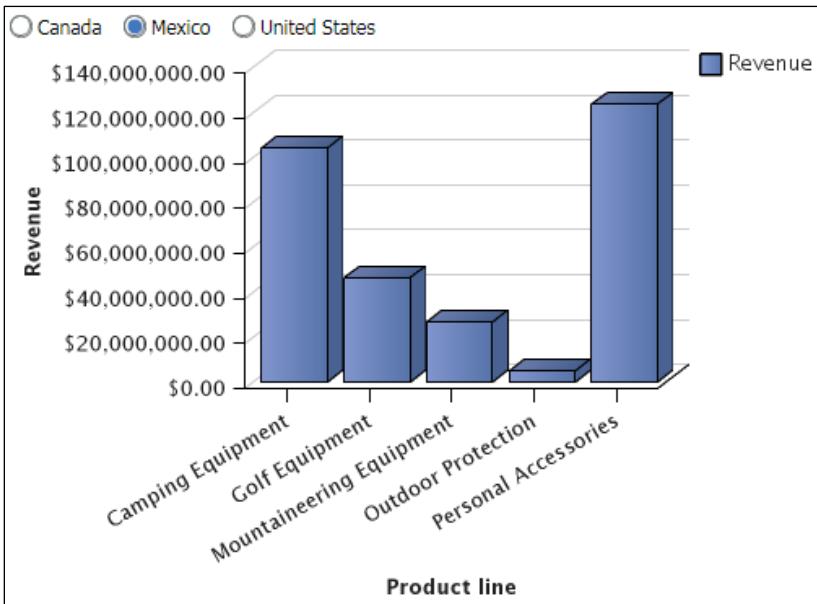
1. On the Application bar, click **Run options** , then click **Run Active Report**. The results appear as follows:



The chart displays data for revenue from Canada.

2. Select **Mexico**.

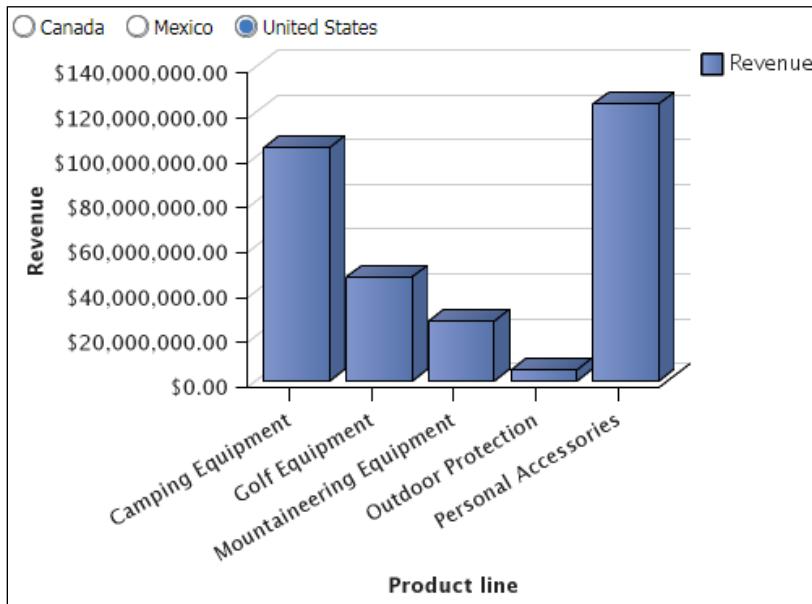
The results appear as follows:



The chart displays data for revenue from Mexico.

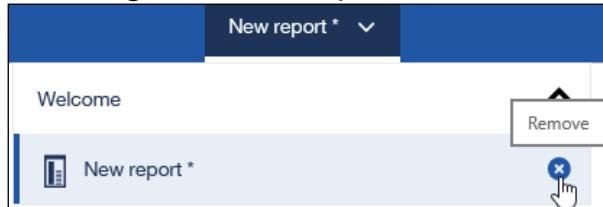
3. Select **United States**.

The results appear as follows:



The chart displays data for revenue from the United States.

4. Close the rendered report tab.
5. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



6. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You used a Deck with separate queries to create an optimized report to display chart data for a subset of the available countries in your database without storing data that is not related to the available country selections.

Demonstration 3

Optimize a Data deck report for size



The screenshot shows a Data deck report interface. On the left, there's a sidebar with dropdown menus for 'Australia' and 'Global Product Info'. Below that is a table titled '1 for 1 Sports shop' with columns for 'Product line' and 'Revenue'. The table lists five categories: Camping Equipment (\$1,331,569.87), Mountaineering Equipment (\$792,886.58), Outdoor Protection (\$145,302.07), Personal Accessories (\$4,162,491.80), and an overall total of \$6,432,250.32. To the right of this is a larger table titled 'Australia Revenue Data by Product Line' with columns for 'Revenue' and years '2011', '2012', and '2013'. This table provides more granular data for each category.

Revenue	2011	2012	2013
Camping Equipment	\$9,752,591.01	\$19,175,957.20	\$13,007,383.98
Golf Equipment	\$4,094,643.54	\$8,482,438.67	\$6,502,474.22
Mountaineering Equipment	\$2,691,279.15	\$5,861,253.12	\$5,380,587.79
Outdoor Protection	\$600,956.77	\$367,636.38	\$171,750.41
Personal Accessories	\$8,467,573.03	\$13,912,451.77	\$10,830,992.10

Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Demonstration 3: Optimize a Data deck report for size

Demonstration 3: Optimize a Data deck report for size

Purpose:

You have been asked to create a dashboard for mobile users containing both country-specific and generic revenue information. You will examine the impact on the size based on how you arrange controls and content in your report.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create a Data deck and Data drop-down list to display country-specific information

1. Using the **GO sales (query)** package, create a new **Blank active report**.
2. Click the **Toolbox**, then from under the ACTIVE REPORT section, drag a **Data drop-down list** to the page body.
 - Name: **Country_Data_DDL**
 - Query Name: **Country_Data_DDL_Query**
3. From the **Data**  tab, drag Retailers > **Retailer country** onto **Country_Data_DDL**.
4. From the **Toolbox** tab, drag a **Data deck** under the Data drop-down list:
 - Name: **Country_Data_Deck**
 - Query Name: **Country_Data_Deck_Query**
5. From the **Data**  tab, drag Retailers > **Retailer country** to the Deck Items drop zone of **Country_Data_Deck**.

Task 2. Create a connection between the Data drop-down list and the Data deck

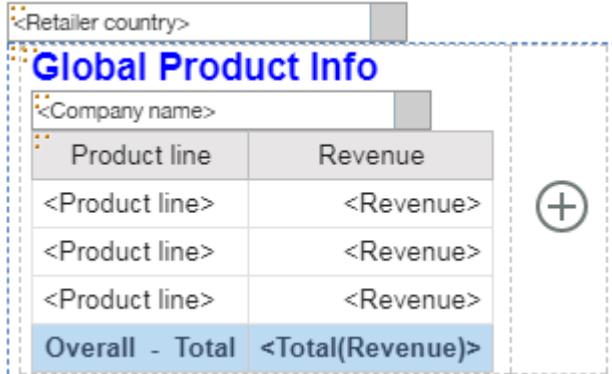
1. Click the **Country_Data_DDL**, then in the Toolbar, click **Create a new connection**, and specify the following:
 - Country_Data_DDL (Retailer country) > Select > Country_Data_Deck (Retailer country)
 - New variable: **vCountry**
 - Default value: **Australia**

2. Click Connect.

Task 3. Add global information to the Deck card

1. From the **Toolbox** tab, drag a **Table** to the **Country_Data_Deck** drop zone.
2. Accept the default settings by clicking **OK**.
3. From the **Toolbox** tab, drag a **Text item** to the left table cell.
4. Type **Global Product Info**, and then click **OK**.
5. On the Toolbar, click **Font A**.
6. Set **Size** to **14pt**, **Weight** to **Bold**, **Foreground Color** to **Blue**, then click **OK**.
7. From the **Toolbox** tab, drag a **Data drop-down list** after the Text item:
 - Name: **Company_Data_DDL**
 - Query Name: **Company_Data_DDL_Query**
8. From the **Data**  tab, drag Retailers > **Company name** onto **Company_Data_DDL**.
9. From the **Toolbox** panel, drag a **List** after the **Company_Data_DDL** control:
 - Name: **Company_Revenue_List**
 - Query Name: **Company_Revenue_List_Query**
10. From the **Data** tab, drag the following query items to **Company_Revenue_List**:
 - **Products > Product line**
 - **Sales > Revenue**
11. Click **<Revenue>**, then from the Toolbar, click **Summarize** > Summarize in Output > **Total**.

The result appears as follows:



The screenshot shows a 'Deck card' interface. At the top, there is a header cell containing the text '<Retailer country>'. Below it is a table with a single row labeled 'Global Product Info'. This row contains a cell with the text '<Company name>' and another cell with a plus sign (+) inside a circle, indicating a dropdown menu. The main body of the card contains a table with the following data:

Product line	Revenue
<Product line>	<Revenue>
<Product line>	<Revenue>
<Product line>	<Revenue>
Overall - Total	<Total(Revenue)>

Task 4. Create a connection between the Data Drop-Down

List control and the list

1. Click the **Company_Data_DDL** control, then from the Toolbar, click **Create a new connection**, and specify the following:
 - Company_Data_DDL (Company name) > **Filter** > **Company_Revenue_List** (Copy Data Item from source)
 - New variable: **vCompany**
 - Default value: **1 for 1 Sports shop**
2. Click **Connect**.

Task 5. Add country-specific information to the Deck card

3. From the **Toolbox** tab, under the ACTIVE REPORT section, drag a **Variable text item** to the right table cell.
4. Select **vCountry**, and then click **OK**.
5. From the **Toolbox** tab, drag a **Text Item** to the right **<%vCountry%>**.
6. Type a space, type **Revenue Data by Product Line**, and then click **OK**.
7. Click the **Global Product Info** text item, and then on the Toolbar, click **Pick up style** .
8. Select the **<%vCountry%>** variable text item, then on the toolbar, click **Apply style** .
9. Select the **Revenue Data by Product Line** text item, then on the toolbar, click **Apply Style**.
10. From the **Toolbox**, drag a **Crosstab** to the right table cell under the Text item:
 - Name: **Country_Revenue_Crosstab**
 - Query Name: **Country_Rev_Crosstab_Query**
11. From the **Data**  tab, drag the following query items to the Crosstab:
 - Rows: Products > **Product line**
 - Columns: Time > **Year**
 - Measures: Sales > **Revenue**

Task 6. Filter the crosstab by a country parameter

1. Select the **Country_Revenue_Crosstab** control.
2. On the Toolbar, click **Filters** > **Edit Filters**.
3. Click **Add**, then click **Advanced**, and then click **OK**.

4. Under **Available Components**, expand **Sales (query)** and **Retailers**, and then drag **Retailer country** to the **Expression Definition** pane.
5. Edit the expression to the following:
 - **[Sales (query)].[Retailers].[Retailer country]=?pCountry?**
6. Click **OK**.
7. Click **OK**.

Task 7. Create a Master detail relationship between the country variable and the crosstab

1. Select the **Country_Revenue_Crosstab**.
2. On the Toolbar, click **More** > **Master Detail Relationships**.
3. Click **New Link**.
4. Click **pCountry** under **Parameters** to define the link, then click **OK**.

Task 8. Save the report, run it, and then save an MHT file to check the file size

1. On the Application bar, click **Save**.
 2. Save the report as **Dashboard_1** in **My content**.
 3. On the Application bar, click **Run options** , then click **Run Active Report**. Note that it may take a while for the report to display.
- The results appear as follows:

Global Product Info		Australia Revenue Data by Product Line			
Product line	Revenue	Revenue	2011	2012	2013
Camping Equipment	\$1,331,569.87	Camping Equipment	\$9,752,591.01	\$19,175,957.20	\$13,007,383.98
Mountaineering Equipment	\$792,886.58	Golf Equipment	\$4,094,643.54	\$8,482,438.67	\$6,502,474.22
Outdoor Protection	\$145,302.07	Mountaineering Equipment	\$2,691,279.15	\$5,861,253.12	\$5,380,587.79
Personal Accessories	\$4,162,491.80	Outdoor Protection	\$600,956.77	\$367,636.38	\$171,750.41
Overall - Summary	\$4,686,775,768.85	Personal Accessories	\$8,467,573.03	\$13,912,451.77	\$10,830,992.10

4. Close the browser tab.
5. On the Application bar, click **Run options** , then click **Run Active Report**.
6. From the Application bar click **Run options**, then click **Download Active Report**.
7. At the prompt to save **Dashboard_1.mht**, click the down arrow beside **Save**, and click **Save as**.
8. Save the report as **C:\Edcognos\B6098\Dashboard_1.mht**.
9. Click **Open folder**, and then take note of the size of **Dashboard_1.mht**.

The size of the file is a little large for a small dataset, and might not scale well.

You will now change the report to reduce the size of the MHT file.

10. Close **Windows Explorer**, and then close the browser tab.

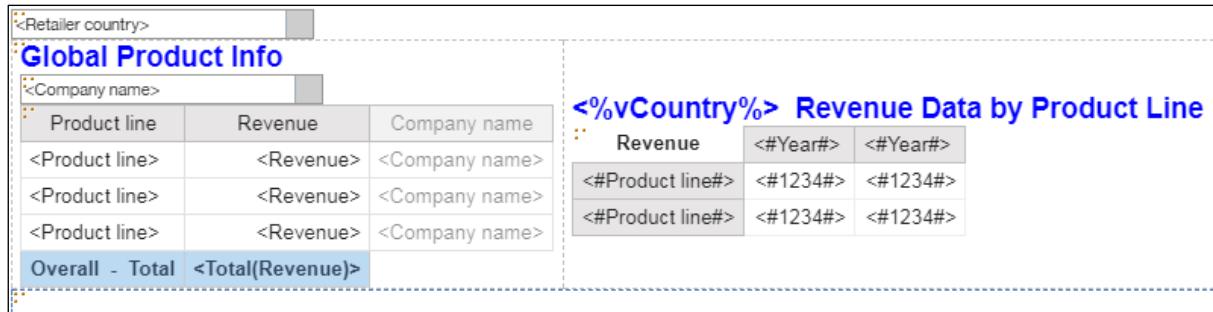
Task 9. Save a new copy of the report, and then move controls that are not related to country out of the Data deck

Because many of the controls in this report do not change as the country changes, we can move these controls out of the Data deck to avoid duplicating these controls for each card in the Data deck.

1. On the Application bar, click the down arrow beside **Save**  , and then click **Save as**.
2. Save the report as **Dashboard_2** in **My content**.
3. Select the Table, then drag it after the **Country_Data_DDL**, before the **Country_Data_Deck**.

Hint: To select the Table, you may need to click the **Company_Revenue_List**, then on the Toolbar, click **Select ancestor**  , then click **Table**.

The result appears as follows:



The screenshot shows a report layout with two main sections. On the left, there is a table titled "Global Product Info" with columns "Product line", "Revenue", and "Company name". The table has four rows of data and an "Overall - Total" row at the bottom. On the right, there is a table titled "<%vCountry%> Revenue Data by Product Line" with columns "Revenue", "#Year#", and "#Year#". This table also has four rows of data and an "Overall - Total" row at the bottom. Both tables have a dashed border, indicating they are separate components.

4. In the Properties pane, under **BOX**, double-click **Margin**.
5. For the **Top margin**, type **10**, then click **OK**.
6. Click **<%vCountry%>**, Ctrl+click the **Revenue Data by Product Line** Text item beside it, and Ctrl+click the **Country_Revenue_Crosstab**.
7. Drag the selection onto the **Country_Data_Deck**.
8. Click **Select Country_Data_Deck**  , then drag it to the right cell of the Table.

The result appears as follows:

The screenshot shows a report titled "Global Product Info" with a table of revenue data. The table has columns for "Product line", "Revenue", and "Company name". It includes rows for "Overall - Total" and "Total(Revenue)". To the right of the main table is a secondary table titled "<%vCountry%> Revenue Data by Product Line", which also displays revenue data for different product lines and years.

Task 10. Save the report, run it, and then save an MHT file to check the file size

1. On the Application bar, click **Save**.
2. On the Application bar, click **Run options** , then click **Run Active Report**.

The results appear as follows:

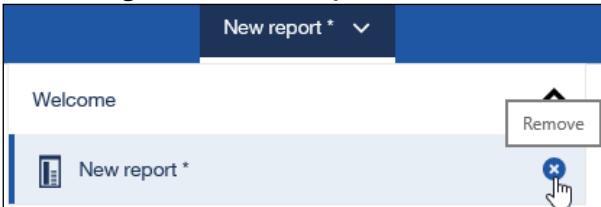
The screenshot shows the "Global Product Info" report for "Australia". It includes a dropdown menu for "Australia" and a table of revenue data for "1 for 1 Sports shop". To the right is a separate table titled "Australia Revenue Data by Product Line", which provides detailed revenue data for various product categories over three years (2011, 2012, 2013).

Product line	Revenue	2011	2012	2013
Camping Equipment	\$1,331,569.87	\$9,752,591.01	\$19,175,957.20	\$13,007,383.98
Mountaineering Equipment	\$792,886.58	\$4,094,643.54	\$8,482,438.67	\$6,502,474.22
Outdoor Protection	\$145,302.07	\$2,691,279.15	\$5,861,253.12	\$5,380,587.79
Personal Accessories	\$4,162,491.80	\$600,956.77	\$367,636.38	\$171,750.41
Overall - Total	6,432,250.32	\$8,467,573.03	\$13,912,451.77	\$10,830,992.10

The results appear the same as when the report was run previously. Next, you will generate an MHT file and compare the new file's size to the old one.

3. Close the browser tab.
 4. On the Application bar, click **Run options**, then click **Download Active Report**.
 5. At the prompt to save **Dashboard_2.mht**, click the down arrow beside **Save**, and click **Save as**.
 6. Save the report as **C:\Edcognos\B6098\Dashboard_2.mht**.
 7. Click **Open folder**, and then compare the size of **Dashboard_2.mht** to the size of **Dashboard_1.mht**.
- The new MHT file is significantly smaller, which will make it faster to download and display on a mobile device.
8. Close **Windows Explorer**.
 9. Close the rendered report tab.

10. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



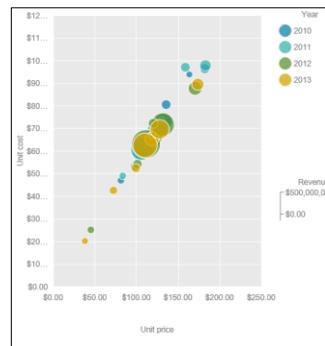
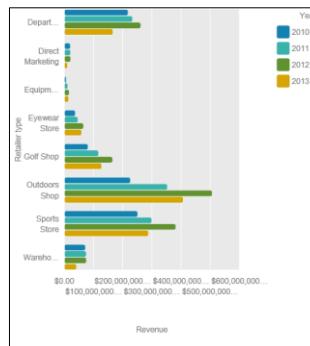
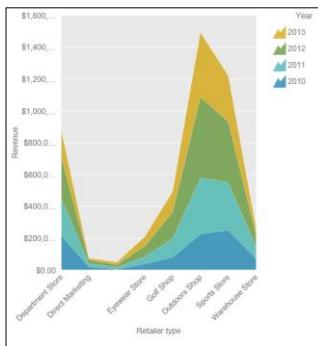
11. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You created a basic dashboard for mobile users, and explored changes that can optimize the size of the resulting MHT file.

Rapidly Adaptive Visualization Engine (RAVE)

- Interactive
- Animated
- Variety of chart types
- Customizable
- Import additional visualizations



Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Rapidly Adaptive Visualization Engine (RAVE)

To get more flexibility when designing Active Reports, report authors can incorporate RAVE visualizations into their reports to complement or enhance views of data already provided by standard chart controls.

MHT files generated for Active Reports contain an engine to generate RAVE visualizations, so visualizations in Active Reports can be rendered on the client side dynamically based on user selections, or they can be rendered on the server as a static image when the MHT file is first created.

RAVE visualizations can also be animated to dynamically highlight aspects of the data, in order to provide a more engaging experience to report consumers.

Traditional charts vs. RAVE visualizations

- Traditional charts:
 - stored as images (static)
 - deck or query used to filter content
- RAVE visualizations:
 - extensible
 - rendered on client or server (dynamic or static)
 - can be filtered
 - graphical animations

Traditional charts vs. RAVE visualizations

Both traditional charts and RAVE visualizations can be used in Active Reports, and each has its own characteristics and advantages.

Traditional charts are generated at the time the Active Report is first generated, so subsequent runs of the MHT will be faster.

By using a Deck, or assigning unique queries to different charts, content displayed to the user can be customized, even though traditional filters can be used.

RAVE visualizations can be customized and imported to the IBM Cognos Analytics server, allowing access to a dynamically growing library of visualizations.

RAVE visualizations can be generated on the client side in the reprot, allowing content to be dynamically updated for more interactivity. Alternately, visualizations can be rendered on the server side, creating a static image, similar to traditional charts.

Content in a RAVE visualization can be dynamically filtered at the MHT file's run time, and can also include graphical animations to enhance the report user's experience.

Connections and RAVE visualizations

- Active Report connections support RAVE visualization controls:
 - control -> filter/select -> visualization
 - visualization -> filter/select -> control
 - visualization -> filter/select -> visualization

Connections and RAVE visualizations

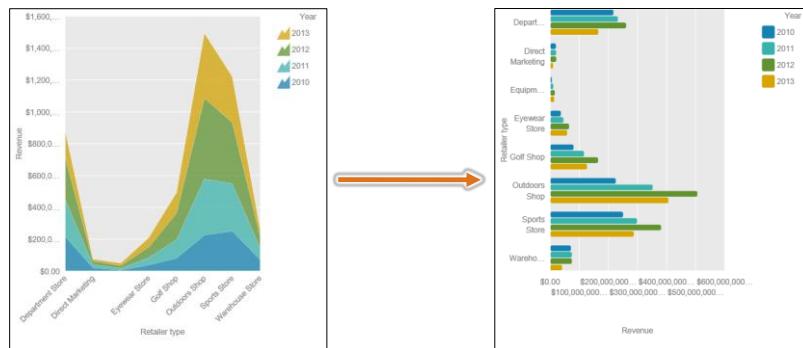
Because RAVE visualizations are also Active Report controls, their behavior can be controlled by creating a connection from an Active Report control to filter or select the Visualization control.

It is also possible for a connection to be set up from a RAVE visualization to another control, or even a second RAVE visualization.

For example, a report could be created where clicking on a bar in a RAVE bar chart visualization highlights the rows in a List object that contribute to the bar chart's data.

Converting visualizations in a report

- Convert between different types of visualizations.
- Data from the old visualization is mapped to the new visualization.



Converting visualizations in a report

As a report author works with visualizations in a report, there are times when a different visualization might provide better insights than the one currently in the report, using the same or similar underlying data.

By clicking a visualization in the report and using the On-demand toolbar, the report author can choose to convert the visualization type to another visualization in the library.

When IBM Cognos Analytics converts the visualization, it also attempts to map the data from the original visualization to data slots in the new visualization, so that the author does not need to add the same data again. The report author can then move data around or add more data to fine-tune the appearance of the new visualization.

Rendering RAVE visualizations

- Client-side
 - visualization rendered at run-time
 - full interactivity
 - slightly smaller file size
- Server-side
 - visualization rendered at file generation time
 - static images
 - slightly better performance

Rendering RAVE visualizations

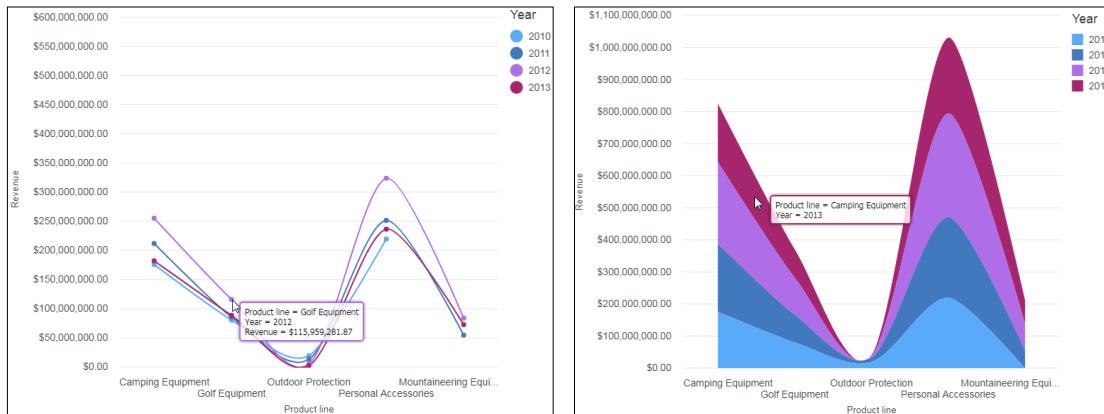
There are two options available for rendering visualizations when using RAVE visualizations in an Active Report, depending on the requirements of the report consumer.

If a RAVE visualization is set to use client-side rendering, then code in the report will dynamically render the visualization in the browser at run-time. This allows for full interactivity with the visualization, and can result in a slightly smaller file size.

If a RAVE visualization is set to use server-side rendering, then the visualization is rendered at the server when the report is first created, using static images, like traditional charts. Since code is not executed in this case, there may be a slight increase in performance.

Demonstration 4

Create an Active Report using RAVE visualizations



Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Demonstration 4: Create an Active Report using RAVE visualizations

Demonstration 4:

Create an Active Report using RAVE visualizations

Purpose:

You want to create an interactive Active Report using animated charts from RAVE visualizations. You will create a report displaying data using different visualizations to better engage the consumer.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create an Active Report with a List

1. Using the **GO sales (query)** package, create a new **Blank active report**.
2. From the **Toolbox**, expand **ACTIVE REPORT**, then drag a **Data toggle button bar** object to the top table cell.
 - Name: **Region_Data_TBB**
 - Query Name: **Region_Data_TBB_Query**
3. From the **Data** tab, drag Retailers > Region to the **Region_Data_TBB** drop zone.
4. On the Application bar, click **Show properties**  to open the Properties pane.
5. In the Properties pane, under GENERAL, double-click **Button gap**, enter 5 and click **OK**.
6. From the **Toolbox** tab, drag a **Table** under the Data button bar, and click **OK**.
7. In the left Table cell, click **Add** , then click **List** 
 - Name: **Revenue_List**
 - Query Name: **Revenue_List_Query**
8. From the **Data** tab, drag the following to the List:
 - **Sales (query) > Retailers: Region, Retailer country**
 - **Sales: Revenue and Quantity**.
9. Click **<Revenue>**, and Ctrl-click **<Quantity>**, then from the Toolbar, click **Summarize > Default summary > Total**.

The result appears as follows:

<Region>	<Region>	<Region>	
Region	Retailer country	Revenue	Quantity
<Region>	<Retailer country>	<Revenue>	<Quantity>
<Region>	<Retailer country>	<Revenue>	<Quantity>
<Region>	<Retailer country>	<Revenue>	<Quantity>
Overall - Total		<Total(Revenue)>	<Total(Quantity)>



Task 2. Connect the Data toggle bar to the List

- Click the **Region_Data_TBB**, then from the Toolbar, click **Add a new connection**, and specify the following:
 - Region_Data_TBB (Region) > Select > Revenue_List (Region)
 - Create a new variable: **vRegion**
 - Click **Connect**.

The vRegion variable will be used later to filter the data displayed in the RAVE visualization, but here you want to continue to display all data in the List, but highlight the data in the List that is displayed in the RAVE visualizations.

- From the Application bar, click **Run options** , then click **Run Active Report**.
- Select **Americas** from the **Region_Data_TBB** control.

The result appears as follows:

Americas	Asia Pacific	Central Europe	Northern Europe	Southern Europe
Region	Retailer country	Revenue	Quantity	
Asia Pacific	Australia	\$109,299,969.14	2,000,781	
Southern Europe	Austria	\$143,709,405.32	2,742,824	
Central Europe	Belgium	\$113,031,617.63	2,124,791	
Americas	Brazil	\$138,276,679.05	2,591,989	
Americas	Canada	\$306,159,341.39	5,722,733	
Asia Pacific	China	\$317,244,910.76	6,110,945	
Northern Europe	Denmark	\$62,013,023.51	1,301,136	
Northern Europe	Finland	\$188,575,310.13	3,603,492	
Central Europe	France	\$286,569,502.15	5,529,613	
Central Europe	Germany	\$262,313,028.79	5,084,611	
Southern Europe	Italy	\$186,648,098.68	3,545,695	
Asia Pacific	Japan	\$357,446,606.95	6,787,127	
Asia Pacific	Korea	\$200,725,307.09	3,902,092	
Americas	Mexico	\$167,187,013.22	3,175,752	
Northern Europe	Netherlands	\$184,321,676.63	3,448,760	
Asia Pacific	Singapore	\$197,622,385.04	3,788,595	
Southern Europe	Spain	\$165,066,464.22	3,171,715	
Northern Europe	Sweden	\$95,411,435.54	1,681,811	
Central Europe	Switzerland	\$100,731,875.51	1,822,191	
Central Europe	United Kingdom	\$277,509,546.56	5,378,361	
Americas	United States	\$826,912,571.54	15,722,077	
Overall - Total		\$4,686,775,768.85	89,237,091	

- Close the rendered reports' tabs.

Task 3. Add a RAVE Visualization to the report.

1. In the right table cell, click **Add** , then click **Visualization** .
2. From the top right corner, click the visualization filter, then click **New visualizations**.
3. In the categories on the left, click **Line**, then double-click **Smooth line**.
 - Visualization name: **Revenue_Visualization**
 - Data container name: **Revenue_Visualization_Data**
 - Query Name: **Revenue_Visualization_Query**

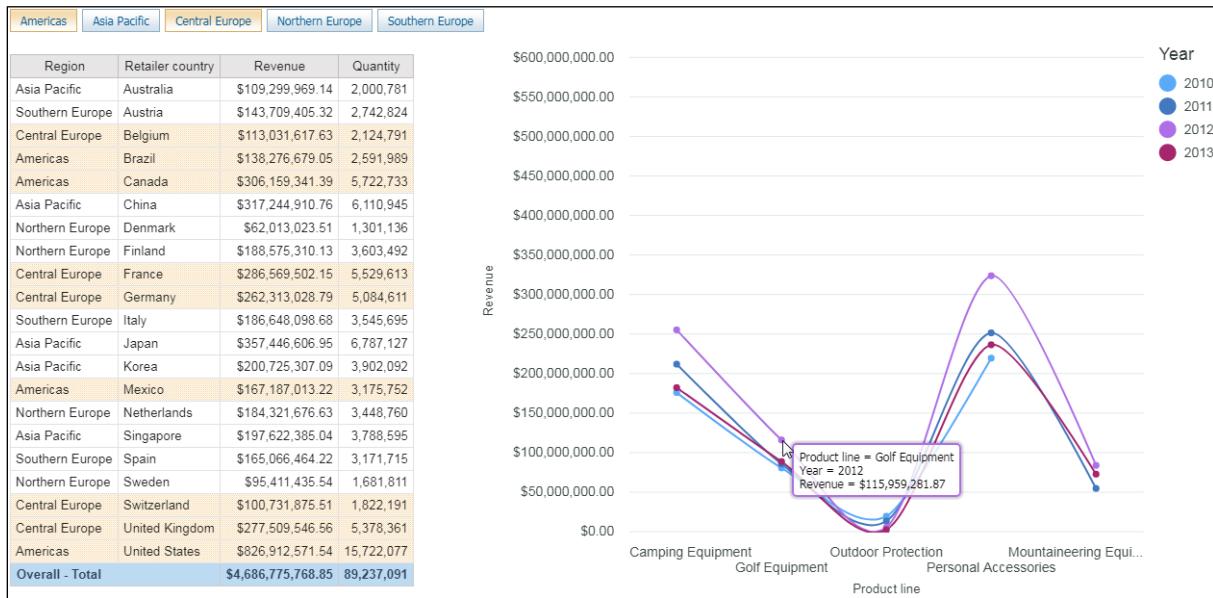
Note that we do not need to put the visualization inside a Deck, because the RAVE visualization engine dynamically updates the visualization content, rather than storing a static copy of all possible visuals.

4. From the **Data** tab, drag the following items to the specified drop zones:
 - Values > Value: Sales > **Revenue**
 - Categories > X axis: Products > **Product line**
 - Color: Time > **Year**
5. Click the Visualization, then in the Properties pane, under COMMON, set **Width** to **1000**, and Height to **600**.

Task 4. Connect the Data toggle button bar to the Visualization

1. Select the **Region_Data_TBB**, then in the Toolbar, click **Create a new connection**, and specify the following:
 - Region_Data_TBB (Region) > **Filter** > **Revenue_Visualization** (Region)
 - Reuse an existing variable: vRegion
 - Click **Connect**.
2. On the Application bar, click **Run options** , then click **Run Active Report**.
3. Select **Americas** and **Central Europe** from the **Region_Data_TBB** control.

The result appears as follows:

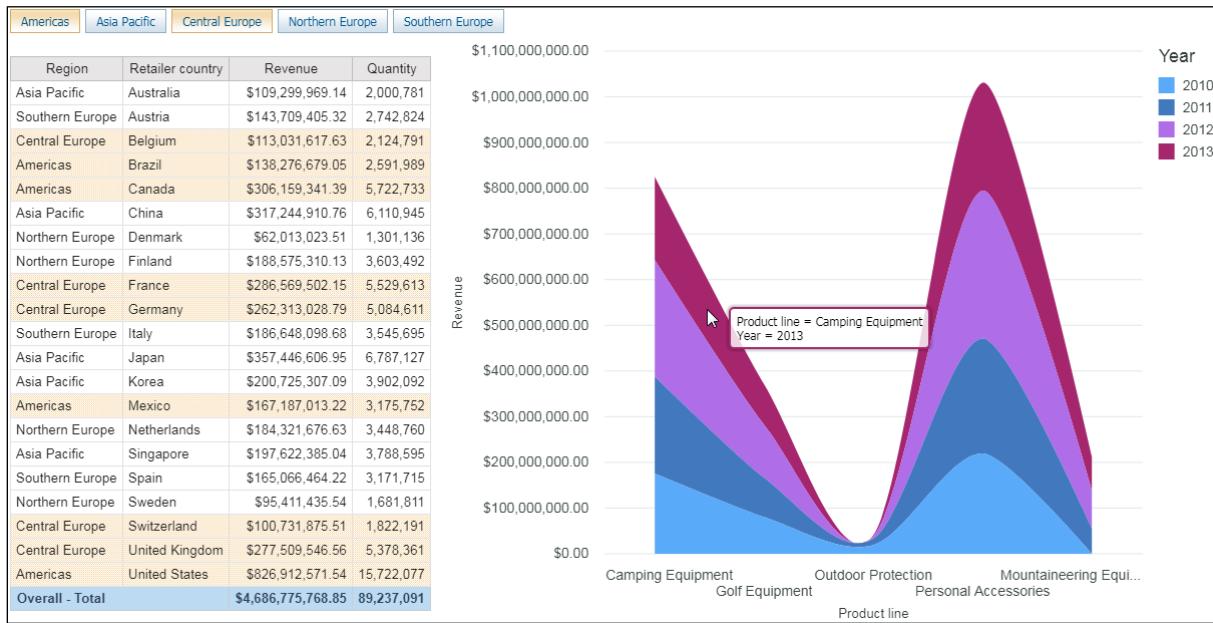


- Close the rendered report tab.

Task 5. Convert the line Visualization to an area Visualization

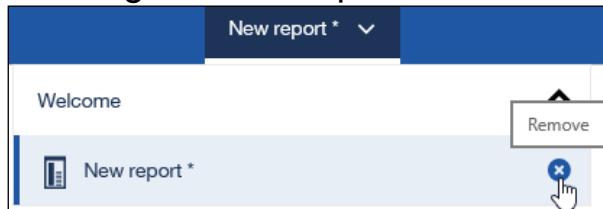
- Click the **Revenue_Visualization**, then on the Toolbar, click **More ... > Convert visualization**.
- In the categories list on the left, click **Area**, then double-click **Smooth area**. IBM Cognos Analytics populates the new visualization based on the data that was in the old visualization, so Revenue is added to the Values zone, Product line is added to the X categories zone, Region is added to the Extra Categories zone, and Year is added to the Color zone.
- In the **Properties** pane, under CLIENT BEHAVIOR, set **Animation** to **Transition**.
- On the Application bar, click **Run options** , then click **Run Active Report**.
- Select **Americas** and **Central Europe** from the **Region_Data_TBB** control.

The results appear as follows:



As you change selections, the colored sections of the area chart resize to reflect the new filtered data.

6. Close the rendered reports' tabs.
7. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



8. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You created a dynamic Active Report using RAVE visualizations to present an interactive experience for report users, and converted a visualization to provide a different graphical experience for the user.

Demonstration 5

Create a professional-looking Active Report (Optional)



Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Demonstration 5: Create a professional-looking Active Report (Optional)

Demonstration 5: Create a professional-looking Active Report (Optional)

Purpose:

You have a request to create a report that will be used by executives to present to their sales teams. The report will contain key metric information on Revenue, Planned revenue, Quantity and Gross profit for Australian sales in 2012. The report should be visually appealing.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO data warehouse (query)

Namespace: Sales (query)

Note that this demonstration is fairly advanced, and time-consuming. It should be used only once the other material in this course have been completed, and can be safely skipped if time is short.

Task 1. Create and format an Active Report

1. Using the **GO data warehouse (query)** package, create a new **Blank active report**.
 Note that this is a different package from the one used in other demonstrations in this course.
2. In the page body, click **Add** , then click **Block** .
3. On the Application bar, click **Show properties**  to open the Properties pane.
4. In the Properties pane, under BOX, double-click **Padding**.
5. Set **Top padding** to **10px**, and **Bottom padding** to **10px**, then click **OK**.
6. In the Properties pane, under BOX, double-click **Margin**.
7. Set **Bottom margin** to **15px**.
8. Click **OK**.
9. In the **Properties** pane, under COLOR & BACKGROUND, double-click **Background effects**.
10. Select the **Fill** check box.
11. Under **Colors**, click the color box, and then click **Color**.
12. In the Basic Colors tab, click **White**, and then click **OK**.
13. Click **New** , and then click **Color**.
14. In the **Basic Colors** tab, double-click **Teal**.
15. Under **Angle**, type **270**, and then click **OK**.
16. In the Properties pane, under POSITIONING, double-click **Size & overflow**.

17. Set **Width** to **1000px**, and then set **Height** to **125px**.
18. Click **OK**.
19. In the Properties pane, under **FONT & TEXT**, set **Horizontal alignment** to **Center**.
20. From the Toolbox, expand **ACTIVE REPORT** section, and drag a **Data radio button group** into the Box:
 - Name: **ProductLine_Data_RBG**
 - Query name: **ProductLine_Data_RBG_Query**
21. From the **Data**  tab, drag **Sales and Marketing (query) > Sales (query) > Products > Product line** to the **ProductLine_Data_RBG** control's drop zone.

Task 2. Display key metrics

1. From the Toolbox, drag a **List** onto the Block, after **ProductLine_Data_RBG**:
 - Name: **Sales_List**
 - Query name: **Sales_List_Query**
2. In the Properties pane, under **BOX**, double-click **Margin**, set **Top margin** and **Bottom margin** to **15px**.
3. From the **Data**  tab, drag the following query items to the List:
 - Sales fact > **Revenue**, **Planned revenue**, **Quantity**, and **Gross profit**.
4. In the Properties pane, under **GENERAL**, change **Column Titles** to **Hide**.
5. Click **<Revenue>**, then on the Toolbar, click **Select ancestor** , then click **List columns body style** .
6. In the Toolbar, click **Border** .
7. Under **Style**, click **None**, then click **Apply All Borders** , then click **OK**.
8. In the Properties pane, under **POSITIONING**, double-click **Size & overflow**.
9. Set **Width** to **250px**, and then click **OK**.
10. In the Toolbar, click **Font**.
11. Set **Size** to **20pt**, **Weight** to **Bold**, and then click **Foreground Color**.
12. Double-click **Blue**, then click **OK**.
13. In the Properties pane, under **FONT & TEXT**, set **Horizontal alignment** to **Center**.

The results appear as follows:

<Revenue>	<Planned revenue>	<Quantity>	<Gross profit>
<Revenue>	<Planned revenue>	<Quantity>	<Gross profit>
<Revenue>	<Planned revenue>	<Quantity>	<Gross profit>

14. In the Properties pane, under DATA, double-click **Data format**.
15. Under **Format type**, select **Currency**, and define the following properties:
 - Number of decimal places: **0**
 - Scale: **-6**
 - Pattern: **\$### M**
16. Click **OK**.
17. Click **<Quantity>**, then on the Toolbar, click **Filters > Edit Filters**.
18. Click **Add**, click **Advanced**, and then click **OK**.
19. In the **Expression Definition** pane, define the following expression:
 - **[Sales (query)].[Retailers].[Retailer country]='Australia'**
20. Click **OK**.
21. Click **Add**, click **Advanced**, and then click **OK**.
22. In the **Expression Definition** pane, define the following expression:
 - **[Sales (query)].[Time].[Year]=2012**
23. Click **OK**.
24. Click **OK**, to close the Filters dialog.
25. In the Toolbar, click **Data format**
26. Under **Format type**, select **Number**.
27. Define the following properties:
 - Scale: **-3**
 - Pattern: **### K**
28. Click **OK**.
29. From the Toolbox, drag a **Table** into the Block, below the **Sales_List** control.
30. Set **Number of columns** to **4**, and then click **OK**.
31. In the Properties panel, under **FONT & TEXT**, set **Horizontal alignment** to **Center**.

32. In the Toolbar, click **Font**.
33. Set **Size** to **12pt**, **Weight** to **Bold**.
34. Click **Foreground Color**, then double-click **White**.
35. Click **OK** to close the Font dialog.
36. In each of the Table cells, click **Add** then click **Text item**, and type the following:
 - Column 1: **Revenue**
 - Column 2: **Planned Revenue**
 - Column 3: **Quantity**
 - Column 4: **Gross Profit**
37. Ctrl-click the 4 table cells, then in the Properties pane, under POSITIONING, double-click **Size & overflow**.
38. Under **Width**, type **250px**, and then click **OK**.
39. Click the **ProductLine_Data_RBG** control.
40. In the Toolbar, click **Create a new connection**, and specify the following:
 - **ProductLine_Data_RBG** (Product line) > **Filter** > **Sales_List** (Copy Data Item from Source)
 - Create a new variable: **vProductLine**
 - Default value: **Camping Equipment**
41. Click **Connect**.
42. On the Application bar, click **Run options** , then click **Run Active Report**.
43. Select **Personal Accessories**.

The result appears as follows:

<input type="radio"/> Camping Equipment	<input type="radio"/> Golf Equipment	<input type="radio"/> Mountaineering Equipment	<input type="radio"/> Outdoor Protection	<input checked="" type="radio"/> Personal Accessories
\$14 M	\$14 M	258 K	\$6 M	

44. Close the rendered report tab.

Task 3. Add the Data deck to the report

1. From the **Toolbox**, drag a **Data deck** below the Block:
 - Name: **ProductLine_Data_Deck**
 - Query name: **ProductLine_Data_Deck_Query**
2. From the **Data** panel, drag **Products > Product line** into the **ProductLine_Data_Deck** object's **Deck items** zone.

Now you will specify interaction between **ProductLine_Data_RBG** and the **ProductLine_Data_Deck** Data deck

3. Click the **ProductLine_Data_RBG**.
4. In the Toolbar, click **Create a new connection**, and specify the following:
 - ProductLine_Data_RBG (Product line) > Select > **ProductLine_Data_Deck** (Product line)
 - Reuse an existing variable: **vProductLine**
5. Click **Connect**.
6. From the **Toolbox** tab, drag a **Table** onto the **ProductLine_Data_Deck**.
7. Set **Number of rows** to **2**, uncheck **Maximize width**, and then click **OK**.
8. In the Properties panel, under POSITIONING, double-click **Size & overflow**, set **Width** to **1000px**.
9. Select the top 2 table cells.
10. In the Toolbar, click **Font** .
11. Set **Size** to **12pt**, then click **Foreground Color**.
12. Double-click **Teal**, then click **OK** to close the Font dialog.
13. In the Properties pane, under FONT & TEXT, set **Horizontal alignment** to **Center**.
14. In the Properties pane, under POSITIONING, double-click **Size & overflow**, set **Width** to **50%**.
15. In the top-left table cell, click **Add** , then click **Text item**.
16. Type **2012 Revenue by Retailer Type**, and then click **OK**.
17. In the top-right table cell, click **Add** , then click **Text item**.
18. Type **2012 Margin by Retailer Type**, and then click **OK**.

Task 4. Add revenue chart to the report

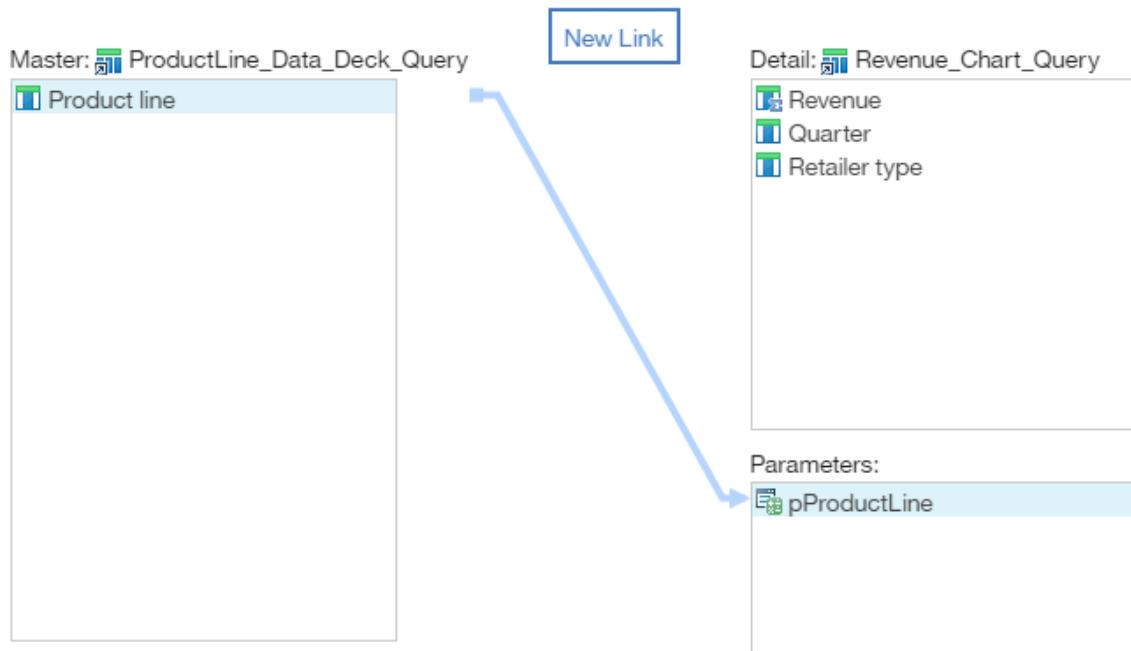
1. In the bottom-left Table cell, click **Add** , then click **Visualization** .
2. In the top right corner of the Visualization gallery dialog, click **Charts**.
3. In the **Find** box, type: **Cylinder**, then double-click the **Clustered Cylinder** chart.
4. In the Properties pane, under MISCELLANEOUS, set **Name** to **Revenue_Chart**.
5. In the Toolbar, click **More** , then click **Go to query**.
6. In the Properties pane, under MISCELLANEOUS, set **Name** to **Revenue_Chart_Query**.

7. From the side bar, click **Pages** , and then click **Page1**.
8. From the **Data** panel, drag **Sales fact > Revenue** to **Default measure (y-axis)**.
9. Expand the **Time** query subject, and then drag **Quarter** to **Categories (x axis)**.
10. Expand **Retailer type**, and then drag **Retailer type** to **Series (primary axis)**.
11. In the Properties pane, under POSITIONING, double-click **Size & overflow**.
12. Set **Width** to **500px** and **Height** to **400px**, and then click **OK**.
13. On Toolbar, click **Filters > Edit Filters**.
14. Click **Add**, click **Advanced**, and then click **OK**.
15. In the **Expression Definition** pane, define the following expression:
 - **[Sales (query)].[Retailers].[Retailer country]='Australia'**
16. Click **OK**.
17. Click **Add**, click **Advanced**, and then click **OK**.
18. In the **Expression Definition** pane, define the following expression:
 - **[Sales (query)].[Time].[Year]=2012**
19. Click **OK**.
20. Click **Add**, click **Advanced**, and then click **OK**.
21. In the **Expression Definition** pane, define the following expression:
 - **[Sales (query)].[Products].[Product line]=?pProductLine?**
22. Click **OK** to close the Detail filter expression dialog.
23. Click **OK** to close the Filters dialog.

Now you will link the query of the **ProductLine_Data_Deck** to specify the value of the **pProductLine** parameter; to be used in the filter expression.
24. On the Toolbar, click **More** , then click **Master Detail Relationships**.
25. Click **New Link**.
26. Under **Parameters**, click **pProductLine** to create a link between Product line from **ProductLine_Data_Deck_Query** and the parameter **pProductLine**.

The result appears as follows:

Master detail relationships



27. Click **OK**.

Task 5. Add margin Chart to the report

1. In the bottom-right Table cell, click **Add** , then click **Visualization** .
2. From the top right corner, click the visualization filter, then click **New visualization**.
3. Double-click the **Clustered Cylinder Bar** chart.
4. In the Properties pane, under MISCELLANEOUS, set **Name** to **Margin_Chart**.
5. In the Toolbar, click **More** , then click **Go to query**.
6. In the Properties pane, under MISCELLANEOUS, set **Name** to **Margin_Chart_Query**.
7. In the **Queries** tab, expand **Margin_Chart_Query**, then click **Margin_Chart**.
8. From the **Data** tab, drag **Gross margin** to **Default measure (x-axis)**.
9. Drag **Time > Quarter** to **Categories (y-axis)**.
10. Drag **Retailer type > Retailer type** to **Series (primary axis)**.
11. In the Properties pane, under POSITIONING, double-click **Size & overflow**.
12. Set **Width** to **500px** and **Height** to **400px**, and then click **OK**.
13. On the Toolbar, click **Filters**, and then click **Edit Filters**.
14. Click **Add**, click **Advanced**, and then click **OK**.
15. In the **Expression Definition** pane, define the following expression:
 - **[Sales (query)].[Retailers].[Retailer country]='Australia'**

16. Click **OK**.
17. Click **Add**, click **Advanced**, and then click **OK**.
18. In the Expression Definition pane, define the following expression:
 - **[Sales (query)].[Time].[Year]=2012**
19. Click **OK**.
20. Click **Add**, click **Advanced**, and then click **OK**.
21. In the **Expression Definition** pane, define the following expression:
 - **[Sales (query)].[Products].[Product line]=?pProdLine?**
22. Click **OK** to close the Detail filter expression dialog.
23. Click **OK** to close the Filters dialog.

Now you will link the query of the **ProductLine_Data_Deck** to specify the value of the **pProductLine** parameter; to be used in the filter expression.
24. On the Toolbar, click **More** , then click **Master Detail Relationships**.
25. Click **New Link**.
26. Under **Parameters**, click **pProductLine** to create a link between Product line from **ProductLine_Data_Deck_Query** and the parameter **pProductLine**.
27. Click **OK**.
28. Click **<Gross margin>**, then from the Properties pane, under DATA ITEM, set **Detail aggregation** to **Average**.

Task 5. Run the Active Report

1. On the Application bar, click **Run options** , then click **Run Active Report**.
2. Select **Personal Accessories**.

The result appears as follows:



3. Close the rendered report's tabs.
4. In the Application bar, click the report dropdown menu, then click **Remove** to the right of New report to close it. Then click **OK** to confirm your action.



5. Leave the IBM Cognos Analytics portal open for the next exercise.

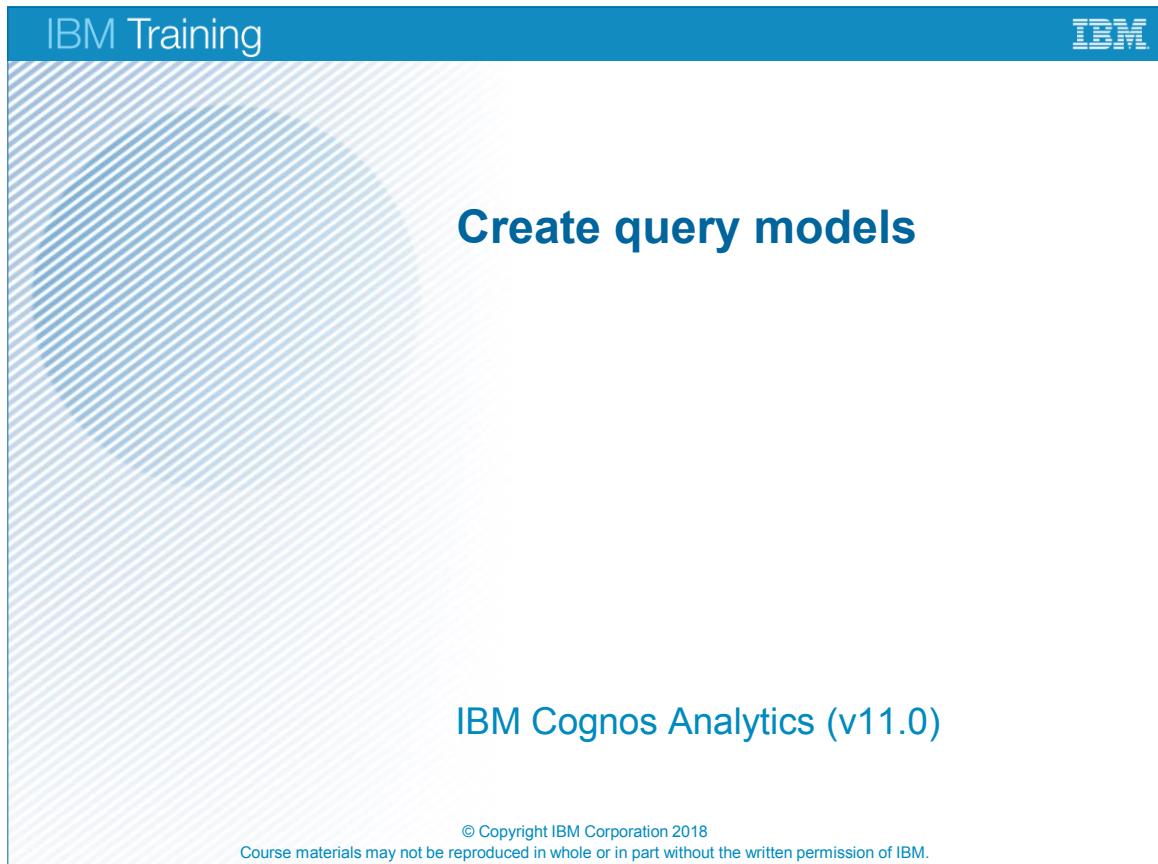
Results:

You created a professional-looking formatted Active Report that provides insight on sales in Australia for 2012.

Summary

- Describe characteristics of traditional charts in Active Reports
- Control data display using decks and data decks
- Use decks and data decks to display traditional charts
- Optimize decks for performance
- Describe characteristics of RAVE Visualizations

Unit 1 Create query models



The slide features a blue header bar with "IBM Training" on the left and the IBM logo on the right. The main content area has a light gray diagonal striped background. The title "Create query models" is centered in large blue text. Below it, the text "IBM Cognos Analytics (v11.0)" is displayed in blue. At the bottom, a copyright notice reads: "© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM."

Create query models

IBM Cognos Analytics (v11.0)

© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM.

Unit objectives

- Build query models and then connect them to the report layout
- Edit an SQL statement to author custom queries
- Add filters and prompts to a report using the query model

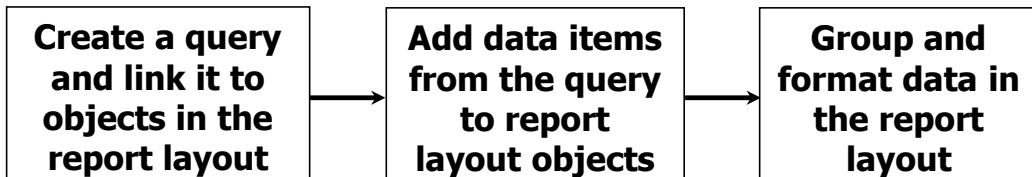
[Create query models](#)

© Copyright IBM Corporation 2018

Unit objectives

Create queries manually

Process to create a query manually and then create reports using the query



Create queries manually

It is a personal preference whether you work in the Pages tab (report layout) to create a report, or if you work in the Queries tab (data access layer of the report) by first creating the query and then later applying the data items and calculations to the report layout.

When building queries manually, you can:

- add data items to specify the data a query extracts from the data source
- specify data item properties
- define filters and slicers to exclude unnecessary data from the query
- specify how a query will retrieve data by customizing its SQL or MDX statement
- combine results from different queries

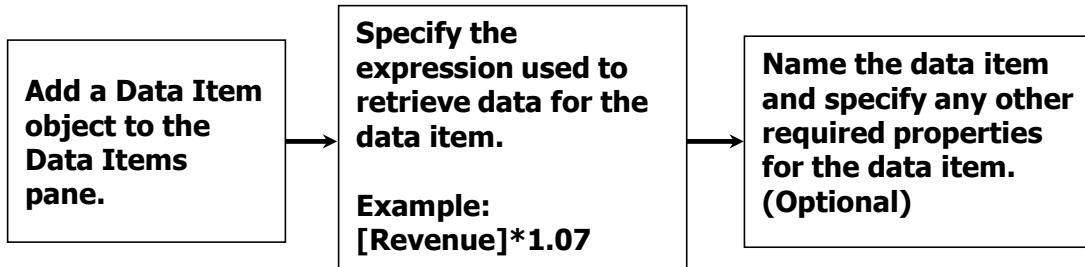
Notes:

- Slicers are context filters that are used when reporting from dimensional data sources.
- SQL (Structured Query Language) is the standard query language for relational data sources.
- MDX (Multi-Dimensional eXpressions) is the standard query language for dimensional data sources (OLAP)

Add calculated data items to queries

- If the package you are using does not contain the data you require, you can create a calculated data item that will retrieve the required data.

Process to Add a Calculated Data Item to a Query



Add calculated data items to queries

When you add a calculated data item using the Queries explorer (in the Queries tab), the item is added directly to the query and you can then choose whether you want to add it to the report layout, or not.

If you are working with many calculated data items, giving the items meaningful names helps you keep track of which data items contain which data.

It is useful to give a meaningful name the calculated data item because if the data item appears in the report layout, its name is used as the title by default if you don't specify the Label property. This will be used for example, in the List column title where the data item appears.

Demonstration and exercise start point information

This section describes in detail how to use the start point information included with the demonstrations and exercises in this course. It is particularly important for students in a self-paced learning environment to review this information before proceeding with the course.

Before you begin the steps of a demonstration or exercise, you will see start point information to help you set your environment for the tasks that you will perform in that demonstration or exercise. The start point format appears like the following:

Portal:	http://vclassbase/ibmcognos
User/Password:	brettonf/Education1
Package:	Team content\Samples\Models\GO data warehouse (query)
Folder:	Sales and Marketing (query)
Namespace:	Sales (query)

This information provides you with a unique starting point for that demonstration or exercise. It tells how you will access IBM Cognos Analytics through the browser. It provides the user ID and password to use, the package to use, and within it, the folder and namespace for items to be used in your report as you build it. Use this to set your environment before beginning the first task of the demonstration or exercise.

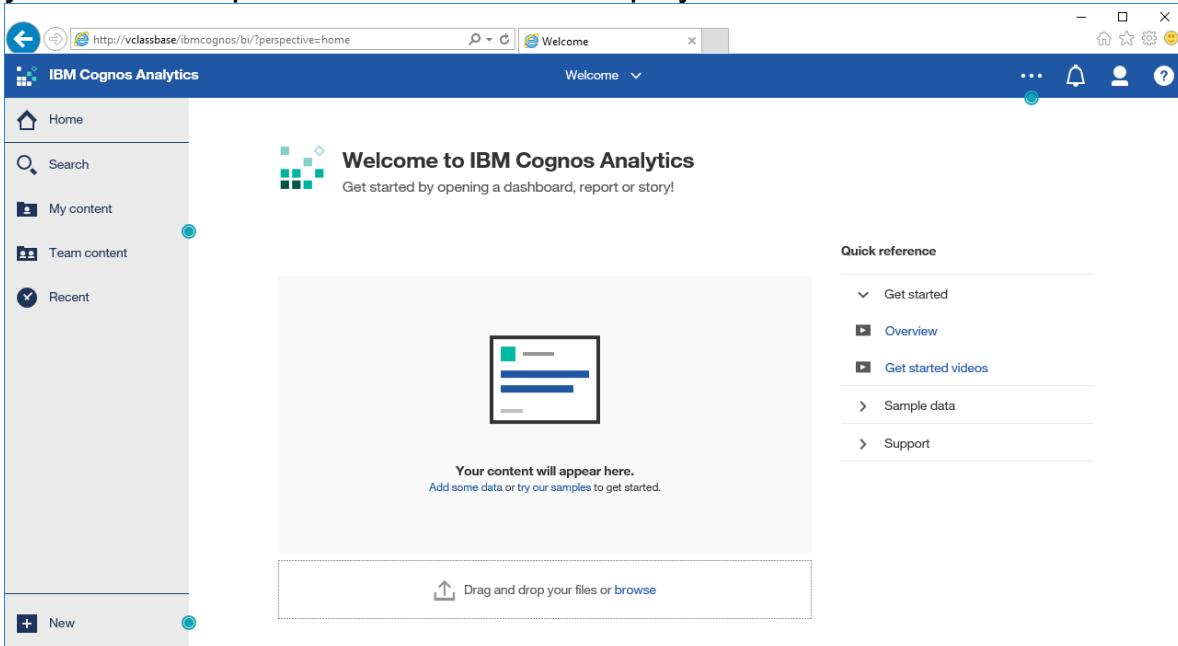
For example, if your demonstration provided the start point information above, you would do the following steps to create the List report using the Sales and Marketing (query)\Sales (query) namespace:

1. From the Windows taskbar, click the **Internet Explorer** icon to launch it.
2. In the **Address** bar, type the portal address **<http://vclassbase/ibmcognos>** and then press **Enter**.
The login page appears in a browser tab, with a Log on dialog box prompting for a User ID and Password. You will log on with the credentials listed in the start point information.
3. In the **User ID** box type **brettonf**, in the **Password** box type **Education1**.

4. Click **Sign in**.



The IBM Cognos Analytics portal displays the features of the application that your user has permission to use are displayed.



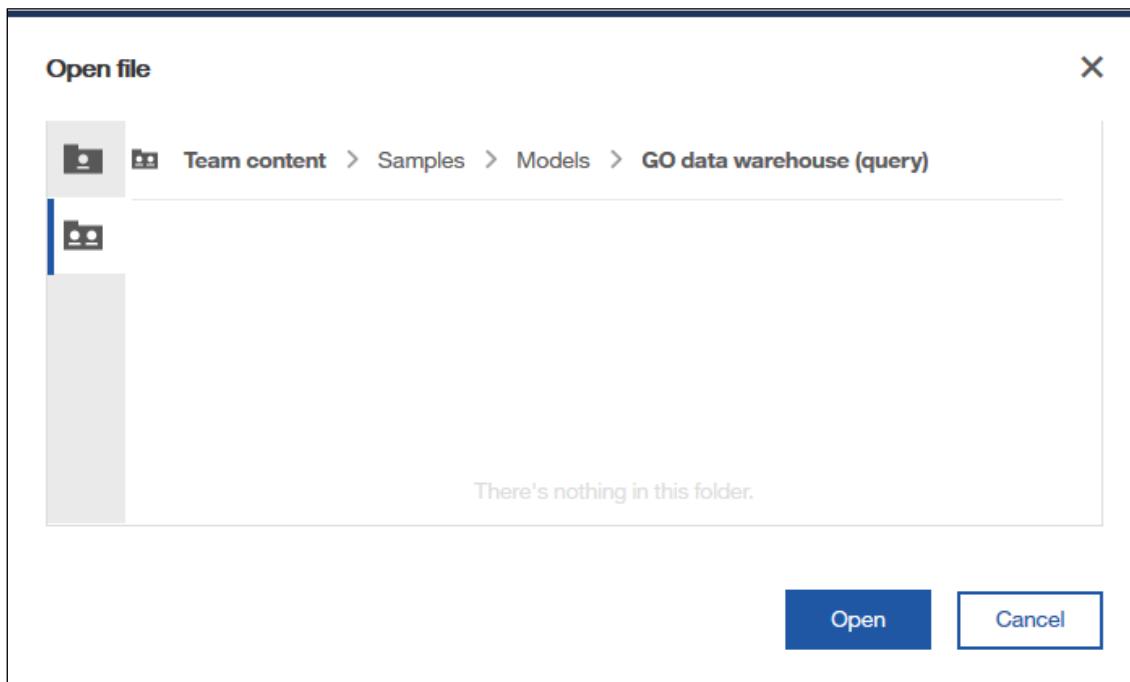
5. Click **+ New**, then click **Report**.

The Templates and themes dialog opens and allows you to choose a report template and theme.

6. Select the **Blank** template, then click **OK**.

7. In the **Data** tab, click **Add report data**

8. Browse to the required package by clicking **Samples > Models > GO data warehouse (query)**, then click **Open**.

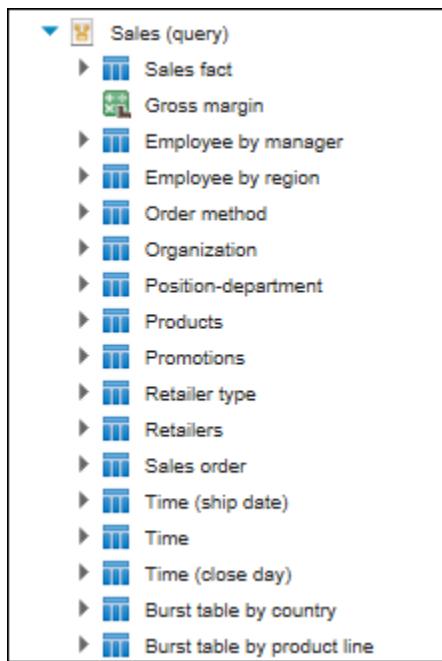


Now you have two tabs on the left: Source and Data items. The Source tab includes a package explorer for using the data in the package that you have selected, while the Data items tab contains a list of data items that are used in your report.

9. In the Source tab, expand the **Sales and Marketing (query)** folder.
The expanded folder displays the namespaces that are available to you in this package. The starting point information in this example will work with the Sales (query) namespace.

10. Expand the **Sales (query)** namespace.

The results appear as follows:



From here, you would work with the metadata of query subjects, query items, and facts within this selected namespace unless otherwise mentioned.

If you have performed the steps for the example start point here, then you are ready to begin the tasks in Demonstration 1, as the start point for Demonstration 1 was used in this example.

At the end of most of the demonstrations and exercises, you will be asked to close the report you designed. This is a best practice; to free up browser resources.

Demonstration 1

Build a query and connect it to a report

Product line	Product type	Revenue	Projected Revenue
Camping Equipment	Cooking Gear	272,835,984.18	291,934,503.0726
	Lanterns	126,925,660.64	135,810,456.8848
	Packs	351,880,402.84	376,512,031.0388
	Sleeping Bags	309,172,888.35	330,814,990.5345
	Tents	528,221,728.02	565,197,248.9814
Camping Equipment - Total		1,589,036,664.03	1,700,269,230.5121

Demonstration 1: Build a query and connect it to a report

Demonstration 1: Build a query and connect it to a report

Purpose:

Management wants to compare revenue generated by each product line. They also want to examine information about how much revenue each product line will generate if revenue increases by 7%. You will create a query that retrieves this data and then add data from the query to the report layout.

In the interest of time, most demonstrations and exercises in this course will not specifically instruct you to save the report. You can save your reports from demonstrations and exercises if you would like. A good place to save reports would be in My content, or in the Team content\B6059 folder of the IBM Cognos Analytics portal.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Clear the Automatic group and summary behavior for lists

Before beginning step 1, ensure that you have logged onto the portal with the user and password identified in the starting point information above, created a new Blank report and selected the correct package, etc. If you are unsure of how to do this, please refer to page 1-6, *Demonstration and exercise start point information* for detailed steps.

1. On the Application bar, click **More** ..., and then click **Options**.
2. Click the **Report** tab, then ensure that **Automatic group and summary behavior for lists** is deselected.
3. Click **OK** to close the Options dialog.

Task 2. Create a query

By starting with a blank report, IBM Cognos Analytics - Reporting has not created a query for this report. You will begin by adding a query to this report, and then add data to the query.

1. On the side bar, click **Queries** , then click **Queries** .
2. On the side bar, click **Toolbox** , then drag a **Query**  to the work area.
3. In the work area, double-click **Query1**.
4. On the side bar, click **Data** .
5. Expand the **Products**  query subject.

6. Click **Product line**, Ctrl+click **Product type**, and then drag both to the **Data Items**  pane.
7. On the **Source** tab, expand **Sales fact**, and then drag **Revenue** to the **Data Items** pane.
To display projected revenue generated if revenue increases by 7%, you will add a calculated data item that retrieves Revenue multiplied by 1.07.
8. Click **Toolbox**, and then drag a **Data Item**  object to the **Data Items** pane. The **Data item expression** dialog opens.
9. In the **Name** box, replace **Data Item1** with **Projected Revenue**.
You want to use the Revenue data item that is already in the query, rather than the Revenue item from the source, so you will add Revenue to the expression from the Data Items tab.
10. In the **Data item expression** dialog box, create and validate the following expression:
[Revenue]*1.07
Hint: Drag Revenue from the Data items tab, then complete the expression.
11. Click **OK** to close the **Data item expression** dialog box.
You have added the required data items to the query. Before using this query on report objects, you want to check the data this query will retrieve, regardless of the layout.
12. On the side bar, click **Queries**, right-click **Query1**, then click **View Tabular Data**.
You can view the data that the items in the query retrieve.
A section of the results appears as follows:

Product line	Product type	Revenue	Projected Revenue
Personal Accessories	Binoculars	130,834,653.2	139,993,078.924
Mountaineering Equipment	Climbing Accessories	81,096,582.48	86,773,343.2536
Camping Equipment	Cooking Gear	272,835,984.18	291,934,503.0726
Personal Accessories	Eyewear	867,125,198.48	927,823,962.3736
Outdoor Protection	First Aid	12,429,699.12	13,299,778.0584
Golf Equipment	Golf Accessories	51,514,343.88	55,120,347.9516

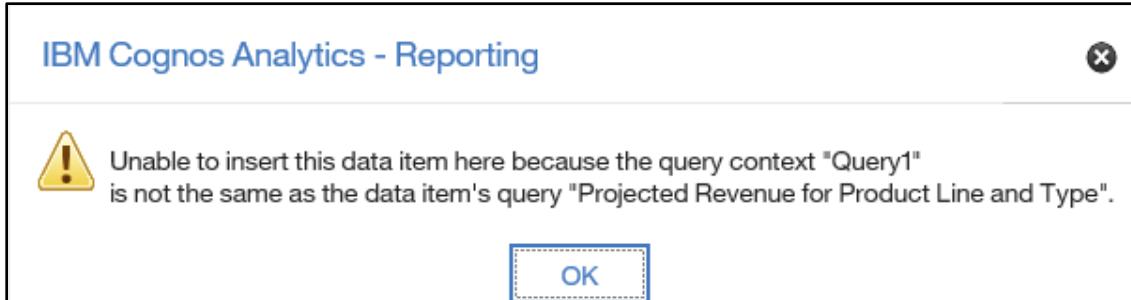
13. Close the rendered report tab.

Task 3. Name the query, and then link it to a List

1. In **Query explorer**, click the **Queries** folder, then in the work area, click **Query1**.
2. From the Application bar, click **Show properties** .

3. In the **Properties** pane, under the MISCELLANEOUS section, change the **Name** property from **Query1** to **Projected Revenue for Product Line and Type**.
 4. On the side bar, click **Pages** , and then click **Page1**.
 5. In the center of the work area, click **Add** , then click **List** .
 6. In the Object and query name dialog, accept the defaults by clicking **OK**.
 7. On the side bar, click **Queries** .
- Notice that the report now contains two queries: **Projected Revenue for Product Line and Type** and **Query1**. When you added the List object to this report and accepted the defaults, IBM Cognos Analytics - Reporting created a second query for this object.
8. On the side bar, click **Pages** , then click **Page1**.
 9. On the side bar, click **Data**, and then click the **Data items** tab.
 10. Under **Projected Revenue for Product Line and Type**, drag **Product line** to the List object.

An error message appears explaining that IBM Cognos Analytics - Reporting cannot insert this data item into the List because the query context (Query1) is not the same as the data item's query (Projected Revenue for Product Line and Type).



- Before you can add items from the first query to the List, you must link the first query to the List.
11. Click **OK** to close the message box.
 12. Click **Select List1** button  to select the entire List data container.
- In the Properties pane, under the DATA section, notice that Query1 appears in the Query property. This means that the List is linked to Query1. When you link this List to the first query, IBM Cognos Analytics - Reporting will then delete Query1 from the report.
13. In the **Properties** pane, in the **DATA** section, click the **Query** property, and then from the list, select **Projected Revenue for Product Line and Type**.
 14. On the side bar, click **Queries**.

Observe that the report now contains only the **Projected Revenue for Product Line and Type** query.

15. Expand the **Projected Revenue for Product Line and Type** query.

You can see that the query is currently used by the List object, List1.

Task 4. Add data to the List

1. In side bar, click **Data**, in the **Data items** tab, click the first item to select it, Shift-click the last item to select all the items in **Projected Revenue for Product Line and Type**, and then drag them all to the List.
 2. Click the **<Product line>** List column body, then from the On-demand toolbar, click **Group / Ungroup** .
- The List is now grouped by Product line.
3. Click **<Revenue>**, then from the Toolbar, click **Summarize** , then click **Total**.
 4. Click **<Projected Revenue>**, then from the Toolbar, click **Summarize > Total**.

Summary footers are included in the List for the selected columns.

The report now appears as follows:

Product line	Product type	Revenue	Projected Revenue
<Product line>	<Product type>	<Revenue>	<Projected Revenue>
<Product line> - Total		<Total(Revenue)>	<Total(Projected Revenue)>
<Product line>	<Product type>	<Revenue>	<Projected Revenue>
<Product line> - Total		<Total(Revenue)>	<Total(Projected Revenue)>
Overall - Total		<Total(Revenue)>	<Total(Projected Revenue)>

5. On the Application bar, click **Run options** , and then click **Run HTML**.

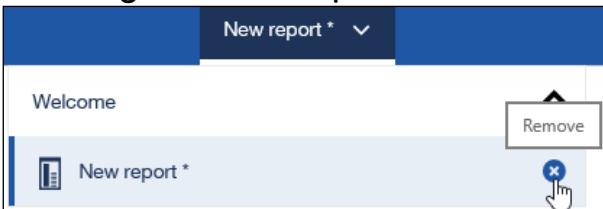
A section of the result appears as follows:

Product line	Product type	Revenue	Projected Revenue
Camping Equipment	Cooking Gear	272,835,984.18	291,934,503.0726
	Lanterns	126,925,660.64	135,810,456.8848
	Packs	351,880,402.84	376,512,031.0388
	Sleeping Bags	309,172,888.35	330,814,990.5345
	Tents	528,221,728.02	565,197,248.9814
Camping Equipment - Total		1,589,036,664.03	1,700,269,230.5121

IBM Cognos Analytics - Reporting retrieves the data you specified in the query model for the Projected Revenue for Product Line and Type query, and displays it in the List report.

6. Close the rendered report tab.

7. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



8. Leave the IBM Cognos Analytics portal open for the next demonstration.

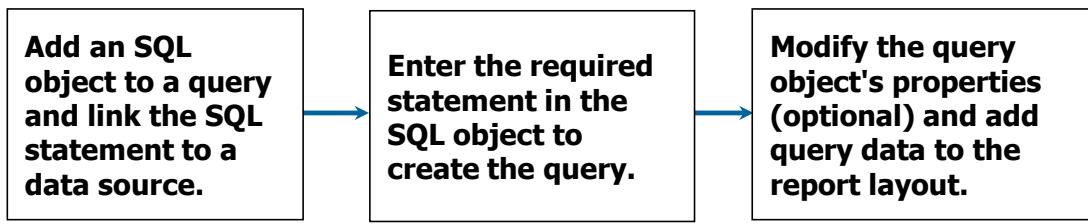
Results:

You built a query containing Product line, Product type, and Revenue items. You added a calculated data item to determine values for a 7% increase in revenue and then linked this query to a list report. You grouped the data in the list report, added aggregate data, and then ran the report.

Create custom queries using SQL

- You can create a SQL query statement that retrieves the data you require.
- You must specify the data source from which the query will retrieve data.

Process to Create a Custom Query Using SQL or MDX



Create custom queries using SQL

So far, you created reports that use the model (package) as a data source. But there are times when you want to customize the SQL statement that Cognos Analytics generates, e.g. to use an advanced syntax in your backend database system that Cognos Analytics is not using, or to optimize the SQL statement that Cognos Analytics is generating.

IBM Cognos Analytics - Reporting allows you to create a Query object that uses a custom SQL (Structured Query Language) statement that you define.

Or you can convert an existing Query object to use a SQL object instead of using the model (the package) as a data source. If you convert a query to use SQL, you cannot convert the statement back to a Query object that uses the model.

You can only add or remove data items from a SQL object by editing the SQL statement. You cannot drag data items from the package; as the Query does not refer to the package in this case.

Demonstration 2

Create a report using a SQL statement

Year: 2010		
Product Type	Product Name	Average Forecasted Cost
Binoculars	Opera Vision	51.08
	Ranger Vision	83.09
	Seeker 35	79.72
	Seeker 50	103.37
	Seeker Extreme	81.91
	Seeker Mini	45.46
Binoculars - Average		74.105

Demonstration 2: Create a report using a SQL statement

Demonstration 2: Create a report using a SQL statement

Purpose:

You have been asked to create a report using an SQL statement that displays the production cost of each product. The report should also display the average production costs for all products in each product type.

You will convert a query to SQL, and then you will use the SQL in a report to retrieve the necessary data in the report.

As a report author, by default you will not have the capabilities to execute this report from SQL. The administrator must grant capabilities to execute the specification and to execute user-defined SQL. This capability has been configured by the administrator for authors in your environment.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Product forecast (query)

Task 1. Create a query, and then convert the query to SQL

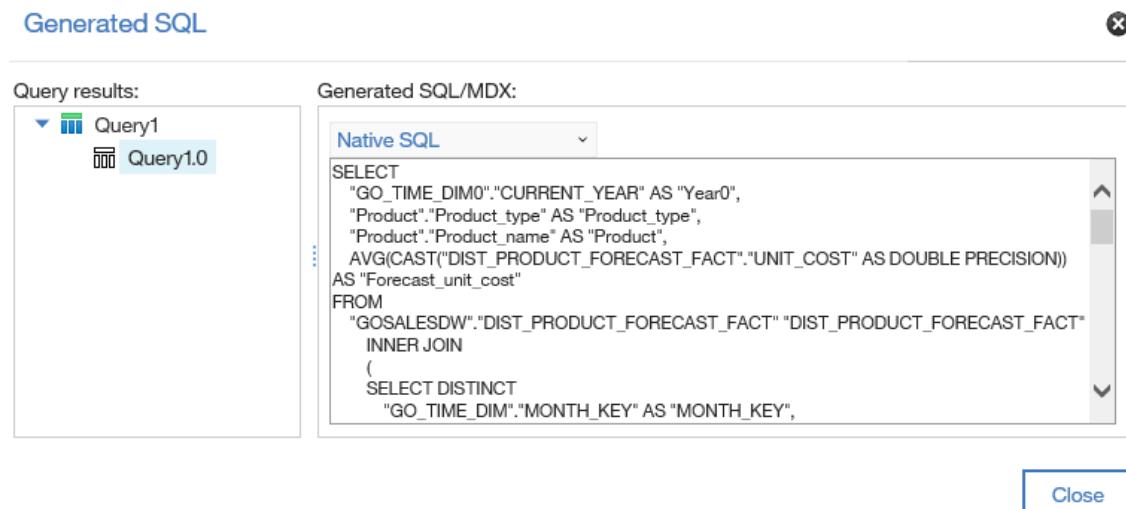
1. Navigate to **Team content > Samples > Models**.
2. Right-click the **GO data warehouse (query)** package, and then click **Create report** .
3. In the Templates and themes dialog, double-click the **Blank** template.
4. From the **Data\Source** tab, drag the following query items to the page from the **Product forecast (query)** namespace:
 - Time: **Year**
 - Products: **Product type**, **Product**
 - Product forecast fact: **Forecast unit cost**

The result appears as follows:

Year	Product type	Product	Forecast unit cost
<Year>	<Product type>	<Product>	<Forecast unit cost>
<Year>	<Product type>	<Product>	<Forecast unit cost>
<Year>	<Product type>	<Product>	<Forecast unit cost>

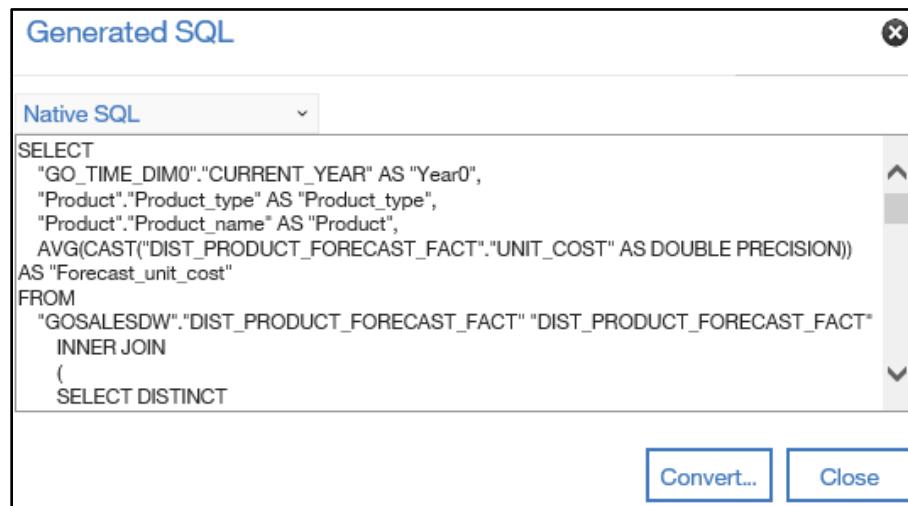
5. On the side bar, click **Queries** .
6. Right-click **Report**, then click **Show generated SQL/MDX**.

The results are displayed in the Generated SQL dialog box. The default format is Native SQL.



7. Under **Generated SQL/MDX**, change **Native SQL** to **IBM Cognos SQL**. Notice the different presentation format of the code. You will work with the Native SQL in this demonstration.
8. In the Query explorer tab, click **Query1**.
9. From the Application bar, click **Show properties** .
10. Under DATA, double-click the **Generated SQL** property.
11. Click **OK** to skip the information message.

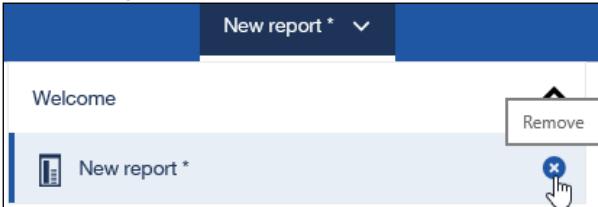
The results appear as follows:



You can click Convert to convert Query1 to use a SQL object instead of the model. But you want to review and edit the SQL statement first.

12. Click anywhere in the code, press Ctrl+A to select all the code, and then press Ctrl+C to copy the selection to the clipboard.
13. From the Windows taskbar, click **Start > Windows Accessories > Notepad**.

14. Press Ctrl+V to paste the code into **Notepad**, and then save this file to **C:\Training\B6059\Create Query Models SQL.txt**.
15. In **IBM Cognos Analytics - Reporting**, close the Generated SQL dialog box.
16. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



17. Leave the IBM Cognos Analytics portal open for the next demonstration.

Task 2. Create a List report, add an SQL object, and specify a data source

1. Create a new **Blank** report using the **GO data warehouse (query)** package.
2. In the side bar, click **Queries**, then click the **Queries** folder.
You have the SQL needed to retrieve the data for this report, so you will add an SQL object and then paste the SQL statement into it.
3. On the side bar, click **Toolbox** , then drag an **SQL** object to the work area.
The result appears as follows:



The SQL code that you have saved in Task 1 is native to the data source you will be retrieving data from, so you will leave the SQL Syntax property for the SQL object set to Native.

You need to specify that you will be retrieving data from the `great_outdoors_warehouse` database.

4. With the new **SQL** object selected, on the Application bar, click **Show properties** to open the **Properties** pane.
5. In the **Properties** pane, double-click the **Data source** property.
6. In the **Data source** dialog, click `great_outdoors_warehouse`, then click **OK**.

Task 3. Add an SQL statement, modify it, and then view the data retrieved

You will copy the SQL statement from Task 1 to the clipboard.

1. Switch to **Notepad**, where the **Create Query Models SQL.txt** code is displayed.

Examine the SQL and observe that near the beginning, this code specifies that the query retrieves four data items aliased as "Year0", "Product_type", "Product", and "Forecast_unit_cost" from the data source.

You want to modify the way this statement names the data items to make the final report easier to read.

When you add an SQL statement to a report, the report is using custom SQL and it can no longer be supported by IBM Cognos Customer Support.

Database administrators may be concerned with the performance of standard reports and may provide report authors with the SQL statements to use to create reports if they are concerned about the size or complexity of reports.

If your query uses a SQL object as its source, then in the Insertable Objects pane, you will not be able to take advantage of any relationships or other modeling data modelers have set up before publishing the package.

2. Make the following changes to the alias names (not the data item names), in the first 5 lines of:
 - "Year0" to "Year"
 - "Product_type" to "Product Type"
 - "Product" to "Product Name"
 - "Forecast_unit_cost" to "Average Forecasted Cost"

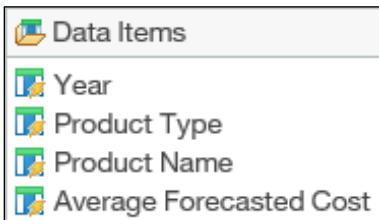
The first 5 lines of the code appear as follows:

```
SELECT
  "GO_TIME_DIM0"."CURRENT_YEAR" AS "Year",
  "Product"."Product_type" AS "Product Type",
  "Product"."Product_name" AS "Product Name",
  AVG(CAST("DIST_PRODUCT_FORECAST_FACT"."UNIT_COST" AS DOUBLE PRECISION)) AS "Average Forecasted Cost"
```

These changes specify that the four data items in the query be named Year, Product Type, Product Name, and Average Forecasted Cost.

3. At the end of the code, delete the following text:
 - **FOR FETCH ONLY**
4. Press Ctrl+A to select all the code, then press Ctrl+C to copy it to the clipboard.
5. In **IBM Cognos Analytics - Reporting**, double-click the **SQL** object.
6. Press Ctrl+V to paste the code.
7. In the **SQL** dialog box, click **Validate**, and then once the statement has been validated, click **OK** to close the "Validation Successful" message.
8. Click **OK** to close the **SQL** dialog box.
9. In the work area, double-click **Query1**.

The four data items retrieved by the data source appear in the Data Items pane for Query1. The data items are named as you specified in the SQL statement.



You will now rename the query.

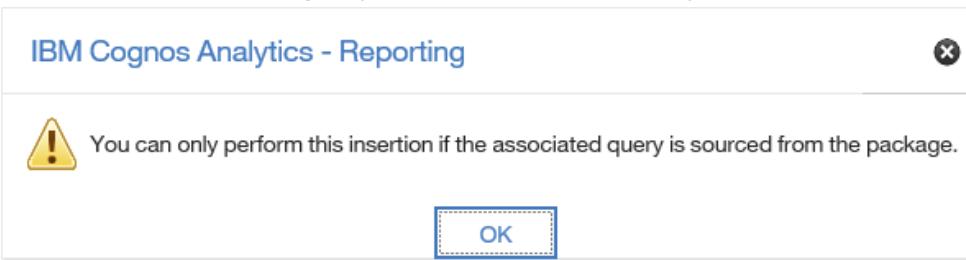
10. In the side bar, click **Queries**, then click **Query1**, then in the **Properties** pane, under the **MISCELLANEOUS** section, change the **Name** property from **Query1** to **Product Type Forecasted Costs**.

You will now add data from the **Product Type Forecasted Costs** query to a List on the report.

Task 4. Add data to the report

1. On the side bar, click **Pages** , then click **Page1**.
 2. In the work area, click **Add** , then click **List** .
 3. In the **Object and query name** dialog, under **Query Name**, select **Product Type Forecasted Costs**, then click **OK**.
- Now this List object is linked to the **Product Type Forecasted Costs** query, which means that you can add data from the query to this List.
4. On the side bar, click **Data**.
 5. On the **Data\DataSource** tab, expand **Sales and Marketing (query)\Product forecast (query)\Products** query subject, and then drag **Product line** query item to the List.

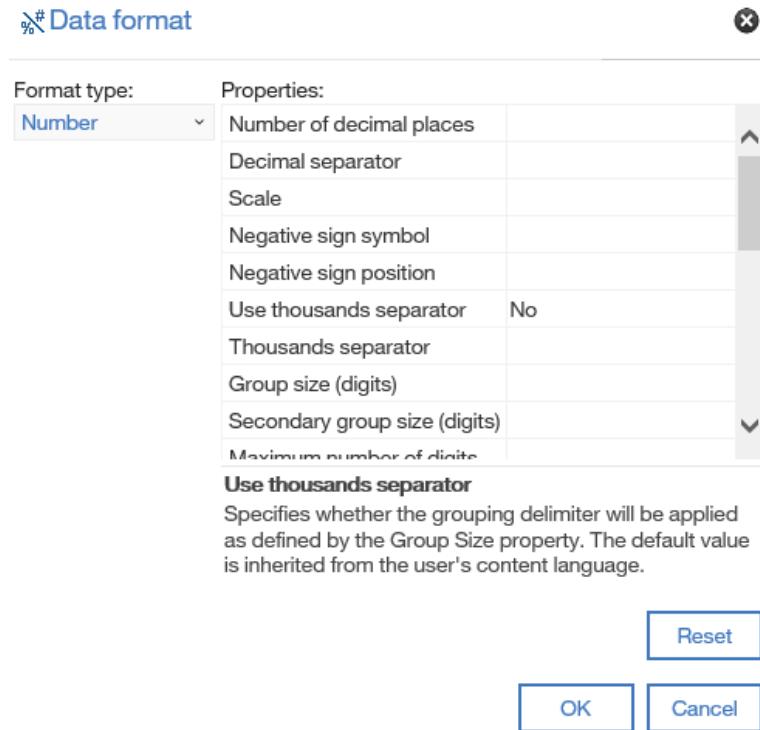
A warning message appears. Because the query was sourced from SQL and not from the package, you cannot add query items from the Source tab.



6. Click **OK** to close the message, and then click the **Data items** tab.
- The Product Type Forecasted Costs query, and the data items that you added to the query using the SQL statement, appear in the pane.
7. Under **Product Type Forecasted Costs**, click the first data item, Shift+click the last data item to select all of them, and then drag the selected items to the List.
 8. Click **<Year>**, then from the Toolbar, click **Section / unsection > Section / unsection** .

You have sectioned the report by Year.

9. From the Application bar, click **More** , and then click **Locked**  to unlock the report authoring environment.
10. Click **<Year>**, then from the Toolbar, click **Data format** .
11. In the **Format type** list, select **Number**.
12. Under **Properties**, set the Use thousands separator property to No.



13. Click **OK** to close the **Data Format** dialog box.
14. From the Application bar, click **More**, and then click **Unlocked**  to lock the report authoring environment again.
15. Click **<Product Type>**, and then from the Toolbar, click **Group / Ungroup**. The List is grouped by Product Type.

Year: <  Year>		
Product Type	Product Name	Average Forecasted Cost
<  Product Type>	<Product Name>	<Average Forecasted Cost>
<Product Type>	<Product Name>	<Average Forecasted Cost>

Task 5. Modify data item aggregation properties, and then run the report

You will include data about the average production cost for all products in each product type and the average production cost for all product types.

1. Click <Average Forecasted Cost>, and then from the Toolbar, click **Summarize** Σ , then click **Average**.

Notice that only one aggregate row appears at the bottom of the report. You also want the report to include aggregate rows for each product type. To accomplish this, you will need to change the Aggregate Function property for the Product Name data item.

2. From the Application toolbar, click **Undo** .
3. In the side bar, click **Queries**, then click the **Product Type Forecasted Costs**
4. In the **Data Items** pane, click **Product Name**.

In the Properties pane, notice that the Detail aggregation property is set to Default. To achieve the result you want in this report, you will set this property to None.

5. In the Properties pane, in the DATA ITEM section, click the **Detail aggregation** property, and then in the drop-down list, select **None**.
6. In the side bar, click **Pages**, and then click **Page1**.
7. Click the <Average Forecasted Cost>, then from the Toolbar, click **Summarize**, and then click **Average**.

Summary rows appear for each Product Type and a summary row appears at the bottom of the report for all the Product Types.

The result appears as follows:

Year: <Year>		
Product Type	Product Name	Average Forecasted Cost
<Product Type>	<Product Name>	<Average Forecasted Cost>
<Product Type> - Average		<Average(Average Forecasted Cost)>
<Product Type>	<Product Name>	<Average Forecasted Cost>
<Product Type> - Average		<Average(Average Forecasted Cost)>
Overall - Average		<Average(Average Forecasted Cost)>

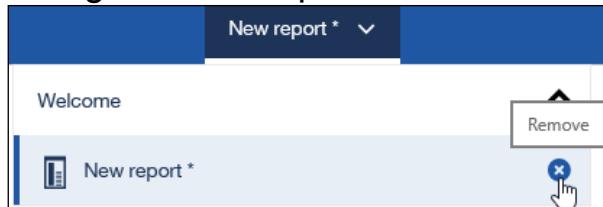
8. Run the report in **HTML**.

A section of the result appears as follows:

Year: 2010		
Product Type	Product Name	Average Forecasted Cost
Binoculars	Opera Vision	51.08
	Ranger Vision	83.09
	Seeker 35	79.72
	Seeker 50	103.37
	Seeker Extreme	81.91
	Seeker Mini	45.46
Binoculars - Average		74.105

The data items that you added using SQL are displayed, and the grouping and aggregation you specified in the report layout is applied.

9. Close the rendered report tab.
10. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.

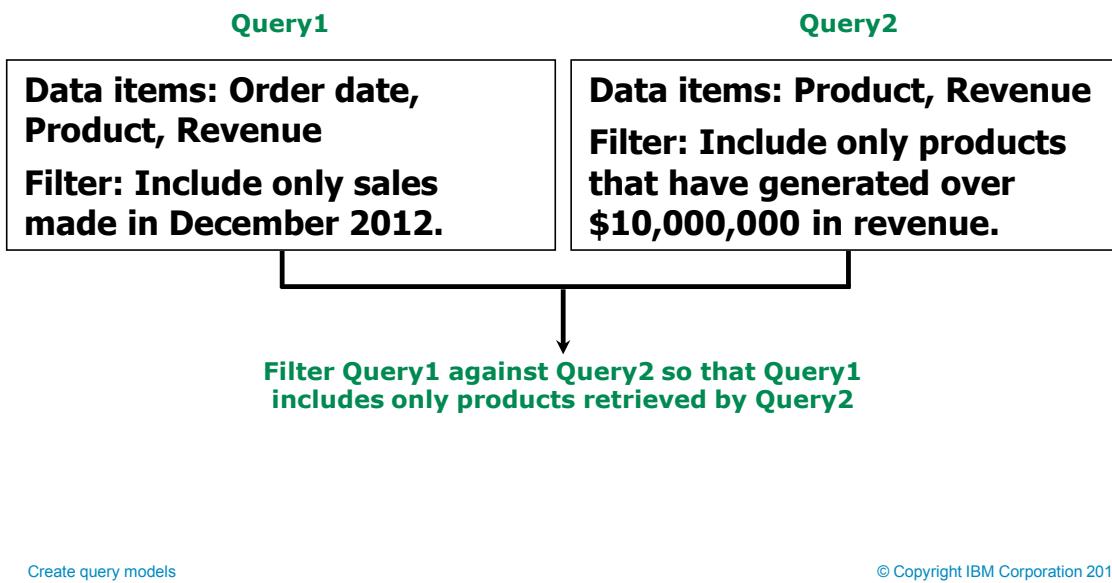


11. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You used an SQL statement to create a report that displays the production cost of each product. You modified the SQL statement to change the way data items are named and then added query data to the report layout. You sectioned, grouped, and formatted the data using the report layout and added aggregate data to the report to display the average production costs of all products in each product type and the average production cost of all product types.

Filter query data against data in a separate query



Filter query data against data in a separate query

When you filter Query1 against data in a second query, each query retrieves a separate set of data.

According to the filter you create, Query1 will only retrieve data retrieved by the second query.

Demonstration 3

Answer a business question by referencing data in a separate query

834905	Zone	755.55	23
834905 - Total		755.55	23
834922	Zone	1,790.1	54
834922 - Total		1,790.1	54
Overall - Total		9,451,826.18	165,443

Demonstration 3:

Answer a business question by referencing data in a separate query

Purpose:

For any given month, management wants to know which products were ordered that have lifetime revenues exceeding a specified amount. You will create a report to answer this question.

For example, management could use the report to view data for products ordered during December 2012 that have generated lifetime revenue of at least \$150 million.

To create this report, you will create one query containing prompts for the order month and year, and create a second query containing a prompt for lifetime revenue. You will then filter the first query against the data retrieved by the second query.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Create a List report containing data about orders and add prompts

1. Create a new Blank report using the **GO data warehouse (query)** package.
2. From the **Data\Source** tab, drag the following query items to the page:
 - Sales order: **Order number**
 - Time: **Year, Month (numeric)**
 - Products: **Product**
 - Sales fact: **Revenue, Quantity**

Order number	Year	Month (numeric)	Product	Revenue	Quantity
<Order number>	<Year>	<Month (numeric)>	<Product>	<Revenue>	<Quantity>
<Order number>	<Year>	<Month (numeric)>	<Product>	<Revenue>	<Quantity>
<Order number>	<Year>	<Month (numeric)>	<Product>	<Revenue>	<Quantity>

You will create a prompt for users to select a year to view data for.

3. On the Toolbar, click **Filters** , and then click **Edit Filters**.
4. In the **Filters** dialog box, ensure that the **Detail Filters** tab is selected, and then click **Add** .

You want to create a simple filter condition based on only one data item, so you could use the option of Custom based on data item, and select Year. However, the prompt that is displayed from this option offers the user the opportunity to select more than one year, so you will create an advanced filter to control this.

5. Select **Advanced**, and then click **OK**.
6. Create and validate the following expression:

[Year]=?Year?

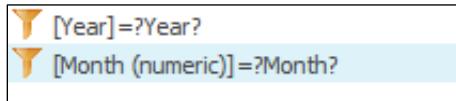
Hint: Drag Year from the **Data items** tab. Validate using 2012 when prompted.

7. Click **OK** to close the **Detail filter expression** dialog box.
8. Repeat steps 4 through 7, to create another detail filter for **Month (numeric)**:

[Month (numeric)]=?Month?

Provide **12** as the value for month number, when validating the expression.

You now have two filters.



9. Click **OK** to close the **Filters** dialog box.

Task 2. Format and then test the report

1. Click **<Order number>**, Shift-click **<Month (numeric)>**, then from the Toolbar, click **Group / Ungroup**.
2. Click **<Revenue>**, Ctrl-click **<Quantity>** then from the Toolbar, click **Summarize > Total**.

A section of the result appears as follows:

Order number	Year	Month (numeric)	Product	Revenue	Quantity
<Order number>	<Year>	<Month (numeric)>	<Product>	<Revenue>	<Quantity>
		<Month (numeric)> - Total		<Total(Revenue)>	<Total(Quantity)>
		<Month (numeric)>	<Product>	<Revenue>	<Quantity>
		<Month (numeric)> - Total		<Total(Revenue)>	<Total(Quantity)>
		<Year> - Total		<Total(Revenue)>	<Total(Quantity)>

You will test the report by viewing data for December 2012.

3. Run the report in **HTML**.
4. In the Prompt page, enter **2012** for **Year**, and **12** for **Month (numeric)**
5. Click **OK**.

A section of the result appears as follows:

Order number	Year	Month (numeric)	Product	Revenue	Quantity		
104579	2012	12	Canyon Mule Weekender Backpack	80,393.6	296		
			Firefly 4	7,507.2	255		
			Flicker Lantern	7,737.75	225		
			Mountain Man Analog	5,891.7	123		
			Polar Ice	2,090	19		
			Polar Sports	6,296.93	52		
			Polar Sun	1,174.96	19		
			Star Gazer 2	142,198.1	257		
			12 - Total	253,290.24	1,246		
2012 - Total				253,290.24	1,246		
104579 - Total				253,290.24	1,246		

6. In the report, click **Bottom**, and then scroll to the end of the report.
You can see the total revenue generated by orders made in December and the total quantity of products sold.
7. Close the rendered report tab.

Task 3. Add a third filter to prompt for a minimum lifetime revenue amount

You want this report to display only data for orders that have life-to-date revenues of a minimum specified amount, selected using a prompt at run time. You will add a filter to the report to achieve this result.

1. Click anywhere on the List, from the Toolbar click **Filters > Edit Filters**.
2. Click **Add**, select **Advanced**, and then click **OK**.
3. Create and validate the following expression:
[Revenue]>=?Minimum revenue?
Hint: Drag Revenue from the Data items tab.
Validate the expression with any number as the minimum revenue prompt.
4. Click **OK** to close the **Detail filter expression** dialog box.
5. In the **Filters** dialog box, with the Revenue filter still selected, in the **Application** section, select **After auto aggregation**, and then click **OK**.
You will test this prompt by viewing data for products sold in December 2012 that generated lifetime revenue of at least 150 million dollars.
6. Run the report in **HTML**, ensure that **2012** is the **Year**, **12** is the **Month (numeric)**, **150000000** (150 million) is the **Minimum revenue**, and then click **OK**.

The report contains no data since all filters combined eliminated all data from the query.

To retrieve the data you require, you will remove the third filter from Query1 and will create a second query containing this filter. You will then create a query reference from the first query. You will first give Query1 a more meaningful name.

7. Close the rendered report tab.
8. On the side bar, click **Queries**, then click **Query1**.
9. Open the **Properties** pane, and then in the **MISCELLANEOUS** section, change the **Name** property to **Selected month**.
10. In the **Detail Filters** pane, click the **[Revenue]>=?Minimum revenue?** filter, and then press **Delete**.

Task 4. Add a second query containing a filter to prompt for a minimum lifetime revenue amount

1. Click on **Queries** folder to view of all queries in the report.
2. From the **Toolbox**, drag a **Query** object below the **Selected month** query in the work area, and then double-click the newly added **Query1**.
3. In the **Properties** pane, in the **MISCELLANEOUS** section, change the **Name** property to **Minimum revenue**.
You will add items to your query.
4. On the **Data\DataSource** tab, under the **Products** query subject, double-click **Product**, and then under the **Sales fact** query subject, double-click **Revenue** - to add them to the **Data Items** pane.
You will add a filter to this report to prompt users to select a minimum lifetime revenue for all products retrieved by the query.
5. From the **Data Items** pane, drag **Revenue** into the **Detail Filters** pane.
6. Create and validate the following expression:
[Revenue]>=?Minimum revenue?
You want to apply this filter to all revenue generated by a product rather than the revenue generated by individual orders.
7. Click **OK**.
8. With the filter you just created still selected, in the **Properties** pane, under the **GENERAL** section, click the **Application** property, and then from the list select **After Auto Aggregation**.
You will preview these results to see which products have a life-to-date revenue of at least 150 million.
9. In the side bar, click **Queries** .

- Right-click the **Minimum revenue** query, then click **View Tabular Data**, ensure **150000000** (150 million) appears in the **Minimum revenue** box, and then click **OK**.

The result appears as follows:

Revenue	Product
168,191,550.48	Star Lite
157,369,344.95	Zone

You can see that two products, Star Lite and Zone, generated lifetime revenue of at least one hundred and fifty million dollars.

- Close the rendered report tab.

Task 5. Filter the first query by referencing data in the second query

You will specify that only products that meet the conditions of the Minimum revenue filter should be included in the Year to date query.

- Click the **Selected month** query.

You will filter the Product name data item in the Selected month query, against the Product name data items retrieved by the Minimum revenue query.

- From the **Data Items** pane, drag **Product** to the **Detail Filters** pane.
- At the end of the expression, type **IN (**, and then in the **Available Components** pane, click the **Queries**  tab.
- From the **Available Components** pane, under **Minimum revenue**, drag **Product** to the end of the expression.
- At the end of the expression, type **)**.

The final expression appears as follows:

[Product] IN ([Minimum revenue].[Product])

- Validate the expression, then click **OK** to close the **Detail filter expression** dialog box.
- Run the report in **HTML**.
- Ensure that **150000000** (150 million) is the **Minimum revenue**, **2012** is the **Year**, **12** is the **Month (numeric)**, and then click **OK**.

A section of the result appears as follows:

Order number	Year	Month (numeric)	Product	Revenue	Quantity
104614	2012	12	Star Lite	210,972	600
		12 - Total		210,972	600
		2012 - Total		210,972	600
104614 - Total				210,972	600

The report contains data for the orders in December 2012 of the two products that have lifetime revenue of at least 150 million dollars (Star Lite and Zone).

- Close the rendered report tab.

Task 6. Format the report

You do not need to include the Year and Month columns in the List; as the user is filtering on a specific year and month.

- On side bar, click **Pages** , then click **Page1**.
- Click <Year>, and Ctrl-click <Month (numeric)>, then press **Ctrl-X** to cut them.
- Delete the <Month (numeric)> and <Year> summary rows; as they are no longer needed.

The results appear as follows:

Order number	Product	Revenue	Quantity
<Order number>	<Product>	<Revenue>	<Quantity>
<Order number> - Total		<Total(Revenue)>	<Total(Quantity)>
<Order number>	<Product>	<Revenue>	<Quantity>
<Order number> - Total		<Total(Revenue)>	<Total(Quantity)>
Overall - Total		<Total(Revenue)>	<Total(Quantity)>

- Run the report in **HTML**, with **150000000** as the **Minimum revenue**, **2012** as the **Order year**, and **12** as the **Month (numeric)**.
- Click **Bottom** and then scroll down.

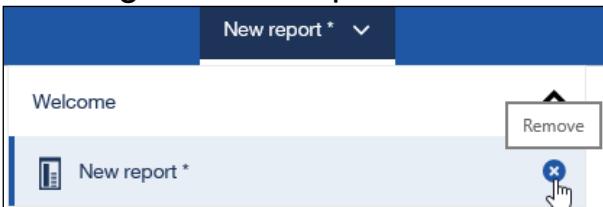
A section of the result appears as follows:

834905	Zone	755.55	23
834905 - Total		755.55	23
834922	Zone	1,790.1	54
834922 - Total		1,790.1	54
Overall - Total		9,451,826.18	165,443

You can see that for December 2012, 165,443 units of these two products were sold and they generated 9,451,826.18 in revenue.

- Close the rendered report tab.

7. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



8. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

By filtering one query against another, you created a report that lets users choose to view order data for a specified month for products whose lifetime revenue exceeds a specified amount.

Unit summary

- Build query models and then connect them to the report layout
- Edit an SQL statement to author custom queries
- Add filters and prompts to a report using the query model

[Create query models](#)

© Copyright IBM Corporation 2018

Unit summary

Unit 2 Create reports based on query relationships

IBM Training



Create reports based on query relationships

IBM Cognos Analytics (v11.0)

© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM.

Unit objectives

- Create reports by merging query results
- Create reports by joining queries
- Combine data containers based on relationships from different queries

Create reports based on query relationships

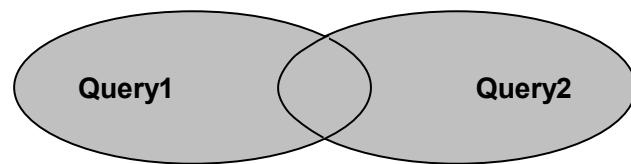
© Copyright IBM Corporation 2018

Unit objectives

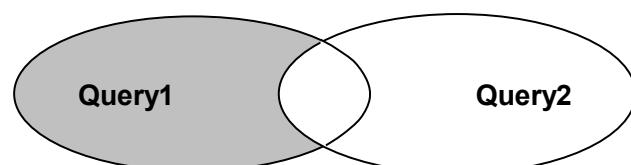
Merge Query Results Using Set Operations (1 of 2)

- To retrieve only the data from separate queries that meets a specific relationship requirement, use set operations.

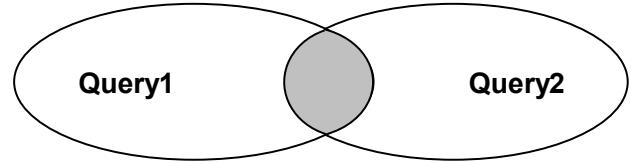
Union Operator



Except Operator



Intersect Operator



Create reports based on query relationships

© Copyright IBM Corporation 2018

Merge Query Results Using Set Operations

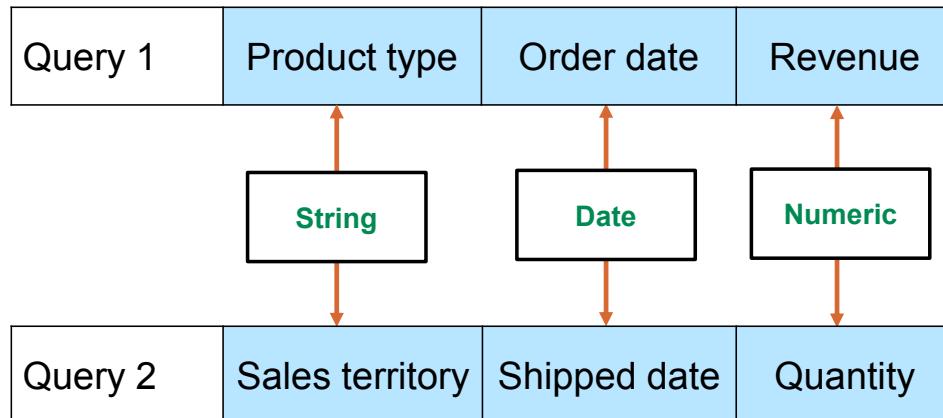
Set operations can be used to merge results of two queries into one result set. Three set operations are available:

- The Union operator combines two queries into one query including all data retrieved by any of the combined queries.
- The Except query combines two queries into one query including only data retrieved by the first query that is not also retrieved by the second query.
- The Intersect operator combines two queries into one query including only data that is common to the combined queries.

A Join is not a set operator; because it does not create a projection list. When you use a Join, two full query sets are linked together through a common data item. We will discuss Joins later in this unit.

Merge Query Results Using Set Operations (2 of 2)

- Combined queries must contain the same number of data items, the data items must be compatible in the same order



Two queries that can be combined using set operations

Using set operations to merge data from two queries may produce intermixed rows. For example, you may have one query that extracts order method data and another query that extracts sales representative data. The sales rep names and order methods will likely not be sorted into distinct groups of rows.

You can choose to keep the data from the two queries distinct (separate). For example, so that all sales rep names appear first, then all order methods appear below.

To do this, add a sort key data item to each of the queries combined by the set operator, then add this sort key as a data item in the merged query.

Demonstration 1

Answer a business question by creating a Union between two queries

Sales rep/Order method	Revenue
Kolina Nilsson	151,605.59
Henri Bacchus	344,978.59
Johan Mäkinen	747,352.68
Beatriz Couto	905,704.48
Frida Armfelt	1,186,849.91
Akira Hashimoto	1,206,901.22
Wanda Coleman	1,226,558.46
Morela Castro	1,533,562.39
Jake Cartel	1,582,666.1
Frank Jever	1,841,730.57
Special	1,006,100.01
Mail	6,905,730.44
Fax	13,445,559.93
E-mail	23,701,042.57

Demonstration 1: Answer a business question by creating a Union between two queries

Demonstration 1: Answer a business question by creating a Union between two queries

Purpose:

Management wants to know which sales reps and order methods did not achieve their yearly targets in 2012. Sales reps have a yearly target of 2 million, and order methods have a yearly target of 30 million.

To answer this question, you will create two queries because you have different filters for each target. Then you will merge the two queries into a single result set using a Union set operation. You will then use a sort key to keep the data from the two queries separated.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

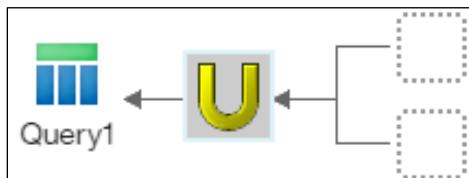
Folder: Sales and Marketing (query)

Namespace: Sales (query)

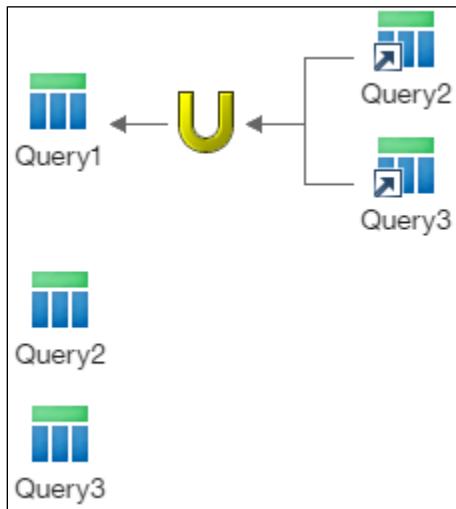
Task 1. Add a Union object and queries to a blank report

1. Create a new Blank report using the **GO data warehouse (query)** package.
2. On the side bar, click **Queries** , then click the **Queries** folder.
3. On the side bar, click **Toolbox** , then drag a **Union**  object to the work area.

A new Query is created, and two drop zones appear to the right of the Union object.



4. From the **Toolbox**, drag a **Query** object to each of the two drop zones. Query2 and Query3 appear in the work area and shortcuts to each query appear in the drop zones to the right of the Union object. You will add data to each of these queries then union the results.



Task 2. Add data and filters to the Sales reps query

1. On the Application bar, click **Show properties**.
2. In the work area, double-click **Query2** or its shortcut.
3. In the Properties pane, under MISCELLANEOUS, change the **Name** property to **Sales reps query**.
4. Click the **Data\Source** tab, expand the **Sales and Marketing (query)** folder and the **Sales (query)** namespace.
5. From the **Employee by region** query subject, drag **Employee name** to the **Data Items** pane.
6. From the **Sales fact** query subject, drag **Revenue** to the **Data Items** pane.
You want this query to focus on sales made during 2012.
7. From the **Data\Source** tab, expand **Time**, then drag **Year** to the **Detail Filters** pane.
8. In the Detail filter expression dialog, complete the expression to be:
[Sales (query)].[Time].[Year]=2012
9. Click **OK** to close the Detail filter expression dialog.
You also want this query to include only sales reps who achieved the yearly target for a sales rep.
10. From the **Data Items** pane, drag **Revenue** to the **Detail Filters** pane to start creating a filter using the Revenue.
11. In the Detail filter expression dialog, complete the expression to be:
[Revenue]<=2000000
12. Click **OK** to close the Detail filter expression dialog.
13. In the Properties pane, under GENERAL, change the **Application** property to **After Auto Aggregation**.

By selecting After Auto Aggregation, the filter will apply to the total revenue generated by each sales rep, rather than the revenue per order.

14. To test the query, click the **Queries** tab, then right-click **Sales rep query**, then click **View tabular data**.
15. An information message appears to tell you that you are running the query only, regardless of any layout that is using it. Click **OK** to close the message.

The result appears as follows:

Employee name	Revenue
Akira Hashimoto	1,206,901.22
Beatriz Couto	905,704.48
Frank Jever	1,841,730.57
Frida Armfelt	1,186,849.91
Henri Bacchus	344,978.59
Jake Cartel	1,582,666.1
Johan Mäkinen	747,352.68
Kolina Nilsson	151,605.59
Morela Castro	1,533,562.39
Wanda Coleman	1,226,558.46

16. Close the viewer tab to return to the report.

Task 3. Add data and filters to the Order methods query

1. Click the **Queries** tab, then click the **Queries** folder.
2. In the work area, double-click **Query3** or its shortcut.
3. In the Properties pane, under MISCELLANEOUS, change the **Name** property to **Order methods query**.
4. Click the **Data\DataSource** tab, expand the **Order method** query subject, then drag **Order method type** to the **Data Items** pane.
5. From the **Sales fact** query subject, drag **Revenue** to the **Data Items** pane.
You want this query to focus on sales made during 2012.
6. From the **Data\DataSource** tab, expand **Time**, then drag **Year** to the **Detail Filters** pane.
7. In the Detail filter expression dialog, complete the expression to be:
[Sales (query)].[Time].[Year]=2012
8. Click **OK** to close the Detail filter expression dialog.
You also want this query to include only order methods that achieved the yearly target for an order method.

9. From the **Data Items** pane, drag **Revenue** to the **Detail Filters** pane to start creating a filter using the Revenue.
10. In the Detail filter expression dialog, complete the expression to be:
[Revenue]<30000000
11. Click **OK** to close the Detail filter expression dialog.
12. In the Properties pane, under GENERAL, change the **Application** property to **After Auto Aggregation**.
13. To test the query, click the **Queries** tab, then right-click **Order methods query**, then click **View tabular data**.
14. Click **OK** to close the information message.

The result appears as follows:

Order method type	Revenue
E-mail	23,701,042.57
Fax	13,445,559.93
Mail	6,905,730.44
Special	1,006,100.01

15. Close the viewer tab to return to the report.

Task 4. Combine data items from the two queries in the Result query

1. Click the **Queries** tab, then click **Query1**.
2. In the Properties pane, under MISCELLANEOUS, change the **Name** property to **Result query**.
3. Click the **Union** object, then in the Properties pane, under DATA, double-click the **Projection list** property.
4. In the **Projection list** dialog, select **Manual**.
5. In the list, double-click **Employee name**.
6. In the Edit dialog, change the existing text to: **Sales rep/Order method**.
7. Click **OK** to close the Edit dialog, then again to close the Projection list dialog.
8. Double-click the **Result query**.
9. From the **Data/Source** tab, drag both **Sales rep/Order method** and **Revenue** to the **Data Items** pane.
10. On the side bar, click **Pages**, then click **Page1**.
11. From the **Data\Data items** tab, drag **Result query** to the work area to get a List using that query.
12. On the Application bar, run the report as HTML.

The result appears as follows:

Sales rep/Order method	Revenue
Kolina Nilsson	151,605.59
Henri Bacchus	344,978.59
Johan Mäkinen	747,352.68
Beatriz Couto	905,704.48
Special	1,006,100.01
Frida Armfelt	1,186,849.91
Akira Hashimoto	1,206,901.22
Wanda Coleman	1,226,558.46
Morela Castro	1,533,562.39
Jake Cartel	1,582,666.1
Frank Jever	1,841,730.57
Mail	6,905,730.44
Fax	13,445,559.93
E-mail	23,701,042.57

This is not the required result; as records of order methods are merged with records of sales reps, and they are not separated. You will use a sort key to sort the result, so that all sales reps appear together then all order methods appear together.

13. Close the rendered report tab.

Task 5. Add a sort key data item to all queries and use it to sort the merged result

You will add a Sort key data item with an expression of 'A' to the Sales reps query and will add a Sort key data item with an expression of 'B' to the Order methods query. You can then use this data item to sort the data in the **Sales rep/Order method** column of the report.

1. On the side bar, click **Queries**, then click **Sales reps query**.
2. On the side bar, click the **Toolbox** , then drag a **Data item** to the **Data Items** pane under existing data items.
3. In the Data item expression dialog, set **Name** to **Sort key**, and for the Expression Definition, type '**A**'.
4. Click **OK** to close the Data item expression dialog.
5. On the side bar, click **Queries**, then click **Order methods query**.
6. From the **Toolbox**, drag a **Data item** to the **Data Items** pane under existing data items.
7. In the Data item expression dialog, set **Name** to **Sort key**, and for the Expression Definition, type '**B**'.

8. Click **OK** to close the Data item expression dialog.
9. Click the **Queries** tab, then click the **Queries** folder.
10. In the work area, select the **Union** object, then in the Properties pane, under the DATA section, double-click the **Projection list** property.
11. In the Projection list dialog, select **Manual**, then click **Add** .
12. Double-click **Data Item1**, then in the Edit box, type **Sort key**, then click **OK**.
13. Click **OK** to close the Projection list dialog.
14. In the **Queries** pane, click **Result query**.
15. From the **Date/Source** tab, drag **Sort key** to the **Data Items** pane.

Task 6. Use the Sort key data item to sort the List

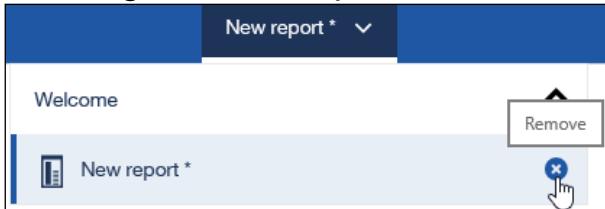
1. With the **Sort key** column selected, in the Properties pane, in the DATA ITEM section, change then **Pre-sort** property to **Sort ascending**.
2. On the side bar, click **Pages**, then click **Page1**.
3. Click the List's container selector , then in the **Properties** pane, under DATA, double-click the **Properties** property.
4. Select the **Sort key** checkbox, then click **OK** to close the dialog.
5. Run the report in **HTML**.

The result appears as follows:

Sales rep/Order method	Revenue
Kolina Nilsson	151,605.59
Henri Bacchus	344,978.59
Johan Mäkinen	747,352.68
Beatriz Couto	905,704.48
Frida Armfelt	1,186,849.91
Akira Hashimoto	1,206,901.22
Wanda Coleman	1,226,558.46
Morela Castro	1,533,562.39
Jake Cartel	1,582,666.1
Frank Jever	1,841,730.57
Special	1,006,100.01
Mail	6,905,730.44
Fax	13,445,559.93
E-mail	23,701,042.57

The sales reps that did not achieve their yearly target in 2012 appear at the top of the Sales rep/Order method column, and the order methods that did not achieve their yearly targets in the same year appear at the bottom of the report.

6. Close the rendered report tab.
7. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



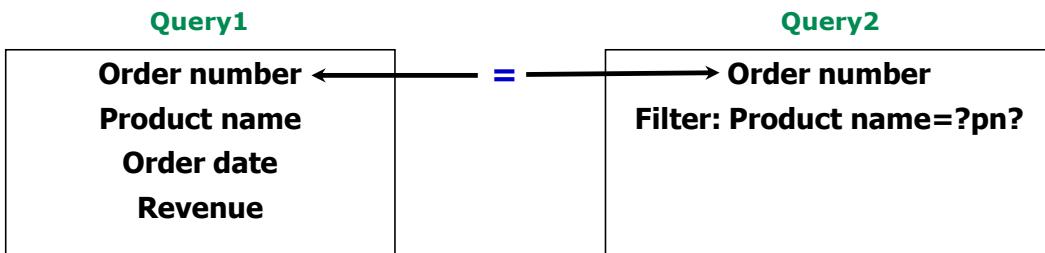
8. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You created a report that shows the sales reps that did not achieve their yearly targets, as well as order methods that did not achieve their yearly targets for the same year. While the yearly target for sales reps is different from that of order methods. You used a Union set operator in addition to a sort key to keep the data from the two queries distinct.

Create join relationships between queries (1 of 2)

- Create join relationships between queries when you need to create a relationship between data in separate queries to answer a specific business question.



Create reports based on query relationships

© Copyright IBM Corporation 2018

Create join relationships between queries

You can create join relationships when working with relational or dimensionally modeled relational data.

If you want to edit the expression used to create a join relationship, you can convert the join relationship to an expression and then make the required changes.

Join objects let you create both inner joins and outer joins between queries.

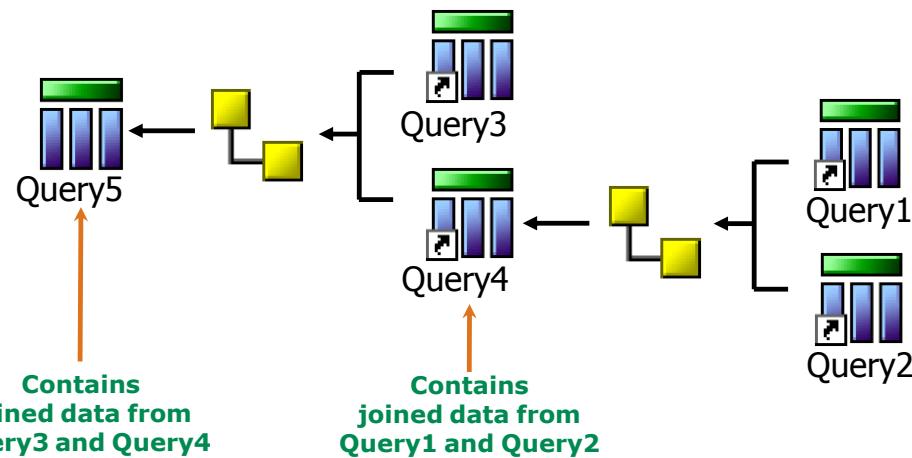
An outer join retrieves all rows returned by an inner join (matching rows) as well as all the rows from one table that do not match any row from the other table. Outer joins can be left, right, or full joins.

You can create a join relationship if you want to create a list report displaying sales target data for each month, but there is no relationship between the tables in which the Month and Sales Target query items are contained in the model you are using.

After creating the join, you can add both the Month and the Sales target query items to the joined query, and can then link the joined query to a List data container and add the two query items to the list.

Create join relationships between queries (2 of 2)

- To retrieve the data you require, you can join a query or create multiple joined queries.



When working with multiple joined queries, it is useful to view the relationships that exist among queries in the report.

Demonstration 2

Analyze product sales by joining two queries

Order number	Date	Product	Quantity	Revenue
100183	Feb 9, 2010	Bear Edge	216	8,402.4
		EverGlow Kerosene	220	6,732
		Star Lite	287	98,791.14
		TrailChef Cook Set	540	28,177.2
		TrailChef Single Flame	477	29,621.7
		TrailChef Utensils	415	7,764.65
100183 - Total			2,155	179,489.09

Demonstration 2:

Analyze product sales by joining two queries

Purpose:

To help the sales department understand purchasing trends, you have been asked to create a report displaying products that were purchased in orders that included a specific product, selected by users at run time. To achieve this result, you will create a join relationship between two queries.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

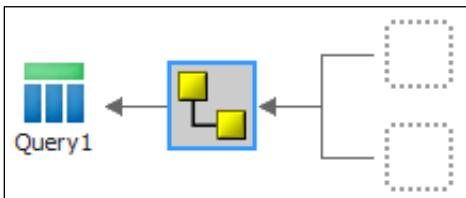
Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Add queries to a blank report

1. Create a new Blank report using the **GO data warehouse (query)** package.
 2. On the side bar, click **Queries** , then click the **Queries**  folder.
 3. From the **Toolbox** in side pane, drag a **Join** object to the work area.
- A Query is created and two drop zones appear to the right of the Join object.



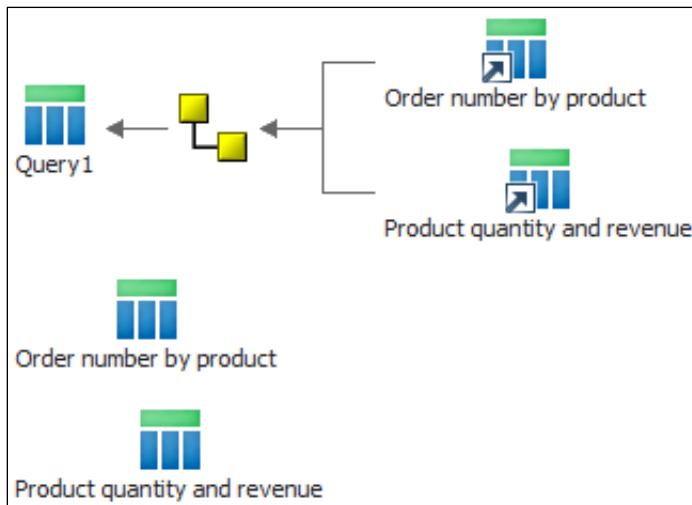
4. From the **Toolbox**, drag a **Query** object to each of the two drop zones. Query2 and Query3 display in the work area and shortcuts to each query appear in the drop zones to the right of the Join object. You will add data to each of these queries and then join the results.

Task 2. Add data to the queries

1. In the work area, double-click **Query2**.
 2. On the **Data\DataSource** tab, expand the **Sales and Marketing (query)** folder and the **Sales (query)** namespace.
 3. From the **Sales order** query subject, drag **Order number** to the **Data Items** pane.
- You want this query to include only orders that contained a specific product selected at run time.
4. From the **Products** query subject, drag **Product** to the **Detail Filters** pane.
 5. Create and validate the following filter expression:

[Sales (query)].[Products].[Product]= ?Product?

6. Click **OK** to close the **Detail filter expression** dialog box.
7. On the Application bar, click **Show properties**  to open the Properties pane.
8. With the Product detail filter selected, in the Properties pane under the GENERAL section, change the **Application** property to **After Auto Aggregation**.
9. In the Queries  folder in the content pane, click **Query2**.
10. In the Properties pane under the MISCELLANEOUS section, change the **Name** of the Query to **Order number by product**.
You will add data to Query3.
11. In the Queries  folder in the content pane, click **Query3**.
12. Add the following data items from the **Data\Source** tab, to the **Data Items** pane:
 - Sales order: **Order number**
 - Time: **Date**
 - Products: **Product**
 - Sales fact: **Quantity, Revenue**
13. In the Queries  folder in the content pane, click **Query3**.
14. In the Properties pane under the MISCELLANEOUS section, change the **Name** of the Query to **Product quantity and revenue**.
15. Click the **Queries**  folder in the content pane.

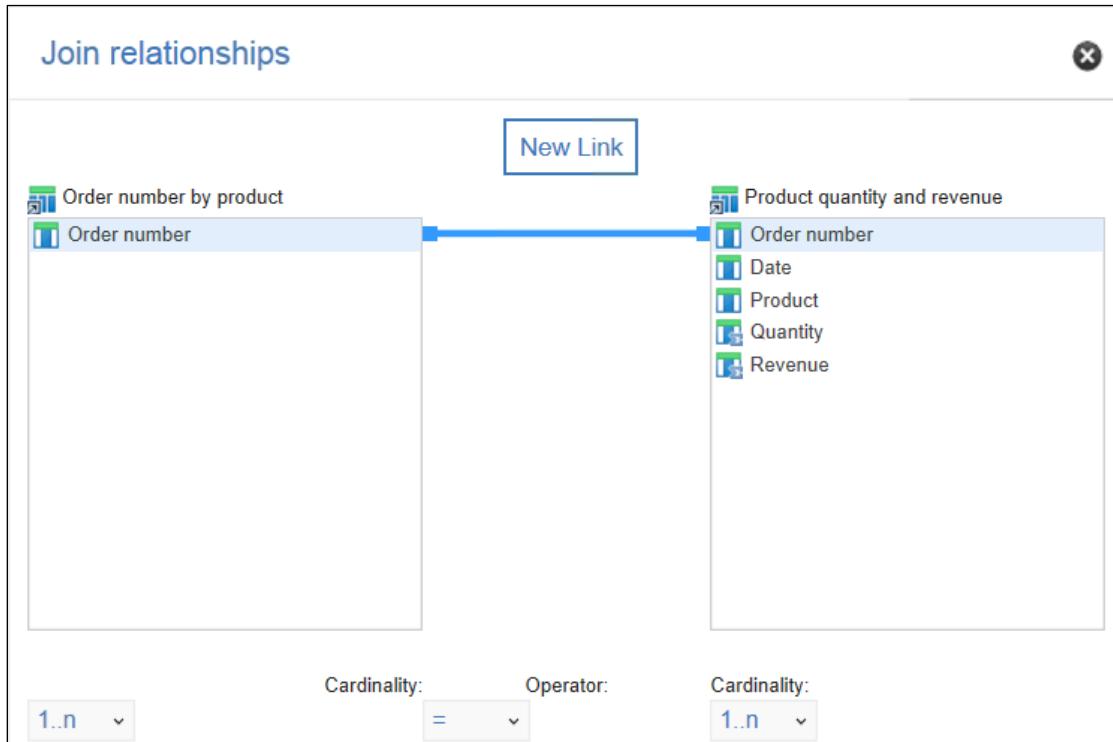


Task 3. Create a join relationship between the two named queries, and then add data items to Query1

1. In the work area, double-click the **Join** object.
2. In the **Join relationships** dialog box, click **New Link**.

You want to specify that the joined query will only contain order numbers that appear in both of the named queries.

3. Ensure that **Order number** is selected in both the **Order number by product** query list and the **Product quantity and revenue** query list, and that the **Cardinality** on the left and the right is **1..n**.



4. Click **OK** to close the **Join relationships** dialog box, and then on the **Queries**, click **Query1**.
5. From the **Data/Data items** tab, click and drag the **Product quantity and revenue** query to the **Data Items** pane.
6. In the Queries folder in the content pane, click **Query1**.
7. In the Properties pane, under the MISCELLANEOUS section, rename the query to **Select product quantity and revenue by order number**.
8. In the Data Items pane, click **Product**, and then on the Properties pane, change the **Pre-sort** property to **Sort ascending**.

Task 4. Add data to the list, and then run the report

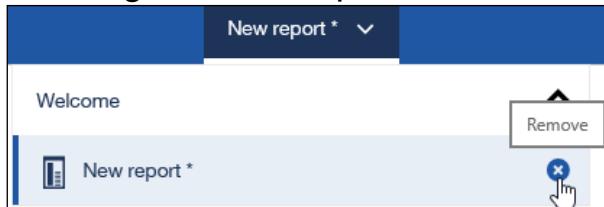
1. Click **Pages** in side bar, then click **Page1**.
2. Open the **Data** tab, then click the **Data Items** tab.
3. Under **Select product quantity and revenue by order number**, click the first item, Shift-click the last item to select all of the data items, and then drag them to the work area.

4. Click the <Order number> List column body, Ctrl-click the <Date> List column body, and then from the Toolbar, click **Group / Ungroup**.
5. Click the <Quantity> List column body, then from the Toolbar, click **Summarize**, then click **Total**.
6. Click the <Revenue> List column body, from the Toolbar, click **Summarize**, and then click **Total**.
7. Run the report in **HTML**.
8. In the **Product** value box, select **Bear Edge**, and then click **OK**.

A section of the result appears like the following:

Order number	Date	Product	Quantity	Revenue
100183	Feb 9, 2010	Bear Edge	216	8,402.4
		EverGlow Kerosene	220	6,732
		Star Lite	287	98,791.14
		TrailChef Cook Set	540	28,177.2
		TrailChef Single Flame	477	29,621.7
		TrailChef Utensils	415	7,764.65
100183 - Total			2,155	179,489.09

9. Click **Page down** and observe that only orders that included the **Bear Edge** product appear in the report.
10. Close the rendered report tab.
11. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



12. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

To help your sales department understand purchasing trends, you created a report displaying products that were purchased in orders that included a specific product, selected by users at run time. You created a join relationship between two queries to ensure that the final report only displayed information for products sold in orders that included the product users selected in the prompt.

Create a report within a report

- You can create a report that delivers information that would otherwise require two or more reports.

Product line	Revenue	Product line Sales Trends
Camping Equipment	\$4,578,321	
Mountaineering Equipment	\$4,343,692	
Personal Accessories	\$3,178,613	

Master list



Detail chart inside a list



Create a report within a report

By using master-detail relationships, you ensure that the detail data container only displays information relevant for the row in which it appears in the master data container.

The master and the detail report containers each have their own separate queries. You must create a relationship between the queries based on a data item that is common to both queries.

The common data item used to create the link does not need to appear in both reports but must be included in both underlying queries.

Demonstration 3

Create a revenue report using a Master-detail relationship

Country	Planned revenue	Revenue	Revenue by Quarter						
			Revenue	Q1	Q2	Q3	Q4	2012 Total	
Australia	41,383,351.34	38,968,802.62	Melbourne	Alice Walter	2,019,726.71	2,119,397.51	2,992,620.71	3,374,957.17	10,506,702.1
				Dave Smythe	2,186,537.08	1,882,231.16	1,800,068.74	1,047,201.83	6,916,038.81
				Donald Ward	2,178,861.5	4,002,379	1,926,622.95	1,293,386.63	9,401,250.08
				Jackie Fulford	2,074,084.88	1,559,967.22	2,133,314.29	2,427,298.86	8,194,665.25
				Jake Cartel	207,467.54	377,394.22	103,522.89	894,281.45	1,582,666.1
				John Sinden		762,028.46	1,208,888.04	396,563.78	2,367,480.28

Create reports based on query relationships

© Copyright IBM Corporation 2018

Demonstration 3: Create a revenue report using a Master-detail relationship

Demonstration 3:

Create a revenue report using a Master-detail relationship

Purpose:

Management wants a report that shows planned revenue and actual revenue for 2012. They would also like to see the generated revenue broken out by quarters for each sales rep in each city. To do this, you will create a master-detail relationship on Country between a list and a crosstab.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Create a List report

1. Create a new **Blank** report using the **GO data warehouse (query)** package.
2. Add a **List** to the work area.
3. From the **Data\DataSource** tab, expand **Sales and Marketing (query)** folder, and then expand **Sales (query)** namespace.
4. Add the following query items to the List:
 - Employee by region: **Country**
 - Sales fact: **Planned revenue, Revenue**

Country	Planned revenue	Revenue
<Country>	<Planned revenue>	<Revenue>

5. Click the **<Country>** List column body, then from the Toolbar, click **Group / Ungroup**.
6. With **Country** still selected, click **Sort**, and then click **Ascending**.
7. From the Toolbar, click **Filters**, and then click **Edit Filters**.
8. Click **Add**, then select **Advanced**, then click **OK**.
9. In the **Available Components** pane, expand the **Sales and Marketing (query)** folder, the **Sales (query)** namespace, and the **Time** query subject.
10. Drag **Year** to the **Expression Definition** pane.
11. Create and validate the following detail filter expression:
[Sales (query)].[Time].[Year]=2012
12. Click **OK** to close the **Detail filter expression** dialog box.
13. Click **OK** to close the **Filters** dialog box.

Task 2. Add a Crosstab to the List

- From the **Toolbox**, drag a **Crosstab** object to the last column of the List.
- In the **Query Name** box, replace the **Query2** text with **Employee revenue**.
- Click **OK** to close the dialog.

A section of the result appears as follows:

- From the **DataSource** tab, add the following query items to the crosstab:
 - Rows:
 - Employee by region: **City**, **Employee name** (nested as a child of **City**)
 - Columns:
 - Time: **Quarter**
 - Measures:
 - Sales fact: **Revenue**
- Click the **<#Quarter#>** column header, then from the Toolbar click **Sort**, and then click **Ascending**.
- Click the **<#Employee name#>** row header, from the Toolbar click **Sort**, and then click **Ascending**.
- On the Application bar, click **Show properties** to open the **Properties** pane.
- Click the **<#Quarter#>** column header, then from the Toolbar click **Summarize**, and then click **Total**.
- Click the **Total** column header, then in the Properties pane, under the **TEXT SOURCE** section, change the following properties:
 - Source Type: **Text**
 - Text: **2012 Total**
- From the Toolbar, click **Filters**, and then click **Edit Filters**.

11. Click **Add**, click **Advanced**, and then click **OK**.
12. Create and validate the following detail filter expression:
[Sales (query)].[Time].[Year]=2012
Hint: Drag Year from the Source tab, under Time\Year.
You have created the same filter as in the List so that the report only shows data from 2012.
13. Click **OK** to close the **Detail filter expression** dialog box.
14. Click **OK** to close the **Filters** dialog box.

Task 3. Create a Master-detail relationship, and name the queries

Although Country is not shown in the crosstab, you need to add it to the query so that a link can be created on Country between both data containers.

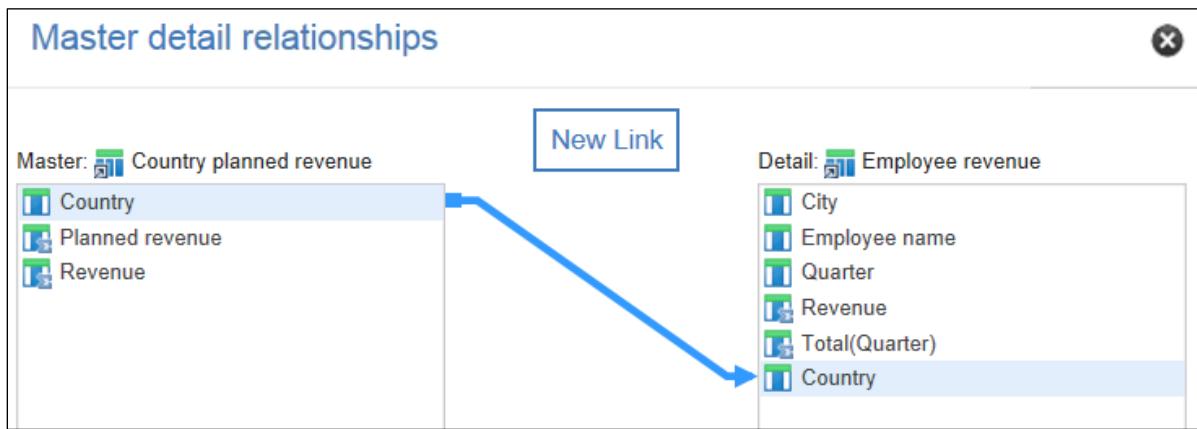
1. Click the **Queries**  tab on the side bar.
2. Click the **Employee revenue** query.
3. From the **Data/Source** tab, under the **Employee by region** query subject, drag **Country** to the **Data Items** pane.

A section of the result appears as follows:

Data Items		Detail Filters
	Employee name	
	City	
	Quarter	
	Revenue	
	Total(Quarter)	
	Country	

4. Click the **Queries**  tab on the side bar.
5. In the **Queries**  folder, click **Query1**.
6. In the Properties pane under the MISCELLANEOUS section, rename the query to **Country planned revenue**.
7. Click the **Pages**  tab in side bar, then click **Page1**.
8. Click anywhere in the Crosstab.
9. In the Toolbar, click **More** , then click **Master Detail Relationships**.
The Master Detail Relationships menu option is only active when you have clicked inside the detail report object.
10. Click **New Link**, and then click **Country** in each query list.

A section of the result appears as follows:



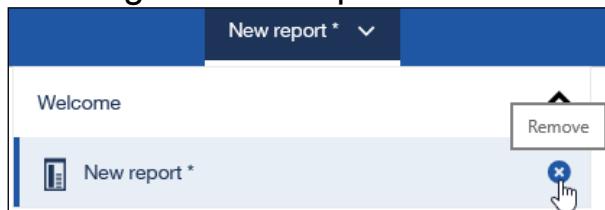
11. Click **OK** to close the Master detail relationships dialog.
12. In the List, double-click the **Crosstab** column header.
13. Change the text to **Revenue by Quarter**, and then click **OK**.
14. Run the report in **HTML**.

A section of the results appears as follows:

Country	Planned revenue	Revenue	Revenue by Quarter						
			Revenue	Q1	Q2	Q3	Q4	2012 Total	
Australia	41,383,351.34	38,968,802.62	Melbourne	Alice Walter	2,019,726.71	2,119,397.51	2,992,620.71	3,374,957.17	10,506,702.1
				Dave Smythe	2,186,537.08	1,882,231.16	1,800,068.74	1,047,201.83	6,916,038.81
				Donald Ward	2,178,861.5	4,002,379	1,926,622.95	1,293,386.63	9,401,250.08
				Jackie Fulford	2,074,084.88	1,559,967.22	2,133,314.29	2,427,298.86	8,194,665.25
				Jake Cartel	207,467.54	377,394.22	103,522.89	894,281.45	1,582,666.1
				John Sinden		762,028.46	1,208,888.04	396,563.78	2,367,480.28

You can see the details of the revenue generated in each quarter, by each sales rep, for each country.

15. Close the rendered report tab.
16. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



17. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You created a revenue report using Master-detail relationships. You linked the master list with the detailed crosstab by using Country, which is a data item common to both queries.

Unit summary

- Create reports by merging query results
- Create reports by joining queries
- Combine data containers based on relationships from different queries

Create reports based on query relationships

© Copyright IBM Corporation 2018

Unit summary

Unit 3 Create advanced dynamic reports

IBM Training



Create advanced dynamic reports

IBM Cognos Analytics (v11.0)

© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM.

Unit objectives

- Filter reports on session parameter values
- Navigate a briefing book using a table of contents
- Create dynamic headers and titles that reflect report data
- Create a customer invoice report

Filter reports using query macros

- A query macro is an expression that is embedded directly in the SQL of a query and dynamically alters the SQL at runtime.

[Sales rep] = #sq(\$account.defaultName)#

This macro gets the name of the user currently logged into IBM Cognos Analytics to filter the data

Filter reports using query macros

Query macros can be included anywhere in an SQL statement as long as the resulting expression is valid. You can add macros to report objects that pass expressions to the query model, such as filters and calculated query items.

Session parameters are made available to you through your authentication source. Depending on your authentication source, different session parameters are available.

Query macros can be used both in native SQL and Cognos SQL.

Examples of different authentication sources include NTLM, LDAP, or Siteminder.

In this example, you create a filter that takes the name of the person who has logged into IBM Cognos Analytics, and displays only that person's results to them. The same result as in this example could be achieved through bursting.

While it is better practice to place macros into the Framework Manager model, they are useful in IBM Cognos Analytics - Reporting, especially where the Framework Manager project cannot be modified or the development-time model is not available.

Demonstration 1

Control report output using a query macro

Sales Information for Bart Scott		
Product line	Product type	Revenue
Camping Equipment	Cooking Gear	3,096,705.62
	Lanterns	1,597,289.18
	Packs	4,521,701.21
	Sleeping Bags	4,412,665.17
	Tents	6,078,370.27
Camping Equipment - Total		19,706,731.45
Golf Equipment	Golf Accessories	753,380.1

Demonstration 1: Control report output using a query macro

Demonstration 1: Control report output using a query macro

Purpose:

Management wants to see a report that outlines the total revenue produced by all product lines for each sales representative. They would like sales representatives in the field to be able to generate this report as needed to view their latest figures, but each rep should be able to see only their own data. You will create a report that includes revenue figures for every sales rep but filters on the current user's identity and displays only data appropriate to them.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Create a report with revenue by sales rep.

1. Create a new **Blank** report using the **GO data warehouse (query)** package.
2. Add a **List** to the work area.
3. From the **Data/Source** tab, expand the **Sales and Marketing (query)** folder and the **Sales (query)** namespace.
4. Add the following query items to the list data container:
 - Employee by region: **Employee name**
 - Products: **Product line**, **Product type**
 - Sales fact: **Revenue**

Employee name	Product line	Product type	Revenue
<Employee name>	<Product line>	<Product type>	<Revenue>

5. Click **<Employee name>**, Ctrl-click **<Product line>** List column body, and then from the Toolbar click **Group / Ungroup**.
 You can use Ctrl-click or Shift-click methods to select multiple columns.
 You want to include a footer for the employee name in your list, and then summarize the revenue column.
6. Click the **<Employee name>** List column body, then on the Toolbar, click **Headers & footers**, and then click **Create footer**.

7. Click the <Revenue> List column body, then on the Toolbar click **Summarize**, and then click **Total**.

A section of the result appears as follows:

Employee name	Product line	Product type	Revenue
<Employee name>	<Product line>	<Product type>	<Revenue>
<Product line> - Total			<Total(Revenue)>
<Product line>	<Product type>		<Revenue>
<Product line> - Total			<Total(Revenue)>
<Employee name>			
<Employee name> - Total			<Total(Revenue)>

8. On the Application bar, click **Run options** , then click **Run HTML**.

A section of the result appears as follows:

Employee name	Product line	Product type	Revenue	
Aaltje Hansen	Personal Accessories	Binoculars	1,346,151.9	
		Eyewear	12,278,798.75	
		Knives	1,835,433.2	
		Navigation	1,119,025	
		Watches	8,419,443.6	
		Personal Accessories - Total	24,998,852.45	
Aaltje Hansen				
Aaltje Hansen - Total			24,998,852.45	

This report displays revenue figures for every available employee who is a sales rep. You will filter on the Employee name to include only data for the user that is currently logged into IBM Cognos Analytics.

9. Close the rendered report tab.

Task 2. Create a filter that uses a query macro.

- Click anywhere on the List data container, then from the Toolbar, click **Filters**, and then click **Edit Filters**.
- Click **Add**, click **Advanced**, and then click **OK**.
- Create and validate the following expression:

[Employee name]=#sq(\$account.defaultName)#.

Hint: Drag **Employee name** from the **Data items**  tab. Drag the **sq()** function from the **Macros**  tab, from the Macro Functions folder. And the **account.defaultName** variable from the Session parameters folder in the same tab.

The pound symbols (#) enclose the query macro expression. The variable **account.defaultName** retrieves the **defaultName** session parameter from the

current IBM Cognos session. The sq expression encloses the variable in single quotes. The expression is case sensitive.

4. Click **OK** to close each open dialog box.
5. On the Application bar, click **Run options** , then click **Run HTML**.
The report displays no data; because you are logged in as a report author (Frank Bretton), not as a sales rep, so you do not see any corresponding sales data. You need to log in to IBM Cognos using one of the sales rep accounts.
6. Close the rendered report tab, and then save the report in the **Team content\B6059** folder as **Sales Rep Revenue**.
7. On the Application bar, click **Frank Bretton** , and then click **Sign out**.
8. Sign in using **smythed/Education1**.
9. On the **IBM Cognos Analytics** portal page, navigate to **Team content\B6059**.
10. Click **Sales Rep Revenue**, to run the report.

A section of the result appears as follows:

Employee name	Product line	Product type	Revenue
Dave Smythe	Camping Equipment	Cooking Gear	1,742,564.12
		Lanterns	820,939.9
		Packs	2,463,562.46
		Sleeping Bags	1,209,711.63
		Tents	4,745,739.82
	Camping Equipment - Total		10,982,517.93
	Golf Equipment	Golf Accessories	147,287

The report now only displays sales data for Dave Smythe.

11. Sign out of the **IBM Cognos Analytics** portal.
Since the report is designed to display data for only one sales rep at a time, there is no need to include the Employee name column. You can now log back in to IBM Cognos Analytics - Reporting as the report author and perform this and other formatting functions.

Task 3. Format the report.

1. On the **Sign in** page, sign in as **brettonf/Education1**.
2. On the **IBM Cognos Analytics** portal page, navigate to **Team content\B6059**, then right-click the **Sales Rep Revenue** report, and then click **Edit report**.
3. In the List data container, click the **<Employee name>** List column body, and then from the Toolbar, click **More**, then click **Cut**.
4. Click the **<Employee name>** summary row (not the text item), and then click **Cut**.

A section of the result appears as follows:

Product line	Product type	Revenue
<Product line>	<Product type>	<Revenue>
<Product line> - Total		<Total(Revenue)>
<Product line>	<Product type>	<Revenue>
<Product line> - Total		<Total(Revenue)>
<Employee name> - Total		<Total(Revenue)>

5. Click the List's **Container Selector**  to select the entire List.
6. On the Application bar, click **Show properties**  to open the Properties pane.
7. From the Properties pane, under DATA, double-click **Properties**.
8. Select **Employee name** to make **Employee name** a property of the Query.
9. Click **OK** to close the dialog.
10. Create report page header with a Text item object.
11. Change the title to **Sales Information for**, press the space bar, and then click **OK**.
12. Click anywhere in the Page, then in the **Toolbar**, click **Select Ancestor**  , and then click **Page**.
13. In the Properties pane, in the DATA section, change the **Query** property to **Query1**.
14. On the **Toolbox**, expand **TEXTUAL**, and then drag a **Layout calculation** object to the end of the report title.
The expression appears as [Query1].[Employee name].
15. In the Report expression dialog box, drag **Employee name** to the Expression Definition pane.
16. Click **Validate**, and then click **OK**.
17. Click to the left of the title to select the header box, and then from the toolbar, click **Left**  to left align the title.
18. Save the report, and then sign out of the **IBM Cognos Analytics** portal.

Task 4. View the report for various sales reps.

1. On the **Sign in** page, sign in as **smythed/Education1**.
2. On the **IBM Cognos Analytics** portal page, navigate to the **Team content/B6059** folder.
3. Click **Sales Rep Revenue** to run the report.

A section of the result appears as follows:

<u>Sales Information for Dave Smythe</u>		
Product line	Product type	Revenue
Camping Equipment	Cooking Gear	1,742,564.12
	Lanterns	820,939.9
	Packs	2,463,562.46
	Sleeping Bags	1,209,711.63
	Tents	4,745,739.82
Camping Equipment - Total		10,982,517.93
Golf Equipment	Golf Accessories	147,287

4. Sign out of **IBM Cognos Analytics**, sign in as **scottb/Education1**, and then run the report again.

A section of the results appears as follows:

<u>Sales Information for Bart Scott</u>		
Product line	Product type	Revenue
Camping Equipment	Cooking Gear	3,096,705.62
	Lanterns	1,597,289.18
	Packs	4,521,701.21
	Sleeping Bags	4,412,665.17
	Tents	6,078,370.27
Camping Equipment - Total		19,706,731.45
Golf Equipment	Golf Accessories	753,380.1

The query changes each time the report is run by a different user.

5. Sign out, and then sign in as **brettonf/Education1**.
 6. Leave the **IBM Cognos Analytics** portal open for the next demonstration.

Results:

You created a report that outlines the total revenue produced by all product lines for each sales rep. You then filtered this report using a query macro to display sales data for the user currently signed into IBM Cognos Analytics. After viewing the report while signed in as a particular sales rep, you formatted the report layout to include the name of the sales rep in the header. Finally, you examined this new report from the perspective of two different sales reps.

Use a table of contents to navigate reports in a report book



Use a table of contents to navigate reports in a report book

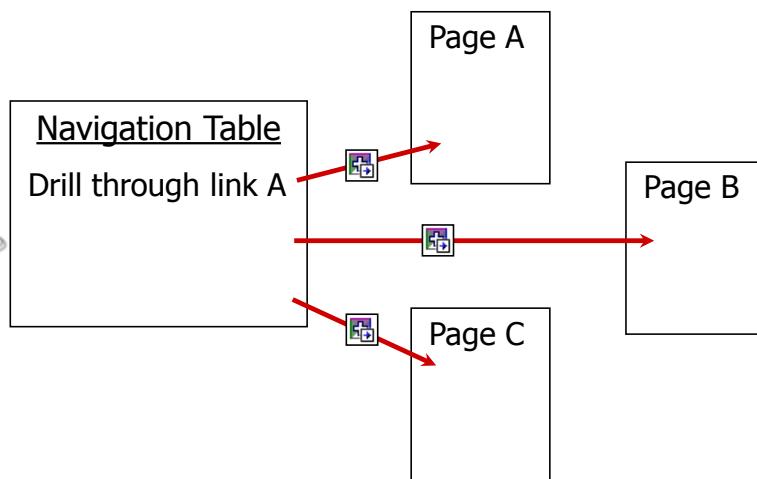
You can create a table of contents to quickly navigate reports with:

- sectioned items
- grouped items
- multiple pages

Set up drill through to different locations in a report

- You can let users navigate to bookmarks in PDF or HTML report outputs.

Use bookmarks to let users navigate to different pages in a report using an interactive navigation table



Set up drill through to different locations in a report

Set up drill-through access to bookmarks to let users navigate:

- to related information within the same report
- from one report to the relevant section of a second report, without filtering the second report

Drill-through access to bookmarks will work only with saved report outputs such as HTML and PDF.

An example of drill-through access to bookmarks is to let users drill through from a list containing images of products to more details about each product contained in another location in the report.

An example of setting up drill-through access for a package to bookmarks in a target report: Your target report displays sales made using different order methods and has a bookmark in the heading where each new order method begins in the report. The source type for this bookmark is data item values. You could set up drill-through access to this report so that data item values are passed to the bookmarks in the target report. If someone drilled through from Fax, for example, they would be taken to the location in the target report where data about Fax orders begins.

Demonstration 2

Create a table of contents for a report book, and add bookmarks for navigation

Table of Contents

1 Quantity By Product Line and Order Method	2
2 Quantity By Product Line and Country	3
2.1 Camping Equipment	3
2.2 Golf Equipment	3
2.3 Mountaineering Equipment	4
2.4 Outdoor Protection	4
2.5 Personal Accessories	5
3 Quantity By Product Line and Retailer Type	6

Demonstration 2: Create a table of contents for a report book, and add bookmarks for navigation

Demonstration 2:

Create a table of contents for a report book, and add bookmarks for navigation

Purpose:

You have been asked to create a report book that contains the following reports: quantity by product line and order method, quantity by product line and country, and quantity by product line and retailer type. This report needs to be distributed in PDF format. You will create a table of contents and add bookmarks to let users navigate quickly to the desired report page and information within the report.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

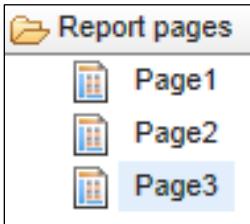
Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Create three List report pages

1. Create a new report using the **GO data warehouse (query)** package and the **1 column** template.
2. Click to the left of the title text to select the header Box.
3. On the Toolbar, click the arrow beside **Horizontal alignment**  , then click **Left** .
4. On the Toolbar, click **Font**  , then set **Family** to **Arial Black**, **Size** to **20pt**, and **Weight** to **Bold**, under Effects, select **Underline**, then click **OK**.
5. Click in the middle cell of the page, then from the Toolbar, click the arrow beside **Vertical alignment**  , then click **Top** .
6. In the middle cell of the page, click **Add**  , then click **List** .
7. In the Object and query name dialog, click **OK** to accept the defaults.
You will copy the first report page to two other pages so that they will have a header, footer and List already on them. You will then add data and a name to each report page to represent what is on the report.
8. On the side bar, click **Pages**  , then click **Report pages** .
9. Ctrl-click and drag **Page1** below **Page1** to create a copy of it.

10. Repeat step 9 so that you have three pages.



11. On the Application bar, click **Show properties** to open the Properties pane.
12. For each page, in the Report pages pane, click each of the pages to select it, and then in the Properties pane, under the MISCELLANEOUS section, change the **Name** property as follows:
 - **Page1: Quantity by Product Line and Order Method**
 - **Page2: Quantity by Product Line and Country**
 - **Page3: Quantity by Product Line and Retailer Type**

Task 2. Populate the List on each page

1. Double-click the **Quantity by Product Line and Order Method** page to open it.
2. Double-click the page header text to change it, then type **Quantity by Product Line and Order Method**.
3. From the **Data/Source** tab, drag the following data items to the List:
 - Products: **Product line**
 - Order method: **Order method type**
 - Sales fact: **Quantity**
4. Click **<Product line>**, then from the Toolbar, click **Group / Ungroup** .

The results appear as follows:

Quantity by Product Line and Order Method		
Product line	Order method type	Quantity
<Product line>	<Order method type>	<Quantity>
<Product line>	<Order method type>	<Quantity>

5. On the side bar, click **Pages** , then click the **Quantity by Product Line and Country** page.
6. Change the page header text to **Quantity by Product Line and Country**.
7. On the **Data/Data items** tab, drag the following query items to the List:
 - **Product line**
 - **Quantity**

8. On the **Data/Source** tab, from the **Retailers** query subject, drag **Retailer country** between the Product line column and the Quantity column.
 9. Click <Product line>, then from the Toolbar, click **Group / Ungroup** .
- The results appear as follows:

Quantity by Product Line and Country		
Product line	Retailer country	Quantity
<Product line>	<Retailer country>	<Quantity>
<Product line>	<Retailer country>	<Quantity>

10. On the side bar, click **Pages** , then click the **Quantity by Product Line and Retailer Type** page.
 11. Change the page header text to **Quantity by Product Line and Retailer Type**.
 12. On the **Data/Data items** tab, drag the following query items to the List:
 - **Product line**
 - **Quantity**
 13. On the **Data/Source** tab, from the **Retailer type** query subject, drag **Retailer type** between the **Product line** column and the **Quantity** column.
 14. Click <Product line>, then from the Toolbar, click **Group / Ungroup** .
- A section of the result appears as follows:

Quantity by Product Line and Retailer Type		
Product line	Retailer type	Quantity
<Product line>	<Retailer type>	<Quantity>
<Product line>	<Retailer type>	<Quantity>

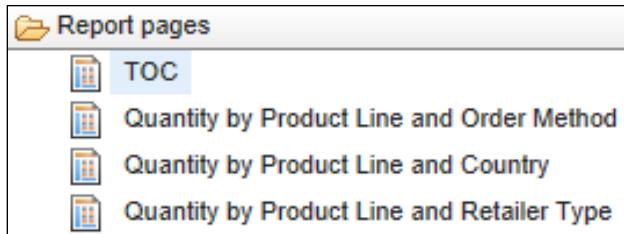
Task 3. Create a table of contents page

You will create a page for the table of contents by copying one of the existing pages, renaming the page and title, and deleting the list object. You will then create a table of contents on the page.

1. On the side bar, click **Pages** , then click **Report pages** .
2. In the Report pages pane, Ctrl-click and drag **Quantity by Product Line and Order Method** to the top of the list, to create a copy.

3. In the Properties pane, under the MISCELLANEOUS section, change the **Name** property to **TOC**.

The Report pages pane displays the four named pages.

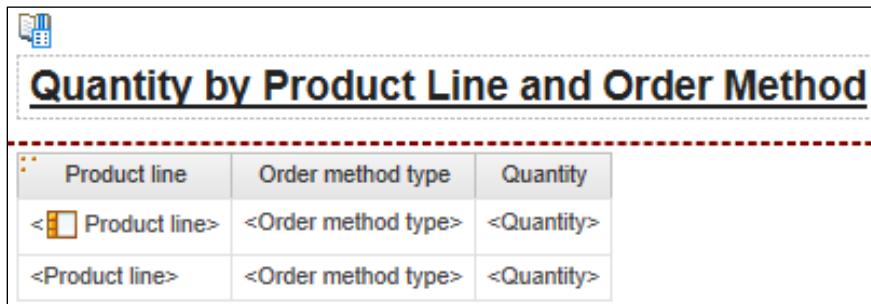


4. In the Report pages pane, double-click the **TOC** page.
5. Click the List's **Container Selector** to select the List, then press **Delete**.
6. Change the page header text to **Table of Contents**.
7. On the side bar, click **Toolbox** , expand **ADVANCED**, and then drag a **Table of contents** object to the work area.

Task 4. Add a Table of Contents Entry to the report pages

1. On the side bar, click **Pages** , then click the **Quantity by Product Line and Order Method** page.
2. From the **Toolbox**, expand **ADVANCED**, and then drag a **Table of contents entry** object to the top left corner of the page, above the title in the header.

The results appear as follows:



3. Repeat steps 1 and 2 to include a **Table of Contents Entry** object on the **Quantity by Product Line and Country** and the **Quantity by Product Line and Retailer Type** pages.
4. On the side bar, click **Pages** , then click **TOC**.

The TOC itself is constructed of TOC entries, each of which is a 3 cell table constructed with text, dashed lines and page numbers.

You have the choice of typing the name of the page, or you can change the text to a report expression to include the PageName. If you type the page name, you have to maintain the page name manually if the pages are re-ordered. If

you use PageName, then the page name will be updated dynamically even if you re-order the page.

You will use the PageName method in the next few steps.

5. Click the first text item, and then in the **Properties** pane under **TEXT SOURCE**, change the **Source type** property from **Text** to **Report expression**.
6. In the Properties pane, under TEXT SOURCE, double-click the **Report expression** property.
7. On the **Functions** tab, expand **Report Functions**, double-click **PageName**, and then click **OK** to close the **Report expression** dialog box.
8. Click the **<%PageName()%>** text that you have just created, Ctrl+click and drag a copy of the function to each row of the table just in front of the the text.
9. Delete **Double-click to edit text** from each row.
10. On the Application bar, click **Run options** , then click **Run PDF**.

Table of Contents works only for reports produced in PDF and non-interactive HTML format (when viewing saved report outputs).

The result appears as follows:

<u>Table of Contents</u>	
Quantity by Product Line and Order Method	2
Quantity by Product Line and Country	3
Quantity by Product Line and Retailer Type	6

Three things to note:

- The PageName now appears in the Table of Contents. This is the page name that you see in Page explorer.
- The page numbers reflect the actual starting page of the item so as more or less data appears in the report, the page numbers will always be correct.
- When you move the mouse cursor over each item, it changes shape to indicate a live link. You can click the link to go to that section of the report.

11. Close the rendered report tab.

Task 5. Add numbering to the Table of Contents

You will format the Table of Contents to have numbers for each entry.

1. On the **TOC** page, from the **Toolbox** tab, expand **TEXTUAL**, and then drag a **Layout calculation** object into the **TOC**, before the first `<%PageName()%>`, as follows:

Table of Contents	
<code><%PageName()%></code>
<code><%PageName()%></code>
<code><%PageName()%></code>

2. In the **Expression Definition** pane, create and validate the following expression:

TOCHeadingCount(1)

Hint: **TOCHeadingCount** is on the Functions tab, in the Report Functions folder. You may also type the function expression, but be careful of syntax.

The value 1 represents the Heading level property found on a Table of contents entry. When you create the Table of contents entry, you can choose which level the entry will be accounted for in the table of contents.

The functionality of the Heading level property will be more clearly understood after a second level of numbering is added to the TOC.

3. Click **OK** to close the **Report expression** dialog box.
4. Click the new layout expression, and then Ctrl+click and drag a copy into each of the rows.
5. With the first `<%PageName()%>` item selected, Ctrl+click the other `<%PageName()%>` items to select multiple items, in the Properties pane, under the **BOX** section, change the **Padding** property to **Left padding of 5 px**, and then click **OK**.
6. On the Application bar, click **Run options** , then click **Run PDF**.

The result appears as follows:

Table of Contents

1 Quantity by Product Line and Order Method	2
2 Quantity by Product Line and Country	3
3 Quantity by Product Line and Retailer Type	6

The TOCHeadingCount () function returns the TOC1 entry count. Since you added three TOC entries, each belonging to Table of Contents 1, there are three entries that display on this page.

7. On the right, click page 3.

You are brought to the Quantity by Product Line and Country report on page 3, as indicated in the Table of Contents.

8. Close the rendered report tab.

Task 6. Add data driven TOC entries to the Table of Contents page

1. On the side bar, click **Pages** , then click the **Quantity by Product Line and Country** page.
2. On the Application bar, under **More**, click **Locked** to unlock the report.
3. On the side bar, click the **Toolbox** , expand **ADVANCED**, and then drag a **Table of contents entry** object into the List, before <Product line>.

The result appears as follows:

Product line	Retailer country	Quantity
 <Product line>	<Retailer country>	<Quantity>
 <Product line>	<Retailer country>	<Quantity>

4. On the Application bar, click **Run options** , then click **Run PDF**.

The table of contents now displays the data values that are in the report.

The results appear as follows:

Table of Contents

1 Quantity by Product Line and Order Method	2
2 Quantity by Product Line and Country	3
Camping Equipment.....	3
Golf Equipment.....	3
Mountaineering Equipment.....	4
Outdoor Protection	4
Personal Accessories	5
8 Quantity by Product Line and Retailer Type	6

The Product line entries do not yet have an explicit number at the start, because you will add a second level of numbering to the table of contents, and indent the Product line entries.

5. Close the rendered report tab.

Task 7. Add a second level of numbering and indent the product line on the Table of Contents page

You will build a second level of numbering based on, (<Level1>.<Level2> <Product line>). For example: 2.1 Camping Equipment.

1. With the report still unlocked, click the **Table of contents entry** beside **<Product line>**.
2. In the Properties pane, under GENERAL, change the **Heading level** property to **2**.
3. On the Application bar, under **More**, click **Unlocked** to lock the report.
4. On the side bar, click **Pages** , then click **TOC**.
5. From the **Toolbox**, expand **TEXTUAL**, and then insert a **Layout calculation** in front of the **<Product line>** entry.
6. In the **Expression Definition** pane, create and validate the following expression:
TOCHeadingCount(2)
7. Click **OK**.
8. From the **Toolbox**, drag a **Text item** in front of the new Layout calculation, type a period (.) and then click **OK** to close the **Text** dialog box.
9. Insert a **Layout calculation** in front of the Text item you just added.
10. In the **Expression Definition** pane, create and validate the following expression:
TOCHeadingCount(1)
11. Click **OK**.
12. With this last layout calculation selected (**<%TOCHeadingCo...%>**), in the Properties pane, under the **BOX** section, set the **Left padding** property to **20px**, and then click **OK**.

13. Click <Product line>, and then set **Left padding** to **5px**.

A section of the report layout appears as follows:

```
<%TOCHeadingCo...%> <%PageName ()%>
<%TOCHeadingCo...%> <%PageName ()%>
<%TOCHeadingCo...%> <%TOCHeadingCo...%> <Product line>
<%TOCHeadingCo...%> <%PageName ()%>
```

14. On the Application bar, click **Run options** , then click **Run PDF**.

The result appears as follows:

<u>Table of Contents</u>		
1	Quantity by Product Line and Order Method	2
2	Quantity by Product Line and Country	3
2.1	Camping Equipment	3
2.2	Golf Equipment	3
2.3	Mountaineering Equipment	4
2.4	Outdoor Protection	4
2.5	Personal Accessories	5
3	Quantity by Product Line and Retailer Type	6

The second level of numbering appears and the entries are indented.

15. Close the rendered report tab.

Task 8. Add a Top link and a bookmark for better navigation

You want to create a text item at the bottom of the list that will have a drill-through definition for users to return to a bookmark the beginning of the list.

1. On the side bar, click **Pages** , then click **Quantity by Product Line and Order Method**.
 2. From the **Toolbox**, drag a **Text item** outside of the List, to the right of the List.
 3. In the **Text** dialog box, type **Top**, and then click **OK**.
 4. Click the **Top** text item, from the on demand toolbar, under **More**, click **Drill-Through Definitions**.
 5. Click **New Drill-Through Definition** , and then click the **Bookmark** tab.
 6. In the **Source type** list, select **Text**, and then to the right of the **Text** box, click the ellipsis.
 7. Type **Top of Product Line and Order Method**, click **OK** to close the **Text** dialog box, and then click **OK** to close the **Drill-through definitions** dialog box.
- Hint: If you copy the text from the Text dialog box while it is open, you can paste it instead of typing it again at step 9.

The bookmark that you referenced in the drill-through definition does not yet exist. You will create the bookmark of the top of the report.

8. On the **Toolbox**, under **ADVANCED**, drag a **Bookmark** to the right of the **Table of contents entry** object, in the header.
9. With the Bookmark item selected, in the Properties pane, under the **BOOKMARK SOURCE** section, change the **Label** property to **Top of Product Line and Order Method**.
10. Click **OK**.

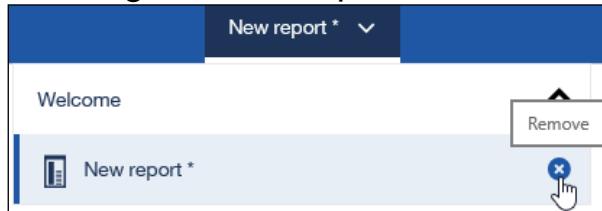
A section of the report layout appears as follows:

The screenshot shows a portion of a report layout. At the top left is a 'Bookmark' icon. To its right is the title 'Quantity by Product Line and Order Method'. Below the title is a table of contents entry. To the right of the entry is a blue callout bubble containing the word 'Top' with a small plus sign icon. The table of contents entry has three columns: 'Product line', 'Order method type', and 'Quantity'. Under each column heading is a yellow icon. Below each row is a placeholder '<Product line>'.

Product line	Order method type	Quantity
<Product line>	<Order method type>	<Quantity>
<Product line>	<Order method type>	<Quantity>

Optionally, repeat steps 1 through 10 on the pages that you would like to have this Top navigation functionality, changing the bookmark and drill-through definitions to have unique names, such as Top of Product Line and Country and Top of Product Line and Retailer Type.

11. On the Application bar, click **Run options** , then click **Run PDF**.
12. Navigate through the report using the table of contents and using bookmark navigation to return to the top of a page.
13. Close the rendered report, then remove the report without saving.
14. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



15. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You created a report book with a table of contents and bookmarks for users to quickly navigate to the desired report. You added numbering and data driven table of content entries that used a second level of numbering.

Create dynamic data items and titles

- Use calculated data items to:
 - create dynamic column titles and report titles
 - display user prompt selections

Dynamic titles set by a prompt

Summary of 2012 Product Sales and Growth		
Product	2012 Revenue	Growth
Aloe Relief	\$101,812.41	-43.77%
Astro Pilot	\$13,072,475.00	139.40%
Auto Pilot	\$3,388,230.00	*** no prior data ***
Bear Edge	\$3,960,860.01	46.57%
Bear Survival Edge	\$3,320,876.50	59.34%

Dynamic calculation

Create dynamic data items and titles

To create a dynamic column header:

1. Create a calculated data item in the query that will display the title you want.
2. Change the column title Source type property to Data item value.
3. Change the column title Data item value property to the calculated item.

For report titles, follow the same steps, but set the Source type property to Report Expression (not Data Item), and set the value of the report expression to the calculated item.

By default, a List column title displays the name (label) of the data item it is associated with. In the Properties pane, you can change the Content property of the List Column Title to Value, and the column title will display the first data value retrieved by the data item. In crosstabs, column and row titles display the data values returned by the data item associated with it. For both lists and crosstabs, you can associate the column or row title with a different data item (and also change the content to Label or Value) to display different information in the title. You can associate the title with any data item in the query.

Demonstration 3

Create a dynamic revenue growth report

Summary of 2012 Product Sales and Growth

Product	2012 Revenue	Growth
Aloe Relief	\$101,812.41	-43.77%
Astro Pilot	\$13,072,475.00	139.40%
Auto Pilot	\$3,388,230.00	*** no prior data ***
Bear Edge	\$3,960,860.01	46.57%
Bear Survival Edge	\$3,320,876.50	59.34%

Demonstration 3: Create a dynamic revenue growth report

Demonstration 3:

Create a dynamic revenue growth report

Purpose:

Management wants to see a report that outlines the total revenue generated for each product in any given year. The report should also show the percentage growth in revenue between the year of interest and the prior year. To do this, you will use calculated data items to retrieve and derive data based on the year chosen by the user, and to name the column headers and report title accordingly.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Create a Query

You will create a query containing the date you require and a parameter that users can use to select an order year at runtime.

1. Create a new **Blank** report using the **GO data warehouse (query)** package.
2. Add a **List** to the work area.
3. On the side bar, click **Queries** , then click **Query1**.
4. Add the following data item to the **Data Items pane**: Products: **Product**. You will add a parameter for users to select an order year to view data when they run the report.
5. From the **Toolbox**, drag a **Data Item** to the **Data Items** pane.
6. In the Data item expression dialog box, update the **Name** field to **Order year_prompt**.
7. In the **Expression Definition** pane, create and validate the following expression:
?Order year?
8. Click **OK** to close the Data item expression dialog box. You will add a data item that can be used to return only data for the year that the user selects in the prompt.
9. From the **Toolbox**, drag a **Data Item** to the Data Items pane.
10. In the **Data item expression** dialog box, update the **Name** field to **Y2 Revenue**.

11. In the Expression Definition pane, create and validate the following expression:
if ([Sales (query)].[Time].[Year]=[Order year_prompt]) then ([Sales (query)].[Sales fact].[Revenue]) else (0)

Hints:

- **Source tab: Time: Year**
- **Data items tab: Order year_prompt**
- **Source tab: Sales fact: Revenue**

This data item will retrieve every record from the Time dimension query subject and examine the year of each one. If the year is equal to that specified in the prompt, then the Revenue data is retrieved and aggregated, otherwise the number 0 is added to the running Revenue total.

12. Click **OK** to close the Data item expression dialog box.

Task 2. Add additional data to the query

You will add a second data item to retrieve data for revenue of the previous year.

1. In the **Data Items** pane, double-click the **Y2 Revenue** data item, copy the expression in the **Expression Definition** pane, and then click **OK**.
2. From the **Toolbox** tab drag a **Data Item** to the **Data Items** pane.
3. In the **Data item expression** dialog box, in the **Name** field, type **Y1 Revenue**.
4. In the **Expression Definition** pane, create and validate the following expression:

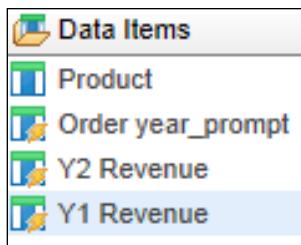
if ([Sales (query)].[Time].[Year]=[Order year_prompt]-1) then ([Sales (query)].[Sales fact].[Revenue]) else (0)

Hint: At the end of **[Order year_prompt]**, type **-1**.

This expression is the same as the one you created earlier except it will retrieve revenue for the previous order year.

5. Click **OK** to close the **Data item expression** dialog box.

Four data items are currently in the query.



Task 3. Add calculated items and a filter to the query

You will create a calculated item for the percentage growth over the previous year.

1. From the **Toolbox**, drag a **Data Item** object to the **Data Items** pane.
2. In the **Data item expression** dialog box, in the **Name** field, type **Growth**.
3. In the **Expression Definition** pane, create and validate the following expression:

([Y2 Revenue]-[Y1 Revenue])/[Y1 Revenue]

Hint: Use the Data items tab for the data items in the expression.

This data item uses the aggregated Revenue totals calculated by the query to determine the percentage change in total revenue between the year specified in the prompt and the previous year.

4. Click **OK** to close the **Data item expression** dialog box.
5. From the **Data/Source** tab, from the **Time** query subject, drag **Year** to the **Detail Filters** pane, and then create the following expression:

[Sales (query)].[Time].[Year] between ?Order year?-1 and ?Order year?

This filter is not necessary for this report to function properly, but it makes the query more efficient. By having the filter, the IF-THEN-ELSE statements on Y1 Revenue and Y2 Revenue will not have to evaluate all four years in the database. Instead, it will only have to evaluate one year.

6. Click **Validate**, and then click **OK**.

The work area appears as follows:

Data Items	Detail Filters
Product	
Order year_prompt	
Y2 Revenue	
Y1 Revenue	
Growth	
	[Sales (query)].[Time].[Year] between ?Order ye...

Task 4. Build the report layout

1. On side bar, click **Pages** , then click **Page1**.
 2. On the **Data/Data items** tab, click **Product**, Ctrl-click **Y2 Revenue** and **Growth**, and then drag the selected items to the List.
 3. Click the **<Product>** List column body, then from the Toolbar, click **Sort**, then click **Ascending**.
 4. On the Application bar, click **Show properties**  to open the Properties pane.
 5. Click the **<Growth>** List column body, and then in the Properties pane, under the DATA section, double-click the **Data format** property.
- The Data format dialog box appears.
6. In the **Format type** list, select **Percent**, and then under Properties, set **Number of Decimal Places** to **2**.

7. In the dialog box, scroll down and change the **Missing value characters** property to ***** no prior data *****.
8. Click **OK** to close the **Data format** dialog box.
9. Click the **<Y2 Revenue>** list column body, and then in the Properties pane, under the DATA section, double-click the **Data format** property.
10. In the **Format type** list, select **Currency**, and then in the **Properties** pane, change the **Currency** property to **\$ (USD) - United States of America, dollar**.
11. On the Application bar, click **Run options** , then click **Run HTML**.
12. In the Year prompt box, type **2012**, and then click **OK**.

A section of the result appears as follows:

Product	Y2 Revenue	Growth
Aloe Relief	\$101,812.41	-43.77%
Astro Pilot	\$13,072,475.00	139.40%
Auto Pilot	\$3,388,230.00	*** no prior data ***
Bear Edge	\$3,960,860.01	46.57%
Bear Survival Edge	\$3,320,876.50	59.34%

Since the report displays data for only the year chosen in the prompt, you will add a dynamic column header and report title that will always reflect the data being displayed.

13. Close the rendered report tab.

Task 5. Create a dynamic column header.

1. In the list data container, click the **Y2 Revenue** column header, and then in the **Properties** pane, under the **TEXT SOURCE** section, change the **Source type** property to **Report expression**.
2. In the Properties pane, under TEXT SOURCE, double-click the **Report expression** property, and create and validate the following expression:
ParamDisplayValue('Order year')
Hint: Drag Order year for the Parameters tab, and then validate with 2012.
3. Click **OK** to close the **Report Expression** dialog box.

In the Properties pane, the TEXT SOURCE section appears as follows:

TEXT SOURCE	
Source type	Report expression
Report expression	ParamDisplayValue('Order year')

4. On the Application bar, click **More** , then click **Locked**  to unlock the report.
5. From the **Toolbox**, drag a **Text item** to the end of the **Y2 Revenue** column header, press the spacebar, type **Revenue**, and then click **OK**.

6. On the Application bar, click **More** , then click **Unlocked**  to lock the report.
7. On the Application bar, under **More**, click **Unlocked** to lock the report
8. On the Application bar, click **Run options** , then click **Run HTML**.
9. In the Year prompt box type **2012**, and then click **OK**.

A section of the report appears as follows:

Product	2012 Revenue	Growth
Aloe Relief	\$101,812.41	-43.77%
Astro Pilot	\$13,072,475.00	139.40%
Auto Pilot	\$3,388,230.00	*** no prior data ***

The title for the revenue column now reflects the data in the report. You will now create a dynamic title for the report that also references the selected order year.

10. Close the rendered report tab.

Task 6. Create a dynamic report title

1. Create page header and Text object.

In the title type **Summary of**, press the spacebar, and then click **OK** to close the **Text** dialog box.

2. From the **Toolbox** tab, under **TEXTUAL**, drag a **Layout calculation** object to the end of the report title.
3. In the Available Components pane, on the **Parameters** tab, drag the **Order year** parameter to the **Expression Definition** pane.

The expression appears as follows:

ParamDisplayValue('Order year')

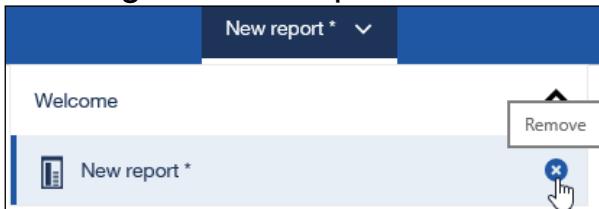
4. Click **OK** to close the **Report expression** dialog box.
5. From the **Toolbox**, drag a **Text item** object to the end of the report title, press the spacebar, type **Product Sales and Growth**, and then click **OK**.
6. In the title, click **Summary of**, and then on the toolbar, click **Pick up Style**.
7. Click the layout calculation, and then from the on demand toolbar, click **Apply Style**.
8. Click **Product Sales and Growth**, and then on the Toolbar, click **Apply Style**.
9. Click the title text block to select it, and then on the Toolbar, click **Left**.
10. On the Application bar, click **Run options** , then click **Run HTML**.

11. When prompted for a year, type **2012**, and then click **OK**.
 A section of the report appears as follows:

Product	2012 Revenue	Growth
Aloe Relief	\$101,812.41	-43.77%
Astro Pilot	\$13,072,475.00	139.40%
Auto Pilot	\$3,388,230.00	*** no prior data ***
Bear Edge	\$3,960,860.01	46.57%
Bear Survival Edge	\$3,320,876.50	59.34%

Now both the revenue column header and the report title reflect the data in the report, based on the year specified in the prompt.

12. Close the rendered report tab.
 13. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



14. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You created a revenue and growth report that uses calculated data items to retrieve data based on the year selected in the prompt. You then formatted a column title and the report title using data items that incorporated the order year specified by the user.

Unit summary

- Filter reports on session parameter values
- Navigate a briefing book using a table of contents
- Create dynamic headers and titles that reflect report data
- Create a customer invoice report

Unit summary

Design effective prompts

IBM Training



Design effective prompts

IBM Cognos Analytics (v11.0)

© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM.

Unit objectives

- Control report displays using prompts
- Specify conditional formatting values using prompts
- Specify conditional rendering of objects based on prompt selection
- Create sorted and filtered reports based on prompt selection

Let users specify criteria used to highlight exceptional data

- You can let users specify the criteria used to conditionally highlight report data by applying conditional formatting that uses prompted values.



Use this calculated data item to conditionally format data in a report depending on the Low gross profit parameter value a user specifies at run time.

Let users specify criteria used to highlight exceptional data

By setting up conditional formatting using prompted values, you can create a report that does not need to be modified when company benchmark values change.

If you do not want the calculated data item to appear in the report, you can either:

- add the calculated data item to the report to test conditional formatting, and then cut the calculated data item from the report layout, or
- add the calculated data item to the query in Query explorer but not to the report layout

When applying conditional formatting, you cannot use parameters directly in a variable expression, but you can reference data items that use parameters.

Demonstration 1

Create a prompt that lets users select conditional formatting values

Country	Product line	Product type	Revenue	Gross profit	LowMedHigh
Australia	Camping Equipment	Cooking Gear	7,464,381.59	2,938,400.28	2
		Lanterns	3,416,036.14	1,495,901.15	2
		Packs	9,063,575.38	3,590,975.57	3
		Sleeping Bags	8,027,884.16	3,217,019.11	3
		Tents	13,964,054.92	4,481,882.33	3
	Golf Equipment	Golf Accessories	1,318,528.37	805,825.44	2
		Irons	6,839,374.25	3,270,992.55	3
		Putters	2,401,092.95	1,150,708.83	2
		Woods	8,520,560.86	4,181,044.42	3
	Mountaineering Equipment	Climbing Accessories	2,646,717.89	1,356,587.33	2
		Rope	4,039,366.36	1,274,262.53	2
		Safety	2,637,772.5	983,979.46	2
		Tools	4,609,263.31	1,930,313.75	2
	Outdoor Protection	First Aid	169,911.38	91,347.2	1
		Insect Repellents	559,867.46	371,574.87	1
		Sunscreen	410,564.72	245,984.02	1
	Personal Accessories	Binoculars	2,214,888.28	756,112.83	2
		Eyewear	2,132,100.87	1,130,134.9	2
		Knives	2,760,632.22	893,852.34	2
		Navigation	3,118,074.7	1,230,319.6	2

Demonstration 1: Create a prompt that lets users select conditional formatting values

Demonstration 1:

Create a prompt that lets users select conditional formatting values

Purpose:

Management has requested a report that lets them identify the total revenue and gross profit generated in every country by each product type. They want to be able to customize this report by highlighting high and low gross profit values based on changing benchmark levels. You will create a report that formats gross profit to appear in different colors if gross profit is exceptionally low or exceptionally high. This formatting will be based on high and low gross profit values users select using prompts at run time.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Create a report with product sales information

1. Create a new **Blank** report using the **GO data warehouse (query)** package.
2. Add a **List** to the work area.
3. From the **Data/Source** tab, add the following query items to the new list data container.
 - Employee by region: **Country**
 - Products: **Product line**, **Product type**
 - Sales fact: **Revenue**, **Gross profit**

Country	Product line	Product type	Revenue	Gross profit
<Country>	<Product line>	<Product type>	<Revenue>	<Gross profit>
<Country>	<Product line>	<Product type>	<Revenue>	<Gross profit>
<Country>	<Product line>	<Product type>	<Revenue>	<Gross profit>

4. Click the **<Country>** List column body, Ctrl-click the **<Product line>** List column body, then on the Toolbar click **Group / Ungroup**.
5. Click the **<Product type>** List column body, then on the Toolbar, click **Sort**, and then click **Ascending**.

Task 2. Add a calculated data item with an expression that

uses parameter values, and then run the report

You will add a calculated data item that compares the gross profit values to those selected in two prompts: Low gross profit and High gross profit. This calculated data item will then assign a value: 1 for gross profit less than the Low gross profit value, 2 for gross profit values greater than the Low gross profit value but less than the High gross profit value, and 3 for gross profit greater than the High gross profit value.

- From the **Toolbox** tab, expand **TEXTUAL**, and then drag a **Query calculation** object to the end of the List.
- In the **Data item expression** dialog box, in the **Name** field, type **LowMedHigh**.
- Create and validate the following expression:

if ([Gross profit] < ?Low gross profit?) then (1) else if ([Gross profit] < ?High gross profit?) then (2) else (3)

Hint: Data Items tab: Gross profit.

Validate using:

- Low gross profit: **600000** (600,000)
- High gross profit: **3000000** (3 million)

This calculation retrieves the prompt values the user specifies at run time and assigns a value of 1, 2, or 3 to each row depending on their gross profit values.

- Click **OK** to close any open dialog box.
- On the Application bar, click **Run options** , then click **Run HTML**.
- When prompted for values, ensure that **Low gross profit** is **600000**, **High gross profit** is **3000000**, and then click **OK**.

The LowMedHigh column displays a value for each row based on the values that you supplied using the prompts. Product types generating below \$600,000 have a value of 1, product types generating between \$600,000 and \$3 million have a value of 2, and product types generating over \$3 million in gross profit have a value of 3.

A section of the result appears as follows:

Country	Product line	Product type	Revenue	Gross profit	LowMedHigh
Australia	Camping Equipment	Cooking Gear	7,464,381.59	2,938,400.28	2
		Lanterns	3,416,036.14	1,495,901.15	2
		Packs	9,063,575.38	3,590,975.57	3
		Sleeping Bags	8,027,884.16	3,217,019.11	3
		Tents	13,964,054.92	4,481,882.33	3

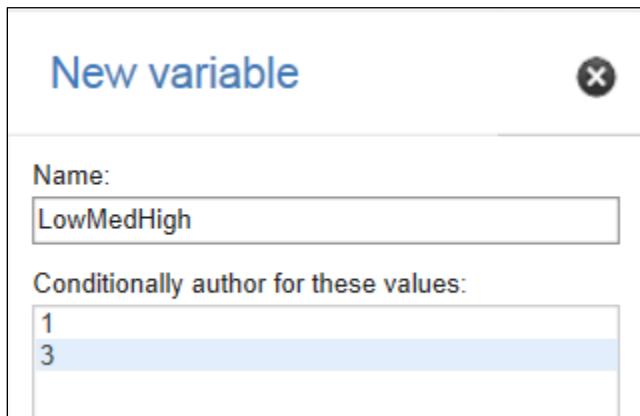
- Close the rendered report tab.

Task 3. Create a variable based on the calculation

You will create a string variable that you can use to conditionally format the Gross profit column of the report. You will use the LowMedHigh query calculation to specify the conditions for this variable.

1. On the Application bar, click **Show properties**  to open the Properties pane.
2. Click the **<Gross profit>** List column body, in the Properties pane, under the CONDITIONAL section, double-click the **Style variable** property, and then in the **Variable** list, select **<New string variable>**.
3. In the **Name** box, type **LowMedHigh**, and then click **Add**.
Since you want to apply conditional formatting only to high and low gross profit values, you will add two values (1 and 3) for which you will be able to apply conditional formatting.
4. In the **Add** box, type **1**, and then click **OK**.
5. Repeat the previous step to add **3**, and then click **OK**.

A section of the result appears as follows:



6. Click **OK** to close the **New variable** dialog box.
7. In the **Report expression** dialog box, on the **Queries** tab, drag the **LowMedHigh** item to the **Expression Definition** pane.
8. Click **Validate**, and then after the expression is validated without any errors, click **OK** to close any open dialog box.

Task 4. Apply conditional formatting based on the LowMedHigh variable, and then run the report

1. With the **<Gross profit>** List column body still selected, click **Queries** on side pane, and then click the **Condition explorer**  tab, and then click **1**.
You want low gross profit values to have a red background, white foreground, and bold weight font.

2. From the Toolbar, click **Background color options**.
3. Click **Red** color.
4. On the **Conditional explorer** pane, verify that **1** is still selected.
5. From the Toolbar, click **Font**.
6. Click **Foreground color**, click **White**, and then click **OK**.
7. Under **Weight**, click **Bold**, and then click **OK**.

You want high gross profit values to have a green background, of a different color tone value than the red, in case the report is printed on a non-color printer.
8. With the **<Gross profit>** List column body still selected, on the **Condition explorer**, click **3**.
9. From the Toolbar, click **Background color options**.
10. Click the **Custom color** tab, type **77FF00** into the **#RGB** box, then click **Apply**.
11. On the **Conditional explorer** pane, verify that **3** is still selected.
12. On the Properties pane, under the **FONT & TEXT** section, double-click the **Font** property, under **Weight** click **Bold**, and then click **OK**.
13. On the **Conditional explorer** tab, click **LowMedHigh** to clear the conditional tool.
14. On the Application bar, click **Run options** , then click **Run HTML**.
15. Ensure that **High gross profit** prompt value is **3000000**, **Low gross profit** prompt value is **600000**, and then click **OK**.

The results appear as follows:

Country	Product line	Product type	Revenue	Gross profit	LowMedHigh
Australia	Camping Equipment	Cooking Gear	7,464,381.59	2,938,400.28	2
		Lanterns	3,416,036.14	1,495,901.15	2
		Packs	9,063,575.38	3,590,975.57	3
		Sleeping Bags	8,027,884.16	3,217,019.11	3
		Tents	13,964,054.92	4,481,882.33	3
	Golf Equipment	Golf Accessories	1,318,528.37	805,825.44	2
		Irons	6,839,374.25	3,270,992.55	3
		Putters	2,401,092.95	1,150,708.83	2
		Woods	8,520,560.86	4,181,044.42	3
	Mountaineering Equipment	Climbing Accessories	2,646,717.89	1,356,587.33	2
		Rope	4,039,366.36	1,274,262.53	2
		Safety	2,637,772.5	983,979.46	2
		Tools	4,609,263.31	1,930,313.75	2
	Outdoor Protection	First Aid	169,911.38	91,347.2	1
		Insect Repellents	559,867.46	371,574.87	1
		Sunscreen	410,564.72	245,984.02	1
	Personal Accessories	Binoculars	2,214,888.28	756,112.83	2
		Eyewear	2,132,100.87	1,130,134.9	2
		Knives	2,760,632.22	893,852.34	2
		Navigation	3,118,074.7	1,230,319.6	2

In the Gross profit column, values of less than six-hundred thousand dollars are highlighted in red, while values of greater than three million dollars are highlighted in green.

16. Close the rendered report tab.

Task 5. Cut the calculated column, and then run the report

You do not want to see the LowMedHigh column in the list report, as the conditional formatting has been applied in another column.

1. On the side bar, click **Pages** , then click **Page1**.
2. Click the **<LowMedHigh>** List column body, then from the Toolbar, under **More**, click **Cut**.

To let the string variable use the LowMedHigh data item values for each row in the report, you need to make the LowMedHigh data item a property of the List object.

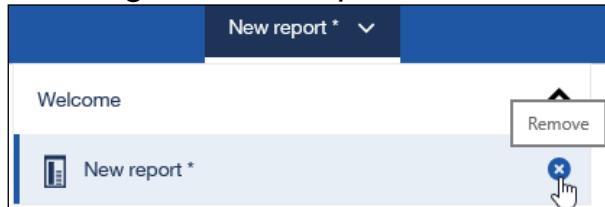
3. Click the List's **Container Selector**  to select the entire List.
4. In the Properties pane, under the DATA section, double-click the **Properties** option, click the **LowMedHigh** check box to select it, and then click **OK**.
5. On the Application bar, click **Run options** , then click **Run HTML**.

6. Ensure that the **Low gross profit** prompt value is **600000**, the **High gross profit** prompt value is **3000000**, and then click **OK**.
A section of the result appears as follows:

Country	Product line	Product type	Revenue	Gross profit
Australia	Camping Equipment	Cooking Gear	7,464,381.59	2,938,400.28
		Lanterns	3,416,036.14	1,495,901.15
		Packs	9,063,575.38	3,590,975.57
		Sleeping Bags	8,027,884.16	3,217,019.11
		Tents	13,964,054.92	4,481,882.33
	Golf Equipment	Golf Accessories	1,318,528.37	805,825.44
		Irons	6,839,374.25	3,270,992.55
		Putters	2,401,092.95	1,150,708.83
		Woods	8,520,560.86	4,181,044.42
	Mountaineering Equipment	Climbing Accessories	2,646,717.89	1,356,587.33
		Rope	4,039,366.36	1,274,262.53
		Safety	2,637,772.5	983,979.46
		Tools	4,609,263.31	1,930,313.75
	Outdoor Protection	First Aid	169,911.38	91,347.2
		Insect Repellents	559,867.46	371,574.87
		Sunscreen	410,564.72	245,984.02
	Personal Accessories	Binoculars	2,214,888.28	756,112.83
		Eyewear	2,132,100.87	1,130,134.9
		Knives	2,760,632.22	893,852.34
		Navigation	3,118,074.7	1,230,319.6

The conditional formatting is now applied to the Gross profit column, even though the LowMedHigh column no longer appears in the report.

7. Close the rendered report tab.
8. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



9. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You created a report that identifies the total revenue and gross profit generated in every country by each product type. You added prompts to let users customize this report by selecting high and low gross profit values to highlight data.

Let users choose how to filter report data

Query Macro Prompt

#prompt ('choose date', 'token', '[Order date] BETWEEN 2012-01-01 00:00:00 AND 2012-12-31 23:59:59')#

Static Choices for Query Macro Prompt

Use	Display
[Order date] BETWEEN 2012-01-01 00:00:00 AND 2012-01-01 23:59:59	All orders in 2012
Cast([Order date]),date)=2012-09-30	Last day 3 rd quarter
[Order date] in_range ?date_range?	Date range

Let users choose how to filter report data

A macro is a fragment of code that you can insert into the Select statement of a query or into an expression. Query macro prompts return different values to a query depending on the prompt option users select.

In query macro prompts, everything between the # signs is replaced with the Use value returned when a user selects a prompt option. You can use query macro prompts when working with relational data or dimensionally-modeled relational data.

Static choices present the user with prompt options that are not found in the data source. Refer to *IBM Cognos Analytics Version 11.0.0 IBM Cognos Analytics - Reporting User Guide: Using Query Macros*, for more information on using query macro prompts.

It is important to know your database when creating the Use values. In the Let users choose how to filter report data slide example, the second static choice casts Order date as a 'date' data type because in the database, the Order date item is stored as a Date & Time data type. If you do not cast it as 'date', you must specify the exact date and time of the order in the static choice.

The Name parameter, which is mandatory, specifies the name of the query macro prompt.

Be sure to use straight quotes, you may need to turn off Microsoft Word's Smart quotes.

Demonstration 2

Create a prompt to let users choose between different filters

Date	Order number	Retailer name	Revenue
May 8, 2012	103576	Great Adventures	185,785.19
May 8, 2012	103581	Todo para el Golf, S.A. de C.V.	318,307.36
May 8, 2012	103601	The Marketplace	237,845.26
May 8, 2012	103605	Tamarack Outfitter Rentals	3,859.1
May 8, 2012	103607	Edward's Department Store	27,064.78
May 8, 2012	103610	I-wear Direct	45,934.56
May 8, 2012	103619	Nature Voyageurs	61,421.94
May 8, 2012	103625	The Sport Pros	254,232.43
May 8, 2012	103627	Extreme Outdoors	361,564.54

Demonstration 2: Create a prompt to let users choose between different filters

Demonstration 2:

Create a prompt to let users choose between different filters

Purpose:

You have been asked to create a report showing the order date, order number, and revenue for sales to all retailers. The report should let users choose entire year of 2012, a specific date of high retail activity in 2012, or any date range of interest to the user. To provide users with these options, you will add a query macro prompt so users can choose which filter to apply to the report.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Add items and a filter containing a query macro prompt

1. Create a new **Blank** report using the **GO data warehouse (query)** package.
2. Add a **List** to the work area.
3. From the **Data/Source** tab, add the following query items to the new list data container:
 - Time: **Date**
 - Sales order: **Order number**
 - Retailers: **Retailer name**
 - Sales fact: **Revenue**

Date	Order number	Retailer name	Revenue
<Date>	<Order number>	<Retailer name>	<Revenue>

4. Click the **<Date>** List column body, on the Toolbar click **Sort**, then click **Ascending**.
5. On the **Queries**, click **Query1**.
6. On the **Application** tab, open the **Properties** pane.
7. In the **Properties** pane, under the **MISCELLANEOUS** section, change the **Name** property to **DateQuery**.

8. From the **Toolbox**, drag a **Filter** object to the **Detail Filters** pane.
You will create a query macro prompt that includes the name of the parameter (choose date), the data type (token), and a default prompt value ([Order date] between 2012-01-01 00:00:00 and 2012-12-31 23:59:59).
You can add parameter maps, session parameters, and macro functions to expressions. In the expression editor, to add a macro, on the Macros tab double-click the parameter maps, session parameters, or macro functions that you want to add to the macro expression.
You can use the Insert macro block to get you started with creating the macro expression. Insert macro block inserts number signs in the expression. Ensure that the macro expression you create is between the number signs.
9. In the **Expression Definition** pane, create and validate the following expression:
#prompt ('choose date', 'token', '[Date] between 2012-01-01 00:00:00 and 2012-12-31 23:59:59')#
Hints:
 - Macros tab: Insert macro block, Macro Functions folder\prompt.
 - Data Items tab: Date.
10. Click **OK** to close the **Detail filter expression** dialog box.
You will create a prompt page with a value prompt that uses the 'choose date' parameter you just created. You will create static choices for the prompt where the Use values are the SQL statements that are passed to the parameter.

Task 2. Create a prompt with static choices

1. On the **Pages**, click "Prompt pages" folder, and then from the **Toolbox**, drag a **Page** object to the **Prompt pages** pane.
2. Double-click **Prompt page1**, from the **Toolbox**, expand **PROMPTING**, and then drag a **Value prompt** object to the work area.
3. In the **Prompt Wizard**, click **Use existing parameter**, from the list select **choose date**, and then click **Finish**.
4. On the prompt page layout, click the **Value prompt** object to select it, and then in the **Properties** pane, under the **DATA** section, double-click the **Static choices** property.
5. In the **Static choices** dialog box, click **Add**.
6. In the **Use** box type **[Date] between 2012-01-01 00:00:00 and 2012-12-31 23:59:59**, in the **Display** box type **All Orders in 2012**, and then click **OK**.

7. Repeat steps **5** and **6** to add the following two static choices for the prompt:

Use	Display
<code>Cast([Date],date)=2012-12-15</code>	December 15, 2012
<code>[Date] in_range ?date_range?</code>	Date Range

In the second option, you cast [Date] as a 'date' data type because in the GO data warehouse database, the Date information is stored as 'date and time'. By converting this to 'date', you will retrieve every order made on December 15th, 2012 - regardless of what time the order was placed.

Use	Display
<code>[Date] between 2012-01-01...</code>	All Orders in 2012
<code>Cast([Date],date)=2012-12-15</code>	December 15 2012
<code>[Date] in_range ?date_range?</code>	Date Range

8. Click **OK** to close the **Static choices** dialog box.

Task 3. Add a Date & time prompt and test the prompt.

You will add a Date & time prompt object to the prompt page, so that users can enter specific dates if they choose the Date range option in the value prompt.

1. From the **Toolbox**, drag a **Date & time prompt** object to the right of the **Value prompt** object in the work area.
2. In the **Create a new parameter** box, type **date_range**, and then click **Finish**.
3. On the prompt page, click the **Date & time prompt** object to select it
4. In the Properties pane, under GENERAL, change the **Range** property to **Yes**.
You will test the prompt to see whether the default date range you specified for this prompt (between 2012-01-01 00:00:00 and 2012-12-31 23:59:59) is used if you do not select a prompt value.
5. On the Application bar, click **Run options** , then click **Run HTML**.
6. On the prompt page, in the **choose date** prompt list, select **All Orders in 2012**, and then click **Finish**.

The report runs. The first dates displayed are early January 2012.

7. In the report, click **Bottom**.

The last dates contained in the report are near the end of December 2012. The query prompt macro is using the default prompt value that you specified.

You will now test the prompt by selecting one of the other prompt options.

8. From the rendered report tab, click **Run Report** to run the report again.
9. In the value prompt list, click **December 15, 2012**, and then click **Finish**.

The report runs and appears on the rendered report tab. The report displays data for December 15, 2012, as expected.

You will test the prompt by selecting a different prompt option.

10. On the rendered report tab, click **Run Report**, to run the report again.
11. In the **choose date** prompt list, click **Date Range**.
12. In the **From** prompt, select **May 1, 2012**, and then in the **To** prompt, select **May 31, 2012**.
13. Click **Finish**.

You may get a second prompting for a date range. This is expected.

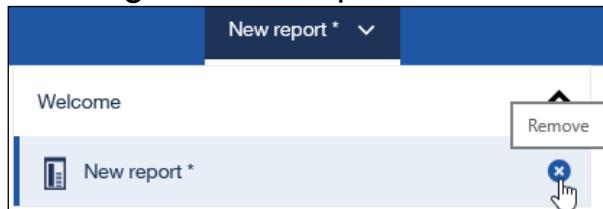
A section of the results appear similar to the following:

Date	Order number	Retailer name	Revenue
May 8, 2012	103576	Great Adventures	185,785.19
May 8, 2012	103581	Todo para el Golf, S.A. de C.V.	318,307.36
May 8, 2012	103601	The Marketplace	237,845.26

The beginning of the report contains data for early in May 2012.

(The rows may not be in the same order as shown)

14. On the report, click **Bottom**.
15. Close the rendered report tab.
16. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



17. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You created a report displaying the order date, order number, and revenue for sales to all retailers. You used a query macro prompt to let users choose to view orders for the entire year of 2012, for December 15th, 2012, or for any date range of interest to the user.

Let users choose which objects appear in reports

Prompt

Display Options	
Crosstab	
Chart	

Static Choices

Use	Display
Crosstab	Crosstab
Chart	Chart

Boolean variable applied to the conditional block:
?Display Options?='Crosstab'

When block value = 'Yes', drag Crosstab to conditional block



When block value = 'No', drag Chart to conditional block



Let users choose which objects appear in reports

Create a list of custom options that will appear at runtime by creating static choices for the prompt.

In this example, the user is prompted to choose Crosstab or Chart to view the report. Static choices represent a list of prompt options presented to the user. A conditional block is created with values that depend on the static choices the user selects in the prompt. Static Choices only apply to Search & Select and Value prompts.

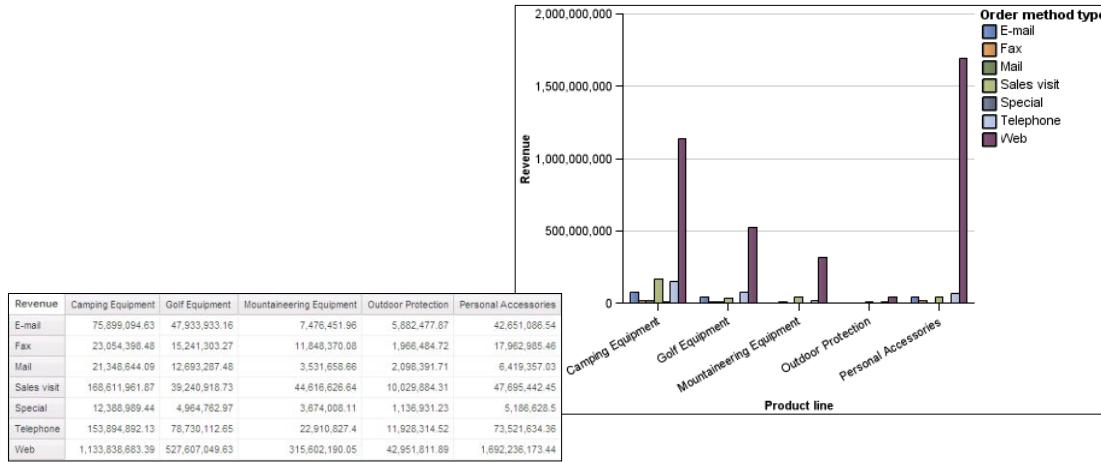
When the report is being created, the block value is set to Yes and the crosstab object is dragged to the conditional block. The block value is then set to No and the chart object is dragged to the conditional block.

If you want to give users the option to select one of two choices, use a Boolean variable. If you want to give users more than two choices, use a string variable.

After creating a report with a conditional block, to view the objects contained in the conditional block for different values, view the page structure for the report page. To do this, from the Application bar, click Page views, and then click Page structure.

Demonstration 3

Create a prompt to let users select a display type



Design effective prompts

© Copyright IBM Corporation 2018

Demonstration 3: Create a prompt to let users select a display type

Demonstration 3:

Create a prompt to let users select a display type

Purpose:

The Marketing department has requested a report that shows the revenue generated for each product line by each order method. Some members of the department prefer a visual representation of the data, and some have requested the data in crosstab format. You will create a report that lets users choose the format that they prefer.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Add a crosstab and a chart to the report.

1. Create a new **Blank** report using the **GO data warehouse (query)** package.
2. Add a **Crosstab** to the work area.
3. From the **Data/Source** tab, add the following query items to the crosstab data container:
 - **Rows:**
 - Order method: **Order method type**
 - **Columns:**
 - Products: **Product line**
 - **Measures:**
 - Sales fact: **Revenue**

Revenue	<#Product line#>	<#Product line#>
<#Order method type#>	<#1234#>	<#1234#>
<#Order method type#>	<#1234#>	<#1234#>

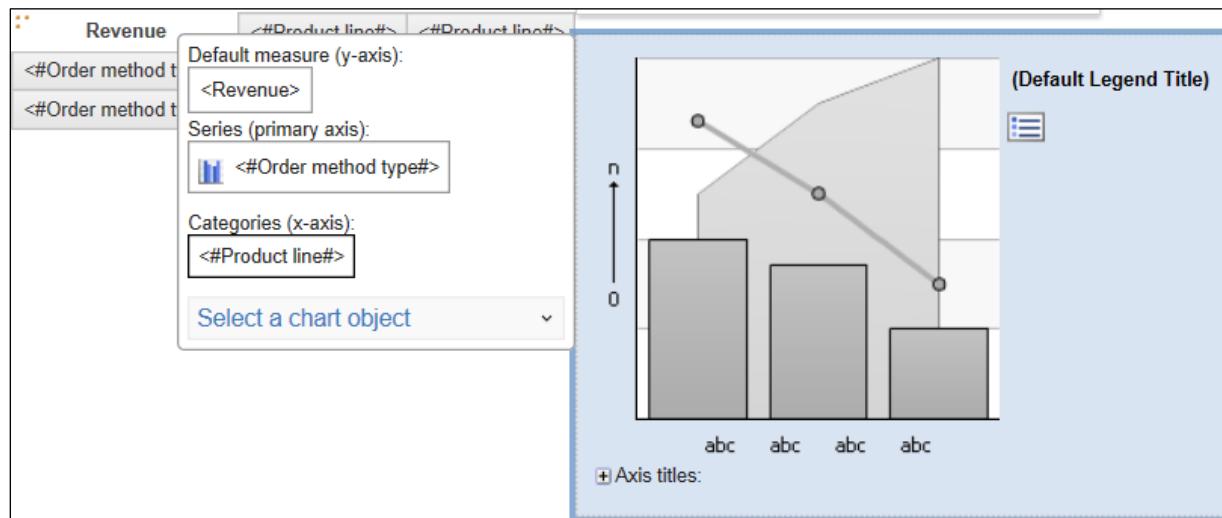
4. From the **Toolbox**, drag a **Visualization** object to the right of the Crosstab.
5. In the Visualization dialog, from the filter menu on the top right corner, choose Charts.
6. Double-click the **Clustered Column** chart.
This should be the first chart option displayed in the **Column** chart options. You want to add the same data that is in the crosstab to the chart.
7. On the Application bar, click **Show properties**  to open the Properties pane.

- In the Properties pane, change the Query property to **Query1**.

You could also have manually added the chart, modified the chart object to use the Query1 association, and then added the data items from Query1 into the chart as follows:

- Default measure (y-axis): **Revenue**
- Series (primary axis): **Order method type**
- Categories (x-axis): **Product line**

The results appear as follows:



Task 2. Add a prompt to the report and add static choices for the prompt

At run time, you want users to choose either a crosstab or a chart, so you will create a value prompt that gives users two display options.

- On side bar, click **Pages** , then click **Prompt pages** folder.
- From the **Toolbox**, drag a **Page** object to the **Prompt pages** pane.
- Double-click **Prompt page1**, and then from the **Toolbox** tab, under **PROMPTING**, drag a **Value prompt** object onto the prompt page.

The Prompt Wizard opens and loads any parameters contained in the report. In this case, the report has no parameters. You will create a parameter for this prompt.

- In the **Create a new parameter** box, type **Display Options**, and then click **Finish**.

You will add two static choices for the prompt (Crosstab and Chart) to let users select a format in which to view data.

- In the work area, click the value prompt object to select it, and then in the Properties pane, under the DATA section, double-click the **Static choices** property.

6. In the **Static choices** dialog box, click **Add**.
7. In the **Use** and **Display** boxes, type **Crosstab**, and then click **OK**.
8. Repeat steps 7 and 8 to add a second static choice, with **Chart** for the **Use** and **Display** values.
9. Click **OK** to close the **Static choices** dialog box.
10. On the **Properties** pane, under **GENERAL**, change **Auto-submit** to **Yes**.

Task 3. Add a conditional block and create a Boolean variable

You will add a Conditional blocks object and apply conditional formatting to this object to determine whether it will contain the chart or the crosstab when the report is run.

1. On side bar, click **Pages** , then click **Page1**.
2. From the **Toolbox**, under **ADVANCED**, drag a **Conditional blocks** object to the right of the chart to add it below the chart.
3. Click the **Conditional blocks** object (that you just added) to select it, and then in the **Properties** pane, under the **CONDITIONAL** section, double-click the **Block variable** property.

You will create a Boolean variable that uses the Display Options parameter so that the block displays differently depending on whether the user selects the Crosstab or the Chart prompt option.

4. In the **Variable** list, select **<New boolean variable>**, in the **Name** box type **DisplayVariable**, and then click **OK**.

You will create an expression that specifies that when users select the Crosstab prompt option, the Yes condition from the Boolean variable is applied to the conditional block.

5. Create and validate the following expression:

ParamDisplayValue('Display Options')='Crosstab'

Hint: Parameters tab: Display Options.

6. Click **OK** to close any open dialog box.

You can conditionally render many objects, including Blocks, Text Items, Lists, List columns, Crosstabs, Charts, Tables, Repeaters, Repeater Tables, Repeater Table Cells, and images. For a complete list, see the IBM Cognos Analytics - Reporting Object and Property Reference section of the *IBM Cognos Analytics Version 11.0.0 IBM Cognos Analytics - Reporting User Guide*.

Task 4. Specify how the conditional block values will display

1. Ensure that the **Conditional blocks** object is selected, in the **Properties** pane, under the **CONDITIONAL** section, change the **Current block** property to **Yes**.
In task 3, you specified that when users select the Crosstab prompt option, the Yes value is applied to the Conditional blocks object. Therefore, with the Yes value selected, you will drag the Crosstab object to the Conditional block.
2. Click the Crosstab's **Container Selector**  to select the entire Crosstab.
3. Drag the crosstab to the Conditional block object.
4. Click the **Conditional blocks** object to select it, in the Properties pane, under the **CONDITIONAL** section, change the **Current block** property to **No**.
In task 3, you specified that when users select the Chart prompt option, the No value is applied to the Conditional Block object. Therefore, with the No value selected, you will drag the Chart object to the Conditional Block.
5. Click the chart background to select the **Combination chart** object, and then drag the chart in to the **Conditional blocks** object.
6. On the Application bar, click **Run options** , then click **Run HTML**.
7. In the **Display Options** prompt, select **Crosstab**.

The results appear as follows:

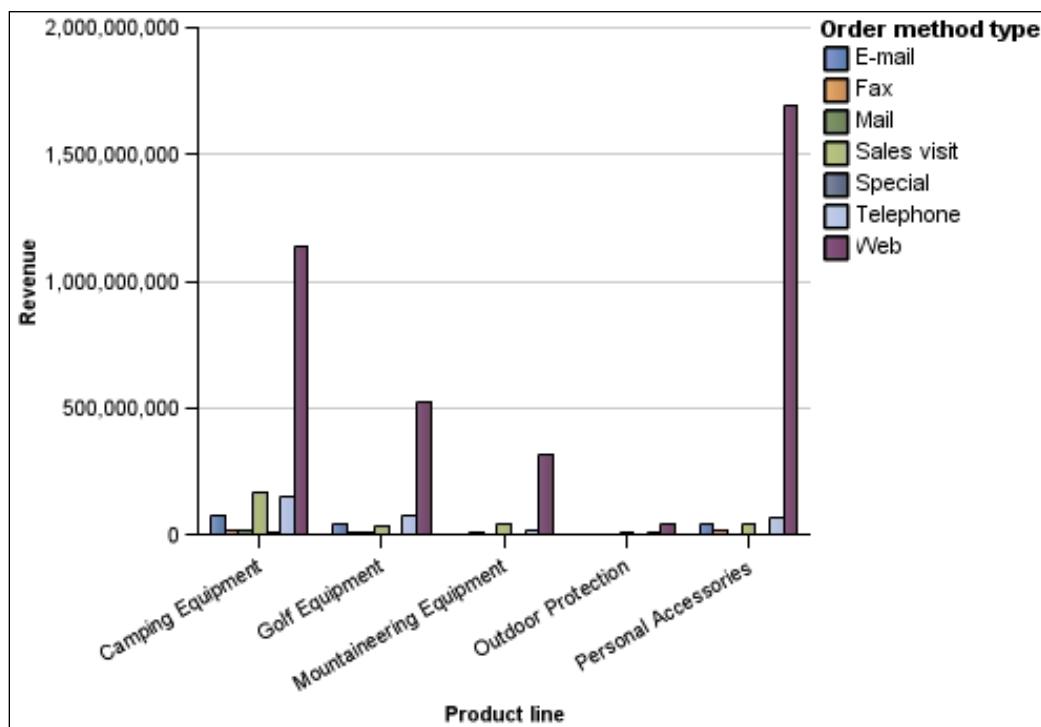
Revenue	Camping Equipment	Golf Equipment	Mountaineering Equipment	Outdoor Protection	Personal Accessories
E-mail	75,899,094.63	47,933,933.16	7,476,451.96	5,882,477.87	42,651,086.54
Fax	23,054,398.48	15,241,303.27	11,848,370.08	1,966,484.72	17,962,985.46
Mail	21,348,644.09	12,693,287.48	3,531,658.66	2,098,391.71	6,419,357.03
Sales visit	168,611,961.87	39,240,918.73	44,616,626.64	10,029,884.31	47,695,442.45
Special	12,388,989.44	4,964,762.97	3,674,008.11	1,136,931.23	5,186,628.5
Telephone	153,894,892.13	78,730,112.65	22,910,827.4	11,928,314.52	73,521,634.36
Web	1,133,838,683.39	527,607,049.63	315,602,190.05	42,951,811.89	1,692,236,173.44

Only the Crosstab appears in the report.

8. On the rendered report tab, click **Run Report**.

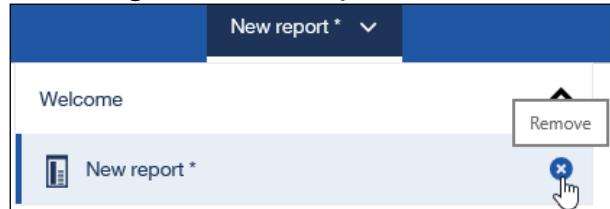
9. In the **Display Options** list, select **Chart**.

The results appear as follows:



10. Close the rendered report tab.

11. In the Application bar, click the report dropdown menu, then click **Remove** to the right of New report to close it. Then click **OK** to confirm your action.



12. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You created a report that lets users choose whether to view data in chart or crosstab format. To create this report, you added a prompt to the report, added static choices for this prompt, and then added a Conditional block object to the report. You created a variable for the Conditional block object with values that depend on the static prompt choice that users select. You specified that different report objects display in the Conditional block object depending on the value applied.

Unit summary

- Control report displays using prompts
- Specify conditional formatting values using prompts
- Specify conditional rendering of objects based on prompt selection
- Create sorted and filtered reports based on prompt selection

Unit 5 Create additional advanced reports

IBM Training



Create additional advanced reports

IBM Cognos Analytics (v11.0)

© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM.

Unit objectives

- Create a report that displays summarized data before detailed data
- Highlight alternate rows in a list report
- Use single data items to summarize report information
- Create a report using an external data file

Display summary data before detailed data

- To let users quickly locate key data, you can create a report that displays summarized information before detailed data.

Product	Quantity	Revenue
Camping Equipment(Orders: 51,715)	27,301,149	1,589,036,664.03
Cooking Gear(Orders: 12,329)	13,400,351	272,835,984.18
TrailChef Canteen	965,723	11,333,518.65
TrailChef Cook Set	813,780	41,184,274.9
TrailChef Cup	1,812,123	5,702,502.7
TrailChef Deluxe Cook Set	442,136	53,195,154.45

Summarized data about sales of all products in this product line and product type

Display summary data before detailed data

You can display summary data before the details when it would be useful to know the summarized values before reviewing the detailed values.

In the slide example, to display the number of orders in the header row cells, the author added a calculated column that counts the number of individual Quantity entries retrieved from the data source. Since each record in the data source represents one order and each order contains a Quantity value, by counting the individual instances of Quantity the query retrieves, the calculation indicates the number of orders made. Any other data item that would be part of each order record (such as Order number, Revenue, and so on) could be used instead of Quantity.

Demonstration 1

Create a report that displays summary data before detailed data

Product Orders		
Total Quantity Sold: 89,237,091		
Total Revenue: \$4,686,775,768.85		
Camping Equipment (Orders: 51,715)	27,301,149	1,589,036,664.03
Cooking Gear (Orders: 12,329)	13,400,351	272,835,984.18
Product	Quantity	Revenue
TrailChef Canteen	965,723	11,333,518.65
TrailChef Cook Set	813,780	41,184,274.9
TrailChef Cup	1,812,123	5,702,502.7
TrailChef Deluxe Cook Set	442,136	53,195,154.45
TrailChef Double Flame	245,559	34,311,174.84
TrailChef Kettle	2,336,950	25,368,496.06
TrailChef Kitchen Kit	866,669	19,535,825.83
TrailChef Single Flame	686,493	43,189,819.56
TrailChef Utensils	922,090	15,958,075.73
TrailChef Water Bag	4,308,828	23,057,141.46
Lanterns (Orders: 14,649)	4,826,755	126,925,660.64

Create additional advanced reports

© Copyright IBM Corporation 2018

Demonstration 1: Create a report that displays summary data before detailed data

Demonstration 1:

Create a report that displays summary data before detailed data

Purpose:

Management wants a report that lets them focus on summary data about total orders made, total quantity sold, and total revenue generated for each product line and product type. You will create a report where totals appear before the details. This report will have group headers that display summary data. To add additional context, below the report title, you will display the total quantity sold and total revenue generated by all products included in the report.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Create a List report and add a Count column

1. Create a new **Blank** report using the **GO data warehouse (query)** package.
2. Add a List  to the work area, for the **Query Name**, enter: **Product Quantity and Revenue**, then click **OK**.
3. From the **Data/Source** tab, add the following query items to the List:
 - Products: **Product line**, **Product type**, **Product**
 - Sales fact: **Quantity**, **Revenue**

Product line	Product type	Product	Quantity	Revenue
<Product line>	<Product type>	<Product>	<Quantity>	<Revenue>
<Product line>	<Product type>	<Product>	<Quantity>	<Revenue>
<Product line>	<Product type>	<Product>	<Quantity>	<Revenue>

You want the Product line and Product type headers to display the total number of orders made for each Product line and Product type.

To obtain this data, you will add a calculated column that counts the number of individual Quantity entries retrieved from the data source. Since each record in the data source represents one order and each order contains a Quantity value, by counting the individual instances of Quantity the query retrieves, you can determine how many orders were made.

4. From the **Toolbox** , under **TEXTUAL**, drag a **Query calculation** to the end of the List.

5. In the **Data item expression** dialog box, in the **Name** field, type **Count Orders**.
6. In the **Expression Definition** pane, create and validate the following expression:

count([Quantity])

Hints:

- You can drag the **count()** function from the Functions tab, in the Summaries folder, if you prefer to not type the function.
- From the Data Items tab, drag **Quantity**.

7. Click **OK**.

Although you chose to use the Quantity measure in the calculation in this task, this was not the only possible choice. For example, you could have used the Revenue measure in the calculation and obtained the same result.

Task 2. Group and summarize items and add headers

1. In the List, click **<Product line>** and Ctrl-click **<Product type>**, then from the Toolbar, click **Group / Ungroup** 

Product line	Product type	Product	Quantity	Revenue	Count Orders
<Product line>	<Product type>	<Product>	<Quantity>	<Revenue>	<Count Orders>
	<Product type>	<Product>	<Quantity>	<Revenue>	<Count Orders>
<Product line>	<Product type>	<Product>	<Quantity>	<Revenue>	<Count Orders>
	<Product type>	<Product>	<Quantity>	<Revenue>	<Count Orders>

2. Click **<Quantity>**, then Ctrl-click **<Revenue>** and **<Count Orders>**, then in the Toolbar, click **Summarize**  > **Total**.
3. Click **<Product line>**, then from the Toolbar click **More**  > **Headers & footers** > **Create header**.
4. With the **<Product line>** list column body still selected, press **Delete** to remove the redundant column from the List. But the header for **<Product line>** remains.
5. Repeat steps 4 and 5 to create a **<Product type>** header and remove the **<Product type>** column.

A section of the result appears as follows:

Product	Quantity	Revenue	Count Orders
<Product line>			
<Product type>			
<Product>	<Quantity>	<Revenue>	<Count Orders>
<Product type> - Total <Total(Quantity)> <Total(Revenue)> <Total(Count Orders)>			
<Product type>			
<Product>	<Quantity>	<Revenue>	<Count Orders>
<Product type> - Total <Total(Quantity)> <Total(Revenue)> <Total(Count Orders)>			
<Product line> - Total <Total(Quantity)> <Total(Revenue)> <Total(Count Orders)>			

Task 3. Move summary data from footers to headers

You want to add summary data to table cells in the header rows, so you will need to unlock the report.

1. Click the **<Product line>** header, then from the **Toolbar**, click **Split Cell** .
2. Click the **<Product type>** header, then from the **Toolbar**, click **Split Cell** .
3. On the Application bar, click **More**  **Locked**  to unlock the report.
4. From the **Toolbox**, drag a **Text item** to the right of **<Product line>**.
5. In the Text dialog, (press a space), then type: **(Orders:**, then (press a space), then click **OK**.
6. From the **<Product line> - Total** footer row, click then drag the **<Total(Count Orders)>** item to the right of the new Text item in the **<Product line>** header cell.
7. From the **Toolbox**, drag another **Text item** to the end of the same **<Product line>** header cell after the **<Total(Count Orders)>** item.
8. In the Text dialog, type: **),**, then press **Enter**.
9. From the **<Product line> - Total** footer row, click then drag the **<Total(Quantity)>** item to the empty Quantity cell in the **<Product line>** header.
10. From the **<Product line> - Total** footer row, click then drag the **<Total(Revenue)>** item to the empty Revenue cell in the **<Product line>** header.

A section of the result appears as follows:

Product	Quantity	Revenue	Count Orders
<Product line> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>	
<Product type>			
<Product>	<Quantity>	<Revenue>	<Count Orders>
<Product type> - Total	<Total(Quantity)>	<Total(Revenue)>	<Total(Count Orders)>
<Product type>			
<Product>	<Quantity>	<Revenue>	<Count Orders>
<Product type> - Total	<Total(Quantity)>	<Total(Revenue)>	<Total(Count Orders)>
<Product line> - Total			

11. Repeat steps 4 through 10 to move the <Total(Count Orders)>, <Total(Quantity)>, and <Total(Revenue)> items from the <Product type> footer to the <Product type> header.
12. From the Application bar, click **More** > **Unlocked**  to lock the report.
13. Click the <Count Orders> List column body, then press **Delete**.

A section of the result appears as follows:

Product	Quantity	Revenue	Count Orders
<Product type> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>	
<Product>	<Quantity>	<Revenue>	<Count Orders>
<Product type> - Total			
<Product type> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>	
<Product>	<Quantity>	<Revenue>	<Count Orders>
<Product type> - Total			
<Product line> - Total			

You no longer require the empty footers, so you will delete them from this report.

14. Click anywhere on the List, then from the Toolbar, click **More**  > **Headers & footers** > **List headers & footers**.
15. Clear the check boxes for all footers: **Product type (footer)**, **Product line (footer)**, and **Overall footer**. Then click **OK**.

A section of the result appears as follows:

Product	Quantity	Revenue
<Product line> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>
<Product type> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>
<Product>	<Quantity>	<Revenue>
<Product type> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>
<Product>	<Quantity>	<Revenue>
<Product line> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>
<Product type> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>
<Product>	<Quantity>	<Revenue>
<Product type> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>
<Product>	<Quantity>	<Revenue>

16. Run the report in **HTML**.

A section of the result appears as follows:

Product	Quantity	Revenue
Camping Equipment (Orders: 51,715)	27,301,149	1,589,036,664.03
Cooking Gear (Orders: 12,329)	13,400,351	272,835,984.18
TrailChef Canteen	965,723	11,333,518.65
TrailChef Cook Set	813,780	41,184,274.9
TrailChef Cup	1,812,123	5,702,502.7

The total quantity sold, revenue generated, and the total number of orders for each product line and product type appears in the headers.

17. Close the rendered report tab.

Task 4. Format the report

You will now format the headers to be more readable and informative.

1. Click the **<Product type>** header cell, and then press the **Tab** key.
2. On the Application bar, click **Show properties**  to open the Properties pane.
In the Properties pane, under the **BOX** section, notice that because you pressed the Tab key, the Padding property is now set to Left: 20px.
3. In the List, click a **<Product>** cell, and then press the **Tab** key twice.
In the Properties pane, under the **BOX** section, observe that the Padding property for the Left padding has changed to: Left 40 pixels.

Task 5. Add a report title and add quantity and revenue summary labels below the report title

You will add data to the report title displaying the total revenue generated and quantity sold by all product lines included in the report.

1. From the Toolbar, click **More ... > Headers & footers > Page header & footer**. Check **Header**, then click **OK**.
2. Click the page header, then from the Toolbar, click **Horizontal alignment > Center** .
3. In the page header, click **Add** , then click **Table** .
4. Set columns to **1**, and rows to **3**, uncheck **Maximize width**, then click **OK**.
5. Select the three Table rows, then on the Toolbar, click **Horizontal alignment > Center** .
6. On the Toolbar, click **Font** .
7. Set **Family** to **Arial**, **Weight** to **Bold**, then click **OK**.
8. From the **Toolbox**, under PINNED, drag a **Text item** onto the first table cell, type **Product Orders**, then click **OK**.

9. Click on title **Product Orders**, then in the Toolbar, click **Font A**.
10. Set **Family** to **Arial**, **Size** to **14pt**, check **Underline**, then click **OK**.
11. From the **Toolbox**, drag a **Text item** onto the second table cell, type **Total Quantity Sold:**, press the spacebar, then click **OK**.
12. From the **Toolbox**, drag a **Text item** onto the second table cell, type **Total Revenue:**, press the spacebar, then click **OK**.

A section of the result appears as follows:

Product Orders	
Total Quantity Sold:	
Total Revenue:	

Task 6. Add quantity and revenue summary data using a singleton

You need to add a query calculation to display the total quantity sold by all records retrieved by the query. To do this, you need to set the Product Quantity and Revenue query as a property of the Page object.

1. From the **Toolbox**  tab, under DATA CONTAINER, drag a **Singleton** to the right of the **Total Quantity Sold:** text in the report header.
- Singletons will be discussed in more detail, later in this unit.
2. From the **Object and query name** dialog box, in the **Query Name** list, select **Product Quantity and Revenue**, and then click **OK**.
3. With the **Singleton** selected, from the **Data/Data items** tab, drag **Quantity** onto the singleton object in the report.
4. Repeat steps 1 through 3 to create the summary total for **Total Revenue:**, using **Revenue**.

The report layout appears as follows:

Product Orders		
Total Quantity Sold: <Quantity>		
Total Revenue: <Revenue>		
Product	Quantity	Revenue
< Product line> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>
< Product type> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>
<Product>	<Quantity>	<Revenue>
<Product type> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>
<Product>	<Quantity>	<Revenue>
<Product line> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>
<Product type> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>
<Product>	<Quantity>	<Revenue>
<Product type> (Orders: <Total(Count Orders)>)	<Total(Quantity)>	<Total(Revenue)>
<Product>	<Quantity>	<Revenue>

Task 7. Format the quantity and revenue summary information

- Click the <Revenue> singleton, then in the Toolbar, click **Data format** .
- Change **Format type** to **Currency**, then click **OK** to close the **Data format** dialog.
- Click anywhere on the page below the List to select the page body, then from the Toolbar, click **Horizontal alignment > Center** .
- Click the List **Container Selector**  to select the List, then in the Properties pane, under GENERAL, change the **Column titles** property to **At start of details**.
- On the Application bar, click **Run options** , then click **Run HTML**.

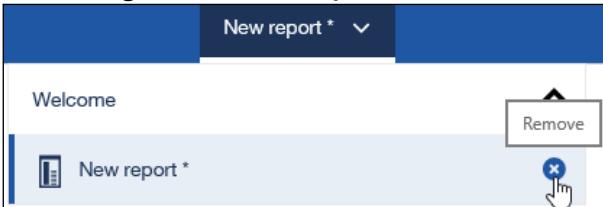
A section of the result appears as follows:

Product Orders		
Total Quantity Sold: 89,237,091		
Total Revenue: \$4,686,775,768.85		
Camping Equipment (Orders: 51,715)	27,301,149	1,589,036,664.03
Cooking Gear (Orders: 12,329)	13,400,351	272,835,984.18
Product	Quantity	Revenue
TrailChef Canteen	965,723	11,333,518.65
TrailChef Cook Set	813,780	41,184,274.9
TrailChef Cup	1,812,123	5,702,502.7
TrailChef Deluxe Cook Set	442,136	53,195,154.45
TrailChef Double Flame	245,559	34,311,174.84
TrailChef Kettle	2,336,950	25,368,496.06
TrailChef Kitchen Kit	866,669	19,535,825.83
TrailChef Single Flame	686,493	43,189,819.56
TrailChef Utensils	922,090	15,958,075.73
TrailChef Water Bag	4,308,828	23,057,141.46
Lanterns (Orders: 14,649)	4,826,755	126,925,660.64

The text under the report title displays the total quantity sold and total revenue generated by all product lines.

- Close the rendered report tab.

7. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



8. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You created a report that lets users quickly locate total quantity sold, total revenue generated, and total orders made for each product type and product line. To give additional context, you added report summary totals for quantity and revenue below the report title.

Highlight alternate rows

Product name	Revenue	Quantity	Calc.
TrailChef Cup	\$218,416	51,834	1
TrailChef Kettle	\$118,595	9,900	0
Star Gazer 2	\$9,123,447	18,262	1
Star Peg	\$85,937	44,686	0
TrailChef Utensils	\$268, 084	15,928	1
Firefly Lite	\$179,749	13,558	0

Use an expression to control the conditional formatting of alternating rows

Highlight alternate rows

To make a report easier to read, you can format a List so that rows appear in two alternating colors. To highlight alternate rows in a report, create a calculated data item that returns 0 for even rows and 1 for odd rows. Then, apply conditional formatting based on the report data.

You may want to leave the calculated column in the report for testing purposes. The calculated column does not need to appear in the final report, although the calculated data item must be a property of the List.

The mod operator, used in an expression such as in the Highlight Alternate Rows slide example, works as follows:

- If you ask the mod function to return the remainder of 1 divided by 2, since the remainder would not be a whole number, the mod function will return the numerator (1).
- $2/2$ will have a remainder of 0
- $3/2$ will have a remainder of 1
- $4/2$ will have remainder of 0, etc.

Because you use the running-count summary function to assign a sequential number to every row, this technique can only be used on List reports, and is not recommended for Crosstabs.

A quick way to apply identical conditional formatting to all columns in a List report is to apply it to the List Columns Body Style. This saves time, and will also apply the formatting to any future columns you may add to the report.

Demonstration 2

Highlight alternate rows in a List report

Employee name	Salary	Bonus value
Aaghie Heiman	60,500	\$6,576.92
Aaltje Hansen	31,730.77	\$0.00
Abel Antunes	63,461.54	\$6,969.23
Abram Ruiz	46,653.85	\$0.00
Ada Morales	29,190.77	\$3,228.40
Adara Cruz	63,461.54	\$7,056.92
Adda Heijman	42,576.92	\$0.00
Adelaide Wiesinger	27,943.08	\$6,406.89
Adeline Arnaud	25,076.92	\$1,883.08
Adelma Ortiz	41,153.85	\$4,753.85
Adriaantje Haanraads	42,576.92	\$0.00
Adriana Iacobucci	154,153.85	\$0.00
Adrien Martin	26,695.38	\$2,085.72
Adrienne Roche	157,846.15	\$0.00

Create additional advanced reports

© Copyright IBM Corporation 2018

Demonstration 2: Highlight alternate rows in a List report

Demonstration 2: Highlight alternate rows in a list report

Purpose:

The Human Resources department has requested a report showing bonus and salary data for each member of the sales staff. To make the report easier to read, every other row must be highlighted in blue. You will create a calculated item that returns 0 for even rows and 1 for odd rows, and will then conditionally format every other row to be highlighted in blue.

Portal: <http://vclassbase/ibmcognos>

User/Password: admin/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: HR (query)

Namespace: Employee summary (query)

Task 1. Create a list report

1. Create a new **Blank** report using the **GO data warehouse (query)** package
2. Add a new List to the work area.
3. From the **HR (query)** folder\Employee summary (query) namespace, add the following items to the list data container.
 - Employee by region: **Employee name**
 - Employee summary fact: **Salary, Bonus value**
4. Sort the **Employee name** column in ascending order.

Employee name▲	Salary	Bonus value
<Employee name>	<Salary>	<Bonus value>
<Employee name>	<Salary>	<Bonus value>
<Employee name>	<Salary>	<Bonus value>

5. On the Application bar, click **Run options** , then click **Run HTML**.

A section of the result appears as follows:

Employee name	Salary	Bonus value
Aaghie Heiman	60,500	\$6,576.92
Aaltje Hansen	31,730.77	\$0.00
Abel Antunes	63,461.54	\$6,969.23
Abram Ruiz	46,653.85	\$0.00
Ada Morales	29,190.77	\$3,228.40
Adara Cruz	63,461.54	\$7,056.92
Adda Heijman	42,576.92	\$0.00
Adelaide Wiesinger	27,943.08	\$6,406.89
Adeline Arnaud	25,076.92	\$1,883.08
Adelma Ortiz	41,153.85	\$4,753.85

To make this report easier to read, you want to highlight every other row in blue. You will create a calculated data item that will return zeros for even rows and ones for odd rows. You can then apply conditional formatting to the report based on this calculated item.

6. Close the rendered report tab.

Task 2. Add a calculated data item

1. On the side bar, click **Queries** , then click  **Queries**, then click **Query1**.
2. From the **Toolbox** tab , drag a **Data Item** object to the **Data Items** pane. You will create a calculation that counts each row in the report that contains a Salary data item value. The calculation will then divide the row number by two and return the remainder. The remainder for even rows will be 0 and the remainder for odd rows will be 1.
3. In the **Data item expression** dialog, in the **Name** field, type **EvenOdd**.
4. Create and validate the following expression:

mod(running-count([Employee name]),2)

Hints:

- Function tab: **mod()** is in the Common Functions folder\M-Q folder; **running-count()** is in the Summaries folder.
- Data Items tab: **Employee name**.

You are using the Employee name because of the possibility that Salary and Bonus may have \$0.00 rows that will not be counted. This would cause a duplication of a highlighted row. Since the Employee name row will always have a value, its rows will always have alternate highlighting.

5. Click **OK**.

6. On the **Application** bar, open the **Properties**  pane.
7. On the **Side pane**, from the **Queries** tab , click **Queries folder**, and then click **Query1**, and then, on the **Properties** pane, rename the query to **Employee Salary Information**.

Task 3. Add the calculated item to the report and then create a conditional variable

You will add the **EvenOdd** data item to the list to see the values returned in the report.

1. On the side bar, click **Pages** , then click  **Report pages**, then click **Page1**.
2. From the **Data/Data items** tab, drag the **EvenOdd** item to the end of the List.
3. On the Application bar, click **Run options** , then click **Run HTML**.

A section of the result appears as follows:

Employee name	Salary	Bonus value	EvenOdd
Aaghie Heiman	60,500	\$6,576.92	1
Aaltje Hansen	31,730.77	\$0.00	0
Abel Antunes	63,461.54	\$6,969.23	1
Abram Ruiz	46,653.85	\$0.00	0
Ada Morales	29,190.77	\$3,228.40	1
Adara Cruz	63,461.54	\$7,056.92	0
Adda Heijman	42,576.92	\$0.00	1
Adelaide Wiesinger	27,943.08	\$6,406.89	0
Adeline Arnaud	25,076.92	\$1,883.08	1

In even rows, the **EvenOdd** value is 0 and in odd rows the value is 1. These values will be used to apply conditional formatting to alternating rows in the report.

You do not want to display the calculated data item in the report layout, however, you need to make this calculated item a property of the list object.

4. Close the rendered report tab.
5. Click the **<EvenOdd>** List column body, then from the Toolbar, click **Select Ancestor** , and then click **List column**.
6. In the Properties pane, under the CONDITIONAL section, change the **Render** property to **No**.

Note: This maintains the EvenOdd column as a property of the list but does not render it at run time.

You will create a Boolean variable that will let you apply formatting to alternate rows depending on whether the EvenOdd calculation is value is 1 or 0.

7. On the **Queries**  tab, click **Conditional explorer** , click Variable, and then on the **toolbox**  tab, double-click **Boolean Variable**.
8. In the **Report expression** dialog box, create and validate the following expression:
[Employee Salary Information].[EvenOdd]=0
Hint: Drag EvenOdd from the Available Components pane.
9. Click **OK** to close the **Report expression** dialog box.
 You created a Boolean variable because there are only two possible choices: the EvenOdd value can be 1 or 0. You can now apply this variable to the report so if the value is Yes (because the row is even and the EvenOdd value equals 0), the row will be highlighted in blue.
10. With the variable still selected, in the Properties pane, under the MISCELLANEOUS section, change the **Name** property to **EvenRows**.
11. Navigate to **Page1**.

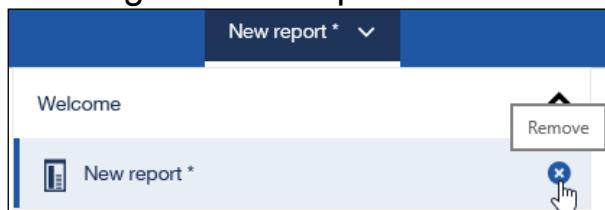
Task 4. Apply conditional formatting and run the report

1. In the list data container, click any list column body in the list, on-demand toolbar, click **Select Ancestor**, and then click **List columns body style**.
2. In the **Properties** pane, under the **CONDITIONAL** section, double-click the **Style variable** property.
 From the Variable list, select **EvenRows**, and then click **OK**.
 You will now specify that even rows be highlighted in blue.
3. On the **Queries**  tab, click **Conditional explorer** tab , and then from under **EvenRows**, click **Yes**.
 A green message box indicates that conditional formatting is turned on and that the EvenRows variable is set to 1.
4. On the **Properties** pane, under **COLOR & BACKGROUND**, double-click **Background color**, and then click the **Color swatch** tab.
5. Click the blue (#99CCFF) box, located five rows from the bottom and seven columns in from the right, and then click **OK**.
 The list columns are highlighted in blue.
6. On the **Queries**  tab, click **Conditional explorer**  tab, click the **EvenRows** variable to turn off conditional formatting mode.
7. On the Application bar, click **Run options** , then click **Run HTML**.
 A section of the result appears as follows:

Employee name	Salary	Bonus value
Aaghie Heiman	60,500	\$6,576.92
Aaltje Hansen	31,730.77	\$0.00
Abel Antunes	63,461.54	\$6,969.23
Abram Ruiz	46,653.85	\$0.00
Ada Morales	29,190.77	\$3,228.40
Adara Cruz	63,461.54	\$7,056.92
Adda Heijman	42,576.92	\$0.00
Adelaide Wiesinger	27,943.08	\$6,406.89
Adeline Arnaud	25,076.92	\$1,883.08
Adelma Ortiz	41,153.85	\$4,753.85
Adriaantje Haanraads	42,576.92	\$0.00
Adriana Iacobucci	154,153.85	\$0.00

Even rows in the list report are highlighted in blue.

9. Close the rendered report tab.
10. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



11. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You created a report in which every other row is highlighted in blue. To do this, you created a calculated item that returned 0 for even rows and 1 for odd rows. You then applied conditional formatting to the list columns so that every other row was highlighted in blue.

Display single data values outside of report data objects

The singleton is in a table, on a report page with no associated query.

Sales Results 2013

Sales Results 2013		
Top Sales Rep		18,649,837.91
Country	Employee name	Revenue
Switzerland	Aaltje Hansen	6,379,097.1
Switzerland	Abram Ruiz	8,910,486.85
Switzerland	Adda Heijman	8,131,762.35
Switzerland	Adriaantje Haanraads	7,578,816.55

Display single data values outside of report data objects

Singletons allow you to display single data values in a report without having to set the page-query association or join unrelated queries. You cannot sort singleton queries, since only one row is returned. You also cannot set 'no data' options on a singleton item.

When you drop a singleton item into a report, it creates a query that retrieves only the first row value.

Singleton queries are not supported when producing report output in CSV format.

Demonstration 3

Use singletons to summarize information in a report

Sales Results 2013		
Top Sales Rep: Helena Lindholm 18,649,837.91		
Country	Employee name	Revenue
Switzerland	Aaltje Hansen	6,379,097.1
Switzerland	Abram Ruiz	8,910,486.85
Switzerland	Adda Heijman	8,131,762.35
Switzerland	Adriaantje Haanraads	7,578,816.55
Spain	Agatha Reyes	8,728,898.43
Switzerland	Agnelo Chavez	7,236,752.15
Switzerland	Agnes Ramos	6,096,683.8
United Kingdom	Aidan Chaplin	4,076,558.81
Japan	Aiko Watanabe	5,065,723.79
Japan	Aimi Tanaka	9,963,809.33

Demonstration 3: Use singletons to summarize information in a report

Demonstration 3:

Use singletons to summarize information in a report

Purpose:

You want to create a list report showing the revenue generated by each sales rep for each year. You then want to create a table using singletons to show the top sales representative for the year 2013. This table will be in the header, so as to be seen on every page, to make it easier for managers to compare results of their sales representatives to the top sales representative of 2013.

Portal: <http://vclassbase/ibmcognos>

User/Password: admin/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Create a list report with a title and table in the header

1. Create a new **Blank** report using the **GO data warehouse (query)** package.
2. Add a new **List** to the work area.
3. On the **Data/Source** tab, add the following items to the list data container:
 - Time: **Year**
 - Employee by region: **Country, Employee name**
 - Sales fact: **Revenue**

Year	Country	Employee name	Revenue
<Year>	<Country>	<Employee name>	<Revenue>
<Year>	<Country>	<Employee name>	<Revenue>
<Year>	<Country>	<Employee name>	<Revenue>

4. In the list, click **Year**, from the on-demand toolbar, under **More** ..., click **Go to query**.
 5. From the **Data Items** pane, drag **Year** to the **Detail Filters** pane, and then create and validate the following expression:
[Year]=2013
 6. Click **OK** to close the **Detail filter expression** dialog box.
 7. On the side bar, click **Pages** , then click **Page1**.
 8. Click the **<Year>** List column body to select it, and then from the Toolbar, under **More** ..., click **Cut**.
- You have removed Year from the report layout, but it remains in the query.

9. From Toolbar, click **More** ..., click **Header & Footer**, click **Page header and footer**, and then check **header**.
10. Click to choose **Text box** , type **Sales Results 2013**, and then left align the title block.
11. From on-demand toolbar, click font to change it to **Arial, 14pt**, and **Bold**.
12. From the **Toolbox** tab, drag a **Block** object below the title text, and then drag a **Table** object into the new block.
13. Accept the default number of columns (2) and rows (1), clear the **Maximize width** check box, and then click **OK**.

A section of the result appears as follows:

Country	Employee name	Revenue
<Country>	<Employee name>	<Revenue>
<Country>	<Employee name>	<Revenue>
<Country>	<Employee name>	<Revenue>

Task 2. Add a singleton to a table

1. From the **Toolbox** tab, drag a **Text item** into the left table cell, type **Top Sales Rep**, press the spacebar, and then click **OK**.
2. On the **Data/Source** tab, from **Employee by region**, drag **Employee name** to the right of the new text item in the left table cell.
3. Click **OK** to accept the new singleton.
4. On the **Application** bar, open the **Properties** pane.
5. From the **Queries tab**, click **Queries**.
6. Click **Query2**, and then change the query name to **Top Sales Rep**.
7. On the Application bar, click **Run options** , then click **Run HTML**. A section of the results appear as follows:

Top Sales Rep: Aaghie Heiman

This is not the actual top sales rep that you want in the table. When you drop a singleton item into a report, it creates a query that retrieves only the first row value. You need to customize the query to display the correct value.

8. Close the rendered report tab.

Task 3. Customize the query

1. On side bar, click **Pages** , then click **Page1**.
2. In the work area, click the **<Employee name>** singleton, and then from the on-demand toolbar, under **More** , click **Go to query**.
3. On the **Data/Source**  tab, from **Sales fact**, drag **Revenue** to the **Data Items** pane.
4. Double-click **Revenue** to open the **Data item expression** dialog box.
5. Edit and validate the expression as follows:
total([Sales (query)].[Sales fact].[Revenue] for [Employee name])
 Hint: Function tab: **total()** is in the Summaries folder.
 Data Items tab: Employee name.
6. Click **OK** to close the **Data item expression** dialog box.
7. From the **Toolbox**  tab, drag a **Data Item** into the **Data Items** pane.
8. In the **Name** field, type **Maximum Sales Rep Revenue**.
9. Create and validate the following expression:
maximum([Revenue] for report)
 Hint: **maximum ()** is in the Summaries folder.
 Data Items tab: Revenue.
10. Click **OK**.
11. From the **Data Items** pane, drag **Revenue** into the **Detail Filters** pane.
12. Create and validate the following expression:
[Revenue]=[Maximum Sales Rep Revenue]
 Hint: Data Items tab: Maximum Sales Rep Revenue.
13. Click **OK**.
14. On the **Data/ Source**  tab, from the **Time** query subject, drag **Year** into the **Detail Filters** pane.
15. Create and validate the following expression:
[Sales (query)].[Time].[Year]=2013
16. Click **OK** to close the dialog box.

Task 4. Add a singleton to the table and format the table items

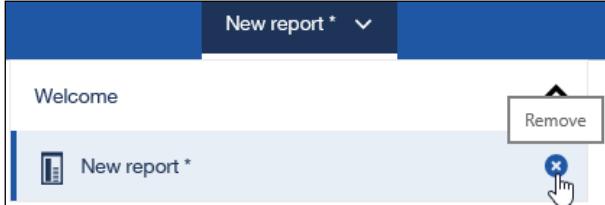
- On side pane, click **Pages** , click **Page1**, and then from the **Data** tab, click the **Data items** tab.
You now have new items available to include in your table. You want to include the new Maximum Sales Rep Revenue item in your header table.
- From the **Data items** tab, drag **Maximum Sales Rep Revenue** to the right table cell, and then click **OK** to accept the singleton.
- Click the **<Maximum Sales Rep Revenue>** singleton in the table, then in the Toolbar, click **Select Ancestor** , then click **Table**.
- In the **Properties**  pane, under the **BOX** section, change the **Padding** property to apply **15** pixels of left padding.
- In the table, click **<Employee name>**, Ctrl-click **<Maximum Sales Rep Revenue>**, and then from the Toolbar, change the font **Weight** to **Bold**.
- On the Application bar, click **Run options** , then click **Run HTML**.

A section of the result appears as follows:

Sales Results 2013		
Top Sales Rep: Helena Lindholm 18,649,837.91		
Country	Employee name	Revenue
Switzerland	Aaltje Hansen	6,379,097.1
Switzerland	Abram Ruiz	8,910,486.85
Switzerland	Adda Heijman	8,131,762.35
Switzerland	Adriaantje Haanraads	7,578,816.55
Spain	Agatha Reyes	8,728,898.43
Switzerland	Agnelo Chavez	7,236,752.15
Switzerland	Agnes Ramos	6,096,683.8
United Kingdom	Aidan Chaplin	4,076,558.81
Japan	Aiko Watanabe	5,065,723.79
Japan	Aimi Tanaka	9,963,809.33

As managers review the pages of the report, they can easily compare their sales rep results with the results of the top sales rep of 2013.

7. Close the rendered report tab.
8. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



9. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You created a list report showing the revenue generated by each sales rep for each year. You then created a table in the page header to display the top sales representative for the year 2013.

Use external data

- External data:
 - is data that is not in the query database
 - extends existing packages with new query subjects.
- Type of external data:
 - Microsoft® Excel (.xls)
 - tab-delimited text (.txt)
 - comma-separated (.csv)
 - XML files

Use external data

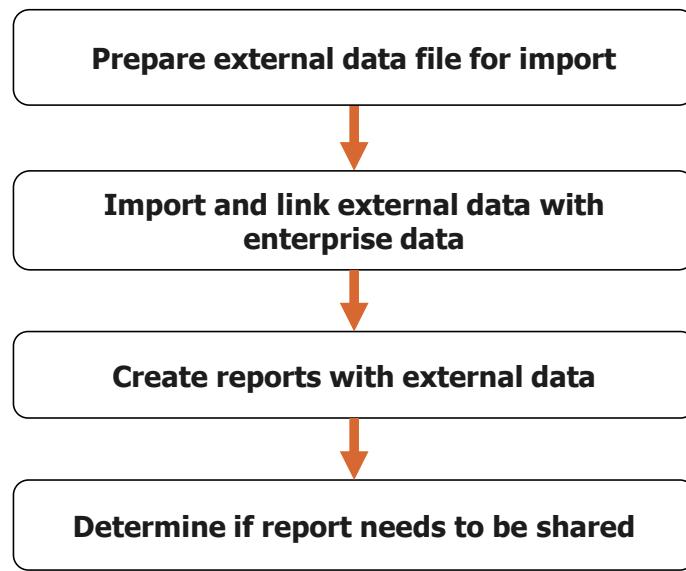
Many times report authors need to add external data to IBM Cognos content to meet report requirements. In many cases, the authors are unable to merge the data from external data sources without intervention from IT. The objective of the External Data functionality is to provide this capability to the report author. This allows the author to do basic data modeling, including creating relationships between the data items and setting cardinality.

External data extends an existing package definition with new query subjects that are in the external data file. The definition involves two steps:

1. Identifying columns from the external data file to become new query subjects.
2. Defining how the External Data file will be related to the existing IBM Cognos content.

At least one query subject in the report must be mapped to a data item from their data. The mapping creates joins (or relationships) between external data and enterprise data. This will ensure that the data is integrated smoothly.

Process to use external data



Create additional advanced reports

© Copyright IBM Corporation 2018

Process to use external data

External data extends an existing package definition by adding new query subjects from the external data source. As you prepare to import, identify which columns of the external data will become new query subjects, and then define how the external data will relate to the existing IBM Cognos content.

To work with external data:

1. Ensure the external data file matches your enterprise data. It is necessary to link at least one column of external data with your enterprise data source.
2. Import your external data file using the wizard.
3. Create reports. After imported and linked, the data appears in the data tree under the Source tab.
4. Determine if you want to share the reports that use your external data.

Before external data can be imported by a user, the IBM Cognos administrator must grant permission for the Allow External Data capability.

Demonstration 4

Create a report and add external data

Country	City	Staff name	Sales staff code	Sales target
Australia	Melbourne	Dave Smythe	10090	\$15,084,300
		Jake Cartel	10092	\$3,786,500
		John Sinden	10093	\$4,462,400
		Alice Walter	10089	\$16,834,700
		Jackie Fulford	10527	\$17,965,800
		Donald Neely	10526	\$997,200
		Donald Ward	10091	\$18,036,200
		Jonathan Farrel	10773	\$2,073,100
		Melbourne - Total		
Australia - Total				\$79,240,200
Austria	Wien	Alexandra Klauser	10489	\$1,577,400
		Sabine Grüner	10124	\$29,116,000
		Jutta Shulz	10765	\$23,595,900
		Thomas Schirmer	10488	\$20,733,300
		Wien - Total		
Austria - Total				\$75,022,600

Create additional advanced reports

© Copyright IBM Corporation 2018

Demonstration 4: Create a report and add external data

Demonstration 4:

Create a report and add external data

Purpose:

You have received a .csv file from the Human Resources manager, which includes updated city and country locations for all employees. You want to create a report that will include the employee details (name, employee code, and sales target numbers) from the corporate data, with the updated employee location information in the external file. To do this, you will create a report using the enterprise data, and then add the external data to the package. This will allow you to create a report that displays the sales targets and current employee location information.

Portal: <http://vclassbase/ibmcognos>

User/Password: admin/Education1

Package: Team content\Samples\Models\GO sales (query)

Namespace: Sales target (query)

Note: The following demonstration must be run using Internet Explorer.¹

Task 1. Create a List report

1. Create a new **Blank** report using the **GO sales (query)** package
2. Add a new List to the work area.
3. From the **Data/Source** tab, add the following query items to the List:
 - Sales staff: **Staff name**, **Sales staff code**
 - Sales target: **Sales target**

Staff name	Sales staff code	Sales target
<Staff name>	<Sales staff code>	<Sales target>
<Staff name>	<Sales staff code>	<Sales target>
<Staff name>	<Sales staff code>	<Sales target>

4. On the Application bar, click **Run options** , then click **Run HTML**.
A section of the results appears similar to the following:

¹ Because other browsers don't expose the full path of a user's file location, see [External Data option "Allow the server to automatically load the file" is disabled](#). And this is required for the demonstration.

Staff name	Sales staff code	Sales target
Élizabeth Michel	10005	\$32,268,800.00
Émile Clermont	10006	\$7,486,000.00
Étienne Jauvin	10007	\$46,673,100.00
Frank Fuhlroth	10014	\$24,085,100.00
Gunter Erler	10015	\$2,755,100.00

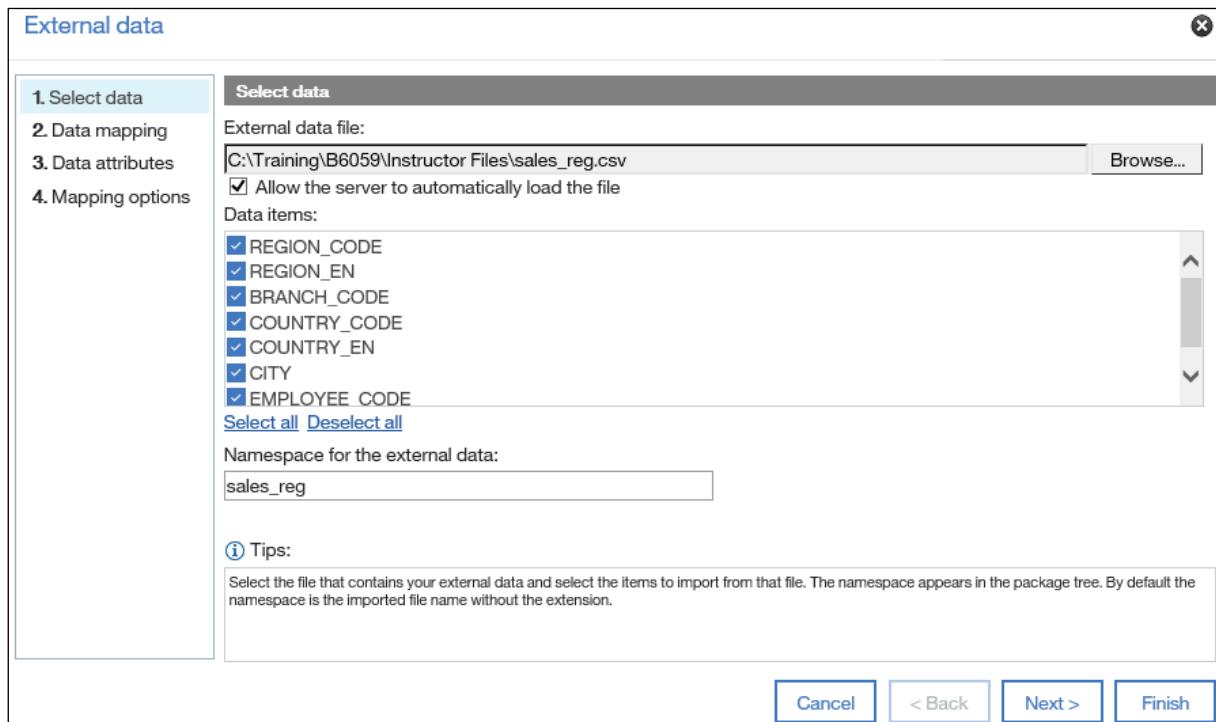
5. Close the rendered report tab, and then save  the report to **Team content\B6059**, as **Staff Sales Targets**.

Task 2. Identify the external data file and columns for the report

- To extend existing packages, in the Data /Source tab, right-click the package, then click **Manage external data** .

The External data wizard opens, to guide you through the process of identifying and mapping the external data. The process begins with selecting the external data.

- In the **External data file** section, click **Choose file**, navigate to **C:\Training\B6059\Instructor Files**, then double-click **sales_reg.csv**.
- Scroll through the list of selected items to review the file.



You will bring in all columns at this time. You could clear check boxes for any columns that you do not require.

- Select the **Allow the server to automatically load the file** check box.

If you deselect this option, when the report is run, the user will be prompted to select the location of the external data file, allowing other users to run your report using their own version of the external data file. If they use their own version, the file must contain the same columns as the original external data file that you used to import the data and create the report. If you always want the report to run using your version of the file, then you should select the **Allow the server to automatically load the file** check box.

Notice the Namespace for the external data box displays the default namespace that will be assigned. The namespace provides a unique name to associate with the data items that you import. The namespace appears in the data tree of the Source tab and is used to organize the data items. By default, the namespace is the imported file name without the extension. If you change the default name for the namespace, you are prompted to select the external data file each time you run the report. To avoid this, select the Allow the server to automatically load the file check box.

5. Click **Next** to proceed to the **Data Mapping** page.

Task 3. Map the external data file with the existing data

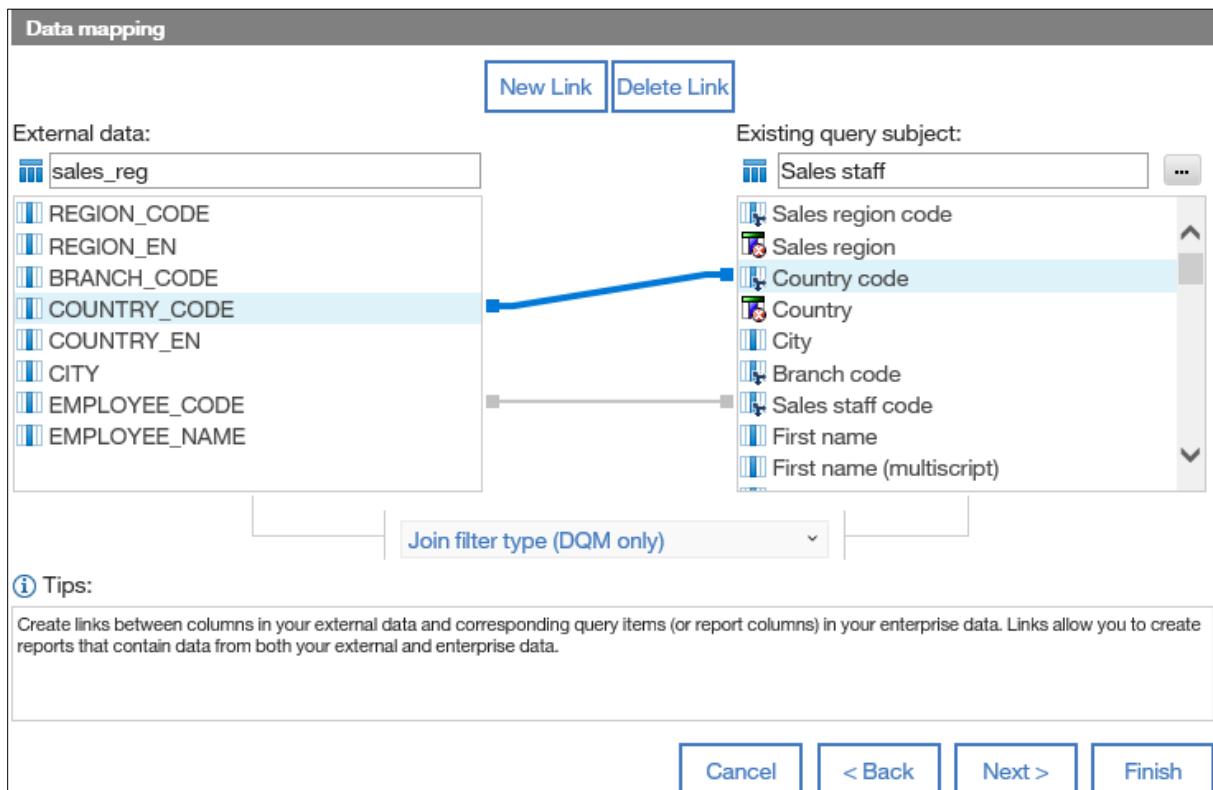
1. On the **Data Mapping** page, click the ellipsis under **Existing query subject / report**, and then click **Choose Query Subject**.
2. Under the **Sales target (query)** namespace, click the **Sales staff** query subject, and then click **OK**.

The Existing query subject pane is populated with the available query items. You will now define the relationships between the query items in the external data and the query items in the corporate data of the existing query subject.

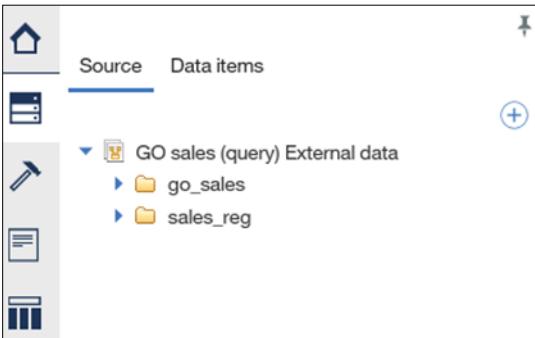
3. Click **New Link**.
A line appears between the panes, defaulting to the first item in each list. You will define the relationship by selecting the items for the new link.
4. In the **External data** list click **EMPLOYEE_CODE**, and then in the **Existing query subject** list click **Sales staff code**.
5. Click **New Link** again, then in the **External data** list click **COUNTRY_CODE**, and then in the **Existing query subject** list click **Country code**.

You have identified joins between the two sources of data.

The result appears as follows:



6. Click **Next** to define the attributes of the external data.
Data item attributes can be modified on this page, if needed. You do not need to modify any attributes for this demonstration.
7. Click **Next** to define your mappings.
A summary of the items is displayed. You will accept the default handling of values in the data source and in the report results.
8. Click **Finish**.
The External Data wizard closes and the Manage external data dialog box displays the new package name. The default name uses the original package name, and appends External data.
9. Once you are satisfied with the package name, click **Publish**.
The new package will be automatically published to the My content folder.
10. Click **Yes** if prompted to replace the existing package.
The system updates the model metadata, and validates the currently open report against the new package.
11. Click **OK** after the report is validated.
Notice that the report design has not changed, but notice that the Source tab has updated to the newly created package that includes the new external data.
The results appear as follows:

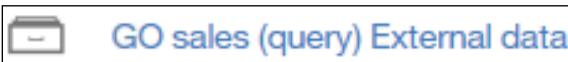


Task 4. Use the external data file in a report

You will look at the new package then return to the report that is now using it.

1. In the side bar, click **Home** , then click **My content**.

You can see the new package.



2. On the Application bar, in the document list, click **Welcome**, then click **Staff Sales Targets *** to switch back to the updated report.
3. From the **Data/Source** tab, expand the **sales_reg** namespace, then expand the **sales_reg** query subject (if not already expanded).
4. Click **COUNTRY_EN**, Ctrl-click **CITY**, then drag them to the beginning of the List, i.e. before <Staff name>.
5. On the Application bar, click **Show properties** .
6. Click <**COUNTRY_EN**>, then in the Properties pane, under DATA ITEM, change **Name** and **Label** to **Country**.
7. Click <**CITY**>, then in the Properties pane, under DATA ITEM, change **Name** and **Label** to **City**.
8. Click <**Country**>, then Ctrl-click <**City**>, then on the Toolbar, click **Group / Ungroup** .
9. Click <**Sales target**>, then on the Toolbar, click **Data format** .
10. In the **Format type** list, select **Currency**.
11. Set the **Number of decimal places** property to **0**.
12. Click **OK** to close the **Data format** dialog.
13. With the <**Sales target**> list column body still selected, on the Toolbar, click **Summarize**  > **Total**.

A section of the report layout appears as follows:

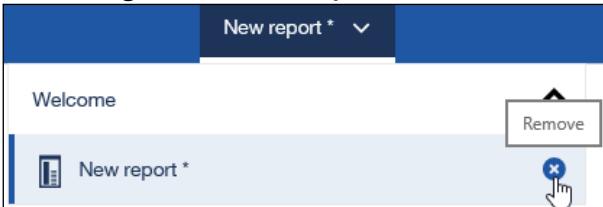
Country	City	Staff name	Sales staff code	Sales target
<Country>	<City>	<Staff name>	<Sales staff code>	<Sales target>
<City> - Total				<Total(Sales target)>
<City>		<Staff name>	<Sales staff code>	<Sales target>
<City> - Total				<Total(Sales target)>
<Country> - Total				<Total(Sales target)>
<Country>	<City>	<Staff name>	<Sales staff code>	<Sales target>
<City> - Total				<Total(Sales target)>
<City>		<Staff name>	<Sales staff code>	<Sales target>
<City> - Total				<Total(Sales target)>
<Country> - Total				<Total(Sales target)>
Overall - Total				<Total(Sales target)>

14. On the Application bar, click **Run options** , then click **Run HTML**.
A section of the result appears as follows:

Country	City	Staff name	Sales staff code	Sales target	
Australia	Melbourne	Dave Smythe	10090	\$15,084,300	
		Jake Cartel	10092	\$3,786,500	
		John Sinden	10093	\$4,462,400	
		Alice Walter	10089	\$16,834,700	
		Jackie Fulford	10527	\$17,965,800	
		Donald Neely	10526	\$997,200	
		Donald Ward	10091	\$18,036,200	
		Jonathan Farrel	10773	\$2,073,100	
Melbourne - Total				\$79,240,200	
Australia - Total				\$79,240,200	
Austria	Wien	Alexandra Klauser	10489	\$1,577,400	
		Sabine Grüner	10124	\$29,116,000	
		Jutta Shulz	10765	\$23,595,900	
		Thomas Schirmer	10488	\$20,733,300	
		Wien - Total			\$75,022,600
Austria - Total				\$75,022,600	

15. Close the rendered report tab.

16. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



17. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You received a .csv file from the Human Resources manager, which included updated city and country locations for all employees. You created a report with employee details using the updated employee location information in the external file.

Unit summary

- Create a report that displays summarized data before detailed data
- Highlight alternate rows in a list report
- Use single data items to summarize report information
- Create a report using an external data file

Unit 6 Examine the report specification

IBM Training



Examine the report specification

IBM Cognos Analytics (v11.0)

© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM.

Unit objectives

- Examine the report specification structure
- Modify a report specification
- Extend existing report templates and themes

Examine the report specification

© Copyright IBM Corporation 2018

Unit objectives

Work with the report specification

- Reports consist of report specifications created in XML.
- The report specification defines the report contents:
 - queries and filters used to retrieve data
 - prompt pages and report pages, including data containers (Lists, Crosstabs, and so on) used to display data
 - other objects in the report layout and the styles used to format these objects

[Examine the report specification](#)

© Copyright IBM Corporation 2018

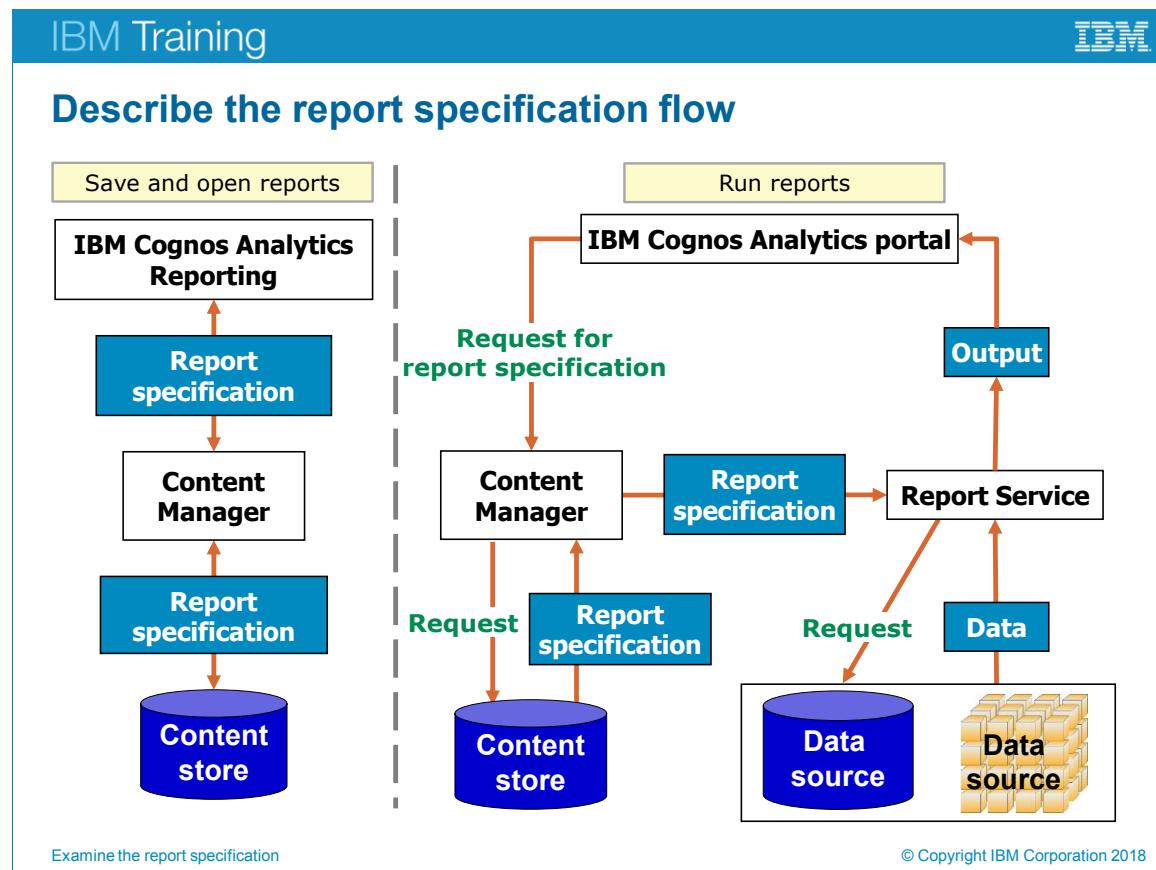
Work with the report specification

When you create a report, you are actually creating a report specification. The report specification is the code that defines the queries and prompts that are used to retrieve data and the layouts and styles used to present the data.

You can use the specification for a report to:

- view the XML code to better understand report contents
- modify a report by editing the XML code in the report specification
- share report specifications with report authors working in different environments
- copy the specification code for a specific report object such as an image or a page header and then reuse this code to add this object to other reports

You can create a report specification entirely in XML, independent of the user interface and then open the report in IBM Cognos Analytics - Reporting. While the report specification can appear complex, it is possible to work with and modify the code with a basic understanding of how it works.



Describe the report specification flow

When you create a report, the specification is created in the browser on the local client machine. When you save a report, IBM Cognos Analytics - Reporting sends the specification to Content Manager. Content Manager then stores the specification in the content store.

When you run a report, the request is sent in two parts:

1. To the Content Manager to retrieve the report specification from the content store.
2. To the business database to retrieve report data.

Once data is returned from the database, and the report specification is returned from Content Manager, the two are displayed in a new browser tab by the report service.

If you edit a previously-saved report, changes to the report specification are applied only on your client machine until you save the report. Once saved, the specification is updated in the content store.

When you run a report in IBM Cognos Analytics - Reporting, Content Manager does not retrieve the specification. Because the specification is already open, IBM Cognos Analytics - Reporting sends the specification.

Describe the report specification structure

- A report specification has four main sections:
 - header section
 - the package it is referring to
 - queries section
 - Queries: data items, query calculations, filters
 - layouts section
 - Report pages
 - Prompt pages
 - Classes
 - report variables section
 - Conditional variables

[Examine the report specification](#)

© Copyright IBM Corporation 2018

Describe the report specification structure

Depending on the report, the report specification may not include these sections.

Demonstration 1

Examine a report specification and copy it to the clipboard

```
- <listColumnBody>
  - <style>
    - <defaultStyles>
      <defaultStyle refStyle="lc"/>
    </defaultStyles>
  </style>
  - <contents>
    - <textItem>
      - <dataSource>
        <dataItemValue refDataItem="Order method type"/>
      </dataSource>
    </textItem>
  </contents>
</listColumnBody>
```

Examine the report specification

© Copyright IBM Corporation 2018

Demonstration 1: Examine a report specification and copy it to the clipboard

Demonstration 1: Examine a report specification and copy it to the clipboard

Purpose:

To learn more about report specifications, you want to create a simple list report and view the XML specification. You will copy this report specification to the clipboard, and then re-open the specification in IBM Cognos Analytics - Reporting.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Create a List report

1. Create a new **Blank** report using the **GO data warehouse (query)** package.
2. Add a **List** to the report page.
3. From the **Data\Source** tab, add the following query items to the List:
 - Order method: **Order method type**
 - Sales fact: **Revenue**

	Order method type	Revenue
<Order method type>	<Revenue>	
<Order method type>	<Revenue>	
<Order method type>	<Revenue>	

Task 2. View the report specification.

1. On the side bar, click **Pages** , right-click **Report** , then click **Show specification**.

The Report Specification for the report displays in XML. The browser window includes controls to expand and collapse the XML elements.

This report specification contains multiple sections: the specification header section, the queries section, the layouts section, and the XMLAttributes section.

The header section specifies the namespace the report uses as well as the language and the package it is using.

```
- <report xmlns="http://developer.cognos.com/schemas/report/14.2/" useStyleVersion="11.4"
expressionLocale="en-us">
  <modelPath type="FM">/content/folder[@name='Samples']/folder
  [@name='Models']/package[@name='GO data warehouse (query)']/model[@name='model']
</modelPath>
  <drillBehavior modelBasedDrillThru="true"/>
```

The queries section contains information about the query used to retrieve data for the List report (as seen in the <queries> section).

The layouts section describes the objects that appear on Page1 of the report (as seen in the <layouts> section).

The XMLAttributes section provides additional information about the XML elements used in the report specification.

2. In the specification, click the - to the left of a tag to collapse it.

The window within the dialog does not allow direct copy and paste of its content, however, the dialog box does allows you to copy the entire content via the



Copy to clipboard button , found in the bottom left corner.

3. Scroll down to the <listColumnBody> section.

The specification for the Order method type column body displays.

The details of the Order method type column body specify that this object is formatted using the "lc" style (list column) and that the data source for this column body is the Order method type data item.

```

- <dataSource>
  <dataItemLabel refDataItem="Order method type"/>
</dataSource>
<textItem>
</contents>
</listColumnTitle>
- <listColumnBody>
  - <style>
    - <defaultStyles>
      <defaultStyle refStyle="lc"/>
    </defaultStyles>
  </style>
  - <contents>
    - <textItem>
      - <dataSource>
        <dataItemValue refDataItem="Order method type"/>
      </dataSource>
    </textItem>
  </contents>
</listColumnBody>

```

4. Close the Report specification XML window.

Task 3. Copy the report specification and open the report from the clipboard.

1. From the **Pages** tab, right-click **Report**, then click **Copy report to clipboard**.
2. From the side bar, click **New** +, and then click **Report** .

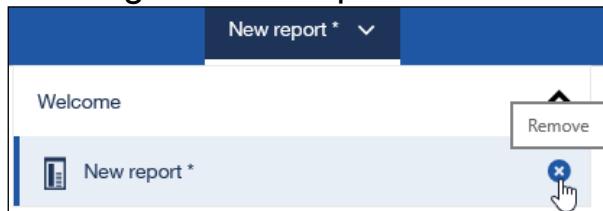
3. Double-click the **Blank** template.
4. On the side bar, click **Pages**, then right-click **Report**, then click **Open report from clipboard**.

IBM Cognos Analytics - Reporting uses the specification you copied to the clipboard, to open the report you created earlier in this demonstration.

A section of the result appears as follows:

Order method type	Revenue
<Order method type>	<Revenue>
<Order method type>	<Revenue>
<Order method type>	<Revenue>

5. Close the rendered report tab.
6. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



7. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You created a simple list report and viewed its XML specification. You then copied the report specification to the clipboard and re-opened it in IBM Cognos Analytics - Reporting.

Considerations when modifying a report specification

- Ensure the XML code creates a valid specification.
- For example:
 - close all tags with </...>
 - ensure XML syntax for each tag is correct
 - preserve the hierarchy of tags
(for example: layouts\layout\reportPages\page)

Examine the report specification

© Copyright IBM Corporation 2018

Considerations when modifying a report specification

Because some XML editors, such as Notepad, do not validate the report specification, it is important to use the correct syntax.

Otherwise, when you edit a report specification in an external editor and then run the report in IBM Cognos Analytics - Reporting, you may receive error messages. The error message describes the problems that were identified and need to be corrected.

If the report specification is written in a language that is not ANSI-supported, you should save the specification file with UTF-8 encoding.

To edit a report in an environment that lets you validate report syntax and structure, edit the report in IBM Cognos Analytics - Reporting instead of working directly with the report specification.

Demonstration 2

Modify a report specification



Sales Performance for Northern Europe
Sales Targets vs. Actual for 2013 Q1



Sales Performance for Northern Europe
Sales Targets vs. Actual for 2013 Q1

Examine the report specification

© Copyright IBM Corporation 2018

Demonstration 2: Modify a report specification

Demonstration 2: Modify a report specification

Purpose:

You want to update a report header to reflect new branding changes. To do this most efficiently, you will modify the report specification.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Application: IBM Cognos Analytics - Reporting

Task 1. Open a report and copy the report specification.

1. Navigate to Team content\SamplesLG\Models\GO Data Warehouse (query)\Report Studio Report Samples.
2. Right-click **Bursted Sales Performance Report**, then click **Edit report**.
Hint: Click OK if you get a message that the report/package has been updated.
3. Click the image in the report title, then from the Toolbar, click **Select ancestor**, then click the first parent **Table cell**.
4. From the Toolbar, click **Border** 
5. Set the **Width** to **2 1/4**, **Color** to **Red**, then under Preview, click **Apply All Borders** , then click **OK**.
6. From the Toolbar, click **Select ancestor** , then click the first parent **Table**.
7. Repeat steps 4 and 5 to apply a border to the title table.

The results appear as follows:



Management has decided that the orange found in the list is more in line with corporate colors and would like you to change the border accordingly. You could select each object change the color manually, but we will explore how to change both objects at one time by modifying the report specification with this change. You will copy the report specification to the clipboard, and paste it into a text editor to make the changes.

8. From the side bar, click **Pages**, then right-click **Report**, then click **Copy report to clipboard**.
9. From the Windows **Start** menu, launch **Notepad**.
Hint: If Notepad is not available directly from the Start menu, click **Windows Accessories**, and then click **Notepad**.
10. From the **Edit** menu, click **Paste**, to paste the report specification from your clipboard.

Task 2. Modify the report specification.

You want to replace the red with orange (#F78E06). There is no other red in the report, so you can replace all occurrences of red in your specification. If there were some instances of red to remain in the report, you could do a search for each instance and decide what to change, based on the element in the report specification.

1. From the **Edit** menu, click **Replace**.
2. In the **Find what** box, type **red**, in the **Replace with** box type **#F78E06**, and then click **Replace All**.

#F78E06 is the hex code for the color used in the report.

3. Close the **Replace** dialog box.

If you had further modifications to the report specification, you could do them now, such as changing the header text or other elements. Ensure that the replacement syntax that you use is valid. At this time, you will incorporate only the change from red borders to orange borders.

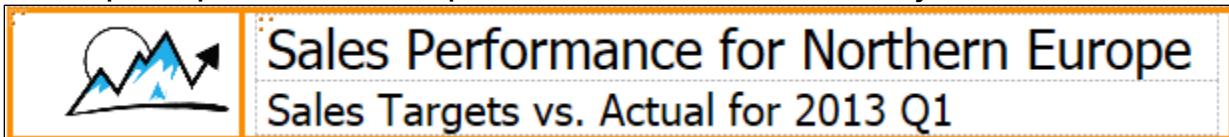
4. From the **Edit** menu, click **Select All**, and then with the specification selected, from the **Edit** menu, click **Copy**.

The modified specification is copied to your clipboard.

Task 3. Open the updated specification from the clipboard.

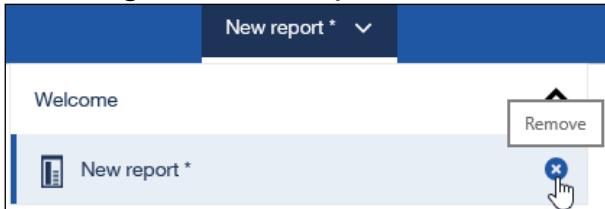
1. In the **Pages** tab, right-click **Report**, then click **Open report from clipboard**.
2. Do not save changes to the currently open report if prompted.

The report opens, with the updated border color on the objects in the header.



You modified the report specification to efficiently update the report.

3. Close **Notepad** without saving changes.
4. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



5. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You updated a report header to reflect new branding changes by modifying the report specification.

IBM Training IBM

Create custom report templates and themes

- Create your own report templates and themes:
 - Convert a report to a template
 - Save templates in **Team content\Templates**
 - Save themes in **Team content\Templates\Style Reference Reports**

Save as

Convert to Template

Convert to Active Report

Examine the report specification

© Copyright IBM Corporation 2018

Create custom toolbox and template objects

If the predefined templates in IBM Cognos Analytics don't meet your requirements, you can define your own set of templates and themes, and make them available to other users.

This allows you to create ready-to-use starting points that keep same look for your corporate-wide reports.

When saving your own template or theme, do not use the name of an existing Cognos Analytics template or theme. Otherwise, your template or theme might be overwritten when Cognos Analytics is upgraded to a later version.

Any report or theme that is saved in the Templates folder appears in the list of templates or themes when you create a new report.

Tip: to give your template or theme an icon in the list of available templates and themes, save the icon as an SVG with the same name as the template or theme, replacing any blank spaces in the name with the underscore (_) character.

For example, if you create a template that is named My template, save the icon as My_template.svg. Then, ask your administrator to add the icon to the folder <install_location>\webcontent\b1\path\images\templateReports for templates, or <install_location>\webcontent\b1\path\images\themeReports for themes.

Unit summary

- Examine the report specification structure
- Modify a report specification
- Extend existing report templates and themes

Examine the report specification

© Copyright IBM Corporation 2018

Unit summary

Distribute reports through bursting

IBM Training



Distribute reports through bursting

IBM Cognos Analytics (v11.0)

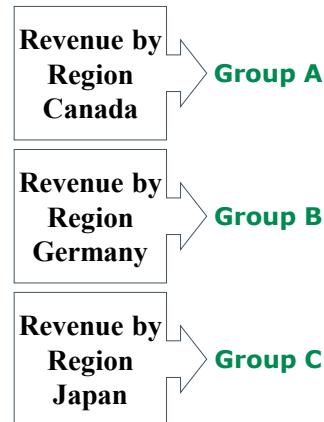
© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM.

Unit objectives

- Distribute reports using bursting
- Create burst keys
- Identify report recipients and data items using burst tables
- Distribute reports using email and IBM Cognos Analytics portal
- Burst dimensional containers

Burst reports

- When you burst a report, you run a report once and divide the results for distribution to multiple recipients
- Each report recipient will only see the subset of data to which they have access to
- Steps
 - 1) Prepare burst recipients in IBM Cognos Analytics - Reporting
 - 2) Set burst options in IBM Cognos Analytics - Reporting
 - 3) Run the report in IBM Cognos Analytics portal with bursting enabled



Distribute reports through bursting

© Copyright IBM Corporation 2018

Burst reports

Bursting is the process of running a report once and then dividing the results for recipients who each view a subset of the data. For example, your company may have sales people in different regions who require the sales target for each country; you can use burst reports to send each salesperson the information that they need.

The burst recipient is a data item in your report that determines who will receive the burst reports. It can come from an existing employee table or a custom burst table that you create.

A burst table maps recipients or groups of recipients to specific data that they are allowed to see and can be a separate table in your data source.

In order to burst a report, you must add a data item to the report that tells IBM Cognos Analytics who the intended recipients are and how you intend to distribute the reports.

You can choose to burst to email addresses or to a directory (in the Burst options). IBM Cognos Analytics leverages the existing security infrastructure in the namespace to dynamically obtain email addresses.

Just as with regular reports, burst reports can be distributed in multiple formats (Excel, PDF, HTML, CSV, XML... etc.).

1) Prepare burst recipients

- You must specify a burst recipient to define who will receive the report.
- You can choose to burst to directories or email addresses or both.
- E-mail can come from:
 - The e-mail address in the authentication source
 - User's profile in IBM Cognos Analytics
 - The data source
 - A calculated field
 - A burst table

Distribute reports through bursting

© Copyright IBM Corporation 2018

Add burst recipients

Burst recipients can be users, groups, roles, email contacts, or email distribution lists.

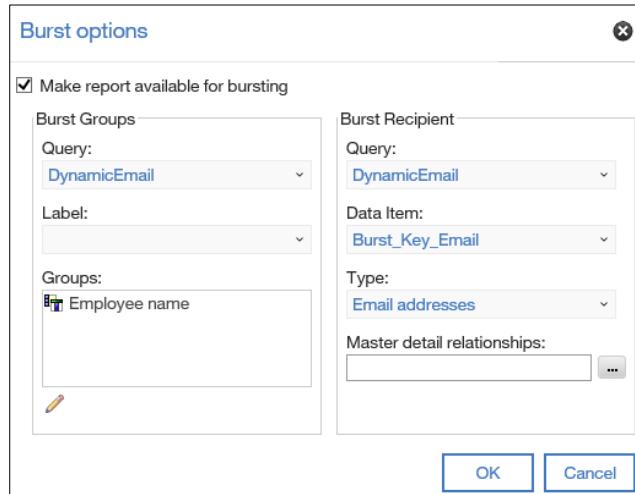
When you burst a report to e-mail, you can:

- reference the e-mail address in the authentication source
- use the e-mail address as it is stored in your data source
- supply the e-mail address in the report itself by creating a calculated field
- reference the appropriate burst table column

If the authentication provider does not contain email information, the email address can be manually entered in the IBM Cognos Analytics portal in the user's personal information. Privileges are set differently at each organization. By default, a report author does not have the necessary rights to make changes to the personal information of other users, in which case your administrator may need to do this for you.

2) Set burst options in IBM Cognos Analytics - Reporting

- Group the report on the data item to burst on
- Burst groups define how the data should be separated
- Burst recipient determines who receives the subset of data
- Burst key defines how the report is burst to its recipients



Distribute reports through bursting

© Copyright IBM Corporation 2018

Set burst options in IBM Cognos Analytics - Reporting

If you burst on a query item that is not a part of the layout, the same report will be distributed to all the burst recipients. This allows you to use the burst functionality to limit who receives the report, even when everyone who has access to the report sees the same thing. This is like drilling through with no parameters in the target.

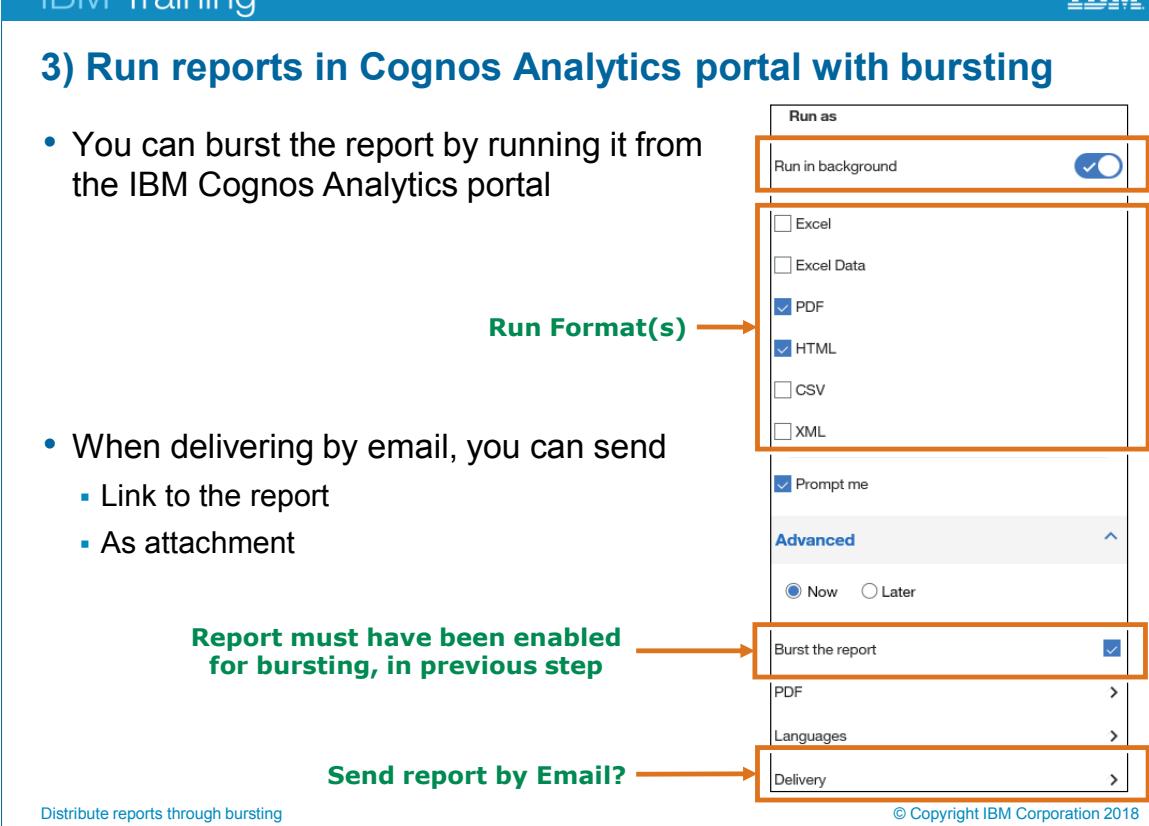
If you only want to e-mail the report to various recipients and not burst to the portal, your burst recipient must be an email address. If you want the report to be emailed as well as burst to the portal, you must specify your Burst Recipient Type to be Directory Entries.

The Label field in the Burst options dialog box determines how the reports are labeled in the IBM Cognos Analytics portal. For example, if you use a country data item as the label, the report will be named after the appropriate country. If you choose the burst key, the report will be named after the burst key value, such as CAMID(":Australia"). If no label is set, the default label is the data item that is being grouped.

IBM Training 

3) Run reports in Cognos Analytics portal with bursting

- You can burst the report by running it from the IBM Cognos Analytics portal



Run Format(s) →

Report must have been enabled for bursting, in previous step →

Send report by Email? →

Distribute reports through bursting © Copyright IBM Corporation 2018

Enable burst reports in IBM Cognos Analytics portal

To burst the report, run the report with options in IBM Cognos Analytics portal. In the advanced options, select the Burst the report check box. This box is only available if the report has the check box selected for Make report available for bursting.

You can burst the report immediately or schedule it to run later.

If you intend to email the reports, you must select the Send the report by email option as well. When emailing the report, you have the option of sending the report as an attachment, sending a link to the report, or both.

IBM Cognos lets administrators disable the ability to send reports as email attachments to ensure that only approved users receive reports. If this is the case, you may still send links to reports. Because these links are accessed through the IBM Cognos system, users must log on to IBM Cognos Analytics portal to access a report.

Demonstration 1

Burst a report to email recipients using a data item

Product Sales Report by Sales Rep				
Product line	Product type	Product	Quantity	Revenue
Alessandra Torta				
Camping Equipment	Cooking Gear	TrailChef Canteen	4,539	55,042.06
		TrailChef Cook Set	8,516	432,851.68
		TrailChef Cup	14,887	48,060.15
		TrailChef Deluxe Cook Set	5,260	634,770.66
		TrailChef Double Flame	2,257	306,593.1
		TrailChef Kettle	8,060	101,092.81
		TrailChef Kitchen Kit	6,234	142,467.02
		TrailChef Single Flame	8,164	512,200.86
		TrailChef Utensils	7,506	140,241.37
		TrailChef Water Bag	24,990	144,791.08
Cooking Gear - Total		90,413	2,518,110.79	

Distribute reports through bursting

© Copyright IBM Corporation 2018

Demonstration 1: Burst a report to email recipients using a data item

Demonstration 1:

Burst a report to email recipients using a data item

Purpose:

Three sales representatives at The Sample Outdoors Company need to know the total number of items that they have sold of each product line and type. However, employees need to see only data pertaining to their own sales. You will burst a product sales report so that each sales representative will see only his or her sales totals.

The mail server must be started before starting this demo. To check the mail server:

1. Click **Start > Windows Administrative Tools > Services**
2. Check that the **hMailServer** service is Running

Refer to the Preface if you need more information about required services.

Portal: <http://vclassbase/ibmcognos>

User/Password: **brettonf/Education1**

Package: **Team content\Samples\Models\GO data warehouse (query)**

Folder: **Sales and Marketing (query)**

Namespace: **Sales (query)**

Task 1. Create a report with product sales information

1. Create a new **Blank** report using the **GO data warehouse (query)** package.
2. Add a **List** to the report page.
3. From the **Data\Source** tab, add the following query items to the List:
 - Employee by region: **Employee name**
 - Products: **Product line**, **Product type**, **Product**
 - Sales fact: **Quantity**, **Revenue**

Employee name	Product line	Product type	Product	Quantity	Revenue
<Employee name>	<Product line>	<Product type>	<Product>	<Quantity>	<Revenue>

4. Select the **<Employee name>**, **<Product line>** and **<Product type>** List column bodies, and then from the Toolbar, click **Group / Ungroup**. You can use Ctrl-click or Shift-click methods to select multiple columns.
5. Click the **<Employee name>** List column body, then from the Toolbar, click **Headers & footers**, and then click **Create header**.
6. With the **<Employee name>** List column body still selected, from the Toolbar, click **More**, then click **Delete** to remove the redundant column.

- Click the <Quantity> and <Revenue> List column bodies, then from the Toolbar, click **Summarize**, and then click **Total**.

A section of the result appears as follows:

Product line	Product type	Product	Quantity	Revenue
<Employee name>				
<Product line>	<Product type>	<Product>	<Quantity>	<Revenue>
		<Product type> - Total	<Total(Quantity)>	<Total(Revenue)>
		<Product type>	<Quantity>	<Revenue>
		<Product type> - Total	<Total(Quantity)>	<Total(Revenue)>
<Product line> - Total			<Total(Quantity)>	<Total(Revenue)>

To include only the data for the three sales reps that need to know what they have sold, you will filter the report.

- From the Toolbar, click **Filters**, **Edit Filters**, and then click **Add**.

- Click **Advanced**, and then click **OK**.

- Create and validate the following expression:

[Employee name] in ('Alessandra Torta', 'Bart Scott', 'John Sinden')

Hint: Drag Employee name from Data items tab. Then type **in**, then click **Select multiple values** .

This filter is important; because the environment does not have mailboxes for all employees. If you ignore it, Cognos will get errors for missing mailboxes.

- Click **OK** to close each open dialog box.

- On the Application bar, click **Run options** , then click **Run HTML**.

A section of the result appears as follows:

Product line	Product type	Product	Quantity	Revenue
Alessandra Torta				
Camping Equipment	Cooking Gear	TrailChef Canteen	4,539	55,042.06
		TrailChef Cook Set	8,516	432,851.68
		TrailChef Cup	14,887	48,060.15
		TrailChef Deluxe Cook Set	5,260	634,770.66
		TrailChef Double Flame	2,257	306,593.1
		TrailChef Kettle	8,060	101,092.81
		TrailChef Kitchen Kit	6,234	142,467.02
		TrailChef Single Flame	8,164	512,200.86
		TrailChef Utensils	7,506	140,241.37
		TrailChef Water Bag	24,990	144,791.08
Cooking Gear - Total				2,518,110.79

- Click **Page down** to scroll through the report, and view the other reps.

The report is filtered to only include sales made by Alessandra Torta, Bart Scott, and John Sinden.

You will now create a burst key data item that will let you distribute this report to the sales reps using their email addresses.

Task 2. Create a burst key

You use the Email data item as a burst key for burst recipients.

1. On the side bar, click **Queries** , then click **Query1**.
2. From the side bar, click the **Data**  tab.
3. From the Source tab, under Employee by region, drag **Email** to the **Data Items** pane.
4. On the side bar, click **Pages** , then click **Page1**.
5. At the top left corner of the List, click **Select List1**  to select the entire List.
6. On the Properties pane, under the DATA section, double-click **Properties** then select the **Email** check box, and then click **OK**.

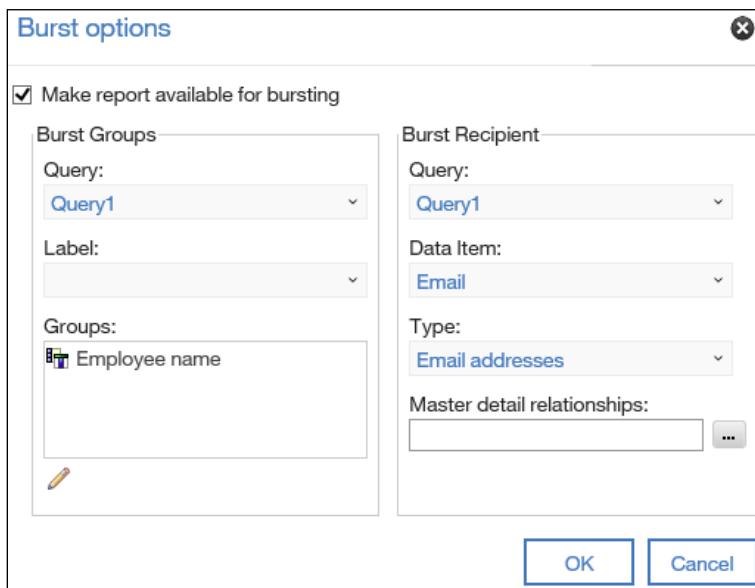
You are making this data item a part of the List.

Task 3. Set the Burst options

1. In the **Pages** tab, click **Report** .
2. On the Properties pane, under RUNNING & VALIDATING, double-click **Burst options**.
3. Check the **Make report available for bursting** check box to select it.
This check box sets a flag for IBM Cognos Analytics portal to allow bursting.
4. In the **Burst Groups** section, from the **Query** list, select **Query1**.
You will not enter a value for the Label, because you will use the default label, Employee name.
5. Below **Groups**, click **Edit**, double-click **Employee name** to add it to the **Groups** folder, and then click **OK** to close the **Grouping & sorting** dialog box.
6. In the **Burst Recipient** section, from the **Query** list, select **Query1**.
7. From the **Data Item** list, select **Email**.

8. From the **Type** list, select **Email addresses**.

A section of the result appears as follows:



With these options, the report will be distributed by email (Type) to the email address specified in Burst Key. The report is grouped by Employee name, and each employee only receives the report with their data.

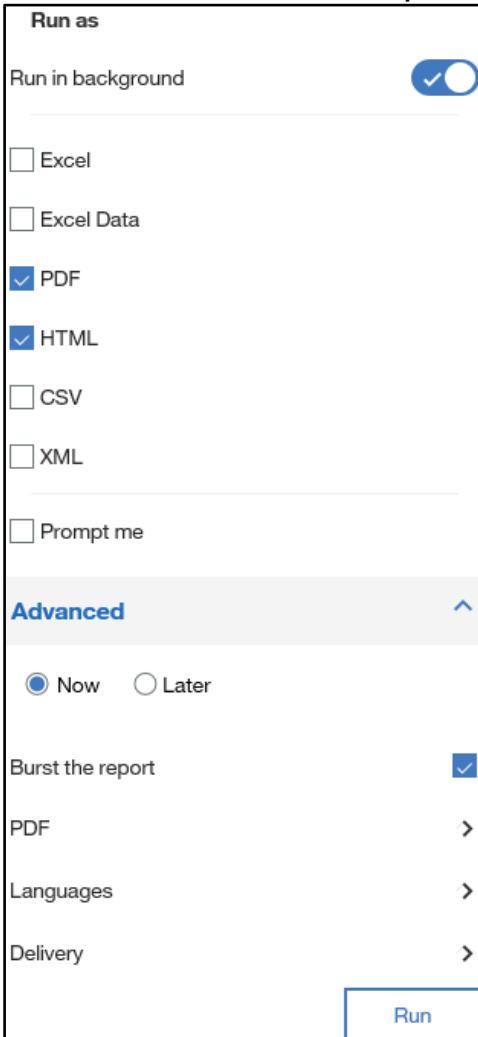
By selecting Email addresses (in the Burst options dialog box), you are indicating that the email address is available in the Email data item.

9. Click **OK**.
10. On the Application bar, click **Save**, and save the report in **Team content\B6059** as **Product Sales Report by Sales Rep**.
11. In the Application bar, click the report dropdown menu, then click **Remove** to the right of the report name to close it.

Task 4. Burst the report

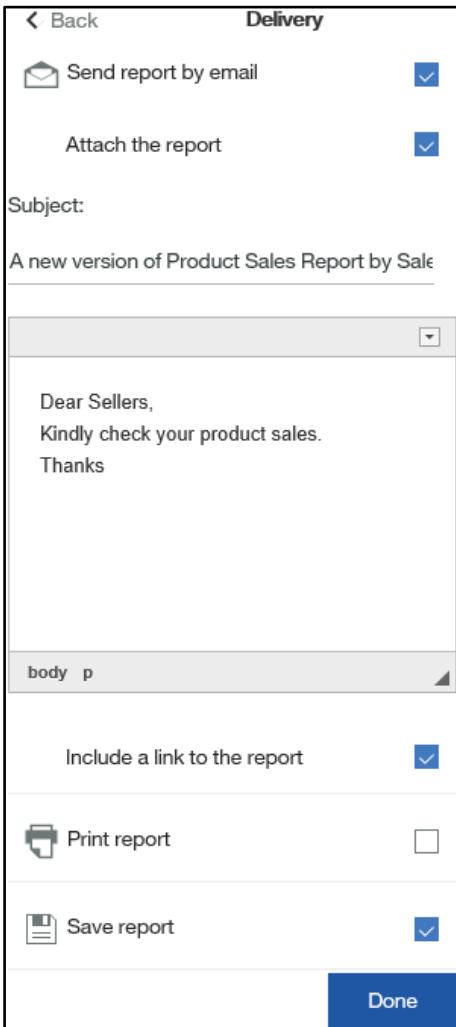
1. In the side bar, click **Home** , then navigate to **Team content\B6059**.
 2. Right-click the **Product Sales Report by Sales Rep**, and then click **Run as**.
 3. Click the **Run in background** button to enable it (should show a checkmark on the toggle switch), then check **HTML** and **PDF**, and then expand **Advanced**.
 4. Uncheck **Prompt me** because the report has no prompts.
 5. Click the **Burst the report** check box to select it.
- This checkbox would not be available if you didn't enable the report for bursting

in *Task 3. Set the Burst options*, step 3.



6. Expand the **Delivery** section.
7. Check both **Send this report by email** and **Attach the report**.
8. In the Subject field write a message to be sent in the email:
Dear Sellers,
Kindly check your product sales.
Thanks

9. Make sure **Include a link to the report** and **Save the report** are checked.



10. Click **Done**.
11. Click **Run**.
12. After a couple of minutes (to allow the report versions to render), click the **Team content\B6059** page, right-click the **Product Sales Report by Sales Rep** report, and then click **View versions**.

13. In the **Versions** flyout, click the most recent timestamp.
The output versions should appear like the following:

The screenshot shows a 'Versions' flyout window. At the top, there are tabs for 'Versions' (which is selected) and 'Archives'. Below the tabs is a checkbox labeled 'Show all history'. The main area lists three versions, each with a preview icon, a timestamp, a name, and two small icons. The first version is for 'Dec 18, 2017 12:02 AM' by 'Alessandra Torta'. The second version is for 'Bart Scott' on the same date. The third version is for 'John Sinden' on the same date.

Timestamp	Name
Dec 18, 2017 12:02 AM	Alessandra Torta
Dec 18, 2017 12:02 AM	Bart Scott
Dec 18, 2017 12:02 AM	John Sinden

Task 5. View the results in the portal

1. Open another (private) browser session.
2. Browse to the IBM Cognos Analytics portal: <http://vclassbase/ibmcognos>
3. Log in using **tortaa/Education1**.
4. Navigate to **Team content\B6059**.
5. Click the **Product Sales Report by Sales Rep** report.

Notice that the report does not run again, it opens the saved version of Alessandra Torta immediately; because it is saved in the portal for her.
A section of the results appears as follows:

The screenshot shows a report titled 'Product Sales Report by Sales Rep' for 'Alessandra Torta'. The report has a blue header bar with a 'Run' button, a dropdown menu, and a '...' button. The main body is a table with columns: Product line, Product type, Product, Quantity, and Revenue. The table shows sales data for Alessandra Torta across various product lines and types, with a total row at the bottom.

Product line	Product type	Product	Quantity	Revenue
Alessandra Torta				
Camping Equipment	Cooking Gear	TrailChef Canteen	4,539	55,042.06
		TrailChef Cook Set	8,516	432,851.68
		TrailChef Cup	14,887	48,060.15
		TrailChef Deluxe Cook Set	5,260	634,770.66
		TrailChef Double Flame	2,257	306,593.1
		TrailChef Kettle	8,060	101,092.81
		TrailChef Kitchen Kit	6,234	142,467.02
		TrailChef Single Flame	8,164	512,200.86
		TrailChef Utensils	7,506	140,241.37
		TrailChef Water Bag	24,990	144,791.08
Cooking Gear - Total			90,413	2,518,110.79

6. Close this browser window (the session of Alessandra Torta).

Task 6. View the results in the e-mail client

1. In the taskbar, click **Start > Mozilla Thunderbird** to open the e-mail client.
2. Under **bscott@grtd123.com** (the e-mail of Bart Scott), click **Inbox**.

The e-mail message appears in the Inbox.

The screenshot shows the Mozilla Thunderbird interface. The left sidebar displays a tree view of email accounts: admin@grtd123.com (Inbox, Sent, Trash), atorta@grtd123.com (Inbox (1), Trash), bscott@grtd123.com (Inbox (1), Trash), jsinden@grtd123.com (Inbox (1), Trash), and Local Folders (Trash, Outbox). The main pane shows an incoming email from FBretton@grtd123.com with the subject "A new version of Product Sales Report by Sales Rep is available". The message body contains a greeting, a call to action ("Kindly check your product sales."), and thanks. Below the message are links to "Bart Scott HTML" and "Bart Scott PDF". At the bottom of the message pane, it says "—Product Sales Report by Sales Rep.html—". A table follows, showing sales data:

Product line	Product type	Product	Quantity	Revenue
Bart Scott				
Camping Equipment	Cooking Gear	TrailChef Canteen	8,053	91,936.17
		TrailChef Cook Set	12,499	652,232.37

At the bottom of the table, there is a link to "2 attachments 153 KB".

You can see the body that you wrote for the email.

The sender of the e-mail is the user who run the report (FBretton@grtd123.com).

At the end of the email you can see links to the report, as well as the report attached as HTML and PDF.

3. Close the e-mail client.
4. Return to **IBM Cognos Analytics** portal, and leave it open for the next demonstration.

Results:

By creating a burst key and setting the appropriate Burst options in IBM Cognos Analytics - Reporting, you were able to burst product sales report to sales reps. Each sales rep received only the data specific to them.

Examine a burst table

- The burst table maps recipients or groups of recipients to specific data that they are allowed to see.
- The burst table typically contains a list of recipients and the data item you will burst on.

CountryCode	Recipients
4	CAMID(":Canada")
3	CAMID(":US")

Examine a burst table

CAMID (Cognos Access Manager ID) is an internal path to users, groups, roles, contacts or distribution lists.

The list of recipients can be either a CAMID, or it could be the alternate search path syntax.

The burst table must also include the data item you intend to burst on. In the slide example, these values are in the CountryCode column. This column will be used to create the appropriate relationship in Framework Manager.

Burst tables are typically created by the Database Administrator (DBA), and the recipient values are provided to the DBA by the report author or system administrator.

After the DBA creates the burst table in the source database, the metadata modeler must import the table's metadata into Framework Manager. The modeler must create a relationship between the burst table query subject and the query subject containing the item that you, the report author, are bursting on.

Examine a mixed recipient list

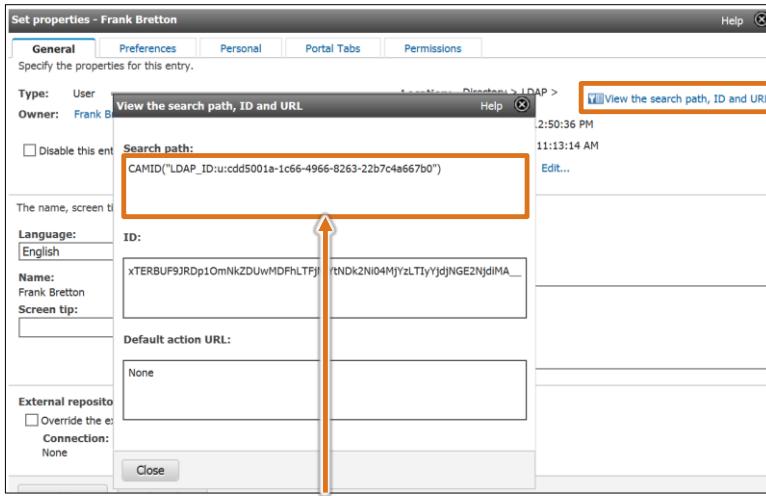
- You can mix the recipient types in your burst table for greater flexibility in your distribution methods.

CountryCode	Recipients
4	CAMID(":Canada")
3	CAMID("/:contact[@name='John Sinden'])
1	CAMID("/:distributionList[@name='European Partners])
4	fretton@grtd123.com
1	CAMID("Local NT ID:u=S-1-5-21-1434109735-2681017343-4103935507-1037")

Examine a mixed recipient list

If you are using the alternate search path syntax, you must set this option to Directory entries. This is because the syntax contains the @ symbol which will be mistaken for an e-mail address.

Obtain burst recipient values for LDAP namespaces



CAMID("LDAP_ID:u:cdd5001a-1c66-4966-8263-22b7c4a667b0")

Alternate Search Path Syntax:

/directory/namespace[@name="LDAP_ID"]//account[@userName="brettonf"]

Obtain burst recipient values for LDAP namespaces

You can obtain values for burst recipients by examining their properties in IBM Cognos Administration.

You can copy the CAMID, or use an alternate search path syntax that supports user-friendly IDs, into the burst recipient expression.

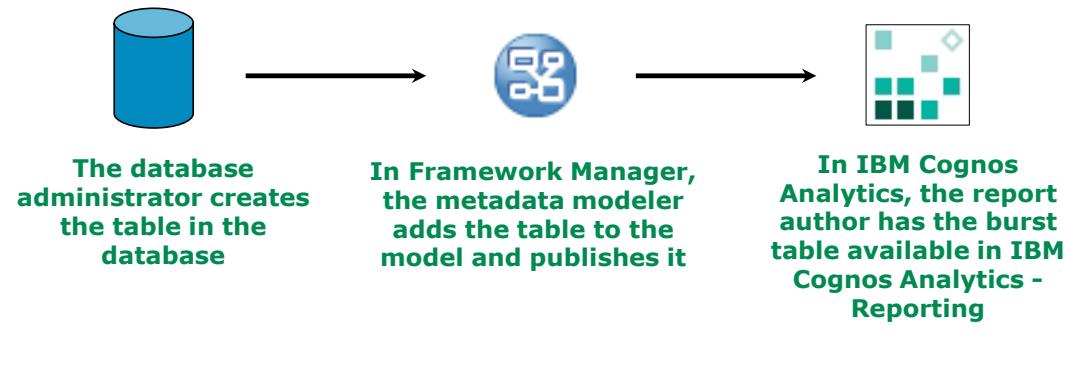
You can burst to both directory and the user's email address if your authentication provider contains the users' email address, or if you enter it manually in IBM Cognos Analytics portal for each user.

The CAMID is accessed from the Properties page of any object. The CAMID is similar to the Member Unique Name (MUN), in that it uniquely identifies objects. The double forward slash preceding the account element indicates that you are searching all accounts under the specified namespace.

If you are using an NTLM or a Series 7 namespace, the user IDs are alphanumeric or numeric. You can create an alternative search path that dynamically generates user IDs based on users' names and not alphanumeric or numeric IDs. Using this CAMID syntax, you can substitute a static user ID (such as brettonf in the slide example) with a dynamically generated user ID when creating your burst key.

The search path for NTLM namespace is of the form: CAMID("Local NT ID: u=S-1-5-21-1004336348-688789844-682003330-1134").

Access a burst table in IBM Cognos Analytics - Reporting



Distribute reports through bursting

© Copyright IBM Corporation 2018

Access a burst table in IBM Cognos Analytics - Reporting

The metadata modeler can set access permissions so that only specific report authors can see the burst table.

Burst tables can be built independently of IBM Cognos Analytics - Reporting or IBM Cognos Analytics using user account information from external authentication providers. This information is extracted from the namespace, arranged to fit the structure of CAMIDs, and then added to a table in a data source.

Burst tables will often be created programmatically since typing each individual record into a database table would be very time consuming.

Most popular programming languages can incorporate libraries that allow them to interact with security software in this way. For example, LDAP security information can be accessed using the Java Naming and Directory Interface (JNDI) libraries.

Considerations when bursting to the portal

- You can burst the report to Team content in the IBM Cognos Analytics portal
- Bursting the report generates report outputs that are specific to the user that is currently logged in

Considerations when bursting to the IBM Cognos Analytics portal

To ensure that a report is burst to the Team content in IBM Cognos Analytics portal, you must specify that you want to burst to Directory entries in the Burst options dialog box in IBM Cognos Analytics - Reporting.

If the Send the report by email option is also selected on the Run with advanced options page, and the email addresses for the recipients are accessible by IBM Cognos Analytics, recipients will also receive the report by email.

After a report is burst, each user will see the data subset specific to them.

To view output versions for a report, you can click the View output versions icon in Team content. Administrators have permission to view all output versions of the report in the Schedule Management tool of the IBM Cognos Analytics portal.

Demonstration 2

Burst a crosstab report to the IBM Cognos Analytics portal using a burst table



Versions	Archives
 Feb 22, 2016 10:22 AM	
CAMID(":Australia")	
EN-US	
CAMID(":Italy")	
EN-US	
CAMID(":US")	
EN-US	

Distribute reports through bursting

© Copyright IBM Corporation 2018

Demonstration 2: Burst a crosstab report to the IBM Cognos Analytics portal using a burst table

Demonstration 2:

Burst a crosstab report to the IBM Cognos Analytics portal using a burst table

Purpose:

You want sales reps to see product revenue that was generated by their country. For example, sales reps from Italy should only see report outputs that contain Italian sales data. To accomplish this, you will use the data in a burst table to burst a report to Team content.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Create a report that displays product revenue by country

1. Create a new **Blank** report using the **GO data warehouse (query)** package.
2. Add a **List** to the report page, and accept the defaults.
3. From the **Data** tab, add the following query items to the List:
 - Employee by region: **Country**
 - Products: **Product line**, **Product type**
 - Sales fact: **Revenue**

Country	Product line	Product type	Revenue
<Country>	<Product line>	<Product type>	<Revenue>

4. Select the **<Country>** and **<Product line>** List column bodies, and then from the Toolbar, click **Group / Ungroup**.
5. Click **<Revenue>**, then from the Toolbar, click **Summarize**, then click **Total**. A section of the result appears as follows:

Country	Product line	Product type	Revenue
< <input checked="" type="checkbox"/> Country>	< <input checked="" type="checkbox"/> Product line>	<Product type>	<Revenue>
<Product line> - Total			<Total(Revenue)>
<Product line>			<Revenue>
<Product line> - Total			<Total(Revenue)>
<Country> - Total			<Total(Revenue)>

6. On the side bar, click **Queries**, and then click **Query1**.

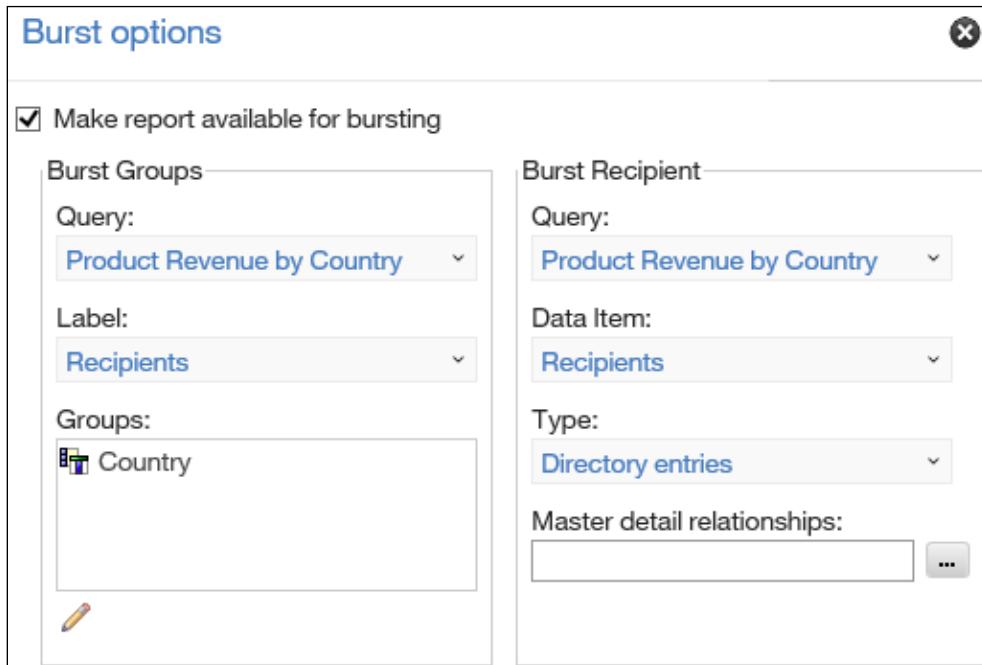
7. On the Application bar, click **Show properties**  to open the Properties pane.
8. In the Properties pane, under the MISCELLANEOUS section, change the Name property to **Product Revenue by Country**.
9. On the **Data\Source** tab, expand the **Burst table by country** query subject, and then drag **Recipients** to the **Data Items** pane.
The Burst table by country table includes the Country CAMID Recipients and the Country Code for each user.
10. On the side bar, click **Pages**, then click **Page1**, and then select the entire List.
11. From the Properties pane, under DATA, double-click **Properties**, and then select the **Recipients** data item check box to make it a property of the List.
12. Click **OK**.

Task 2. Set Burst options

1. On the side bar, click **Pages** , then click **Report** .
2. In the Properties pane, under RUNNING & VALIDATING, double-click **Burst options**, and then select **Make report available for bursting**.
3. In the **Burst Groups** section, in the **Query** list, select **Product Revenue by Country**, and then in the **Label** list, select **Recipients**.
The label identifies the text that appears in the report name when burst to the Web.
4. Below Groups, click **Edit**, double-click **Country** to add it to the **Groups** folder.
5. Click **OK** to close the Grouping & sorting dialog box.

6. In the **Burst Recipient** section, in the **Query** list, select **Product Revenue by Country**, in the **Data Item** list, select **Recipients**, and then in the **Type** list, select **Directory entries**.

A section of the result appears as follows:



A report will be generated for each country, as indicated by the Groups, and will be distributed to each burst recipient - in this case, country. Each report will be labeled by the Recipients, which is the CAMID of a country.

7. Click **OK** to close the **Burst options** dialog box.
8. Save the report in **Team content\B6059** as **Sales by Country**.
9. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of the report name to close it.

Task 3. Burst the report

1. From **IBM Cognos Analytics** portal, navigate to **Team content\B6059**.
2. Right-click **Sales by Country**, then click **Run as**.
3. Enable **Run in background**, and then expand **Advanced**.
4. Click the **Burst the report** check box to select it, and then click **Run**.
5. Navigate to **Team content\B6059**, and right-click the **Sales by Country** report, then click **View versions**.
6. Click the most recent timestamp item to expand.

The result appears similar to the following:

The screenshot shows the 'Versions' tab of a report's history. It lists three output versions, each with a timestamp, a detailed view icon (i >), and a delete icon. The versions are:

- Dec 18, 2017 1:05 AM: CAMID(:Australia) (EN-US)
- Dec 18, 2017 1:05 AM: CAMID(:Italy) (EN-US)
- Dec 18, 2017 1:05 AM: CAMID(:US) (EN-US)

Three output versions were created for this report: one for each of Australia, Italy, and United States. Because you are logged in as Frank Bretton, a report author, you can see all three output versions.

7. On the **Application** bar, click **Frank Bretton** , and then click **Sign out**.
8. Sign in as **scottb\Education1**, and then click **OK**.
9. Navigate to **Team content\B6059**, and then, right-click the **Sales by Country** report, then click **View versions**.
10. Click the most recent timestamp to expand.

A section of the result appears similar to the following:

The screenshot shows the 'Versions' tab of a report's history. It lists one output version, which is expanded to show its details. The version is:

- Dec 18, 2017 1:05 AM: CAMID(:US) (EN-US)

Only one output version is shown, because Bart Scott belongs to the US group.

11. Click the **HTML**  document.

A section of the report appears as follows:

Country	Product line	Product type	Revenue
United States	Camping Equipment	Cooking Gear	43,152,191
		Lanterns	21,597,711.83
		Packs	57,520,074.37
		Sleeping Bags	50,338,776.21
		Tents	90,663,037.65
Camping Equipment - Total			263,271,791.06

12. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of the report name to close it.
13. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

The report that you created shows how much revenue each country generated by product type. You added a burst key to the report, set the Burst options, and burst the report to the Web.

Burst a dimensional container

- You can only add a burst key to a data container with one dimension, such as a List
- If you want to burst a data container with multiple dimensions, such as a Visualization or a Crosstab:
 - create a master-detail relationship between this query and a single dimensional query
 - add the burst key to the single dimensional query

Burst a dimensional container

An easy way to create a master detail relationship for a dimensional data container is to:

1. Create a List object with the item that you want bursting to group on as your first column. This will be the master List
2. Insert your Visualization or Crosstab as the second column of the List. This automatically creates a second query, which will be the detail query.
3. Add the item that you will group on to your Visualization or Crosstab.

In the List, create a section with the item that you want to group.

Demonstration 3

Burst a Crosstab report to the IBM Cognos Analytics portal using a burst table and a master-detail relationship

Country: Australia

Revenue	2011	2012	2013
Camping Equipment	9,752,591.01	19,175,957.2	13,007,383.98
Golf Equipment	4,094,643.54	8,482,438.67	6,502,474.22
Mountaineering Equipment	2,691,279.15	5,861,253.12	5,380,587.79
Outdoor Protection	600,956.77	367,636.38	171,750.41
Personal Accessories	2,131,381.68	5,081,517.25	4,261,477.85

Demonstration 3: Burst a crosstab report to the IBM Cognos Analytics portal using a burst table and a master detail relationship

Demonstration 3:

Burst a crosstab report to the IBM Cognos Analytics portal using a burst table and a master detail relationship

Purpose:

You want sales reps to view a crosstab report of the revenue generated by their country by product line and year. For example, sales reps from Italy will see a report that contains Italian sales data by product line and year. To burst reports to Team content in IBM Cognos Analytics portal, you will use the data in a burst table and create a master detail relationship.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Create a Crosstab within a List

1. Open a new **Blank** report using the **GO data warehouse (query)** package.
2. Add a **List** to the report page, and click **OK** to accept the defaults.
3. From the **Data** tab, add the **Country** query item, from the **Employee by region** query subject, to the List.
4. From the **Toolbox** tab, drag a **Crosstab** to the last column of the List (inside the List).
5. From the **Object and query name** dialog box, rename **Query2** to **Product line Revenue Query**, and then click **OK**.

6. From the **Data\Source** tab, drag the following data items to the Crosstab:
- **Rows:**
 - Products: **Product line**
 - **Columns:**
 - Time: **Year**
 - **Measures:**
 - Sales fact: **Revenue**

The result appears as follows:

Country	Crosstab		
	Revenue	<#Year#>	<#Year#>
<Country>	<#Product line#>	<#1234#>	<#1234#>
	<#Product line#>	<#1234#>	<#1234#>

7. Click **<Country>**, then from the Toolbar, click **Section / Unsection**.
8. On the Application bar, click **Show properties**  to open the Properties pane.
9. Click the **Crosstab** List column title, and then in the **Properties** pane, under the **BOX** section, change the **Box type** property to **None**.
10. In the **Crosstab**, sort **<#Product line#>** and **<#Year#>** in **Ascending** order.
11. On the Application bar, click **Run options** , then click **Run HTML**.

A section of the result appears as follows:

Country: Australia				
Revenue	2010	2011	2012	2013
Camping Equipment	332,986,338.06	402,757,573.17	500,382,422.83	352,910,329.97
Golf Equipment	153,553,850.98	168,006,427.07	230,110,270.55	174,740,819.29
Mountaineering Equipment		107,099,659.94	161,039,823.26	141,520,649.7
Outdoor Protection	36,165,521.07	25,008,574.08	10,349,175.84	4,471,025.26
Personal Accessories	391,647,093.61	456,323,355.9	594,009,408.42	443,693,449.85

Country: Austria				
Revenue	2010	2011	2012	2013
Camping Equipment	332,986,338.06	402,757,573.17	500,382,422.83	352,910,329.97
Golf Equipment	153,553,850.98	168,006,427.07	230,110,270.55	174,740,819.29

Notice that each of the countries has identical data. This is because there is no relationship between the List and the Crosstab, so you see data for all countries. You will add query items to both queries and then create a master-detail relationship to link the queries, which will make the results more meaningful.

12. Close the rendered report tab.

Task 2. Add a query item to the queries

1. On the side bar, click **Queries**, and then click **Query1**.
2. In the Properties pane, under MISCELLANEOUS, rename **Query1** to **Country Query**.
3. From the **Data\Source** tab, add the following items to the **Data Items** pane:
 - Employee by region: **Codes: Country code**
 - Burst table by country: **Recipients**
4. On the sidebar, click **Queries**, then click **Product Line Revenue Query**.
5. From the **Data\Source** tab, add **Country code** from **Employee by region\Codes**.
6. On the sidebar, click **Pages**, and then click **Page1**.

Task 3. Create a master detail relationship and set burst options

1. Click any cell in the Crosstab, and then from the Toolbar, click **More**, click **Master Detail Relationships**.

2. In the **Master detail relationships** dialog box, click **New Link**.

IBM Cognos Analytics - Reporting creates a link between the first items; the Country data item in Country Query and the Product line data item in Product Line Revenue Query. You need to link Country code in Country Query to Country Code in Product Line Revenue Query. You will create the link on Country code instead of Country because Country code is indexed in the database and the queries will run faster.

3. In the **Master: Country Query** pane, click **Country code**.

4. In the **Detail: Product Line Revenue Query** pane, click **Country code**.

[Master detail relationships](#)



5. Click **OK** to close the **Master detail relationships** dialog box.

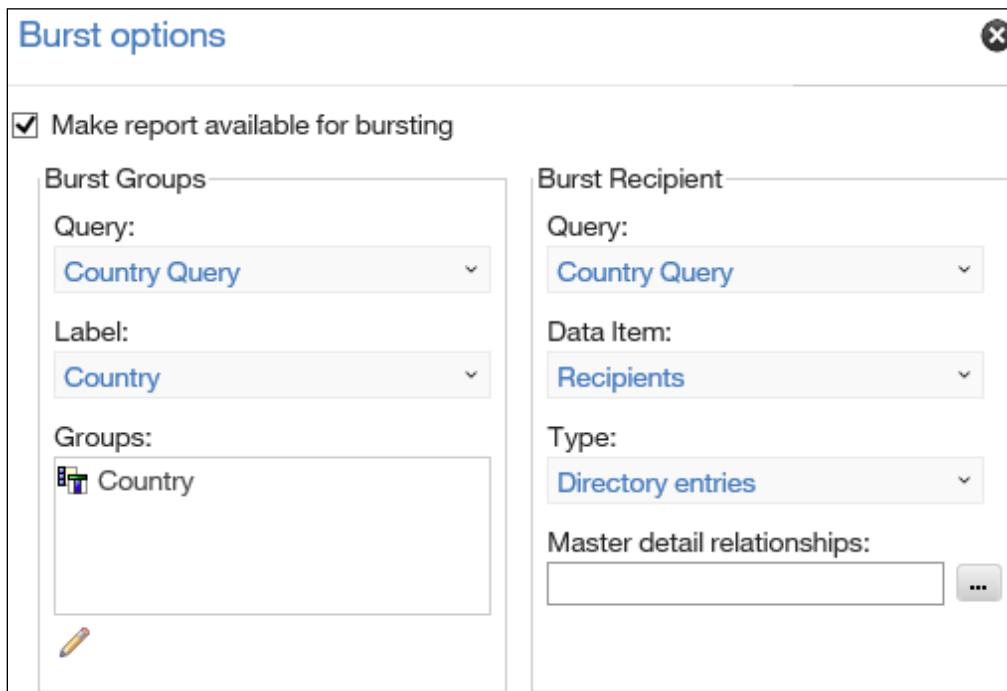
6. On the Application bar, click **Run options** , then click **Run HTML**.

Note that each Crosstab is different now, due to the master detail relationship.

Country: Australia				
Revenue	2011	2012	2013	
Camping Equipment	9,752,591.01	19,175,957.2	13,007,383.98	
Golf Equipment	4,094,643.54	8,482,438.67	6,502,474.22	
Mountaineering Equipment	2,691,279.15	5,861,253.12	5,380,587.79	
Outdoor Protection	600,956.77	367,636.38	171,750.41	
Personal Accessories	2,131,381.68	5,081,517.25	4,261,477.85	

Country: Austria				
Revenue	2010	2011	2012	2013
Camping Equipment	7,431,795.17	9,163,419.93	13,471,100.17	9,731,648.11
Golf Equipment	3,411,617.12	4,465,999.47	6,234,620.98	5,009,903.66

7. Close the rendered report tab.
 8. In the **Pages** pane, click **Report**.
 9. In the Properties pane, under RUNNING & VALIDATING, double-click **Burst options**.
 10. Click the **Make report available for bursting** check box to select it.
 11. In the **Burst Groups** section, in the **Query** list, select **Country Query**, and then in the **Label** list, select **Country**.
Country Query references the List with only one dimension.
 12. Below **Groups**, click **Edit**, double-click **Country** to add it to the **Groups** folder, and then click **OK** to close the **Grouping & sorting** dialog box.
 13. In the **Burst Recipient** section, in the **Query** list, select **Country Query**.
 14. In the **Data Item** list, select **Recipients**.
 15. In the **Type** list, select **Directory entries**.
- A section of the result appears as follows:



16. Click **OK**, and then save the report as **Country Product line Revenue Bursted Report** in **Team content\B6059**.
You are ready to burst the report.

Task 4. Burst the report and view the results

1. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of the report name to close it.
2. In the **IBM Cognos Analytics** portal, navigate to **Team content\B6059**.
3. Right-click the **Country Product Line Revenue Bursted Report** entry, then click **Run as**.
4. Enable **Run in background**, and then expand **Advanced**.
5. Click the **Burst the report** check box to select it, and then click **Run**.
6. Navigate to **Team content\B6059**, and then right-click the **Country Product Line Revenue Bursted Report** entry, then click **View versions**.
7. Click the most recent timestamp to open it.

A section of the result appears similar to the following:

The screenshot shows a list of report output versions under the 'Versions' tab. There is a checkbox labeled 'Show all history'. Below it, three entries are listed:

- Australia**: EN-US, with a globe icon.
- Italy**: EN-US, with a globe icon. To its right is a small trash can icon.
- United States**: EN-US, with a globe icon.

Three report output versions were created: one each for Australia, Italy, and United States. Because you are logged in as Frank Bretton, the report author, you can see all three output versions. You will now log on as a different user.

8. Open another (private) browser session.
 9. Browse to the IBM Cognos Analytics portal: <http://vclassbase/ibmcognos>
 10. Log in using **sindenj/Education1**.
 11. In the **IBM Cognos Analytics** portal, navigate to **Team content\B6059**, and then right-click the **Country Product Line Revenue Bursted Report** entry, then click **View versions**.
 12. Click the most current timestamp to open it.
- Only one output version, for Australia, is displayed.
13. Click the **HTML** report.

The results appear as follows:

Country: Australia			
Revenue	2011	2012	2013
Camping Equipment	9,752,591.01	19,175,957.2	13,007,383.98
Golf Equipment	4,094,643.54	8,482,438.67	6,502,474.22
Mountaineering Equipment	2,691,279.15	5,861,253.12	5,380,587.79
Outdoor Protection	600,956.77	367,636.38	171,750.41
Personal Accessories	2,131,381.68	5,081,517.25	4,261,477.85

This output version contains only Australian data, as you specified in the Burst options.

14. Close this browser window.
15. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

Sales reps can view a crosstab report of the revenue generated in their country by product line and year. To accomplish this, you used the data in a burst table and a master-detail relationship to burst reports to Team content.

Unit summary

- Distribute reports using bursting
- Create burst keys
- Identify report recipients and data items using burst tables
- Distribute reports using email and IBM Cognos Analytics portal
- Burst dimensional containers

Unit 8 Enhance user interaction with HTML

IBM Training



Enhance user interaction with HTML

IBM Cognos Analytics (v11.0)

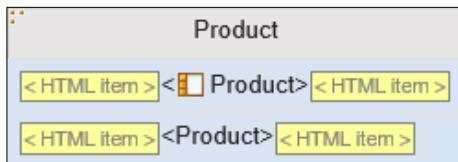
© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM.

Unit objectives

- Enhance reports with HTML
- Know how to add JavaScript to reports
- Create tooltips that clarify report data
- Send emails using Hyperlinks in a report

Create interactive reports using HTML

- Add HTML items to your report to control the behavior of elements of the report.
- Using JavaScript, report authors can restrict/validate user selections in prompts
- You may need to unlock the IBM Cognos Analytics – Reporting environment; to inject HTML inside cells of a List or Crosstab.



Create interactive reports using HTML

An HTML item adds a container in which you can insert custom HTML code, it can be anything the browser can understand, such as links, images, multimedia or JavaScript.

You may need to unlock the IBM Cognos Analytics – Reporting environment; to inject HTML inside cells of a List or Crosstab.

HTML items will only work when the report is rendered in HTML. If you want to display formatted text with other report outputs (such as PDF), you can use a Rich text item.

JavaScript in HTML items is removed when the report is run with full interactivity mode. Instead, you can use the Custom control object to add JavaScript that can run in full interactivity mode. See this technote for more information: [Adding JavaScript to reports](http://www-01.ibm.com/support/docview.wss?uid=swg21990557) (<http://www-01.ibm.com/support/docview.wss?uid=swg21990557>).

Using HTML items is a capability that can be controlled in the Administration console.

The Appendix A - Explore the Prompt API optional unit demonstrates how you can expand report functionality using HTML items and JavaScript.

Include additional information with tooltips

- You can embed HTML code in your report that will create tooltips in selected areas of your layout.
- Example:
 - Revenue
- Tooltips can link to query data and display information not included in the layout.

	Product	Revenue
	Aloe Relief	769,757.23
Features a 23 cm razor edge blade tempered from a special blend of stainless steel. The hollow handle has a compartment containing survival gear including a compass. Overall length: 36cm.		
	Bear Survival Edges	10,329,654.58
	Bella	26,876,327.5

Include additional information with tooltips

A tooltip is a small window of text that appears beside the mouse pointer when it is held over a specific area of the screen.

You can create a tooltip when you want users to have access to detail that you do not want to display directly in the report layout.

You can create tooltips on content by including it between `` and `` HTML tags and using the `title` attribute of the `` tag to specify the tooltip text.

It is also possible to create tooltips using the `<p></p>` tags in HTML, though it is not possible to apply any formatting to the tooltip window.

`` tags are also supported with Rich text items.

Tooltips are already available on charts by setting the Tooltips property to Yes. However, the technique illustrated here, would be suitable for Lists and Crosstabs.

Depending on the content of your HTML item, set the appropriate source type:

- Text: for a static tooltip text
- Data item value: if the tooltip text is equal to the value of a data item
- Report expression: if the tooltip text is a mix of static text and values of data items

The query must include the item you want to use in the tooltip.

Demonstration 1

Add tooltips to a report

	Product	Revenue
	Aloe Relief	769,757.23
Features a 23 cm razor edge blade tempered from a special blend of stainless steel. The hollow handle has a compartment containing survival gear including a compass. Overall length: 36cm.		
	Bear Survival Edge	10,329,654.58
	Bella	26,876,327.5

Demonstration 1: Add tooltips to a report

Demonstration 1: Add tooltips to a report

Purpose:

Management would like a product revenue report to include additional information such as product descriptions and the years included in the revenue value displayed. This information is only supplementary, however, and should not be added directly to the report layout. You will achieve this by including this information in tooltips, which will display at the appropriate points in the report.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: Team content\Samples\Models\GO data warehouse (query)

Folder: Sales and Marketing (query)

Namespace: Sales (query)

Task 1. Create the report

1. Create a new **Blank** report using the **GO data warehouse (query)** package.
2. Add a **List** to the report page.
3. From the **Data\Source** tab, add the following query items to the List:
 - Products: **Product**
 - Products: **Product description**
 - Time: **Year**
 - Sales fact: **Revenue**
4. Group the **<Product>** column, and then cut the **<Product description>** and **Year** columns from the List.

You do not want them to be displayed in the List, but you want to keep them in the query.

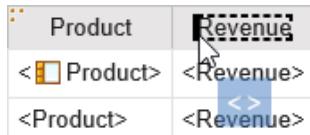
Product	Revenue
<Product>	<Revenue>
<Product>	<Revenue>

5. Run the report in **HTML**.
- Note that as you hover your mouse cursor over the **Product** and the **Revenue** headers and cells, there is no additional information provided to the user.
6. Close the rendered report tab.

Task 2. Create a static text tooltip

You will add a tooltip to the revenue header, to inform users that the revenue displayed is for all years, otherwise they may assume that it is the current year revenue.

1. On the Application bar, click **More**, then click **Locked**  to unlock the report.
1. From the **Toolbox** tab, from under **ADVANCED**, drag an **HTML item** to the left of the **Revenue** List column title, inside the cell itself.



2. Drag another **HTML item** to the right of the **Revenue** List column header title. The Revenue column header title appears between two <HTML item> objects.

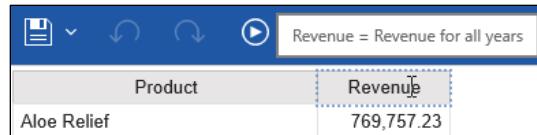


3. Double-click the left <HTML item>, then type the following HTML code in the text box:

```
<span title="Revenue = Revenue for all years">
```

4. Click **OK**, and then double-click the right <HTML item>.
5. In the **HTML** dialog box, type ``, and then click **OK**.
6. On the Application bar, click **Run options** , then click **Run HTML**.

Note that as you hover your mouse cursor over the **Revenue** column header, that the tooltip which you created appears above the mouse cursor, as follows:



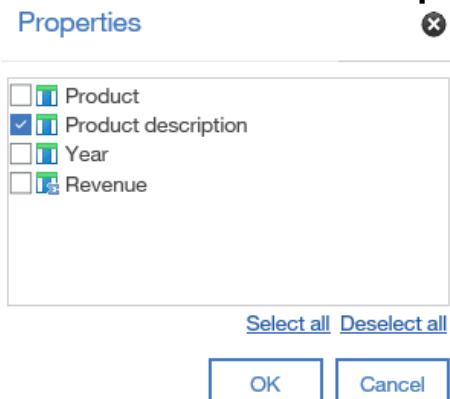
7. Close the rendered report tab.

Task 3. Create a tooltip from a report expression

You will create a tooltip that displays the appropriate product description when the mouse cursor hovers over a product. You need to create an HTML item that uses a report expression to reference product description from the query.

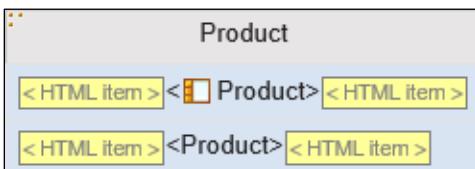
1. Click the List's **Container selector**  to select the entire List.
2. On the Application bar, click **Show properties**  to open the Properties pane.

3. In the Properties pane, under the DATA section, update the **Properties** option to include **Product description**.



4. From the **Toolbox**, under **ADVANCED**, drag an **HTML item** to the left of an item in the **<Product>** List column body, and then drag an **HTML item** to the right of an item in the **<Product>** List column body.

An **<HTML item>** object appears on each side of each data item.



5. Click an **<HTML item>** object to the left of a **<Product>** item to select it.
6. In the Properties pane, under the HTML SOURCE section, change the **Source type** property to **Report expression**, and then double-click the **Report expression** property.
7. In the **Report expression** box, create and validate the following code:
''
The first "" is a double quote followed by a single quote, and the last "" is a single quote followed by a double quote.
8. Click **OK** to close the **Report expression** dialog box.
9. Double-click an **<HTML item>** object to the right of a **<Product>** item, in the **HTML** dialog box, type ****, and then click **OK**.
HTML items are not case sensitive.

Task 4. Test the tooltip

1. On the Application bar, click **Run options** , then click **Run HTML**.
2. Point to **Bear Survival Edge** product to see the tooltip that you created.

A section of the results appear as follows:

	Product	Revenue
	Aloe Relief	769,757.23
Features a 23 cm razor edge blade tempered from a special blend of stainless steel. The hollow handle has a compartment containing survival gear including a compass. Overall length: 36cm.		
	Bear Survival Edgs	10,329,654.58
	Bella	26,876,327.5

3. Point to another product.

The tooltip now displays the description for this product. When IBM Cognos Analytics - Reporting created the HTML version of this report, it created a copy of the tooltip code for every row in the Product column. Each instance of the code passes the Description query item related to the Product to the generic tooltip function.

4. Close the rendered report tab.

Task 5. Create a tooltip from a data item

While the Revenue column header tooltip indicates that the values in that column are based on revenue for all years, not all years contain the same products. Some products are discontinued, and new products are introduced in different years. You will create another tooltip in the Revenue column to display the years that the revenue is based on

1. On the side bar, click **Queries**, and then click **Query1**.
2. From the **Toolbox**, drag a **Data Item** into the **Data Items** pane.
3. In the **Data item expression** dialog box, in the **Name** field, type **Revenue ToolTip**.
4. the **Expression Definition** pane, create and validate the following HTML code:

```
'<span title="Revenue is calculated from the years ' +
cast(minimum([Year]), varchar(4)) + ' through ' + cast(maximum([Year]),
varchar(4)) + "' >'
```

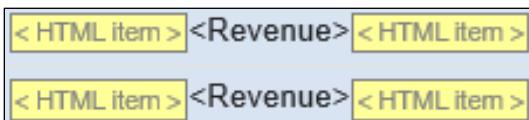
Hints:

- Data Items tab: [Year].
- Functions tab:
 - Common Functions: A-C: cast
 - Summaries: minimum, maximum

This data item references the year data, and for all the years with revenue data for a product, it will determine what the earliest year is, and what the latest year is.

The minimum() function will determine the minimum year, maximum() function determines the maximum year, and the cast() function converts the data expression to a string of 4 characters; so it can be displayed on the page.

5. Click **OK** to close the **Data item expression** dialog box.
6. On the side bar, click **Pages**, and then click **Page1**.
7. From the **Toolbox**, under **ADVANCED**, drag an **HTML item** to the left of an item in the **<Revenue>** List column body, and then drag an **HTML item** to the right of an item in the **<Revenue>** List column body.



8. Click the **<HTML item>** to the left of a **<Revenue>** List column body item, in the Properties pane in the HTML SOURCE section, change the **Source type** property to **Data item value**, and then change the **Data item value** property to **Revenue ToolTip**.
9. Double-click the **<HTML item>** to the right of a **<Revenue>** List column body item, in the **HTML** dialog box, type ****, and then click **OK**.
10. On the Application bar, click **More**, then click **Unlocked** to lock the report.

Task 6. Test the tooltip.

1. On the Application bar, click **Run options** , then click **Run HTML**.
2. Hover the mouse cursor over the revenue for **Auto Pilot** in the third row of the **Revenue** column.

A custom tooltip is displayed to indicate the revenue is derived from sales from 2012 through 2013.

Product	Revenue
Aloe Relief	
Astro Pilot	Revenue is calculated from the years 2012 through 2013
Auto Pilot	\$,961,715

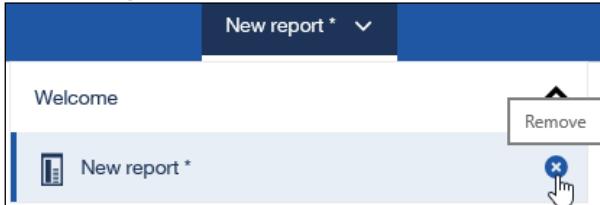
3. Hover the mouse cursor over the revenue for **Bear Edge**.

Product	Revenue
Aloe Relief	769,757.23
Astro Pilot	26,000.00
Auto Pilot	Revenue is calculated from the years 2010 through 2013
Bear Edge	\$12,893,464.19

The tooltip now displays the years used in this revenue value for this specific product.

When IBM Cognos Analytics - Reporting created the HTML version of this report, it created a copy of the tooltip code for every row in the Revenue column. Each instance of the code referenced the Revenue ToolTip data item and passed the content of this item to the generic tooltip function.

4. Close the rendered report tab.
5. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



6. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You created tooltips in your report using HTML items. Your tooltips described the revenue column, both in terms of the general statement for the revenue column that was displayed, details of the years included in the revenue values for each product, as well as a tooltip that displayed descriptions for each product.

Send emails using Hyperlinks in a report

- You can replace a data item with a Hyperlink that creates an email when you click it in the report

The screenshot shows a report configuration interface. On the left is a table with two columns: 'Employee' and 'Revenue'. The 'Employee' column contains values like '<Employee name>' repeated eight times. The 'Revenue' column contains values like '<Revenue>' repeated eight times. To the right of the table is a 'Hyperlink' object properties panel. It has two main sections: 'TEXT SOURCE' and 'URL SOURCE'. In the 'TEXT SOURCE' section, 'Source type' is 'Data item value' and 'Data item value' is 'Employee name'. In the 'URL SOURCE' section, 'Source type' is 'Report expression', 'Open in new window' is 'No', and 'Report expression' is '"mailto:"+[Email]'. Two orange arrows point from the text 'Text that appears as a link in the report' and 'Email address' to the respective 'Data item value' and 'Report expression' fields.

Send emails using Hyperlinks in a report

If the HTML you want to embed in the report is just a hyperlink (and tags), then you can use the Hyperlink object as a shortcut instead of typing the HTML yourself.

The Hyperlink object allows the report author to control the source of the text that appears in the link, as well as the source of the URL itself. Both can be Text, Data item value, Data item label, or Report expression. And it allows the author to choose whether the hyperlink should open in a new window or not.

As per the definition of the HTML tag (hyperlink), if the target (href attribute) starts with "mailto:", then when the user click the hyperlink it should open the default mail client on the system and start a mail draft to the email specified after "mailto:".

Unit summary

- Enhance reports with HTML
- Know how to add JavaScript to reports
- Create tooltips that clarify report data
- Send emails using Hyperlinks in a report

Unit summary

Unit 1 Introduction to IBM Cognos Active Reports

IBM Training



Introduction to IBM Cognos Active Reports

IBM Cognos Analytics (v11.0)

© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM.

Objectives

- Describe Active Reports, and their value
- Save Active Reports
- Use Active Reports as prompt pages
- Convert existing reports to Active Reports
- Explain security considerations in Active Reports
- Debug Active Report behavior

Business drivers for Active Reports

- Interactivity
 - interact with a rich user interface, and easily filter data in reports with prompts
- Performance
 - access and work with information without wait times
- Disconnected functionality
 - access and interact with report content while disconnected from the network
- Distribution
 - access reports as attachments in email rather than through a website
- Authoring
 - pixel-perfect formatting for reports

What is an Active Report?

- Active Report is a report type whose output is an MHT file
- Report consumers can view and interact with an Active Report data without hitting the server and the database again



Introduction to IBM Cognos Active Reports

© Copyright IBM Corporation 2018

What is an Active Report?

An Active Report can be run in the view mode of IBM Cognos Analytics - Reporting like any other report, or it can be downloaded locally as an MHT file.

MHT files are archived HTML files (MHT is the file extension used for MIME HTML file formats). MHT files include all necessary data and images to allow report consumers to interact with Active Report data without connecting to the server and the database again.

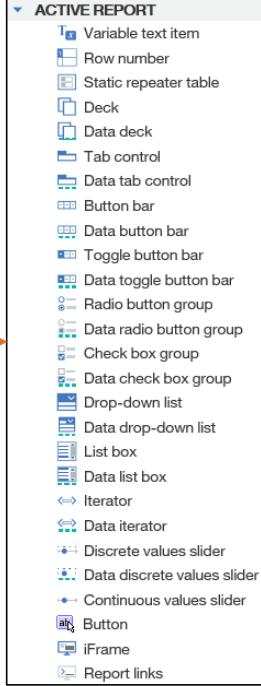
Consumers can explore the report by changing different prompt values and rendering new output in the client. Because all the data is embedded in the report at runtime, when user selections change the data is rendered in the client without going back to the database server, nor the IBM Cognos Analytics server.

IBM Training 

Examine Active Report controls

- When you create an Active Report, there are additional controls available for use.
- Controls can be of two types:
 - static
 - data driven

Active report controls on the Toolbox tab 



The list of controls includes:

- Variable text item
- Row number
- Static repeater table
- Deck
- Data deck
- Tab control
- Data tab control
- Button bar
- Data button bar
- Toggle button bar
- Data toggle button bar
- Radio button group
- Data radio button group
- Check box group
- Data check box group
- Drop-down list
- Data drop-down list
- List box
- Data list box
- Iterator
- Data iterator
- Discrete values slider
- Data discrete values slider
- Continuous values slider
- Button
- iFrame
- Report links

Introduction to IBM Cognos Active Reports 

Examine Active Report controls

There are two versions of most Active Report controls: Static and Data-driven.

Static controls allow you to manually define values.

Data-driven controls have values that are determined by the associated data item.

For example, Drop-down list, Radio button group, and Check box group are all controls whose values are created manually by the report author.

The equivalent data-driven controls are Data drop-down list, Data radio button group, and Data check box group, where each one of these would be populated with the values in a chosen data item.

For more information on available controls for use in Active Reports, please refer to [Active Report Controls](#)

(https://www.ibm.com/support/knowledgecenter/SSEP7J_11.0.0/com.ibm.swg.ba.cognos.ug_cr_rptstd.doc/c_rpt_app_cntrls.html).

IBM Training IBM

Describe Active Report variables

- Active Report variables work with controls to add interactivity and control behavior in reports.

variable

Introduction to IBM Cognos Active Reports © Copyright IBM Corporation 2018

Describe Active Report variables

You can use variables to define the behavior of controls and other report objects. Variables allow you to specify a control's selection behavior or reaction behavior.

For the control selected in the menu (region_Data_BB in the slide example):

- Selection Behavior: upon selection of the control, the value of the Active Report variable is set to the control selection value
- Reaction Behavior: specifies how the control reacts to events on variables, when another control, or an object in the control, is selected.
- There are three types of reaction behavior:
- select an item in a control
- filter an item in a control
- enable a control

Not all reaction behaviors are available for every combination of source and target control, but possible reactions depend on the source and target controls.

Language support in controls

- Create variables that support multiple languages.
- When the report is run in that language, all text will appear as translated.

Label	DB Value
Email	E-mail
Fax	Fax
Post	Mail

Language support in controls

If you want the Active Report to be viewed in different languages, you will have to supply the translated value for any text you create in static controls.

Proven practice: naming conventions

- Demonstrations and exercises in this course explicitly name all Active Report controls, Active Report variables, and queries.

Rename	Default name	Preferred Name
Static Control	Radio Button Group1	Country_RBG
Data Control	Data Drop-Down List1	Region_Data_DDL
Variable	Product line	vProductLine
Parameter	N/A	?pCountry?

Proven practice: naming conventions

Throughout this course, all Active Report controls, Active Report variables, and queries follow a naming standard, to make it easier to identify a control, variable, or query in the report based on its name.

- Control names have a prefix that indicates the primary measure or data item, and a suffix whose abbreviation indicates the data type (i.e. Country_RBG).
- Variables are prefixed with a "v" to denote a variable.
- Parameters are prefixed with a "p" to denote a parameter.

Proven practice: report object sizing

- Plan for scalable reports across different resolutions.
- Percentage-based sizes allow objects to scale to the available resolution.
- Fixed size objects may appear too small on larger resolutions, or may push objects outside the report on smaller resolutions.

Proven practice: report object sizing

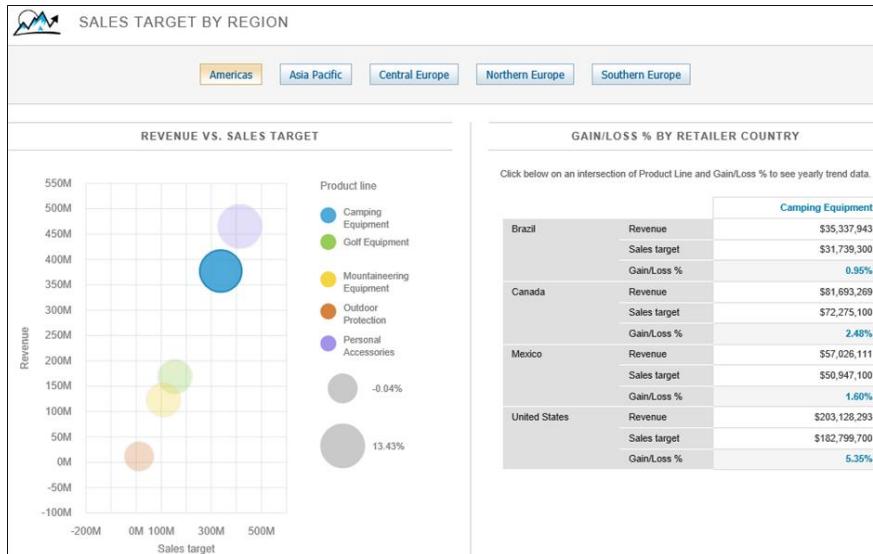
When designing the layout of an Active Report, it is important to consider the different ways the report may be used. For example, reports that are run on a mobile device may have less available space than a report that is run in a browser.

To create a flexible report that can be scaled according to the available resolution, try to size objects as a percentage of the available space, rather than as a fixed size.

As resolutions change, controls that use fixed sizing may have their content clipped (if the resolution is too low), or may leave large unused sections on a larger resolution, which may not be visually appealing to the report consumer.

Demonstration 1

Run an Active Report



Introduction to IBM Cognos Active Reports

© Copyright IBM Corporation 2018

Demonstration 1: Run an Active Report

Demonstration 1: Run an Active Report

Purpose:

You will open an Active Report to examine some of the interactive behavior that is possible in an Active Report.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

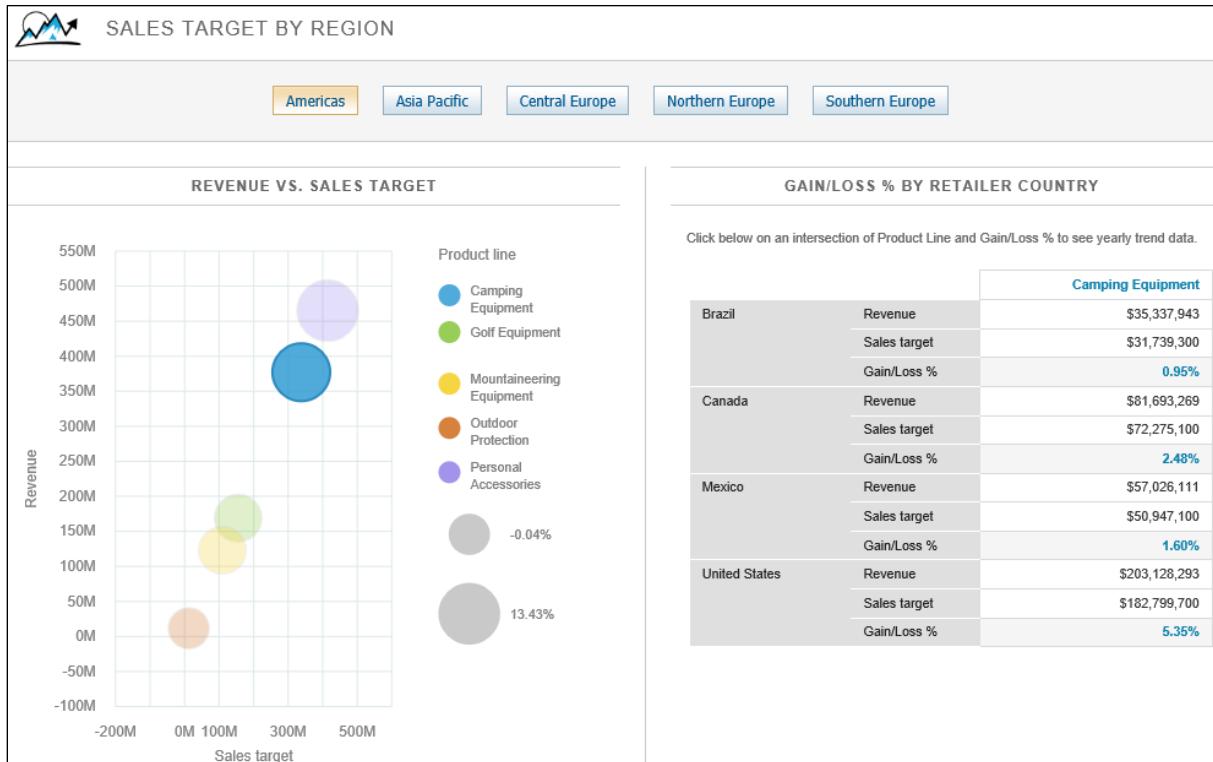
Package: GO sales (query)

Report: Samples > Reports > Active Reports > Sales target by region

Task 1. Open a report in IBM Cognos Analytics

1. On the side panel, click Team content .
2. Click Samples > Reports > Active reports.
3. Click Sales target by region.

The results appear as follows:

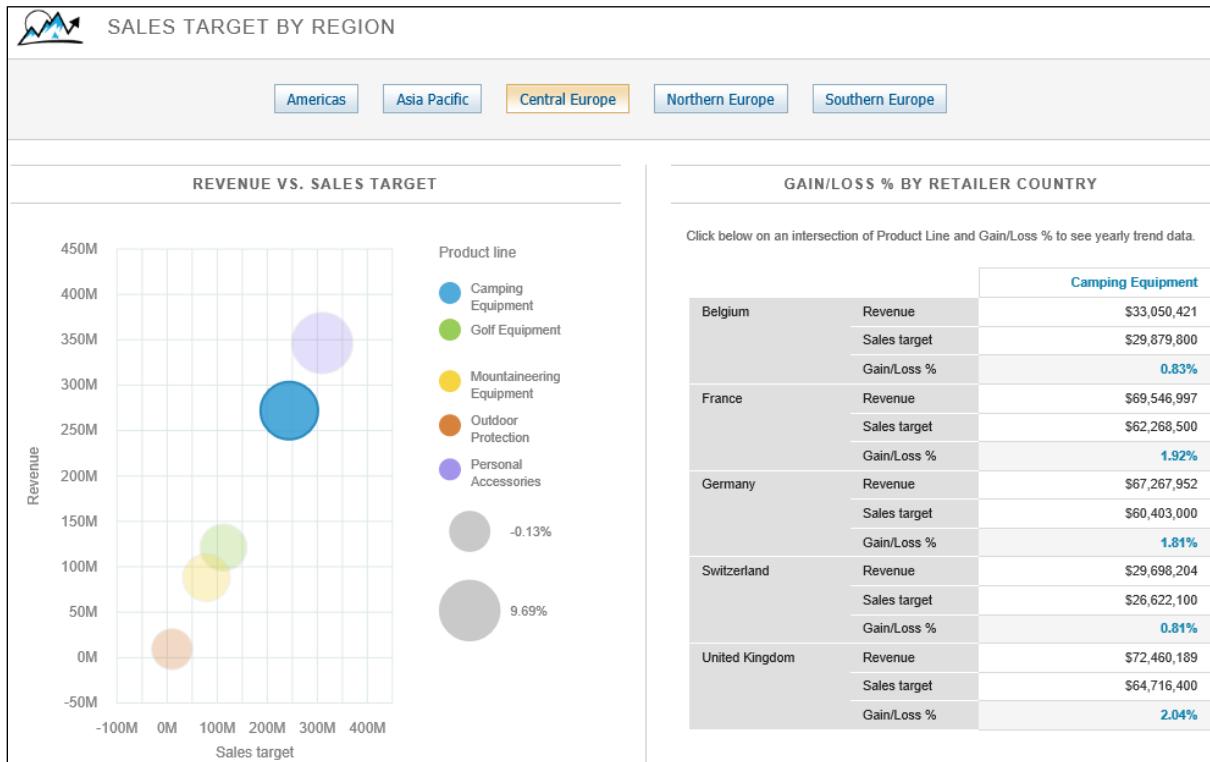


The Active Report opens in a new browser tab.

Task 2. Test the interactive features

1. Click Central Europe.

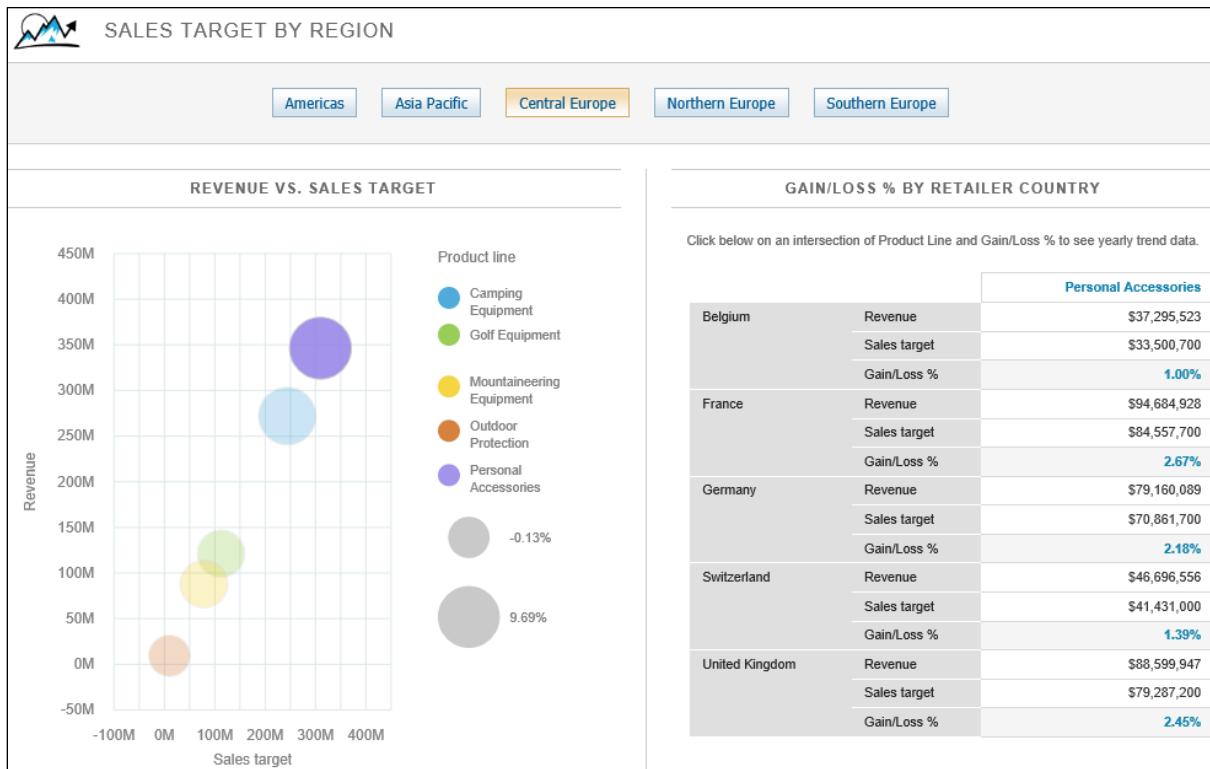
The results appear as follows:



The bubble visualization and Crosstab are updated to display data for Central Europe.

2. In the bubble visualization, click the largest bubble to select **Personal Accessories**.

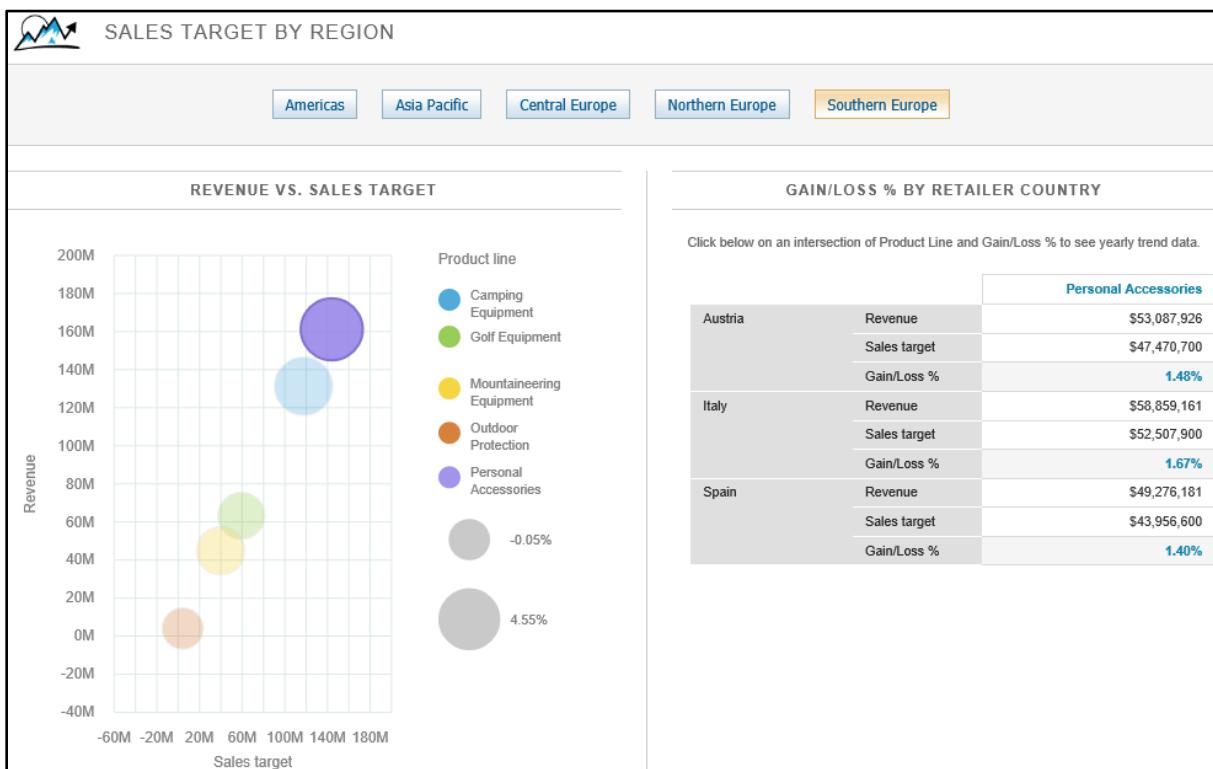
The results appear as follows:



The Crosstab is updated to show data for Personal Accessories.

3. Click Southern Europe.

The results appear as follows:



The visualization and Crosstab are updated to display data for Southern Europe, but the Crosstab continues to display data for Personal Accessories.

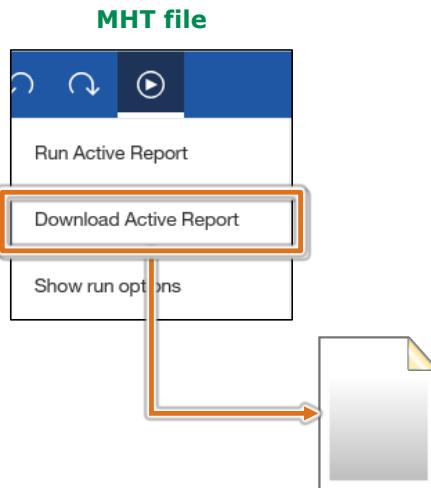
- Close the rendered report tab and leave IBM Cognos Analytics open for the next demonstration.

Results:

You opened and ran an Active Report. And tested some of the interactive behaviors available with this type of report.

Save an Active Report for distribution

- Download the report as MHT output.



Introduction to IBM Cognos Active Reports

© Copyright IBM Corporation 2018

Save an Active Report for distribution

To save an Active Report as an MHT file for distribution, select Download Active Report from the Run options menu. You will be prompted to open or save the report.

Note that if you simply run the report under the IBM Cognos Analytics portal, the MHT file will be unpacked to an XHTML format. To get the actual MHT file, you must save the report and download the saved output.

All data required to execute the Active Report is retrieved at the time the MHT file is generated, which means that the MHT file does not require a connection to the IBM Cognos servers once it has been created.

Because the MHT file generates its data on the initial run, opening the MHT file after the data has changed on the IBM Cognos servers will not update the data in the Active Report. To reflect the latest data, you must run and download the Active Report again.

Demonstration 2

Create a simple Active Report using Static and Data-driven controls

Americas	<input checked="" type="checkbox"/> Email		
Asia Pacific	<input checked="" type="checkbox"/> Fax		
Central Europe	<input type="checkbox"/> Post		
Northern Europe	<input type="checkbox"/> Sales rep		
Southern Europe	<input type="checkbox"/> Promo		
	<input type="checkbox"/> Phone		
	<input type="checkbox"/> Web site		
Retailer country	Retailer site	Order method type	Revenue
Belgium	Brussel	Fax	\$279,876.39
	Liège	Fax	\$2,286,557.32
France	Calais	Fax	\$585,814.01
	Lyon	E-mail	\$65,940.08
	Lyon	Fax	\$2,344,759.22
	Paris	E-mail	\$299,899.71
	Paris	Fax	\$4,430,206.52

Introduction to IBM Cognos Active Reports

© Copyright IBM Corporation 2018

Demonstration 2: Create a simple Active Report using Static and Data-driven controls

Demonstration 2: Create a simple Active Report using Static and Data-driven controls

Purpose:

You want to create a report that allows the consumer to filter a list by one of the available regions. You also want to create labels for order methods whose names are more meaningful to report consumers.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

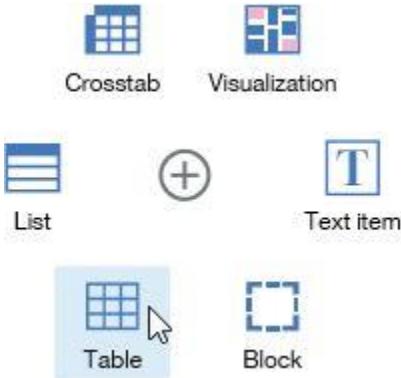
Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create an Active Report with a List

1. From the **Welcome** page, navigate to **Team content** > **Samples** > **Models**.
2. Right-click the **GO sales (query)** package, then click **Create report** .
3. In the Templates and themes dialog, double-click the **Blank active report** template to use it.
4. On the Application bar, click **More** , and then click **Options**.
5. Click the **Report** tab, then uncheck **Automatic group and summary behavior for lists**.
6. Click **OK**.
7. In the work area, click **Add** .

The results appear as follows:



8. Click **Table**, set **Number of rows** to **2**, and then click **OK**.
9. Click the bottom left table cell, Ctrl-click the bottom right table cell, and then on the Toolbar, click **Merge Cells** .
10. On the Toolbar, click **Horizontal alignment options** > **Center** .
11. In the merged table cell, click **Add** , then click **List** .

12. In the **Object and query name** dialog, set **Name** to **Sales_List**, set **Query Name** to **Sales_List_Query**, and then click **OK**.
13. From the **Data**  tab, drag the following query items to the **List**:
 - Retailers: **Retailer country**, and **Retailer site**
 - Sales: **Revenue**
- Click the **<Revenue>** List column body, then on the Toolbar, click **Summarize**  then click **Total**.

The results appear as follows:



Retailer country	Retailer site	Revenue
<Retailer country>	<Retailer site>	<Revenue>
<Retailer country>	<Retailer site>	<Revenue>
<Retailer country>	<Retailer site>	<Revenue>
Overall - Total		<Total(Revenue)>

Task 2. Add a control to filter on regions

You will add a Data button bar to filter the List on the Region data item.

1. On the side bar, click **Toolbox** , and then expand ACTIVE REPORT.
2. Drag a **Data button bar** object into the top left table cell.
3. In the **Object and query name** dialog, for **Name**, type **Region_Data_BB**, and for **Query**, type **Region_Data_BB_Query**, and then click **OK**.
4. On the side bar, click **Data** .
5. From the **Source** tab, under **Retailers**, drag **Region** to the **Region_Data_BB** control's drop zone .

You want to create a connection between the **Region_Data_BB** control and the **Sales_List** control so that as the user clicks a region button, the List will filter the data on one sales region only.

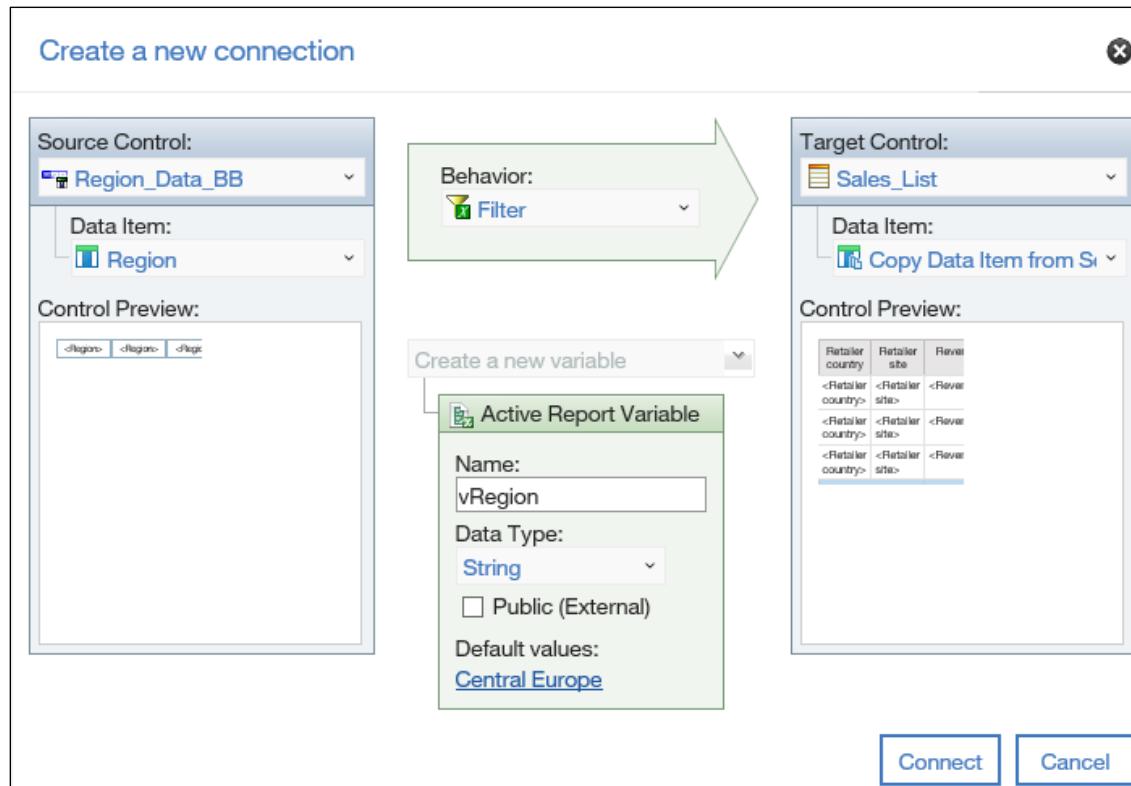
6. In the Toolbar, click **Create a new connection** .

Because there are only two objects in the report, this connection is pre-populated with the correct values to make the connection. On the left is the **Region_Data_BB** control. In the center is the connection method between the two objects, which is a selection by default. On the right is the **Sales_List** control. In the bottom right, **Copy Data Item from Source** will add the data item column to the List as hidden. Once this connection is made, the data button bar

selection will filter the Sales_List control through the use of a variable. You will rename the variable.

7. Under Active Report Variable, rename the new variable as **vRegion**.
8. Under Default values, click **Set default values**.
9. Under Values, double-click **Central Europe** to add it to Selected values.
10. Click **OK** to close the dialog.
11. Under **Behavior**, select **Filter**.

The result appears as follows:



The vRegion variable will be set to the value selected in the Region_Data_BB control. By default, vRegion will be set to Central Europe.

12. Click **Connect**.

Notice that Region is now added to the List because of the connection being created, however it is hidden. Region must be included in the List query so that the filter will work.

13. On the Application bar, click **Run options** , then click **Run Active Report**.
14. Click **Asia Pacific** in the Data button bar.

A section of the results appears as follows:

Americas	Asia Pacific	Central Europe	Northern Europe	Southern Europe
	Retailer country	Retailer site	Revenue	
	China	Beijing	\$15,038,255.58	
	Australia	Brisbane	\$14,990,079.66	
	Australia	Cairns	\$23,445,233.27	
	China	Chaiwan	\$6,105,593.12	
	China	Chengdu	\$8,218,397.39	
	Japan	Chiyoda-ku	\$7,701,778.39	
	Japan	Chuo-ku	\$9,104,505.46	
	Japan	Fukuoka-shi	\$16,328,626.03	
	Australia	Geelong	\$2,155,234.84	
	China	Guangzhou	\$9,952,473.87	
	China	Hangzhou	\$3,235,889.96	
	China	Harbin	\$12,402,588.45	
	Japan	Hiroshima-shi	\$28,518,306.27	
	China	Huhehaote	\$5,515,268.93	
	Korea	Incheon	\$3,872,981.29	
	China	Jinan	\$3,548,805.89	
	Korea	Kwangju	\$2,739,633.83	
	Australia	Melbourne	\$19,238,576.57	
	Japan	Minato-ku	\$5,413,555.03	

Revenue data is displayed for the Asia Pacific region.

15. Close the browser tab.

Task 3. Add a control to filter on order method type

You will add a Button bar control to filter on order method. You will create button labels that are different than the data item values found in the database. The new labels will be helpful for the report consumers who use terminology that is different from the data values in the underlying data source.

1. On the side bar, click **Toolbox**.
2. Drag a **Button bar** object into the top right table cell
3. On the Toolbar, click **Buttons Definition** .

You will be changing the button label names. First you need to create a column to hold the data item values that will be passed when the user clicks the button.

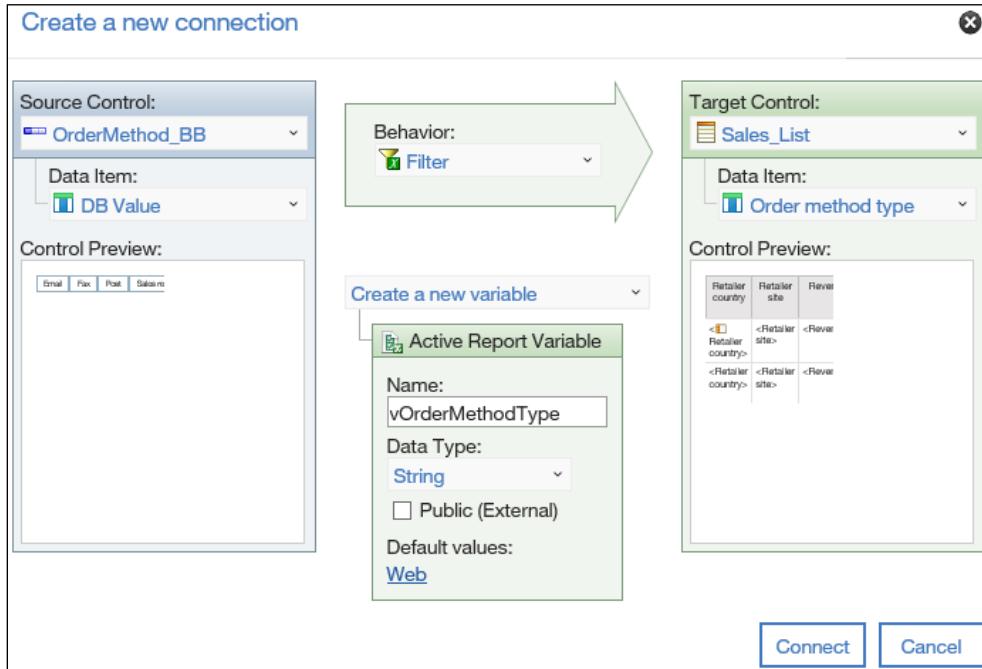
4. Click **New** , type **DB Value**, and then click **OK**.
5. Enter the labels and values as follows:

Label	DB Value
Email	E-mail
Fax	Fax
Post	Mail
Sales rep	Sales visit
Promo	Special
Phone	Telephone
Web site	Web

Tip: Use New  to create new rows.

6. Click **OK**.
7. On the Application bar, click **Show properties**  to open the Properties pane.
8. In the Properties panel, under MISCELLANEOUS, set the **Name** property to **OrderMethod_BB**.
You will create the filter connection between the OrderMethod_BB control and the Sales_List control. First you must add Order method type to the Sales_List control so it will be available when you specify the connection.
9. On the side bar, click **Data**.
10. From the **Source** tab, expand **Order method**, and then drag **Order method type** after the **Retailer site** column.
11. Click the **OrderMethod_BB** control to select it.
12. From the Toolbar, click **Create a New Connection**.
13. Select the following:
 - For **Target Control**, select **Sales_List**.
 - Under **Behavior**, select **Filter**.
 - Create a new variable > Name: **vOrderMethodType**
 - Data Item (for the Source Control): **DB Value**
 - Data Item (for the Target Control): **Order method type**
14. Under **Default values**, click **Set default values**.
15. Under **Values**, double-click **Web**, and then click **OK**.

The result appears as follows:



16. Click **Connect**.
17. On the Application bar, click **Run options** , then click **Run Active Report**.
A section of the results appears as follows:

Americas	Asia Pacific	Central Europe	Northern Europe	Southern Europe	Email	Fax	Post	Sales rep	Promo	Phone	Web site
		Retailer country	Retailer site	Order method type	Revenue						
		Belgium	Antwerpen	Web	\$22,882,658.97						
			Arlon	Web	\$8,160,492.71						
			Brugge	Web	\$7,945,692.93						
			Brussel	Web	\$2,390,137.83						
			Bruxelles	Web	\$21,071,312.35						
			Charleroi	Web	\$884,794.13						
			Leuven	Web	\$2,674,352.37						
			Liège	Web	\$1,492,696.16						
			Louvain-La-Neuve	Web	\$6,422,044.57						
			Sint-Niklaas	Web	\$5,134,408.50						
			Waterloo	Web	\$13,912,554.90						
			Zaventem	Web	\$6,960,811.87						

18. Close the browser tab.

Task 4. Enhance controls and the report

You will convert the controls to enhance the report appearance. You will further enhance the report by grouping the Country column and hiding the Order method type column.

1. Select the **Region_Data_BB** control, and then ensure that the **Properties** pane is showing **Data button bar**.
2. In the Properties pane, under **GENERAL**, set **Orientation** to **Vertical**.

The Region_Data_BB control now runs vertically, eliminating the need to scroll right to access the order method buttons.

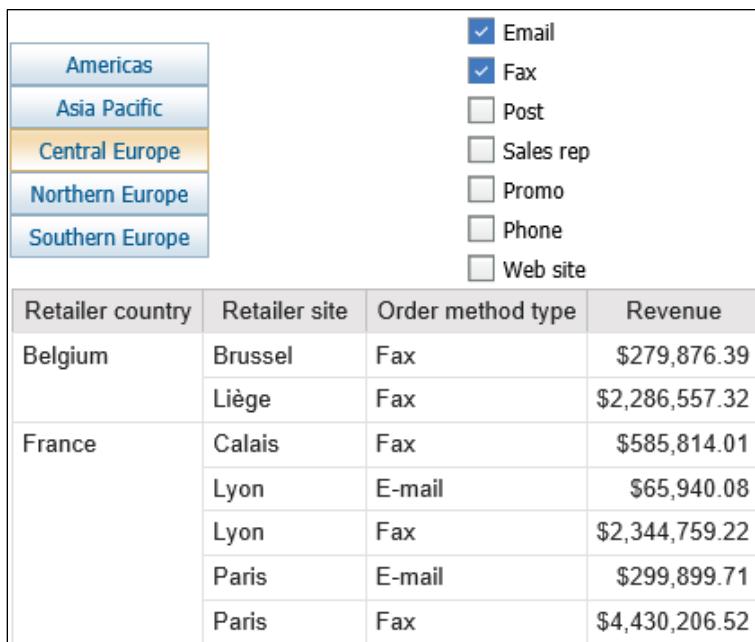
Next, you will convert the OrderMthd_BB control to a Check box group.

3. Select the **OrderMethod_BB** control, then on the Toolbar, click **More**  . Hint: You may need to click the report background to clear the selection first.
 4. Point to **Convert Control**, and then click **Check box group** .
- Note that because you have changed the type of this control, you will also need to change the control's name to reflect that it is now a Check box group. Because this is not a Data-driven control, you do not need to update any query names.
5. Select the **OrderMethod_BB** control, and then in the Properties pane, under MISCELLANEOUS, change the **Name** property to **OrderMethod_CBG**.
 6. Repeat step 2 for the **OrderMethod_CBG** control so that it appears vertically. By converting the control to a Check box group, the user can make multiple selections to filter the List.

To enhance the report, you will group by the Retailer country and Retailer site columns.

7. Click <Retailer country>, then Ctrl-click <Retailer site>, and then on the Toolbar, click **Group / Ungroup** .
8. From the Application bar, click **Run options** , then click **Run Active Report**.
9. Select **Northern Europe** from the **Region_Data_BB** control, and then select **Email** and **Fax** from the **OrderMethod_CBG** control.

A section of the results appears as follows:

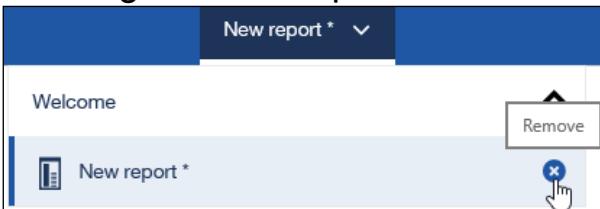


The screenshot shows a report interface. On the left, there is a vertical navigation bar with categories: Americas, Asia Pacific, Central Europe (which is highlighted in orange), Northern Europe, and Southern Europe. To the right of this is a list of order method types with checkboxes: Email (checked), Fax (checked), Post (unchecked), Sales rep (unchecked), Promo (unchecked), Phone (unchecked), and Web site (unchecked). Below this is a table with four columns: Retailer country, Retailer site, Order method type, and Revenue. The data is grouped by Retailer country. For Belgium, there are two rows: one for Brussel (Fax, \$279,876.39) and one for Liège (Fax, \$2,286,557.32). For France, there are five rows: Calais (Fax, \$585,814.01), Lyon (E-mail, \$65,940.08), Lyon (Fax, \$2,344,759.22), Paris (E-mail, \$299,899.71), and Paris (Fax, \$4,430,206.52).

Retailer country	Retailer site	Order method type	Revenue
Belgium	Brussel	Fax	\$279,876.39
	Liège	Fax	\$2,286,557.32
France	Calais	Fax	\$585,814.01
	Lyon	E-mail	\$65,940.08
	Lyon	Fax	\$2,344,759.22
	Paris	E-mail	\$299,899.71
	Paris	Fax	\$4,430,206.52

10. Close the rendered report tab.

11. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



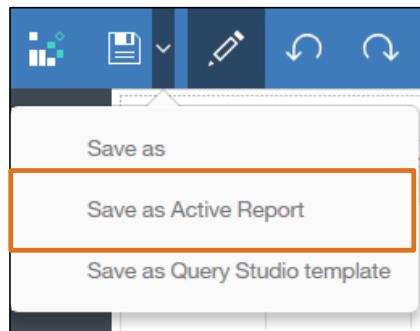
12. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You created a simple Active Report containing Static and Data-driven controls. The Static control allows you to use business terms, rather than the data values, so that consumers will be better able to interact with the report.

Convert an existing report to an Active Report

- Save as Active Report
 - Active Report menu items, objects, and properties become available in Cognos Analytics.
 - objects that are not supported in Active Reports are removed.



Convert an existing report to an Active Report

When you convert an existing report to an Active Report, the report author is still required to modify the report to deliver the full value of an Active Report to consumers. For example, a report author may add a Data Check Box Group to be able to filter on the Product data item.

Objects not supported in IBM Cognos Active report are:

- prompt controls in report pages
- table of contents
- table of contents entries
- bookmarks

Active report security

- Role-based security features supported
 - IBM Cognos Analytics portal
 - scheduled and burst reports
 - mobile distribution (initial access only)
- MHT files do not have IBM Cognos security
- MHT file security can be achieved through:
 - email
 - network-accessible location
 - report-by-report access code
 - (contents not encrypted)



Active report security

IBM Cognos Analytics supports role-based security features whenever the user is connecting to the IBM Cognos servers. This includes reports run in the IBM Cognos Analytics portal, scheduled and burst reports, and reports accessed from a mobile device.

Note that in the case of mobile devices, only the first access to the report undergoes the security check. Once the report has been downloaded, it is viewed on the device without further security verification.

When an Active Report is saved as an MHT file, the file itself should be treated as any other file in the file system. By default, there is no IBM Cognos security protecting the file, although a password can be added to an Active Report on a report-by-report basis.

Active Report MHT files can be secured by sending them using secure e-mail, by storing them in a secured network-accessible location, or by any other precautions that can be taken to secure files containing potentially sensitive data.

IBM Training 

Debug an Active Report

- There are various debugging options available to help you troubleshoot an Active Report in development
- Run Options
 - Enable right-click debug menu

Name	Value
vRegion	Central Europe
vOrderMethodType	Web

Introduction to IBM Cognos Active Reports  © Copyright IBM Corporation 2018

Debug an Active Report

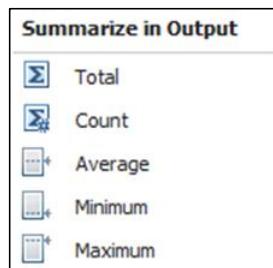
You use debugging options to help you troubleshoot active report behavior. For example, you can track the value of variables while you interact with the report, or reset the report to the default values.

The debug feature is turned on by default. It can be opened from the Application bar, under Run options.

When running the report from IBM Cognos Analytics - Reporting, from the IBM Cognos Analytics portal, or from an MHT file, the debug menu is activated when you right-click the report.

Describe interactive behavior properties

- Several properties control interactive behavior:
 - No Values List
 - Sort
 - Column Visibility
- Summarize measures to aggregate on what is displayed.
 - Summarize in Output options



Describe interactive behavior properties

No Values List - specifies a value to show when no value is selected, for example: <All> can appear at the top of a list. Applies to Drop-Down List and List Box only

Sort: specify whether users can sort data in a List.

Column Visibility: show or hide a column in a List, or a row/column in a Crosstab. It can be based on a condition on an Active Report variable.

Summarize in Output: show the summarized value for what is displayed in an Active Report.

Demonstration 3

Convert an existing report to an Active Report

Falcon Outfitters		<All>	
Region	Retailer country	Order method type	Revenue
Americas	United States	Sales visit	\$1,630,869.35
		Telephone	\$2,489,882.05
		Web	\$9,176,464.06
	Canada	E-mail	\$3,352,349.69
		Mail	\$720,845.68
		Telephone	\$2,441,618.79
		Web	\$12,225,876.72

Demonstration 3: Convert an existing report to an Active Report

Demonstration 3: Convert an existing report to an Active Report

Purpose:

You will create a report that was requested by a client. You will then convert the report to an Active Report, enhance its appearance, and add password protection, in order to exceed the client's expectations.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create a List report

1. From the side panel, navigate to **Team content** > **Samples** > **Models**, then right-click the **GO sales (query)** package, and then click **Create report** .
2. In the Templates and themes dialog, double-click the **Blank** template.
3. In the page body, click **Add** , and then click **List** .
4. In the **Object and query name** dialog, under **Name**, type **Retailer_List**, under **Query**, type **Retailer_List_Query**, and then click **OK**.
5. From the **Data** tab, drag the following query items to the **Retailer_List**:
 - Retailers > **Region**
 - Retailers > **Retailer country**
 - Retailers > **Retailer**
 - Order method > **Order method type**
 - Sales > **Revenue**
6. Click **Region** and Ctrl-click **Retailer country**, then from the Toolbar, click **Group / Ungroup** .

Task 2. Convert the report to an Active Report and add controls

1. Click the down arrow beside **Save** , then click **Convert to Active Report**. Notice that interactive behavior and connection icons appear to the right of the **Retailer_List** if you select the List. The List is considered an interactive control. Other menu items, properties and objects are now available to you for working with an Active Report.

You will create a Data drop-down list control for Retailer and Order method so that a report consumer can filter the Retailer_List control based on their selection.

2. From the **Toolbox**, drag a **Table** below the List, and set **Number of rows** to **2**.
3. Click **OK**.
4. Click the bottom left table cell, Ctrl+click the bottom right table cell, then on the Toolbar, click **Merge Cells** .
5. On the Toolbar, click **Horizontal alignment options** > **Center** .
6. Click then drag the **Retailer_List** object into the bottom table cell.
7. From the **Toolbox**, under **ACTIVE REPORT**, drag a **Data drop-down list** to each of the left and right table cells:
 - Left table cell:
 - Name: **Retailer_Data_DDL**
 - Query Name: **Retailer_Data_DDL_Query**
 - Right table cell:
 - Name: **OrderMethodType_Data_DDL**
 - Query Name: **OrderMethodType_Data_DDL_Query**
8. From the **Data \ Source** tab, drag **Retailers > Retailer** to the **Retailer_Data_DDL** control's drop zone.
9. From the **Source** tab, drag **Order method > Order method type** to the **OrderMethodType_Data_DDL** control's drop zone.

Task 3. Connect the controls to the List

1. Select the **Retailer_Data_DDL**, then from the Toolbar, click **Create a new connection** .
2. Specify the following connections:
 - **Retailer_Data_DDL > Filter > Retailer_List**
 - Create a new variable > Name: **vRetailer**
3. Click **Connect**.
4. Select the **OrderMethodType_Data_DDL**, then from the Toolbar, click **Create a New Connection** .
5. Specify the following connection:
 - **OrderMethodType_Data_DDL > Filter > Retailer_List**
 - Create a new variable > Name: **vOrderMethodType**

6. Click **Connect**.
7. Run the report and select **4 Golf only** and **Web** from the controls.

The result appears as follows:

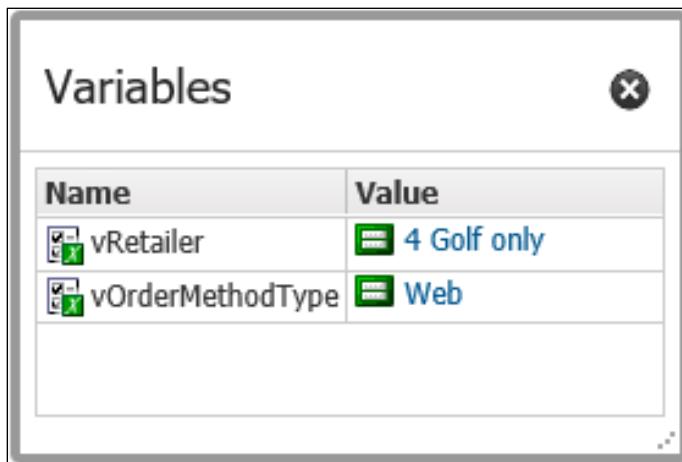
4 Golf only		Web		
Region	Retailer country	Retailer	Order method type	Revenue
Asia Pacific	Australia	4 Golf only	Web	\$2,673,033.19

8. Right-click the report to access the debugging menu.

You can use debug features to help you troubleshoot your report. For example, you can view variable values and reset the default values of controls.

9. Click **Show Variables** to see the current variable values.

The result appears as follows:



10. Close the Variables dialog.
11. Right-click the report, and then click **Reset Variables**.
The variables are reset to their default value, which is no value. And the List is unfiltered again.
12. Close the browser tab.

Task 4. Download the Active Report as an MHT file and open it

1. On the Application bar, click **Run options**, and then click **Download Active Report**.
2. At the prompt to save **New Report.mht**, click the down arrow beside **Save**, and click **Save as**.
3. Save the report as **C:\Training\B6098\ActiveReport.mht**.
Hint: Create the folders in Windows Explorer if they do not exist.
4. Click **Open folder**, and then double-click **ActiveReport.mht**.
The report runs as expected.

5. Close the browser tab.
6. Close **Windows Explorer**.

Next, you will update the report to enhance the content and add some interactivity using Interactive Behavior properties.

Task 5. Define interactive properties

You want to hide the Retailer column, add the ability to sort on Region and Retailer Country columns, and add the default option of <All> to the Order method type drop-down list.

1. In the **Retailer_List**, click <**Retailer**>.
2. On the Application bar, click **Show properties**  to open the Properties pane.
3. In the Properties pane, under INTERACTIVE BEHAVIOR, set **Column Visibility** to **Hidden**.

Next, you will set the Region and Retailer country columns to allow sorting.

4. Click <**Region**>, Ctrl-click <**Retailer country**>, and then in the **Properties** pane, under **INTERACTIVE BEHAVIOR**, set **Allow Sorting** to **Yes**.

Next, you will define a default value of <All> for the **Retailer_List** control.

5. Select the **OrderMethodType_Data_DDL** control.
6. In the Properties pane, under GENERAL, set the **No Value List Item** property to **Show**.

The properties for No Value List Item appear.

7. In the Properties pane, double-click the **Label** property.
8. In the **Default** text box, type <**All**>, and then click **OK**.
9. From the **Run Options** menu, click **Run Active Report**, and then select **Falcon Outfitters** from the **Retailer_Data_DDL** control.

10. Click the **Retailer country** column header twice to sort in descending order.
 The results appear as follows:

Falcon Outfitters		<All>	
Region	Retailer country	Order method type	Revenue
Americas	United States	Sales visit	\$1,630,869.35
		Telephone	\$2,489,882.05
		Web	\$9,176,464.06
	Canada	E-mail	\$3,352,349.69
		Mail	\$720,845.68
		Telephone	\$2,441,618.79
Overall - Summary			\$4,686,775,768.85

The list is sorted by Retailer country, and the Retailer column is no longer visible.

Note that the results may not be identical because the Order method type column is not sorted.

1. Close the browser tab.

Task 6. Add an access code to the Active Report

1. In the side bar, click **Pages** , and then click **Report** .
2. In the Properties pane, under REPORT, double-click **Active reports**.
3. In the **Active Report Properties** dialog, under **Access Code**, type **Education1**.
4. Click **OK**.
5. On the Application bar, click **Run options** , then click **Run Active Report**. You are prompted to enter an access code for the report.
6. Type **Education1**, and then click **OK**.
 The report runs as expected.
7. Close the rendered report tab.
8. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.
9. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

After creating a report that was requested by a client, you converted it to an Active Report, added controls, and saved it as a standalone (MHT) file. You then defined interactive properties to enhance the report, and added a password in order to prevent unauthorized access.

Demonstration 4

Summarize on a filtered list in an Active Report

Mountaineering Equipment		
Product line	Revenue	Quantity
Mountaineering Equipment	\$409,660,132.90	9,900,091
Overall - Total	\$4,686,775,768.85	89,237,091

Mountaineering Equipment		
Product line	Revenue	Quantity
Mountaineering Equipment	\$409,660,132.90	9,900,091
Overall - Total	409,660,132.9	9,900,091

Demonstration 4: Summarize on a filtered list in an Active Report

Demonstration 4: Summarize on a filtered list in an Active Report

Purpose:

You want to create a report with summarized values at the bottom, and ensure that the summarized data accurately reflects what is selected in the active controls. You will use summary options that are specific to Active Reports.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create a List and add an Active Report control

1. Using the **GO sales (query)** package, create a new **Blank active report**.
2. In the work area, click **Add** , and then click **Table** .
3. Set **Number of columns** to 1, **Number of rows** to 2, and then click **OK**.
4. In the bottom table cell, click **Add** , then click **List** .
5. In the **Object and Query Name** dialog, under **Name**, type **Revenue_List**, under **Query Name**, type **Revenue_List_Query**, and then click **OK**.
6. From the **Data**  tab, drag the following query items to the **Revenue_List**:
 - Products > **Product line**
 - Sales > **Revenue, Quantity**
7. Click the **Toolbox**, then expand **ACTIVE REPORT**, and then drag a **Data dropdown list** to the top table cell.
8. On the **Object and Query Name** dialog, under **Name**, type **ProductLine_Data_DDL**, and under **Query Name**, type **ProductLine_Data_DDL_Query**, and then click **OK**.
9. From the **Data** tab, Add **Products > Product line** to the drop zone of **ProductLine_Data_DDL**.
10. In the Toolbar, click **Create a new connection**.
11. Specify the following connection:
 - **ProductLine_Data_DDL > Filter > Revenue_List**
 - New variable: **vProductLine**
 - Data Item (left): **Product line**
 - Data Item (right): **Product line**
12. Click **Connect**.

The results appear as follows:

<Product line>		
Product line	Revenue	Quantity
<Product line>	<Revenue>	<Quantity>
<Product line>	<Revenue>	<Quantity>
<Product line>	<Revenue>	<Quantity>

Task 2. Add a summary row to display aggregated measure values

1. Click <Revenue>, then Ctrl-click <Quantity>.
2. In the Toolbar, click **Summarize** Σ , then under **Default summary**, click **Total**.
3. Run the report.

The results appear as follows:

Product line	Revenue	Quantity
Camping Equipment	\$1,589,036,664.03	27,301,149
Golf Equipment	\$726,411,367.89	5,113,701
Mountaineering Equipment	\$409,660,132.90	9,900,091
Outdoor Protection	\$75,994,296.25	12,014,445
Personal Accessories	\$1,885,673,307.78	34,907,705
Overall - Total	4,686,775,768.85	89,237,091

Note that the total for Revenue is approximately \$4.6B, and for Quantity is approximately 89K.

4. From the **ProductLine_Data_DDL** control, select **Mountaineering Equipment**.

The results appear as follows:

Mountaineering Equipment		
Product line	Revenue	Quantity
Mountaineering Equipment	\$409,660,132.90	9,900,091
Overall - Total	\$4,686,775,768.85	89,237,091

Notice that the totals did not change. The aggregate values being displayed were calculated at run time at the server don't reflecting what is being selected.

To achieve the expected results, you will specify an aggregate that is specific to Active Reports, one that is calculated at the client.

5. Close the rendered report tab.
6. Click the summary cell **Overall - Total**, then press **Delete** in the keyboard.
7. Click <Revenue>, then Ctrl-click <Quantity>.

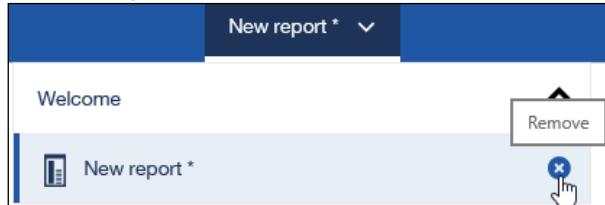
8. In the Toolbar, click **Summarize**, then under **Summarize in Output**, click **Total**.
9. On the Application bar, click **Run options** , then click **Run Active Report**.
10. From the **ProductLine_Data_DDL** control, select **Mountaineering Equipment**.

The results appear as follows:

Mountaineering Equipment		
Product line	Revenue	Quantity
Mountaineering Equipment	\$409,660,132.90	9,900,091
Overall - Total	409,660,132.9	9,900,091

The expected summary values are now displayed for Mountaineering Equipment.

11. Close the rendered report tab.
12. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



13. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You created a report with summarized values that accurately reflect the user's selection in Active Report controls.

Use Active Reports as prompt pages

- Active Reports can be used as a prompt for existing reports.
 - author the target report with parameterized filters.
 - create an Active Report
 - add the Report links object to the Active Report
- Reduce prompt startup time
 - schedule the Active Report to run overnight
- Target report can be an iFrame in the Active Report, or run in a separate viewer.

Use Active Reports as prompt pages

Active Reports can be used as a prompt page to filter an existing report.

To create a report with an Active Report prompt page, first author the target report, using parameterized filters.

Next, create the Active Report, and add a Report links object to the Active Report. This object will allow the Active Report to set parameters for the target report.

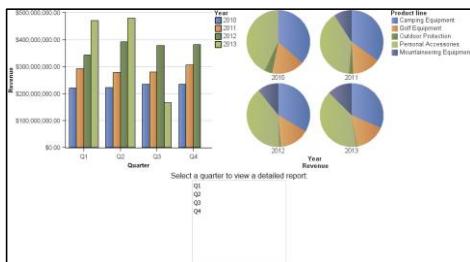
Because Active Reports query their data at the time they are first created, having an Active Report as a prompt page can improve startup performance, because the required data for the prompts has already been queried.

Active Reports can also make use of more advanced features, such as visualizations and charts, to make the prompt page more interactive and visually appealing.

You can configure the Active Report to show the target content in an iFrame within the report, or in a separate viewer.

Demonstration 5

Create an Active Report prompt page



Year	Quarter	Product line	Revenue
2010	Q3	Camping Equipment	\$88,193,533.99
		Golf Equipment	\$38,883,691.26
		Outdoor Protection	\$9,269,980.35
		Personal Accessories	\$99,403,110.65
2011	Q3	Camping Equipment	\$97,718,279.62
		Golf Equipment	\$39,080,737.47
		Mountaineering Equipment	\$25,542,699.11
		Outdoor Protection	\$5,983,942.21
		Personal Accessories	\$112,754,008.54
2012	Q3	Camping Equipment	\$130,749,800.24
		Golf Equipment	\$56,040,710.42
		Mountaineering Equipment	\$41,444,335.89
		Outdoor Protection	\$2,621,653.16
		Personal Accessories	\$147,261,512.83
2013	Q3	Camping Equipment	\$53,939,284.40
		Golf Equipment	\$25,129,748.14
		Mountaineering Equipment	\$21,634,522.83
		Outdoor Protection	\$706,920.16
		Personal Accessories	\$65,031,507.03

Introduction to IBM Cognos Active Reports

© Copyright IBM Corporation 2018

Demonstration 5: Create an Active Report prompt page

Demonstration 5: Create an Active Report prompt page

Purpose:

Your company has requested a filtered report to compare quarterly income for specific quarters across all years. They would like to be able to select the quarter based on regional income

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create the target report

1. Using the **GO sales (query)** package, create a new **Blank active report**.
2. On the report page, click **Add** , then click **List** .
3. Click **OK** to accept the defaults and close the dialog.
4. From the **Data**  tab, Ctrl-click then drag the following query items to the List:
 - Time: **Year, Quarter**
 - Products: **Product line**
 - Sales: **Revenue**
5. Click the **<Year>** List column body, Ctrl-click **<Quarter>**, then on the Toolbar, click **Group / Ungroup** .

The results appear as follows:

Year	Quarter	Product line	Revenue
<Year>	<Quarter>	<Product line>	<Revenue>
	<Quarter>	<Product line>	<Revenue>
<Year>	<Quarter>	<Product line>	<Revenue>
	<Quarter>	<Product line>	<Revenue>

Task 2. Create a parameterized filter for the report

1. Click any cell in the List, then in the Toolbar, click **Filters**  > **Edit Filters**.
2. In the **Filters** dialog, click **Add** , select **Advanced**, then click **OK**.
3. In the **Detail filter expression** dialog, enter the following expression:
[Quarter]=?pQuarter?
 Hint: Drag Quarter from the Data items tab.
4. Click **Validate** , and then under **Provide a value**, select **Q2**, and click **OK**.
5. Click **OK** to close the **Detail filter expression** dialog.

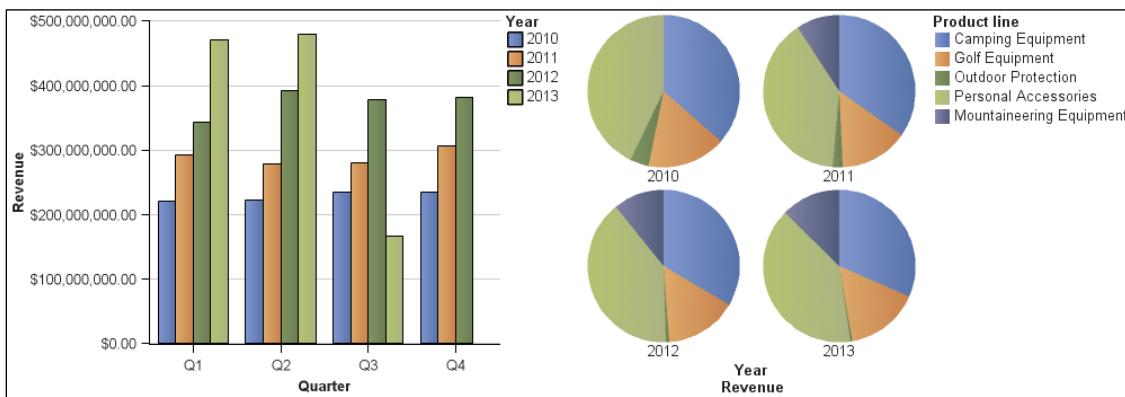
6. Click **OK** to close the **Filters** dialog.
7. On the Application bar, click **Save**, and then save the report under **My content** as **Prompt Target Report**.

Task 3. Create the prompt page report

You will create a prompt report that displays two charts to give a high-level view of the data for report consumers.

1. From the side bar, click **New** , and then click **Report** .
2. Double-click the **Blank active report** template.
3. In the **Data**  tab, click **Add report data** .
4. Navigate to **Team content**  > **Samples** > **Models**, click **GO sales (query)**, and then click **Open**.
5. In the page body, click **Add** , then click **Table** .
6. Set **Number of columns** to **2**, **Number of rows** to **2**, then click **OK**.
7. Click the bottom left table cell, Ctrl+click the bottom right table cell, and then on the Toolbar, click **Merge Cells** .
8. On the Toolbar, click **Horizontal alignment options** > **Center** .
9. In the top-left table cell, click **Add** , then click **Visualization** .
10. From the top right corner, click the visualization filter, then click **Charts**.
11. From the categories on the left, click **Column**, then double-click **Clustered Column**.
12. On the Application bar, click **Show properties**  to open the Properties pane.
13. In the Properties pane, under MISCELLANEOUS, set the **Name** property to **Quarterly_Revenue_Chart**.
14. Populate the Chart with the following data items from the Source tab:
 - Default measure (y-axis): Sales: **Revenue**
 - Series (primary axis): Time: **Year**
 - Categories (x-axis): Time: **Quarter**
15. In the top-right table cell, click **Add** , then click **Visualization** .
16. From the categories on the left, click **Pie, Donut**, then double-click **Pie**.
17. In the Properties pane, under MISCELLANEOUS, set the **Name** property to **Revenue by Product line**.
18. Populate the Chart with the following data items from the Source panel:
 - Default measure: Sales: **Revenue**
 - Series (pie slices): Products: **Product line**

- Categories (pies): Time: Year
19. On the Application bar, click **Run options** , then click **Run Active Report**.
The result appears as follows:



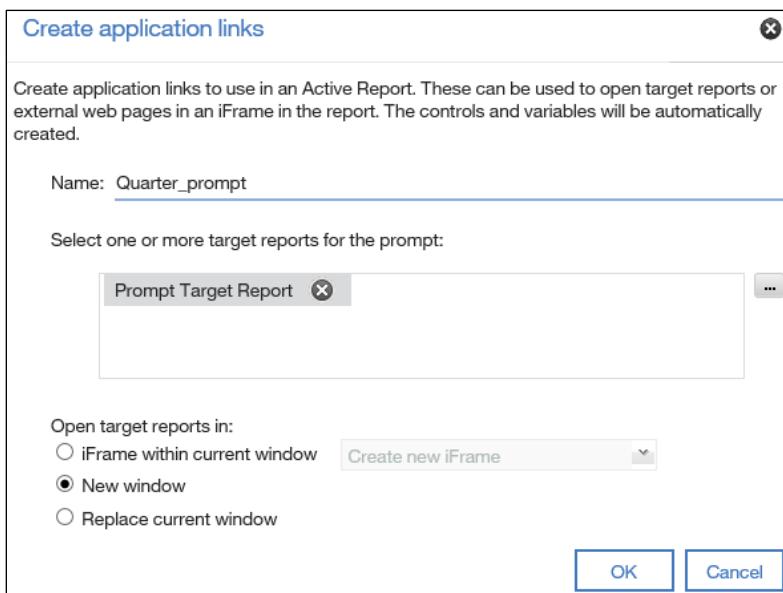
20. Close the rendered report tab.

Next, you will add a Report links object to connect this report to the target report.

Task 4. Add Report links to the report page

- From the **Toolbox**, expand ACTIVE REPORT, then drag a **Report links** object to the bottom table cell.
- In the **Create application links** dialog, beside **Name**, type **Quarter_prompt**.
- Under **Select one or more target reports for the prompt**, click the ellipsis .
- Navigate to **My content**, click **Prompt Target Report**, and then click **Open**.
- Under **Open target reports in**, select **New Window**.

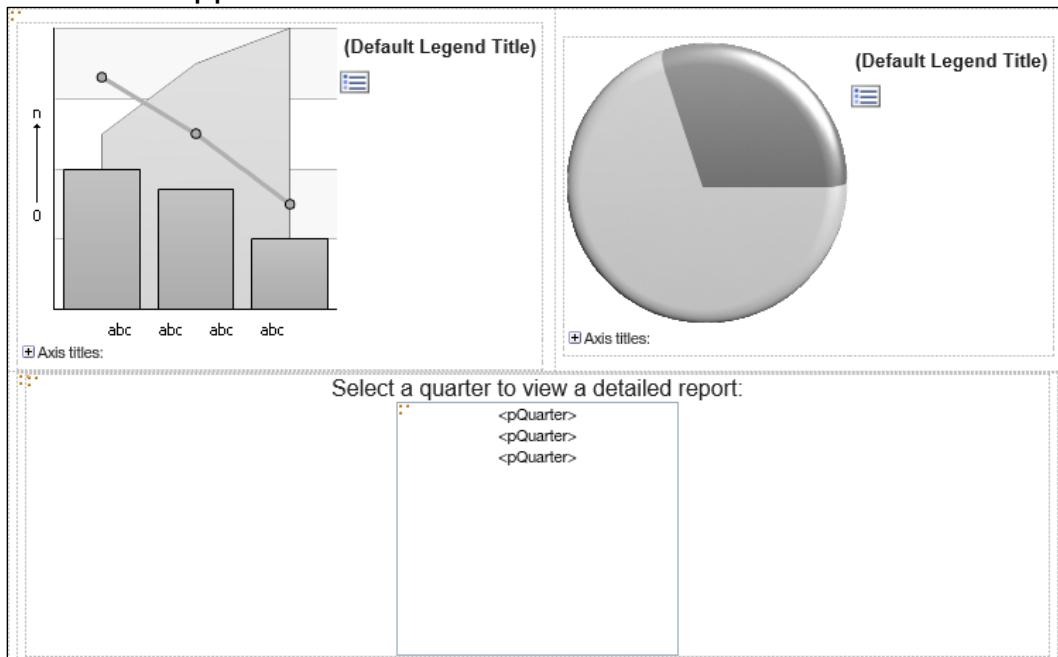
The results appear as follows:



- Click **OK**.
- Click the Data list box that was added.

8. In the Properties pane, under INTERACTIVE BEHAVIOR, set **Auto Select** to **None**.
9. Double-click the **pQuarter** Text item above the Data list box.
10. Type **Select a quarter to view a detailed report:**, type a space, and then click **OK**.

The result appears as follows:



11. From the Application bar, click **Run options** , then click **Run Active Report**.
12. In the Data list box, click **Q3**.

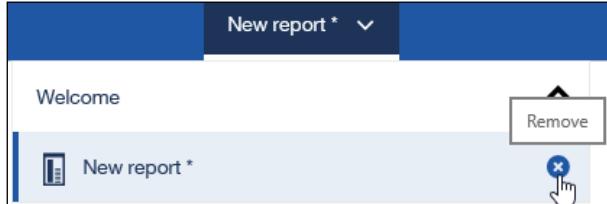
A new browser tab opens with the detailed report.

The results appear as follows:

Year	Quarter	Product line	Revenue
2010	Q3	Camping Equipment	\$88,193,533.99
		Golf Equipment	\$38,883,691.26
		Outdoor Protection	\$9,269,980.35
		Personal Accessories	\$99,403,110.65
2011	Q3	Camping Equipment	\$97,718,279.62
		Golf Equipment	\$39,080,737.47
		Mountaineering Equipment	\$25,542,699.11
		Outdoor Protection	\$5,983,942.21
		Personal Accessories	\$112,754,008.54
2012	Q3	Camping Equipment	\$130,749,800.24
		Golf Equipment	\$56,040,710.42
		Mountaineering Equipment	\$41,444,335.89
		Outdoor Protection	\$2,621,653.16
		Personal Accessories	\$147,261,512.83
2013	Q3	Camping Equipment	\$53,939,284.40
		Golf Equipment	\$25,129,748.14
		Mountaineering Equipment	\$21,634,522.83
		Outdoor Protection	\$706,920.16
		Personal Accessories	\$65,031,507.03

The List displays only data for Q3.

13. Close the rendered report's tab.
14. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



15. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You created an Active Report prompt page with charts and a prompt to select a specific quarter to view in a detailed report. The report links link your selection to the parameter in the target report and filters out data from other quarters.

Summary

- Describe Active Reports, and their value
- Save Active Reports
- Use Active Reports as prompt pages
- Convert existing reports to Active Reports
- Explain security considerations in Active Reports
- Debug Active Report behavior

Unit 2 Use Active Report connections

IBM Training



Use Active Report connections

IBM Cognos Analytics (v11.0)

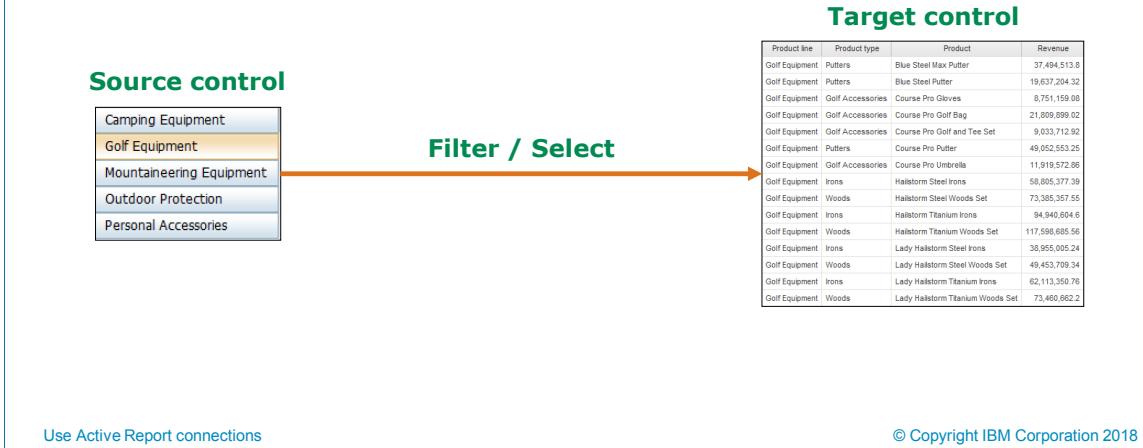
© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM.

Unit objectives

- Describe Active Report connections
- Filter and select Active Report controls
- Modify the interactive behavior of report controls
- Identify Active Report controls and variables
- Use variables to control multiple controls independently
- Control multiple controls from a single variable
- Author and optimize Active Reports for mobile consumption

Examine Active Report connections

- When you create a connection, the following occurs:
 - selection behavior is specified for the source control
 - variable is specified (created or reused)
 - reaction behavior is specified for the target control



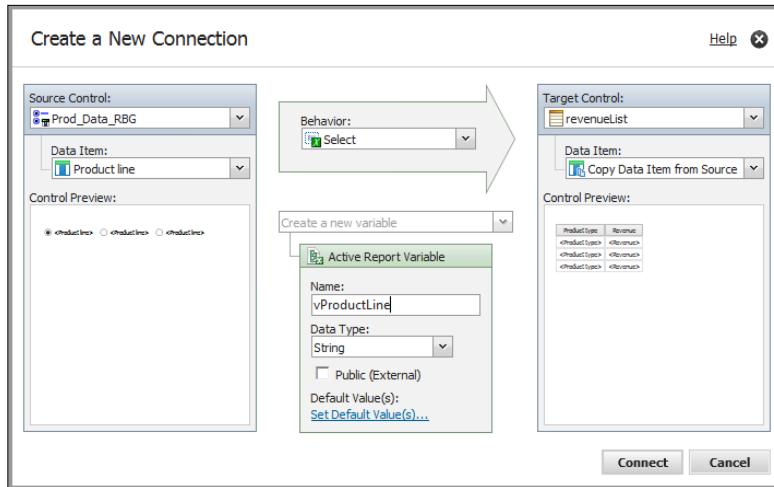
Examine Active Report connections

In this example, the author has created a connection between a Data button bar control and a List control, and has specified the relationship/behavior as Filter. The Data button bar uses a data item that will drive the behavior of the target control. In this case it is Product line. The List uses a data item that will determine what appears in the List. In this case it is Product line. In the report, when the end user clicks a Product line in the Data button bar, the List will be filtered based on what was selected. In this case, the user has clicked Golf Equipment, so the List displays rows for Golf Equipment.

IBM Cognos Analytics automatically sets the reaction behavior when you define a connection between controls using the Create a new connection wizard. However, you may want to change the reaction behavior or specify additional interactive behavior. This can be performed using the Interactive Behavior dialog box.

Create a New Connection wizard

- Use the Create a New Connection wizard to create a connection between two controls



Use Active Report connections

© Copyright IBM Corporation 2018

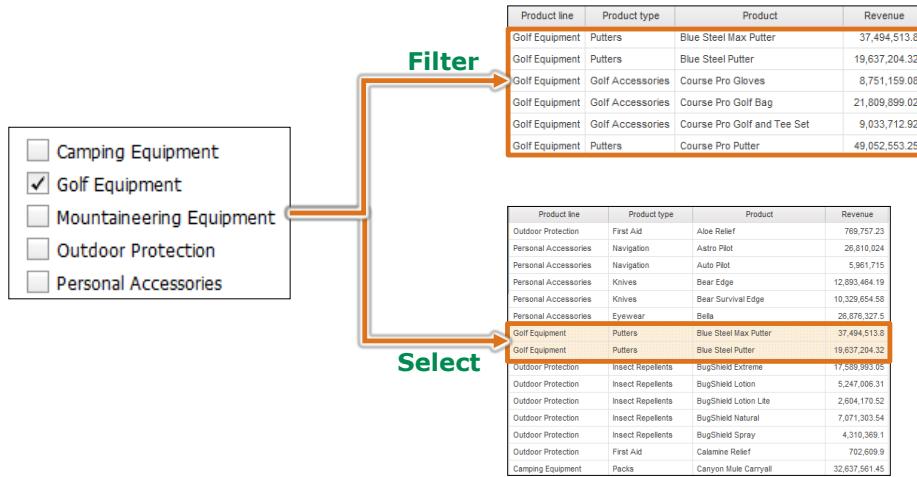
Create a new connection wizard

1. In the Source Control drop-down list, select the source control for which you want to define a connection.
2. Click the Data Item drop-down list under the source control and select the data item that will drive the behavior of the target control.
3. In the Target Control drop-down list, select the target control for which you want to define a connection.
4. Click the Data Item drop-down list under the target control and select the data item that will determine what appears in the target control.
5. Click the Behavior drop-down list between the source and target controls and specify the type of relationship that you want to define between the two controls:
 - Select
 - Filter

6. Below the Behavior area, click the Active Report Variable drop-down list and choose to create a new variable for the connection or use an existing variable.
Tip: If there are no variables defined in the report, you cannot click the Active Report Variable drop-down list.
 - If you create a new variable, type a name for the variable and specify its data type. You can also specify its default value(s)
 - If you choose to use an existing variable, choose the variable that you want. You can also specify its default value(s)
7. Click Connect.

Filter and select behavior

- In the Create a New Connection wizard, specify the behavior between the controls.



Use Active Report connections

© Copyright IBM Corporation 2018

Filter and select behavior

Specify Filter to filter data in the target control based on what will be selected in the source control.

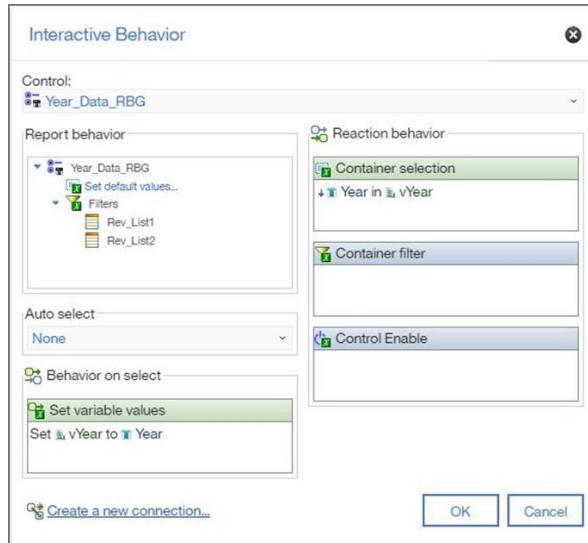
Specify Select to select data in the target control based on what will be selected in the source control.

Select behavior is different depending on the type of target control you are using. If the target is a control that displays data, such as a List or Crosstab, then the data values will be highlighted. If the target control is a container for data, such as a Radio Button Group or Deck, then the active button or card is selected.

After creating a connection, filter and select behavior between controls can be edited in the Interactive Behavior dialog.

Interactive Behavior dialog

- Use the Interactive Behavior dialog to edit behavior between controls.



Use Active Report connections

© Copyright IBM Corporation 2018

Interactive Behavior dialog

This is the Interactive Behavior dialog for an Active Report control:

Report behavior

- Shows the relationship between the control and other controls in the report, if you defined connections between it and other controls. You can specify default values for the variable associated with the control by clicking Set Default Value(s)

Behavior on select

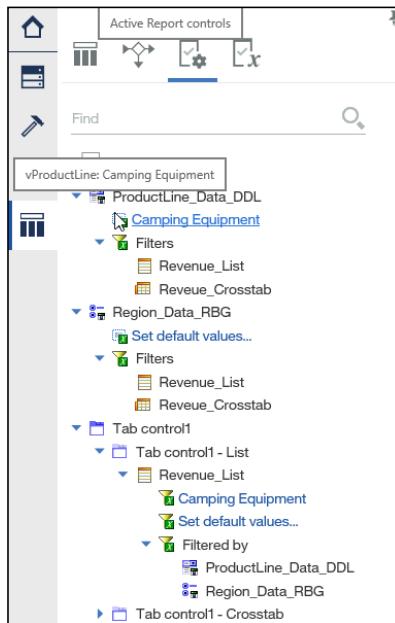
- Shows the effect of existing variables on the control
- Can edit behavior of existing variables for the control
- Create new variables and set behavior for the control

Reaction behavior

- The behavior of the target control is specified through a conditional statement that is applied to the value of the variable that is passed to the target control. Again, the value of the variable is set based on what is clicked or selected in the source control. Here you can:
 - Shows existing conditional statements
 - Can edit existing conditional statements
 - Create new conditional statements
- Container Select: data in the target control will be selected when the condition is met.
- Container filter: data in the target control will be filtered when the condition is met
- Control Enable: the target control will become interactive when the condition is met.

Identify Active Report controls in a report

- Identify all the controls in a report using the Active Report Controls tab



Use Active Report connections

© Copyright IBM Corporation 2018

Identify Active Report controls in a report

To identify all the Active Report controls and their relationships to one another you can use the Active Report controls tab (inside the Queries tab). There, you can also:

- click a control in this tab to quickly locate the control in the report.
- set default values for variables affecting the control.

This is how you interpret the example in the slide:

- Data drop down list named ProductLine_Data_DDL filters two controls: Revenue_List and Revenue_Crosstab, using the variable vProductLine, that has a default value of Camping Equipment, you can click the default value to change it.
- Data radio button group named Region_Data_RB filters two controls: Revenue_List and Revenue_Crosstab, using a variable that does not have a default value. You can click the link to set the default value.
- Tab control1 has two cards (one is called List, and the other is called Crosstab)
- Revenue_List is affected by two variables, one of them has a default value of Camping Equipment (it is vProductLine), and the other has no default value. It is filtered by two other controls: ProductLine_Data_DDL and Region_Data_RB

IBM Training IBM

Identify Active Report variables in a report

- Identify all the variables in a report using the Active Report variables tab.

Use Active Report connections © Copyright IBM Corporation 2018

Identify Active Report variables in a report

Active report variables work with active report controls to add interactivity to the report. If you want to identify all the variables and their behavior you can use the Active Report Variables tab. There you can also:

- create new variables
- set default values for variables
- manage range variables

This is how you interpret the variables in the slide:

- The vProductLine variable:
 - has a default value of Camping Equipment
 - is set by and selects the Data drop down list called ProductLine_Data_DDL
 - filters two controls: a List called Revenue_List and a Crosstab called Revenue_Crosstab
- The vRegion variable:
 - It does not have a default value

The “Manage ranges” link can be used to set the minimum and maximum values for range variables. Range variables are used in association with slider type controls to set the minimum and maximum values for the control. The variables must have the number data type. Two variables are used to define the minimum and maximum values of the

range variable. The two variables are used to define the minimum and maximum values of the slider.

Demonstration 1

Change filtering and selection behavior in a report

		2010	2011	2012	2013		2010	2011	2012	2013		2010	2011	2012	2013			
Year	Product line	Product type	Revenue	Year	Product line	Product type	Revenue	Year	Product line	Product type	Revenue	Year	Product line	Product type	Revenue	Year	Product line	
2010	Camping Equipment	Cooking Gear	\$59,761,536.50	2010	Camping Equipment	Cooking Gear	\$59,761,536.50	2010	Camping Equipment	Cooking Gear	\$59,761,536.50	2010	Camping Equipment	Cooking Gear	\$59,761,536.50	2010	Camping Equipment	
		Lanterns	\$28,662,904.19			Lanterns	\$28,662,904.19				Lanterns	\$28,662,904.19					Lanterns	
		Packs	\$70,296,289.17			Packs	\$70,296,289.17				Packs	\$70,296,289.17					Packs	
		Sleeping Bags	\$65,239,462.96			Sleeping Bags	\$65,239,462.96				Sleeping Bags	\$65,239,462.96					Sleeping Bags	
		Tents	\$109,026,145.24			Tents	\$109,026,145.24				Tents	\$109,026,145.24					Tents	
	Camping Equipment - Total		332,984,338.86		Camping Equipment - Total		332,984,338.86		Camping Equipment - Total		332,984,338.86		Camping Equipment - Total		332,984,338.86		Camping Equipment - Total	
	Golf Equipment	Golf Accessories	\$10,655,401.10		Golf Equipment	Golf Accessories	\$10,655,401.10		Golf Equipment	Golf Accessories	\$10,655,401.10		Golf Equipment	Golf Accessories	\$10,655,401.10		Golf Equipment	Golf Accessories
		Irons	\$54,093,311.24			Irons	\$54,093,311.24				Irons	\$54,093,311.24					Irons	
		Putters	\$29,419,377.82			Putters	\$29,419,377.82				Putters	\$29,419,377.82					Putters	
		Woods	\$59,385,760.82			Woods	\$59,385,760.82				Woods	\$59,385,760.82					Woods	
	Golf Equipment - Total		151,553,850.86		Golf Equipment - Total		151,553,850.86		Golf Equipment - Total		151,553,850.86		Golf Equipment - Total		151,553,850.86		Golf Equipment - Total	
	Outdoor Protection	First Aid	\$6,902,750.07		Outdoor Protection	First Aid	\$6,902,750.07		Outdoor Protection	First Aid	\$6,902,750.07		Outdoor Protection	First Aid	\$6,902,750.07		Outdoor Protection	First Aid
		Insect Repellents	\$17,964,327.13			Insect Repellents	\$17,964,327.13				Insect Repellents	\$17,964,327.13					Insect Repellents	
		Sunscreen	\$11,298,443.87			Sunscreen	\$11,298,443.87				Sunscreen	\$11,298,443.87					Sunscreen	
	Outdoor Protection - Total		36,163,521.87		Outdoor Protection - Total		36,163,521.87		Outdoor Protection - Total		36,163,521.87		Outdoor Protection - Total		36,163,521.87		Outdoor Protection - Total	
	Personal Accessories	Binoculars	\$29,246,444.08		Personal Accessories	Binoculars	\$29,246,444.08		Personal Accessories	Binoculars	\$29,246,444.08		Personal Accessories	Binoculars	\$29,246,444.08		Personal Accessories	Binoculars
		Eyewear	\$154,310,479.02			Eyewear	\$154,310,479.02				Eyewear	\$154,310,479.02					Eyewear	
		Knives	\$36,374,634.09			Knives	\$36,374,634.09				Knives	\$36,374,634.09					Knives	
		Navigation	\$51,598,510.99			Navigation	\$51,598,510.99				Navigation	\$51,598,510.99					Navigation	
		Watches	\$120,117,025.43			Watches	\$120,117,025.43				Watches	\$120,117,025.43					Watches	
	Personal Accessories - Total		391,647,093.61		Personal Accessories - Total		391,647,093.61		Personal Accessories - Total		391,647,093.61		Personal Accessories - Total		391,647,093.61		Personal Accessories - Total	
	2010 - Total		914,352,803.72		2010 - Total		914,352,803.72		2010 - Total		914,352,803.72		2010 - Total		914,352,803.72		Overall - Total	

Use Active Report connections

© Copyright IBM Corporation 2018

Demonstration 1: Change filtering and selection behavior in a report

Demonstration 1: Change filtering and selection behavior in a report

Purpose:

You need to modify the selection behavior within an Active Report so that when the value in a control is selected, data in another control is filtered instead of selected.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create the layout

1. Using the **GO sales (query)** package, create a new **Blank active report**.
2. Click the **Toolbox**  tab, then expand the **ACTIVE REPORT** section, then drag a **Data radio button group**  to the work area.
3. In the **Object and query name** dialog, under **Name**, type **Year_Data_RBG**, under **Query**, type **Year_Data_RBG_Query**, and then click **OK**.
4. From the **Data**  tab, drag **Time > Year**, to the **Year_Data_RBG** drop zone.
5. Click the **Toolbox** tab, then drag a **Table**  to the work area under **Year_Data_RBG**.
6. In the **Insert table** dialog, click **OK**.
7. On the Application bar, click **Show properties**  to open the Properties pane.
8. In the Properties pane, under **BOX**, double-click the **Margin** property to open it.
9. Set **Top margin** to **20**, then click **OK**.
10. Click the left table cell, then Ctrl-click the right cell.
11. From the **Toolbar**, click the small arrow **Vertical alignment options**  , then click **Top** .
12. In the left table cell, click **Add**  , then click **List** .
13. In the **Object and query name** dialog, set **Name** to **Revenue_List1**, and **Query Name** to **Revenue_Query**, then click **OK**.
14. From the **Data**  tab, drag the following data items to **Revenue_List1**:
 - **Time > Year**
 - **Products > Product line, Product type**
 - **Sales > Revenue**
15. Click **<Year>**, then Ctrl-click **<Product line>**, then on the **Toolbar**, click **Group / Ungroup** .

16. Click <Revenue>, then on the Toolbar, click **Summarize** Σ > **Summarize in Output** > **Total** Σ .

17. With **Revenue_List1** still selected, press **Ctrl-C** to copy it.

18. Click the right table cell, then press **Ctrl-V** to paste.

The result appears as follows:

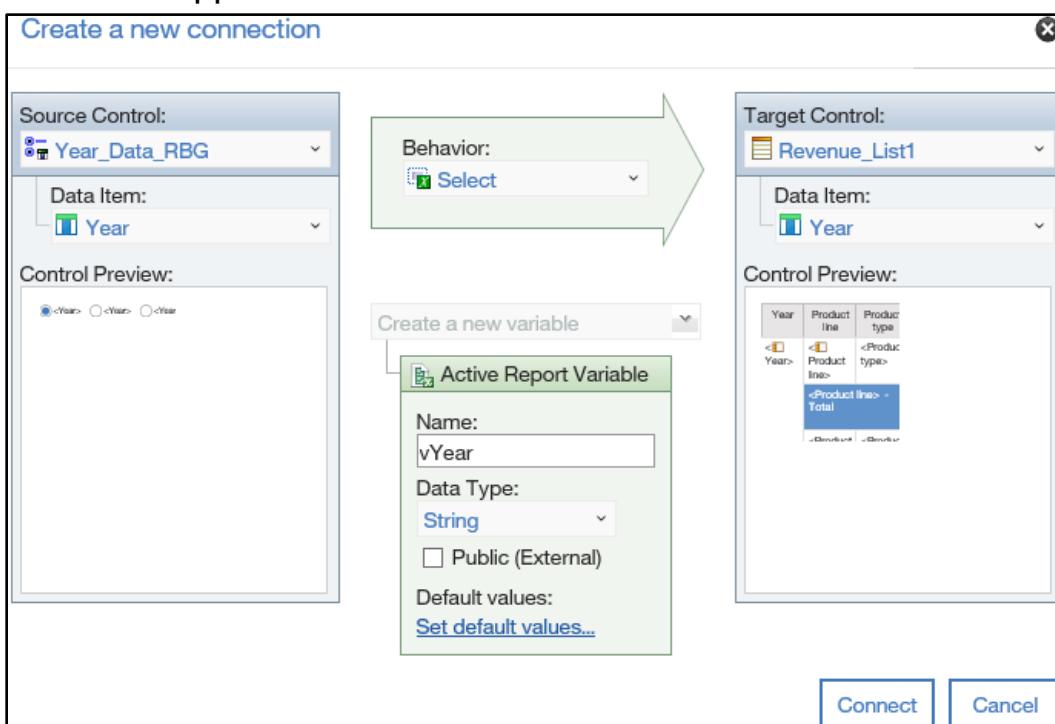
Year	Product line	Product type	Revenue
<Year>	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
<Year> - Total			<Total(Revenue)>
<Year>	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
<Year> - Total			<Total(Revenue)>
Overall - Total			<Total(Revenue)>

Year	Product line	Product type	Revenue
<Year>	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
<Year> - Total			<Total(Revenue)>
<Year>	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
	<Product line>	<Product type>	<Revenue>
	<Product line> - Total		<Total(Revenue)>
<Year> - Total			<Total(Revenue)>
Overall - Total			<Total(Revenue)>

Task 2. Create two connections to the List objects

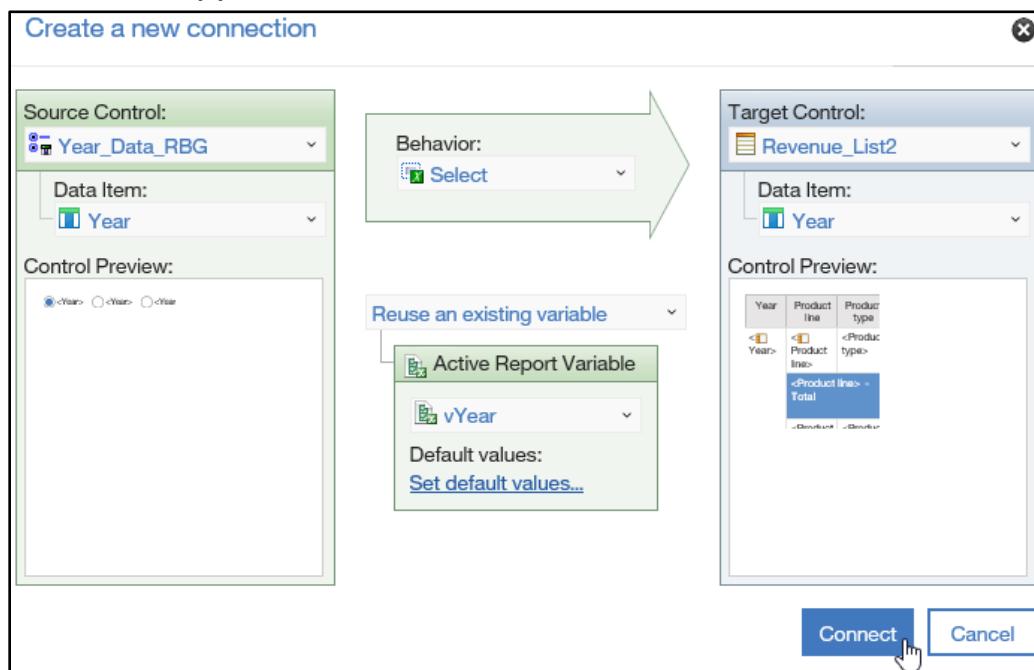
1. Click the **Year_Data_RBG** control, then from the Toolbar, click **Create a new connection** and specify the following connection information:
- **Year_Data_RBG (Year) > Select > Revenue_List1 (Year)**
 - New variable: **vYear**

The result appears as follows:



2. Click **Connect**.
3. In the Toolbar, click **Create a new connection** and specify the following connection information:
 - Year_Data_RBG (Year) > Select > Revenue_List2 (Year)
 - Reuse an existing variable: vYear

The result appears as follows:



4. Click **Connect**.
5. On the Application bar, click **Run options** (play icon), then click **Run Active Report**.
6. In the **Year_Data_RBG** control select **2010**.

The result appears as follows:

2010 2011 2012 2013				
Year	Product line	Product type	Revenue	
2010	Camping Equipment	Cooking Gear	\$59,761,536.50	
		Lanterns	\$28,662,904.19	
		Packs	\$70,296,289.17	
		Sleeping Bags	\$65,239,462.96	
		Tents	\$109,026,145.24	
Camping Equipment - Total		332,986,338.06		
2010	Golf Equipment	Golf Accessories	\$10,655,401.10	
		Irons	\$54,093,311.24	
		Putters	\$29,419,377.82	
		Woods	\$59,385,760.82	
Golf Equipment - Total		153,553,850.98		
2010	Outdoor Protection	First Aid	\$6,902,750.07	
		Insect Repellents	\$17,964,327.13	
		Sunscreen	\$11,298,443.87	
Outdoor Protection - Total		36,165,521.07		
2010	Personal Accessories	Binoculars	\$29,246,444.08	
		Eyewear	\$154,310,479.02	
		Knives	\$36,374,634.09	
		Navigation	\$51,598,510.99	
		Watches	\$120,117,025.43	
Personal Accessories - Total		391,647,093.61		
2010 - Total		914,352,803.72		
Year	Product line	Product type	Revenue	
2010	Camping Equipment	Cooking Gear	\$59,761,536.50	
		Lanterns	\$28,662,904.19	
		Packs	\$70,296,289.17	
		Sleeping Bags	\$65,239,462.96	
		Tents	\$109,026,145.24	
Camping Equipment - Total		332,986,338.06		
2010	Golf Equipment	Golf Accessories	\$10,655,401.10	
		Irons	\$54,093,311.24	
		Putters	\$29,419,377.82	
		Woods	\$59,385,760.82	
Golf Equipment - Total		153,553,850.98		
2010	Outdoor Protection	First Aid	\$6,902,750.07	
		Insect Repellents	\$17,964,327.13	
		Sunscreen	\$11,298,443.87	
Outdoor Protection - Total		36,165,521.07		
2010	Personal Accessories	Binoculars	\$29,246,444.08	
		Eyewear	\$154,310,479.02	
		Knives	\$36,374,634.09	
		Navigation	\$51,598,510.99	
		Watches	\$120,117,025.43	
Personal Accessories - Total		391,647,093.61		
2010 - Total		914,352,803.72		

Both Lists are highlighted based on what is selected in the Data radio button group control.

7. Close the rendered report tab.

You will modify the reaction behavior so that when a value is selected in the Year_Data_RBG control, both lists are filtered instead of selected.

Task 3. Modify the reaction behavior of the List controls

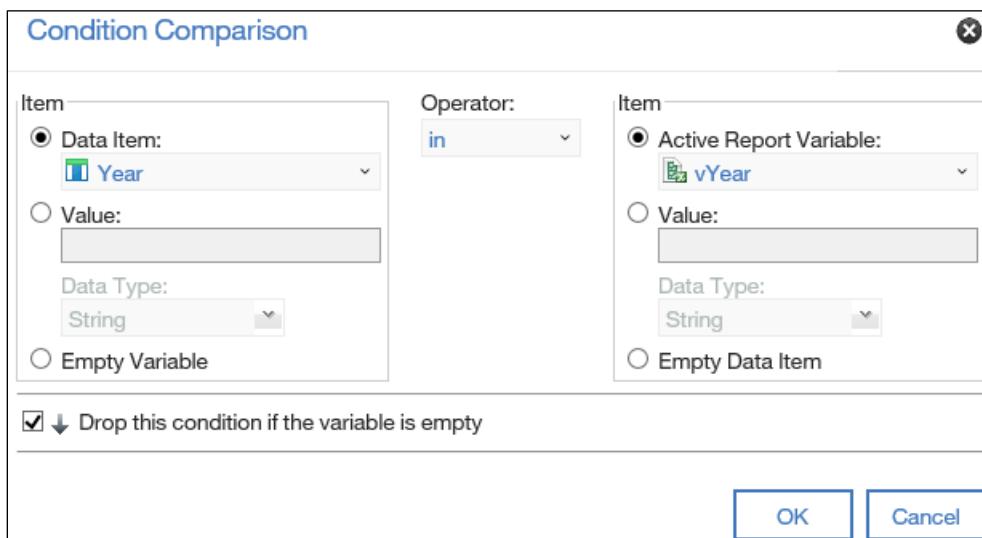
1. Click **Revenue_List1**, then in the **Toolbar**, click **Interactive Behavior** .

Under Report behavior, you can see that Revenue_List1 is currently selected by the Data radio button group control named Year_Data_RBG. You would like to change the reaction behavior of this control so that when a selection is made in Year_Data_RBG, Revenue_List1 is filtered instead of selected.

2. Under **Reaction behavior**, hover the cursor over **Container selection**, and then click **Delete** .

3. Hover the cursor over **Container filter**, and then click **Edit** .

4. Ensure that the **Condition Comparison** box appears as shown below:



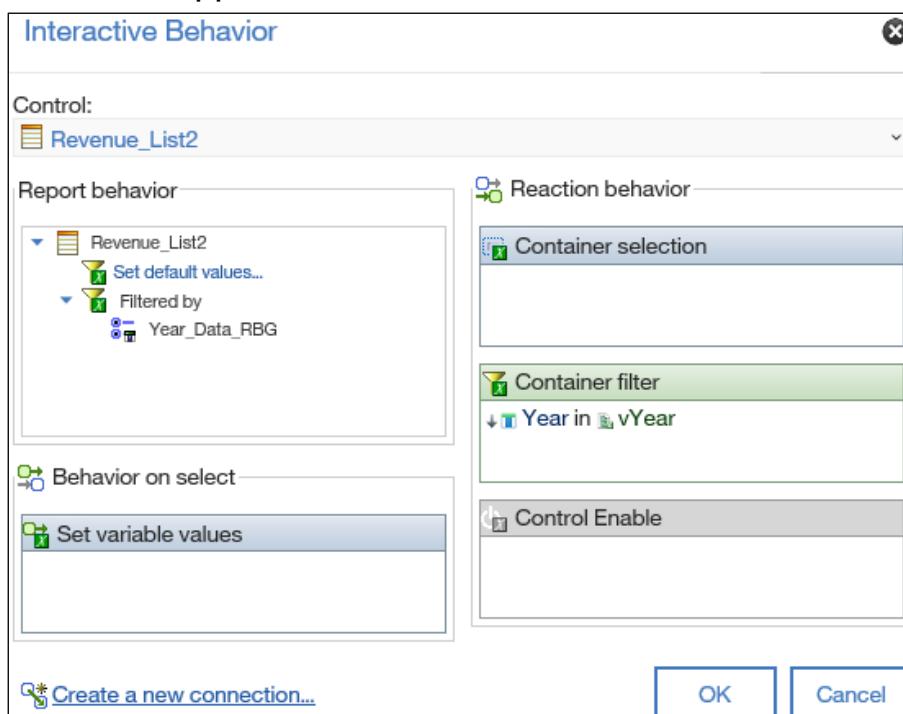
5. Click **OK**, and then click **OK** again.

The condition for Container filter appears as Year in vYear.

Now you would like to change the behavior on Revenue_List2 so that it is the same as Revenue_List1, with respect to how it will react to selections in Year_Data_RBG.

6. From the **Control** combo box, change the selected control from **Revenue_List1** to **Revenue_List2** to see its interactive behavior.
 7. Repeat steps **2 to 5**.

The result appears as follows:



8. Click **OK**.

9. On the Application bar, click **Run options** , then click **Run Active Report**.
10. In the **Year_Data_RB** control, select **2010**.

The result appears as follows:



2010 2011 2012 2013			
Year	Product line	Product type	Revenue
2010	Camping Equipment	Cooking Gear	\$59,761,536.50
		Lanterns	\$28,662,904.19
		Packs	\$70,296,289.17
		Sleeping Bags	\$65,239,462.96
		Tents	\$109,026,145.24
	Camping Equipment - Total		332,986,338.06
2010	Golf Equipment	Golf Accessories	\$10,655,401.10
		Irons	\$54,093,311.24
		Putters	\$29,419,377.82
		Woods	\$59,385,760.82
	Golf Equipment - Total		153,553,850.98
2010	Outdoor Protection	First Aid	\$6,902,750.07
		Insect Repellents	\$17,964,327.13
		Sunscreen	\$11,298,443.87
	Outdoor Protection - Total		36,165,521.07
2010	Personal Accessories	Binoculars	\$29,246,444.08
		Eyewear	\$154,310,479.02
		Knives	\$36,374,634.09
		Navigation	\$51,598,510.99
		Watches	\$120,117,025.43
	Personal Accessories - Total		391,647,093.61
2010 - Total		914,352,803.72	
Overall - Total		914,352,803.72	

Year	Product line	Product type	Revenue
2010	Camping Equipment	Cooking Gear	\$59,761,536.50
		Lanterns	\$28,662,904.19
		Packs	\$70,296,289.17
		Sleeping Bags	\$65,239,462.96
		Tents	\$109,026,145.24
	Camping Equipment - Total		332,986,338.06
2010	Golf Equipment	Golf Accessories	\$10,655,401.10
		Irons	\$54,093,311.24
		Putters	\$29,419,377.82
		Woods	\$59,385,760.82
	Golf Equipment - Total		153,553,850.98
2010	Outdoor Protection	First Aid	\$6,902,750.07
		Insect Repellents	\$17,964,327.13
		Sunscreen	\$11,298,443.87
	Outdoor Protection - Total		36,165,521.07
2010	Personal Accessories	Binoculars	\$29,246,444.08
		Eyewear	\$154,310,479.02
		Knives	\$36,374,634.09
		Navigation	\$51,598,510.99
		Watches	\$120,117,025.43
	Personal Accessories - Total		391,647,093.61
2010 - Total		914,352,803.72	
Overall - Total		914,352,803.72	

The behavior is changed, and now when a value is selected in the **Year_Data_RB** control, both Lists are filtered instead of selected.

11. Close the rendered report tab.

You will use this report as a starting point for a later demonstration, so you will save this report to My content.

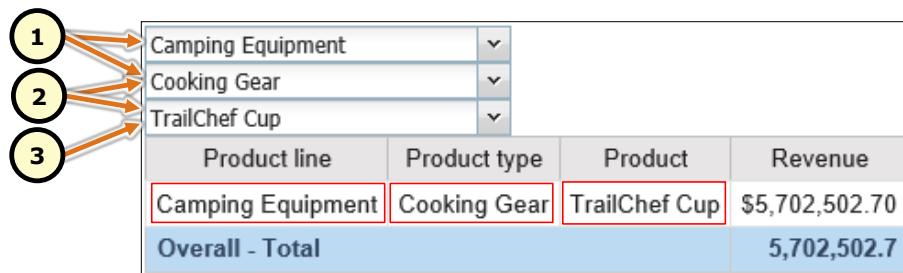
12. Save the report as **Unit 2 Demo 1** in **My content**.
13. Leave **IBM Cognos Analytics - Reporting** open for the next demonstration.

Results:

You modified the selection behavior within an existing Active Report so that when the value in a control is selected, data in another control is filtered instead of selected.

Interactive behavior example - cascading

- Set the interactive behavior between controls to achieve a cascading effect.



Use Active Report connections

© Copyright IBM Corporation 2018

Interactive behavior example - cascading

You can set/clear Active Report variables to achieve cascading.

1. A selection is made in the first Data drop-down list control. Variable values are set, and the second Data drop-down list control is populated based on selection. List is filtered.
2. A selection is made in second Data drop-down list control. Variable values are set, and the third Data drop-down list control is populated based on selection. List is filtered.
3. A selection is made in third Data drop-down list control. Variable values are set, and the List control is filtered based on selections in all Data drop-down list controls.

Demonstration 2

Create an Active Report with cascading controls

Camping Equipment	▼		
Cooking Gear	▼		
TrailChef Cup	▼		
Product line	Product type	Product	Revenue
Camping Equipment	Cooking Gear	TrailChef Cup	\$5,702,502.70
Overall - Total			5,702,502.7

Use Active Report connections

© Copyright IBM Corporation 2018

Demonstration 2: Create an Active Report with cascading controls

Demonstration 2: Create an Active Report with cascading controls

Purpose:

Add interactive behavior to Active Report controls to create a cascading effect between controls.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

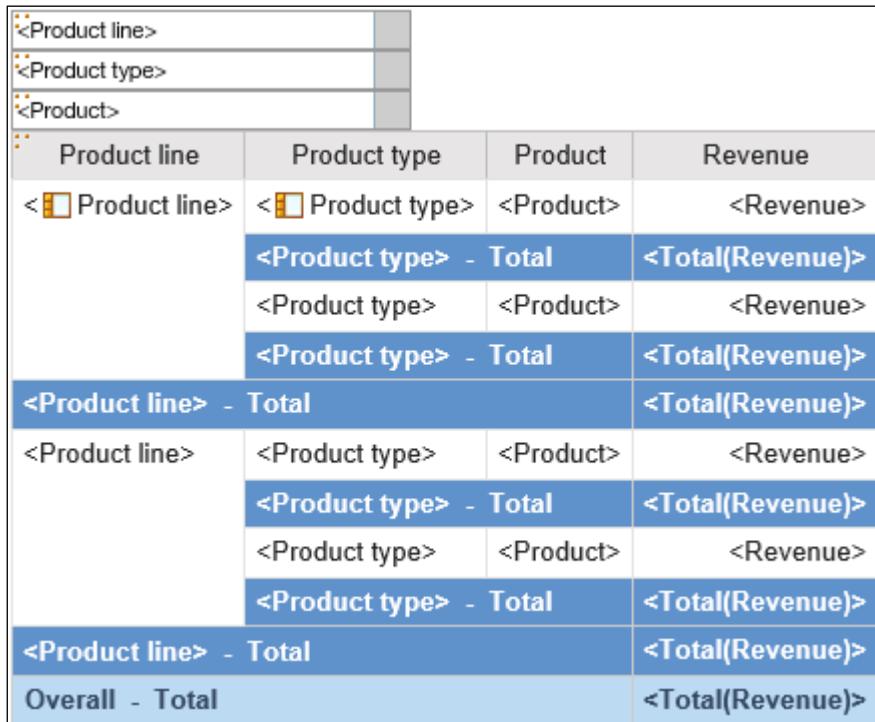
Namespace: Sales (query)

Task 1. Add controls and data to the Active Report

1. Using the **GO sales (query)** package, create a new **Blank active report**.
2. Click the **Toolbox**  tab, then expand the **ACTIVE REPORT** section.
3. Drag a **Data drop-down list**  to the work area. In the **Object and query name** dialog, specify the following properties:
 - Name: **ProductLine_Data_DDL**
 - Query: **Products_Query**
 - Click **OK**.
 You will use the same query (**Products_Query**) for the other Data drop-down lists.
4. From the Toolbox, drag a **Data drop-down list**  to the work area. In the **Object and query name** dialog:
 - Name: **ProductType_Data_DDL**
 - Query: **Products_Query**
 - Click **OK**.
5. From the Toolbox, drag a **Data drop-down list**  to the work area. In the **Object and query name** dialog:
 - Name: **Product_Data_DDL**
 - Query: **Products_Query**
 - Click **OK**.
6. From the **Data**  tab, drag Products > **Product line**, to the **ProductLine_Data_DDL** drop zone.
7. From the **Data**  tab, drag Products > **Product type**, to the **ProductType_Data_DDL** drop zone.
8. From the **Data**  tab, drag Products > **Product**, to the **Product_Data_DDL** drop zone.

9. From the **Data items** tab, Ctrl-click all data items, then drag them to the page under the last Data drop-down list.
This will create List1, containing the three data items.
10. From the **Source** tab, drag Sales > **Revenue** to the end of the List.
11. Click **<Product line>**, then Ctrl-click **<Product type>** then from the Toolbar, click **Group / Ungroup** .
12. Click **<Revenue>**, then from the Toolbar, click **Summarize**  > **Summarize in Output > Total** .

The result appears as follows:



<Product line>			
<Product type>			
<Product>			
Product line	Product type	Product	Revenue
<Product line>	<Product type>	<Product>	<Revenue>
		<Product type> - Total	<Total(Revenue)>
		<Product type>	<Revenue>
		<Product type> - Total	<Total(Revenue)>
<Product line> - Total			<Total(Revenue)>
<Product line>	<Product type>	<Product>	<Revenue>
		<Product type> - Total	<Total(Revenue)>
		<Product type>	<Revenue>
		<Product type> - Total	<Total(Revenue)>
<Product line> - Total			<Total(Revenue)>
Overall - Total			<Total(Revenue)>

Task 2. Connect the Data drop-down lists to the List

1. Click **ProductLine_Data_DDL**, then on the Toolbar, click **Create a new connection**, and specify the following connection properties:
 - **ProductLine_Data_DDL (Product line) > Filter > List1 (Product line)**
 - New variable: **vProductLine**
 - Default values: **Camping Equipment**
2. Click **ProductType_Data_DDL**, then in the Toolbar, click **Create a new connection**, and specify the following connection properties:
 - **ProductType_Data_DDL (Product type) > Filter > List1 (Product type)**
 - New variable: **vProductType**
 - Default values: **Camping Equipment**

3. Click **Product_Data_DDL**, then in the Toolbar, click **Create a new connection**, and specify the following connection properties:
 - **Product_Data_DDL (Product type)** > Filter > **List1 (Product)**
 - New variable: vProduct

Task 3. Add interactive behaviors on the Product type drop-down list

Upon selection of a product line, you want the product type and product drop-down lists to default to empty.

1. Click **ProductLine_Data_DDL**, then in the Toolbar, click **Interactive Behavior**.
2. Under **Behavior on select/Set variable values**, click **Edit**.
3. Beside **vProdType**, select **Set to Empty**.
4. Beside **vProduct**, select **Set to Empty**.
5. Click **OK** twice.

Task 4. Add interactive behaviors on the Product type drop-down list

Upon selection in the product line drop-down list, you want the product type drop-down list to be filtered on the values that belong to the product line selected. You will add product line to the query so the filter can be defined.

1. Click **ProductType_Data_DDL** to select it.
2. From the Data/Data Items tab, drag **Product line** to the **ProductType_Data_DDL/Extra Items** drop zone.
3. Click **ProductType_Data_DDL**, then in the Toolbar, click **Interactive Behavior**.
4. Under **Reaction behavior/Container filter**, click **Edit**.
5. Under **Data Item**, select **Product line**, under **Active Report Variable**, select **vProductLine**, and then click **OK** twice.

In addition, you want the product drop-down list to default to empty.

6. Under **Behavior on select/Set variable values**, click **Edit**.
7. Beside **vProduct**, select **Set to Empty**, and then click **OK** twice.

Task 5. Add interactive behaviors on the Product drop-down list

Upon selection in the product type drop-down list, you want the product drop-down list to be filtered on the values that belong to the product type selected. You will add product type to the query so the filter can be defined.

In addition, you want the product drop-down list to only be enabled once a user selects a product line.

1. Click the **Product_Data_DDL** object to select it.
2. From the Data/Data Items tab, drag **Product type** to the **Product_Data_DDL/Extra Items** drop zone.
3. Click **Product_Data_DDL**, then from the Toolbar, click **Interactive Behavior**.
4. Under **Reaction behavior/Container filter**, click **Edit**.
5. Under **Data Item**, select **Product type**, and then under **Active Report Variable**, select **vProductType**.
6. Click **OK** twice.
7. Under **Reaction behavior/Control Enable**, click **Edit**.
8. Under **Item** (on the left), select **Empty Variable**.
9. Under **Item** (on the right), under **Active Report Variable**, select **vProductType**.
10. Click **OK**, and then click **NOT**.
The expression should now read: NOT (vProductType is empty).
11. Click **OK** twice.

Task 6. Create default values for data drop-down lists

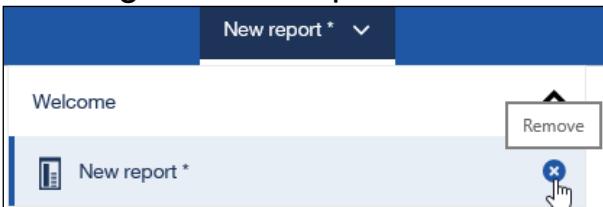
1. Select the **ProductType_Data_DDL** control, and then on the **Application bar**, click **Show properties**.
2. In the Properties pane, under **GENERAL**, set **No Value List Item** to **Show**.
The No Value List Item Properties appear.
3. Double-click **Label**, in the **Default** text box, type **All**, and then click **OK**.
4. Repeat steps 1 and 2 for the **Product_Data_DDL** control.
Next, you want to specify a product line default value.
5. On the side bar, click **Queries**, then click the **Active Report variables**  tab.
6. Expand **vProductLine**, and then click **Set default values**.
7. On the Application bar, click **Run options** , then click **Run Active Report**.
8. Select **Cooking Gear** as product type and **TrailChef Cup** as product.

The result appears as follows:

Product line	Product type	Product	Revenue
Camping Equipment	Cooking Gear	TrailChef Cup	\$5,702,502.70
Overall - Total			5,702,502.7

9. Close the rendered report tab.

10. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



11. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You have added interactive behavior to Active Report controls to create a cascading effect between controls.

Single vs multiple variables (1 of 2)

- Use a single variable to achieve the same data results in multiple controls.

vYear=2013

Year	Product line	Product type	Revenue
2013	Camping Equipment	Cooking Gear	\$58,313,800.35
		Lanterns	\$28,034,475.54
		Packs	\$83,157,796.99
		Sleeping Bags	\$68,730,008.17
		Tents	\$114,674,248.92
Golf Equipment	Golf Accessories	\$11,422,333.75	
	Irons	\$63,606,666.75	
	Putters	\$18,145,999.48	
	Woods	\$81,565,819.31	
Mountaineering Equipment	Climbing Accessories	\$29,483,205.87	

Year	Product line	Product type	Revenue
2013	Camping Equipment	Cooking Gear	\$58,313,800.35
		Lanterns	\$28,034,475.54
		Packs	\$83,157,796.99
		Sleeping Bags	\$68,730,008.17
		Tents	\$114,674,248.92
Golf Equipment	Golf Accessories	\$11,422,333.75	
	Irons	\$63,606,666.75	
	Putters	\$18,145,999.48	
	Woods	\$81,565,819.31	
Mountaineering Equipment	Climbing Accessories	\$29,483,205.87	

Use Active Report connections

© Copyright IBM Corporation 2018

Single vs multiple variables

In this example, two connections have been created on the Data radio button group control. For each of the connections specified, the same variable, vYear, is used to control the display of data in the two List controls. The result is that both Lists are filtered in the same manner, based on what is selected in the Data radio button group control. In this case, 2013 has been selected.

Single vs multiple variables (2 of 2)

- Use multiple variables to achieve different data results in multiple controls.

vPL1=Golf Equipment

Year	Product line	Product type	Revenue
2013	Golf Equipment	Golf Accessories	\$11,422,333.75
		Irons	\$63,606,666.75
		Putters	\$18,145,999.48
		Woods	\$81,565,819.31

vPL2=Mountaineering Equipment, Outdoor Protection

Year	Product line	Product type	Revenue
2013	Mountaineering Equipment	Climbing Accessories	\$29,483,205.87
		Rope	\$39,444,903.89
		Safety	\$28,014,081.20
		Tools	\$44,578,458.74
	Outdoor Protection	First Aid	\$846,984.93
		Insect Repellents	\$2,062,062.11
		Sunscreen	\$1,561,978.22

Use Active Report connections

© Copyright IBM Corporation 2018

In this example, a connection has been created for each of the Data check box group controls to connect to each List control. Each connection uses a different variable to control the display of data in the List controls. The Data check box group control on the left uses the variable PL1, while the Data check box group control on the right uses PL2. Both variables represent the same data item, and in this case it is Product line. The result is that each List is filtered based on which Product line is selected in each of the Data check box group controls. Golf Equipment has been selected on the left, which filters the List on the left. Mountaineering Equipment and Outdoor Protection have been selected on the right, which filters the List on the right.

Demonstration 3

Use single vs. multiple variables

○ 2010 ○ 2011 ● 2012 ○ 2013

Camping Equipment
 Golf Equipment
 Mountaineering Equipment
 Outdoor Protection
 Personal Accessories

Camping Equipment
 Golf Equipment
 Mountaineering Equipment
 Outdoor Protection
 Personal Accessories

Year	Product line	Product type	Revenue
2011	Golf Equipment	Golf Accessories	\$13,251,774.09
		Irons	\$55,116,575.97
		Putters	\$28,923,250.88
		Woods	\$70,714,826.13
	Golf Equipment - Total		168,006,427.07
Overall - Total			168,006,427.07

Year	Product line	Product type	Revenue
2011	Mountaineering Equipment	Climbing Accessories	\$21,876,490.73
		Rope	\$28,655,271.69
		Safety	\$22,505,865.68
		Tools	\$34,062,031.84
	Mountaineering Equipment - Total		107,099,659.94
Personal Accessories - Total			456,323,355.9
Overall - Total			563,423,015.84

Use Active Report connections

© Copyright IBM Corporation 2018

Demonstration 3: Use single vs. multiple variables

Demonstration 3: Use single vs. multiple variables

Purpose:

Create a report to show how Active Report controls can be configured to use a single variable or multiple variables. You will begin by examining a report with a control that uses the same variable to control the display of the same data in different controls. You will then modify the report to include additional controls that use multiple variables to control the display of the same data in different controls.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Examine Active Report controls that are configured to use a single variable

1. On the side panel, click **My content**, then right-click **Unit 2 Demo 1**, then click **Edit report**.
2. Click the **Revenue_List1**, then in the Toolbar, click **Interactive Behavior** .
3. Hover the cursor over **Container filter**, and then click **Edit**.
According to the container filter expression, the behavior for this List control is controlled by a variable named vYear.
4. Click **Cancel** twice.
5. Repeat steps 2 to 4 for the **Revenue_List2** control.
Both List controls are controlled by the vYear variable.
You will now modify the report to include controls that use multiple variables to control the display of data in different controls.

Task 2. Add additional controls that use multiple variables

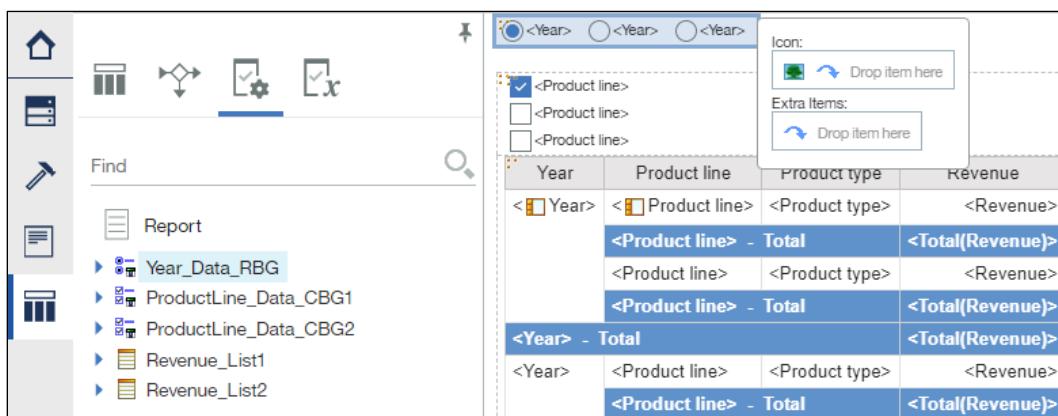
1. Click anywhere in **Revenue_List1**, then from the Toolbar, click **More > Insert > Rows above**.
This will insert a table row before the row containing the two List objects.
2. On the side bar, click **Toolbox**.
3. Drag a **Data check box group** to the left table cell of the new row.
 - Name: **ProductLine_Data_CBG1**
 - Query: **ProductLine_Query**
4. Drag a **Data check box group** to the right table cell of the new row.

- Name: **ProductLine_Data_CBG2**
 - Query: **ProductLine_Query**
5. Click the **ProductLine_Data_CBG1** control to select it, and then in the **Properties** pane, under **GENERAL**, double-click **Orientation** to change from **Horizontal** to **Vertical**.
 6. Repeat step 5 to make the **ProductLine_Data_CBG2** control display vertically.
 7. On the side bar, click **Data**.
 8. On the **Source** tab, from **Sales (query) > Products**, drag **Product line** to the drop zone of the **ProductLine_Data_CBG1** control.
 9. In the Toolbar, click **Create a new connection**, and specify the following connection information:
 - **ProductLine_Data_CBG1 > Filter > Revenue_List1**
 - New variable: **vProductLine1**
 - Click **Connect**
 10. On the **Data items** tab, from **ProductLine_Query**, drag **Product line** to the drop zone of the **ProductLine_Data_CBG2** control.
 11. In the Toolbar, click **Create a new connection**, and specify the following connection information:
 - **ProductLine_Data_CBG2 > Filter > Revenue_List2**
 - New variable: **vProductLine2**
 - Click **Connect**

Task 3. Review controls and variables, and run the report

1. On the side bar, click **Queries**, then click the **Active Report controls**  tab, then click **Year_Data_RBG**.

The result appears as follows:



The screenshot shows the SAP BusinessObjects Web Intelligence interface. On the left, the navigation pane displays several reports: Year_Data_RBG (highlighted in blue), ProductLine_Data_CBG1, ProductLine_Data_CBG2, Revenue_List1, and Revenue_List2. The main work area contains a table with the following data:

Year	Product line	Product type	Revenue
<Year>	<Product line>	<Product type>	<Revenue>
	<Product line> - Total	<Total(Revenue)>	
	<Product line>	<Product type>	<Revenue>
	<Product line> - Total	<Total(Revenue)>	
<Year> - Total			<Total(Revenue)>
<Year>	<Product line>	<Product type>	<Revenue>
	<Product line> - Total	<Total(Revenue)>	

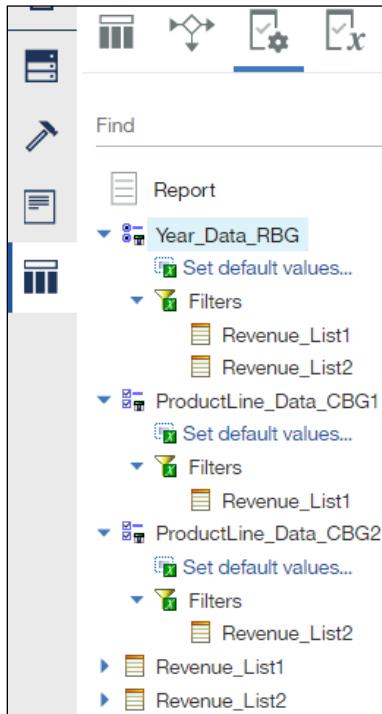
The **Year_Data_RBG** control is highlighted in the work area.

You can use this method to identify all the controls in the report, and select controls in the work area.

2. Expand the following:

- **Year_Data_RBG > Filters**
- **ProductLine_Data_CBG1 > Filters**
- **ProductLine_Data_CBG2 > Filters**

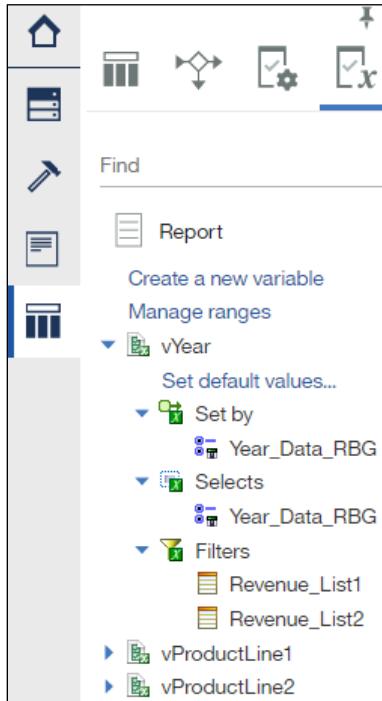
The result appears as follows:



You can see all the controls and their relationship to one another. For example, Year_Data_RBG filters both Revenue_List1 and Revenue_List2. While ProductLine_Data_CBG1 filters only Revenue_List1, and ProductLine_Data_CBG2 filters only Revenue_List2.

3. Click the **Active Report variables** tab, then expand **vYear**, **Set By**, **Selects**, and **Filters**.

The result appears as follows:



You can see all the variables in this report and how they are used. For example, the vYear variable, is set by the Year_Data_RBG control, is used to select a value in that control, and filters both Revenue_List1 and Revenue_List2.

4. Run the report, and then select **2011** in the **Year_Data_RBG** control.
Both lists are filtered based on what is selected in this control.

5. In the **ProductLine_Data_CBG1** control (on the left) select **Golf Equipment**, and then in the **ProductLine_Data_CBG2** control (on the right) select **Mountaineering Equipment** and **Personal Accessories**.

The result appears as follows:

Left List Control (2011):

Year	Product line	Product type	Revenue
2011	Golf Equipment	Golf Accessories	\$13,251,774.09
		Irons	\$55,116,575.97
		Putters	\$28,923,250.88
		Woods	\$70,714,826.13
Golf Equipment - Total		168,006,427.07	
Overall - Total		168,006,427.07	

Right List Control (2011):

Year	Product line	Product type	Revenue
2011	Mountaineering Equipment	Climbing Accessories	\$21,876,490.73
		Rope	\$28,655,271.69
		Safety	\$22,505,865.68
		Tools	\$34,062,031.84
Mountaineering Equipment - Total		107,099,659.94	
Personal Accessories - Total		456,323,355.9	
Overall - Total		563,423,015.84	

For each of the connections created for the Data check box group controls, different variables are used to control the display of data in the List controls. The result is that each List is filtered based on what is selected in each of the Data check box group controls.

6. Close the browser tab.
7. Leave **IBM Cognos Analytics** open for the next demonstration.

Results:

You have created a report that shows how Active Report controls can be configured to use a single variable or multiple variables.

Optimize Active Reports for mobile devices

- Consumers can use the IBM Cognos Mobile on iOS and Android to access Active Reports.
- Controls are automatically optimized to use the multi-touch gestures and style of the application.
- Use the optimal screen sizing for the device to ensure that the entire report is visible without scrolling
- For objects whose data can be difficult to select on a touch interface (i.e. maps), provide an alternate control (i.e. drop-down list).
- Use the iterator control with a deck control to enable the swipe gesture.

Optimize Active Reports for mobile devices

For both iOS and Android, consumers can use either the native or Web versions of IBM Cognos Analytics. You can search your application store for “IBM Cognos Mobile”.

Using the optimal screen sizing for the device improves the end user experience by ensuring that the entire report is visible without scrolling.

Ensure that all objects and containers are sized properly. If an object is larger than its container, the container will be pushed out and the entire report layout will be affected.

Large objects (such as lists) in a smaller block can result in scrollable data. To remain consistent with iOS standards, scrollable areas do not have visible scroll bars. End users should be made aware of this functionality.

Page dots have default colors (Dot Color and Active Dot Color) that you can change if the background color, or image, is too similar.

Demonstration 4

Create an Active Report using slider controls

2010 2011 2012 2013

Regions selected:
Americas, Central Europe, Northern Europe

Americas Asia Pacific Central Europe Northern Europe Southern Europe

Row number	Year	Region	Quantity	Row number
1	2010	Americas	6,439,569	1
2	2010	Central Europe	4,489,368	3
3	2010	Northern Europe	2,195,300	4
4	2011	Americas	7,091,968	6
5	2011	Central Europe	5,421,128	8
6	2011	Northern Europe	2,643,372	9
7	2012	Americas	7,866,166	11
8	2012	Central Europe	5,662,099	13
9	2012	Northern Europe	2,926,050	14
10	2013	Americas	5,814,848	16
11	2013	Central Europe	4,366,972	18
12	2013	Northern Europe	2,270,477	19
Overall - Total			57,187,317	

Use Active Report connections

© Copyright IBM Corporation 2018

Demonstration 4: Create an Active Report using slider controls

Demonstration 4: Create an Active Report using slider controls

Purpose:

You will create an Active Report that allows users to display data based on a range of values for Sales Quantity and Year, as well as the ability to filter data by Region.

Portal: <http://vclassbase/ibmcognos>

User/Password: **brettonf/Education1**

Template: **Blank active report**

Package: Go sales (query)

Namespace: **Sales (query)**

Task 1. Create an Active Report

1. From the **Welcome** page, navigate to **Team content** > **Samples** > **Models**.
 2. Right-click the **GO Sales (query)** package, then click **Create report** .
 3. In the **Templates and themes** dialog, double-click **Blank active report**.
 4. In the page body, click **Add** , then click **Table** .
 5. Set **Number of rows** to **3**, then click **OK**.
 6. From the **Toolbox** tab, drag a **List** to the bottom right cell of the table:
 - Name: **Quantity_List**
 - Query Name: **Quantity_List_Query**
 7. On the side bar, click **Data**, then expand **Sales (query)**.
 8. From the **Source** tab, drag the following query items to the List:
 - **Time > Year**
 - **Retailers > Region**
 - **Sales > Quantity**
 9. Click **<Quantity>**, then from the Toolbar, click **Summarize** > Summarize in Output > **Total**.

The result appears as follows:

Year	Region	Quantity
<Year>	<Region>	<Quantity>
<Year>	<Region>	<Quantity>
<Year>	<Region>	<Quantity>
Overall - Total		<Total(Quantity)>

Task 2. Create a Year filter using a discrete values slider

1. In the **Toolbox** tab, expand **ACTIVE REPORT**, and then drag a **Data discrete values slider** control to the top right table cell:
 - Name: **Year_Data_DVS**
 - Query Name: **Year_Data_DVS_Query**
2. On the side bar, click **Data**.
3. On the **Source** tab, under **Time**, drag **Year** to the **Labels** drop zone of the **Year_Data_DVS** control.
4. In the Toolbar, click **Create a new connection**, and specify the following:
 - **Year_Data_DVS (Year) > Select > Quantity_List (Year)**
 - New variable: **vYear**
 - Click **Connect**.
5. On the **Application bar**, click **Show properties**.
6. In the Properties pane, under **GENERAL**, set the **Range Type** property to **Range**.
The control will now display two thumb controls that let the user set both a minimum and maximum range of values.
7. Run the report, and then drag the right thumb to **2012** and the left thumb to **2011**.
The results appear as follows:



Year	Region	Quantity
2010	Americas	6,439,569
2010	Asia Pacific	5,072,815
2010	Central Europe	4,489,368
2010	Northern Europe	2,195,300
2010	Southern Europe	1,977,678
2011	Americas	7,091,968
2011	Asia Pacific	5,907,433
2011	Central Europe	5,421,128
2011	Northern Europe	2,643,372
2011	Southern Europe	2,460,784
2012	Americas	7,866,166
2012	Asia Pacific	6,617,098
2012	Central Europe	5,662,099
2012	Northern Europe	2,926,050
2012	Southern Europe	2,870,377
2013	Americas	5,814,848
2013	Asia Pacific	4,992,194
2013	Central Europe	4,366,972
2013	Northern Europe	2,270,477
2013	Southern Europe	2,151,395
Overall - Total		89,237,091

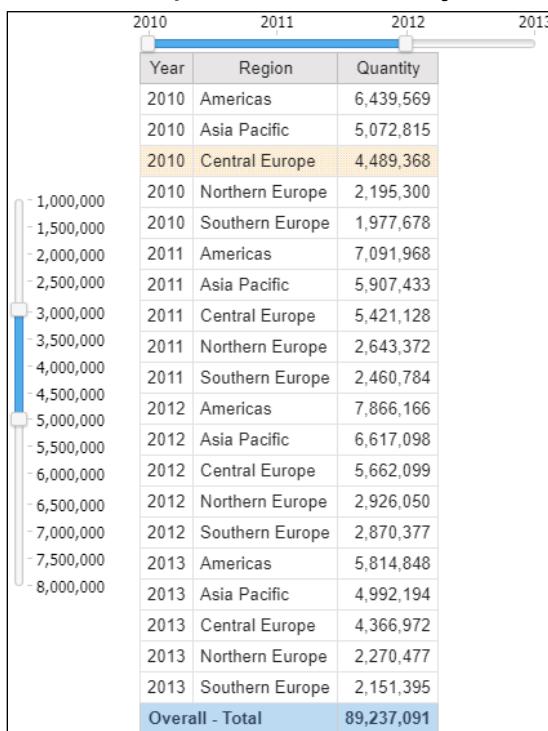
When the Data discrete values slider thumbs are set to 2011 and 2012, only the data for those years is highlighted.

8. Close the browser tab.

Task 3. Create a Quantity filter using a Continuous values

slider

1. From the Toolbox tab, drag a **Continuous values slider** to the bottom left table cell.
 2. In the Properties pane, set the following properties:
 - GENERAL > Orientation: **Vertical**
 - GENERAL > Range Type: **Range**
 - CONTINUOUS SLIDER > Step: **500000** (five hundred thousand)
 - CONTINUOUS SLIDER > Minimum Value: **1000000** (one million)
 - CONTINUOUS SLIDER > Maximum Value: **8000000** (eight million)
 - MISCELLANEOUS > Name: **Quantity_CVS**
- Note: If you do not enter the Step value before you enter the Minimum and Maximum values, an error message displays. You can simply click OK and set the Step value at this point.
3. In the Toolbar, click **Create a new connection**, and specify the following:
 - Quantity_CVS (Continuous Value) > **Select** > **Quantity_List** (Quantity)
 - **Minimum** tab > Active Report Variable > **vQuantity (Minimum)**
 - **Maximum** tab > Active Report Variable > **vQuantity (Maximum)**
 - Click **Connect**.
 4. Click the table cell that contains the **Quantity_CVS** control.
 5. In the **Properties** pane, under **POSITIONING**, set the **Size & Overflow > Width** property to **100px**.
 6. Run the report, and then adjust the slider thumbs to match the following report:



The screenshot shows a report interface with a header containing the years 2010, 2011, 2012, and 2013. Below the header is a horizontal slider with a blue track and a white thumb, positioned between the 2010 and 2011 markers. To the left of the slider is a vertical scale from 1,000,000 to 8,000,000. Below the slider is a table with four columns: Year, Region, and two numerical columns. At the bottom of the table is a row labeled "Overall - Total".

Year	Region	Quantity	Quantity
2010	Americas	6,439,569	
2010	Asia Pacific	5,072,815	
2010	Central Europe	4,489,368	
2010	Northern Europe	2,195,300	
2010	Southern Europe	1,977,678	
2011	Americas	7,091,968	
2011	Asia Pacific	5,907,433	
2011	Central Europe	5,421,128	
2011	Northern Europe	2,643,372	
2011	Southern Europe	2,460,784	
2012	Americas	7,866,166	
2012	Asia Pacific	6,617,098	
2012	Central Europe	5,662,099	
2012	Northern Europe	2,926,050	
2012	Southern Europe	2,870,377	
2013	Americas	5,814,848	
2013	Asia Pacific	4,992,194	
2013	Central Europe	4,366,972	
2013	Northern Europe	2,270,477	
2013	Southern Europe	2,151,395	
Overall - Total		89,237,091	

The highlighted data reflects only the results that match the values of both slider controls.

7. Close the browser tab.

Task 4. Add a Data check box group to filter data by region

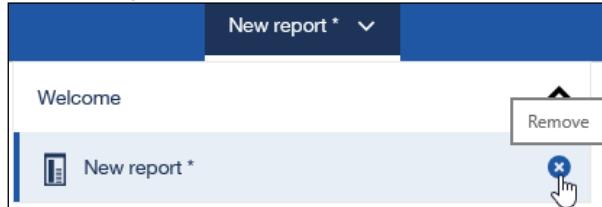
1. From the **Toolbox**, drag a **Data check box group** to the middle-right cell, above the **Quantity_List** control:
 - Name: **Region_Data_CBG**
 - Query Name: **Region_Data_CBG_Query**
2. On the side bar, click **Data**.
3. From the **Source** tab, from **Retailers**, drag **Region** to the **Region_Data_CBG** drop zone.
4. In the Toolbar, click **Create a new connection**, and specify the following:
 - Region_Data_CBG (Region) > **Filter** > **Quantity_List (Region)**
 - New variable: **vRegion**
 - Click **Connect**.
5. On the side bar, click **Toolbox**.
6. From the **Toolbox**, drag a **Variable text item** object to the left of the **Region_Data_CBG** control.
7. Set the **Active Report Variable** to **vRegion**, then click **OK**.
8. From the **Toolbox** tab, drag a **Text item** to the left of the new variable text item, type **Regions selected:**, type a space, and then click **OK**.
9. In the **Toolbox** tab, from the **TEXTUAL** section, drag a **Row number** object to the List control as the last column.
10. In the **Toolbox** tab, from the **ACTIVE REPORT** section, drag a **Row number** object to the List control as the first column.
11. On the Application bar, click **Run options** , then click **Run Active Report**.
12. Adjust the controls as follows:
 - Year_Data_CSV: **2010** to **2012**
 - Quantity_CSV: **2000000** to **5000000** (2 million to 5 million)
 - Region_Data_CBG: **Americas**, **Central Europe**, and **Northern Europe**

The results appear as follows:

	2010	2011	2012	2013
Regions selected:	Americas, Central Europe, Northern Europe	<input checked="" type="checkbox"/> Americas <input type="checkbox"/> Asia Pacific <input checked="" type="checkbox"/> Central Europe <input checked="" type="checkbox"/> Northern Europe <input type="checkbox"/> Southern Europe		
- 1,000,000				
- 1,500,000				
- 2,000,000				
- 2,500,000				
- 3,000,000				
- 3,500,000				
- 4,000,000				
- 4,500,000				
- 5,000,000				
- 5,500,000				
- 6,000,000				
- 6,500,000				
- 7,000,000				
- 7,500,000				
- 8,000,000				
Overall - Total				57,187,317

Note: The values in the ACTIVE REPORT Row Number column reflect the number of visible rows in the report. The values in the TEXTUAL Row Number column reflect the number of rows that the server returned for the query.

13. Close the rendered report tab.
14. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



15. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You created an **Active Report** that allows users to display data based on a range of values for Sales Quantity and Year, as well as the ability to filter data by Region.

Unit summary

- Describe Active Report connections
- Filter and select Active Report controls
- Modify the interactive behavior of report controls
- Identify Active Report controls and variables
- Use variables to control multiple controls independently
- Control multiple controls from a single variable
- Author and optimize Active Reports for mobile consumption

Unit 3 Active Report charts, visualizations, and decks

IBM Training

IBM

Active Report charts, visualizations, and decks

IBM Cognos Analytics (v11.0)

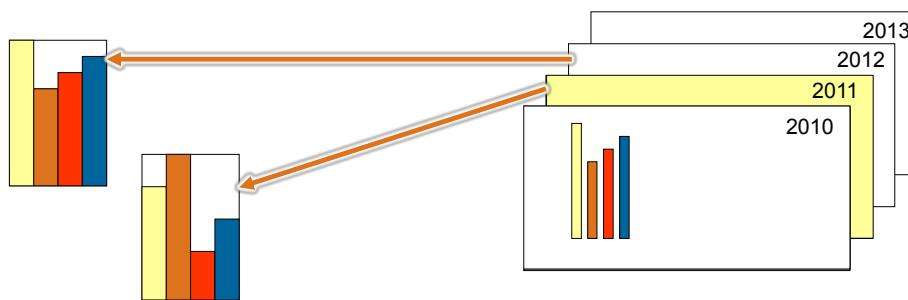
© Copyright IBM Corporation 2018
Course materials may not be reproduced in whole or in part without the written permission of IBM.

Objectives

- Describe characteristics of traditional charts in Active Reports
- Control data display using decks and data decks
- Use decks and data decks to display traditional charts
- Optimize decks for performance
- Describe characteristics of RAVE Visualizations

Traditional charts in Active Reports

- No client-side charting engine
 - chart images generated at runtime
 - all images stored in MHT file
- Use a deck or query to filter content



Traditional charts in Active Reports

When using traditional charts in Active Reports, there are some differences to be aware of compared to charts used in other IBM Cognos Analytics reports.

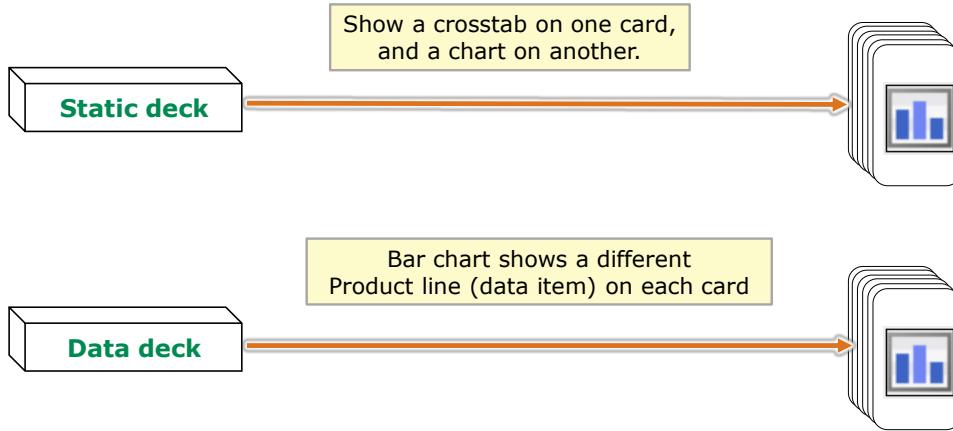
First, the generated MHT file for an Active Report does not contain a charting engine to dynamically generate charts. Instead, IBM Cognos Analytics generates all possible renditions of charts in an Active Report when it is first run, and then stores images of each of those charts in the MHT for later retrieval.

Because the charts are generated at the time the MHT file is first generated, filters cannot be applied to a chart to allow it to update its content based on the user's selections in the Active Report.

In order to create charts that update to reflect the user's selections, you need to either assign each unique chart to its own query, or put the chart inside a Deck to show each chart image based on the card selection in the Deck.

Examine decks

- Decks allow you to show cards containing different objects or data based on user selection in a control.



Examine decks

Decks in Active Reports can be compared to a deck of playing cards. At any time, the top card of the deck is visible to the user, and the user can cycle through each card, one at a time.

You can include objects such as Lists, Crosstabs, or Charts in a Deck.

Data items are added to a Data deck. The data items added to the Deck will define the cards that are created.

In static Decks, you define the number of cards you want, and define the content of each card.

Deck or Tab control?

- Tab control has one selection method (tabs).
- Static decks can link to other selection methods:
 - radio button groups
 - button bars
 - custom image buttons (static repeater table)
 - slider bars

Deck or Tab control?

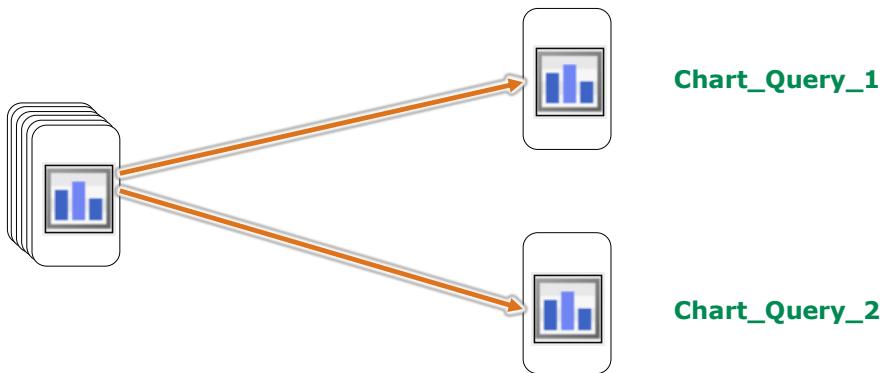
When choosing a control to display different sets of data, you can also use a Tab control. A Tab control can display different content for each tab.

A Deck or Data deck behaves in much the same way, but because Decks do not have a default method for the user to select a card by default, a wide variety of controls can be used to determine which card should be displayed in the report.

For basic report purposes, a Tab control may be sufficient, but for more complex reports and report layouts, a Deck can give more flexibility to the report author.

Traditional charts and static decks

- Deck controls the number of charts generated.
- Use a different query to generate each chart image.



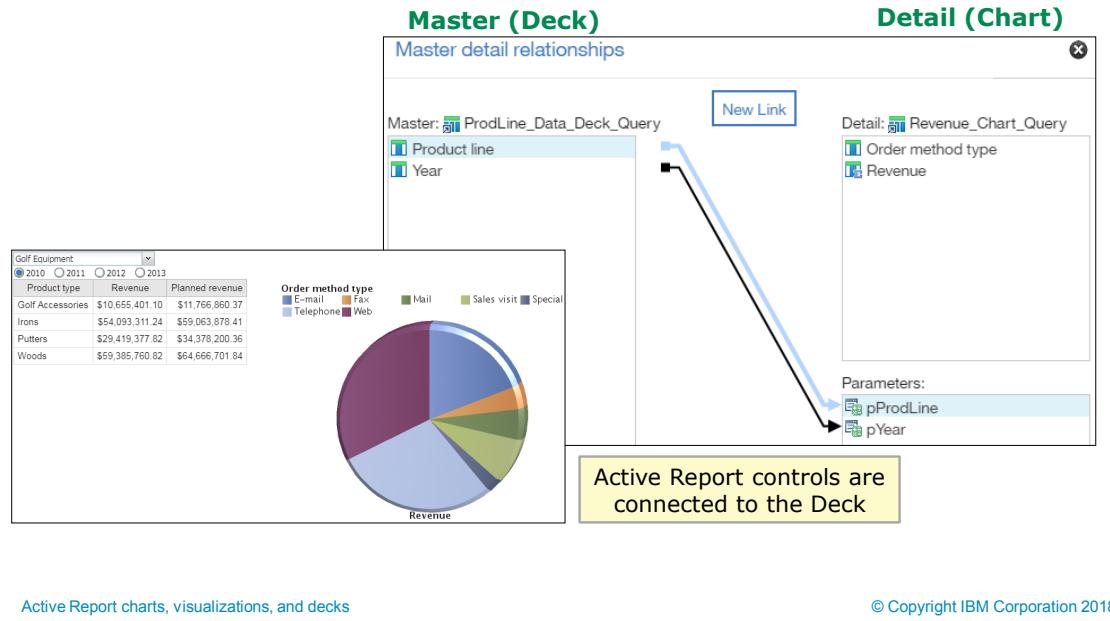
Traditional charts and Static decks

With a static Deck, the number of cards in the Deck is determined ahead of time by the report author. Any Deck card can contain a traditional chart, but in order for traditional charts on different cards to show differently-filtered data, each chart needs to be based on its own unique query with a filter applied to the query.

Charts based on the same query with the same data items will appear the same in the Active Report, regardless of the selected card.

Master detail relationships and decks

- Master detail relationships are used to section a data report object within a data deck.



Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Master detail relationships and decks

Master detail relationships can be used to define a relationship between a Deck and an object contained within the Deck. The purpose of the relationship is to define what data will be shown on the specific card within the Deck. The same technique can be used in a Tab control.

When working with traditional charts, the master detail relationship allows you to connect an Active Report variable to a parameter that is used in a filter on the chart for a particular card in the Deck.

Creating a Data deck with traditional charts

- Add the chart to the Data deck.
- Filter the chart based on a parameter.
 - e.g. [Year]=?pYear?
- Link the parent query to the chart using a master detail relationship.

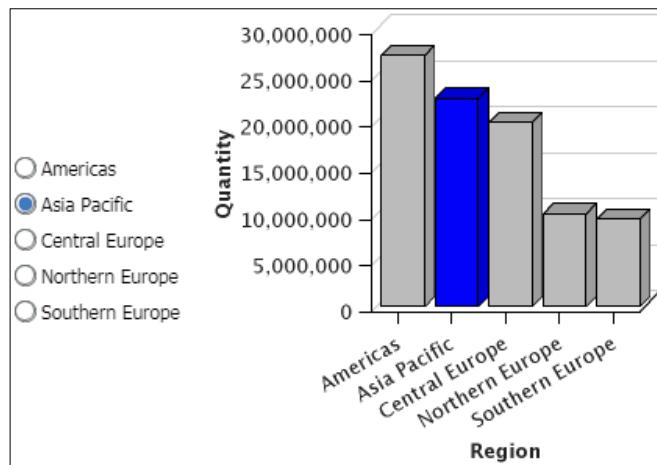
Creating a Data deck with traditional charts

To create a Data deck that contains different traditional charts for each selection, first add the chart to the Data deck, and then filter the chart using a parameter in the filter expression.

Once a parameter has been defined in the chart filter, define a master detail relationship between the parent query and the chart to link the Data deck's control data item to the parameter.

Demonstration 1

Create an Active Report using a Data deck



Demonstration 1: Create an Active Report using a Data deck

Demonstration 1: Create an Active Report using a Data deck

Purpose:

You have been asked to create a column chart showing quantity by region. The chart should be filtered based on a user's region selection. The report may or may not be used in an environment where the IBM Cognos servers are available, so an Active Report is needed. You will use a Data deck to create the various charts that can be displayed.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create an Active Report with a data radio button group control

1. Using the **GO sales (query)** package, create a new **Blank active report**.
2. In the work area, click **Add** , then click **Table** .
3. Click **OK** to accept the defaults (2 columns, 1 row).
4. Click the **Toolbox**  tab, then expand the **ACTIVE REPORT** section.
5. Drag a **Data radio button group**  to the left table cell:
 - Name: **Region_Data_RBG**
 - Query Name: **Region_Data_RBG_Query**
 - Click **OK**
6. From the **Data**  tab, drag Retailers > **Region** to the drop zone of **Region_Data_RBG**.
7. On the Application bar, click **Show properties**  to open the Properties pane.
8. In the Properties pane, under GENERAL, set **Orientation** to **Vertical**.

Task 2. Create a Data deck with a column Chart

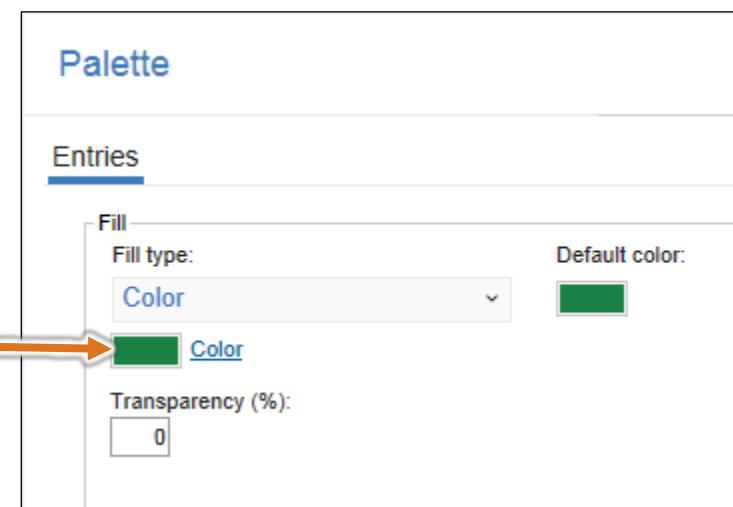
1. Click the **Toolbox**, then drag a **Data deck** to the right table cell.
 - Name: **Region_Data_Deck**
 - Query Name: **Region_Data_Deck_Query**
2. Click the **Data** tab, then drag Retailers > **Region** to the **Region_Data_Deck** object's **Deck Items** drop zone.
3. From the **Toolbox**, drag a **Visualization** to the **Region_Data_Deck** drop zone.
4. From the top right corner, click the visualization filter, then click **Charts**.

5. In the categories, click **Column**, then double-click **Clustered Column with 3D-Effects** .
6. In the Properties pane, change **Name** to **Region_Chart**.
7. Click the **Data** tab, then drag **Retailers > Region** to **Categories (x-axis)**.
8. Drag **Sales > Quantity** to **Default measure (y-axis)**.
9. In the Toolbar, click **More > Go to query**.
10. In the Properties pane, change **Name** to **Region_Chart_Query**.

Task 3. Add a conditional palette to the chart

You will highlight the bar of the selected region and gray out all the other region bars.

1. While the Chart is selected, in the Toolbar, click **More > Conditional Palette**.
2. Click **New Conditional Palette Entry** , and then click **New Advanced Condition**.
3. Click **New advanced condition** , and then in the **Expression Definition** pane, create the following expression:
 - `[Region_Data_Deck_Query].[Region] <> [Region_Chart_Query].[Region]`
4. Click **OK**.
5. Under **Palette Entry**, click **Edit Palette Entry** .
6. Under **Fill**, click the color box beside **Color**:



7. Click the **Basic colors** tab, click **Silver**, and then click **OK**.
8. Click **OK**.
9. Repeat steps 3 to 8, using the following:

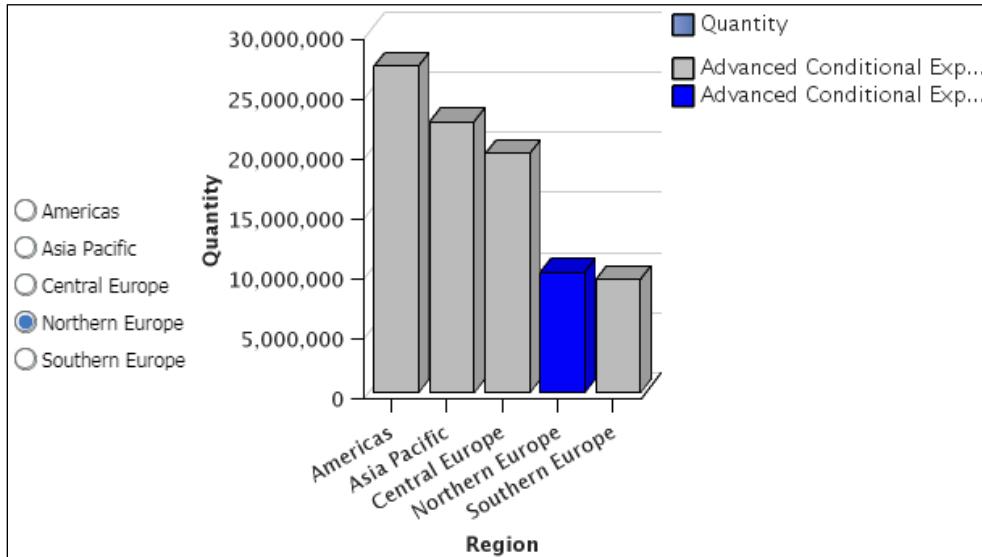
- Expression: **[Region_Data_Deck_Query].[Region] = [RegionChart_Query].[Region]**
 - Color: **Blue**
10. Click **OK**, and then click **OK** again to close the **Conditional palette** dialog.

Task 4. Create a connection between the Radio button group control and the Data deck

By creating this connection, you will be specifying which card to display when the user selects a region.

1. Click the **Region_Data_RBG** control, then in the Toolbar, click **Create a new connection**, and specify the following:
 - Region_Data_RBG (Region) > Select > Region_Data_Deck (Region)
 - New variable: **vRegion** (default value: **Americas**)
 - Click **Connect**.
2. Click the Chart, then in the Toolbar, click **Interactive Behavior**.
You want to specify that when a user clicks a column in the chart, the corresponding region will be selected in the radio button group.
3. Under **Behavior on select**, hover over **Set variable values**, and then click **Edit**.
4. Set **vRegion** to **Region** in the **Value** column, and then click **OK** twice.
5. On the Application bar, click **Run options** , then click **Run Active Report**.
6. Select **Northern Europe** from the **Region_Data_RBG** control.

The results appear as follows:



The report functions correctly. Hover the cursor over a grey bar and observe how it is highlighted.

7. Close the rendered report tab.

Task 5. Enhance the Active Report

The legend shows the conditional palette you created. This does not add value, so you will delete the legend.

1. Select the Chart.
2. In the Properties panel, under CHART ANNOTATIONS, double-click **Legend**.
3. Unselect **Show legend** and then click **OK**.

You will also add an animation effect to the report by specifying the slide direction for the card. To ensure the animation is rendered correctly, you will specify that the chart and the Deck are the same size.

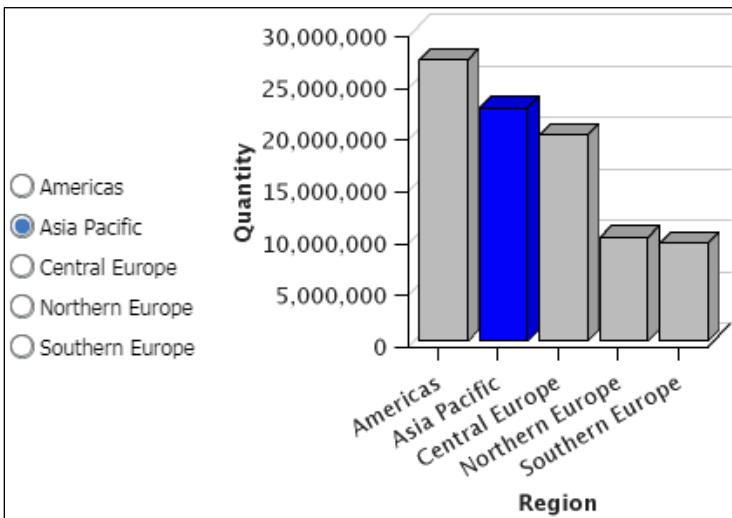
4. Select the **Region_Data_Deck** control, and then in the Properties pane, under General, Set **Slide Animation Direction** to **Auto Vertical**.
5. Select the left table cell, and then in the Properties pane, under POSITIONING, double-click **Size & overflow**, and set **Width** to **25px**.
6. Select the right table cell, and then in the Properties pane, under POSITIONING, double-click **Size & overflow**, and set **Height** and **Width** to **300px** each.

Hint: You may need to select the Data deck, and then in the Toolbar, click **Select Ancestor** to locate the table cell.

7. Select the chart, then in the Properties pane, under POSITIONING, double-click **Size & overflow** and set **Height** and **Width** to **300px** each.
8. On the Application bar, click **Run options** , then click **Run Active Report**.

9. Select **Asia Pacific** from the **Region_Data_RBG** control.

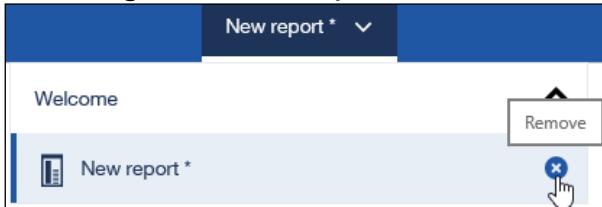
The results appear as follows:



The chart reflects the selection you made using the slide animation direction you specified.

10. Close the rendered reports' tabs.

11. In the Application bar, click the report dropdown menu, then click **Remove** to the right of New report to close it. Then click **OK** to confirm your action.



12. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You created an Active Report using a Data deck. You observed the hover property and defined a selection behavior on the chart.

Examine cards in a Data deck

- Cards are driven by the data values in the Data deck.
- The card displayed depends on the data item selection.

<input checked="" type="checkbox"/>	Camping Equipment	<input checked="" type="checkbox"/>	2010
<input checked="" type="checkbox"/>	Golf Equipment	<input checked="" type="checkbox"/>	2011
<input checked="" type="checkbox"/>	Mountaineering Equipment	<input checked="" type="checkbox"/>	2012
<input checked="" type="checkbox"/>	Outdoor Equipment	<input checked="" type="checkbox"/>	2013
<input checked="" type="checkbox"/>	Personal Accessories		

Product lines (5) * Years (4) = 20 cards

Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Examine cards in a Data deck

The number of cards in a Data deck is determined by the number of data values in the Deck's data items.

Care must be taken when designing Data decks, because the number of cards can increase exponentially as more potential data combinations are added.

Choose proper controls for use in Data decks

- Use single select controls.
- Impact of multi-select controls (e.g. Data check box group):
 - Number of combinations = 2^n
 - Complex logic is required to display the card

Values on multi-select control (n)	Number of cards generated on a deck (2^n)
1	2
2	4
3	8
4	16
5	32
6	64

Choose proper controls for use in Data decks

Single select controls, such as Drop-down lists, Radio button groups, and Button bars will result in a limited number of cards being generated on a Deck that is manageable to display.

With multi-select controls, such as a Check box group, the different combinations of data can result in a deck of 2 to the power of n of cards, where n is the number of values in the data item used in the data control.

The logic required to display the different combinations of card data is very difficult to create with the AND, OR, NOT operators.

Reduce report size with static decks

- Limit selections to a subset of available data.
- Unnecessary charts are not generated / stored.



Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Reduce report size with Static decks

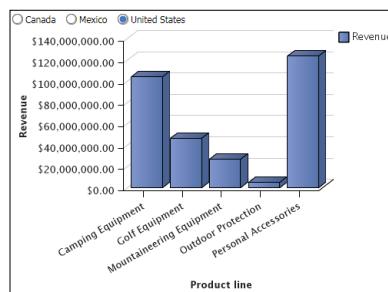
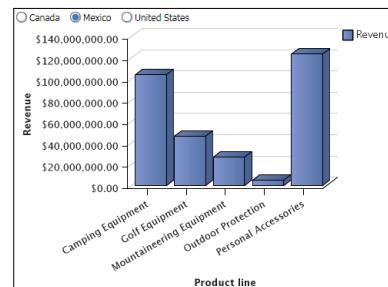
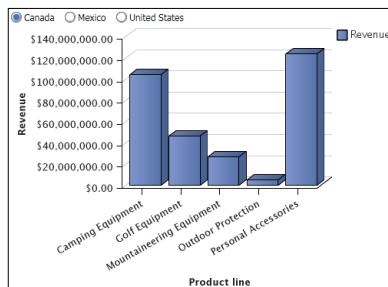
When dealing with a large number of data items, it may be worthwhile to use a Static deck instead of a Data deck, if there is a subset of the data that will not likely be required by the report consumers.

By controlling the number of cards using a static Deck, the report author has more control over how quickly the size of an Active Report grows, as only those data items needed by the users will generate cards for the report.

For example, a report intended only for consumers in a particular geographical region may not need to include all global regions in the Deck. A static Deck using only the regions required by the report consumers can dramatically reduce the overall size of the generated MHT file.

Demonstration 2

Create a Static deck to conserve report size



Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Demonstration 2: Create a Static deck to conserve report size

Demonstration 2:

Create a Static deck to conserve report size

Purpose:

You will create an Active Report for users located in Canada, Mexico, and the United States. You will include charts displaying data for each of these countries, but avoid bringing in other countries' data to optimize the size of the report.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

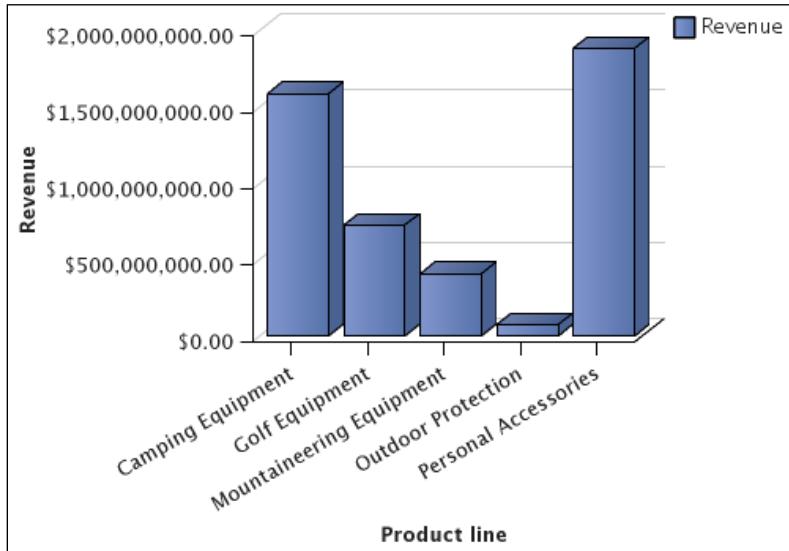
Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create a chart to show revenue by product line

1. From the **Welcome** page, navigate to **Team content** > **Samples** > **Models**.
2. Right-click the **GO Sales (query)** package, then click **Create report** .
3. In the Templates and themes dialog, double-click the **Blank active report** template to use it.
4. In the work area, click **Add** , then click **Table** .
5. In the **Insert table** dialog, set **Number of columns** to 1, **Number of rows** to 2, and then click **OK**.
6. In the bottom table cell, click **Add** , then click **Visualization** .
7. From the top right corner, click the visualization filter, then click **Charts**.
8. In the categories, click **Column**, then double-click **Clustered Column with 3D-Effects** .
9. On the Application bar, click **Show properties** to open the Properties pane.
10. In the Properties pane, under MISCELLANEOUS, set **Name** to **Revenue_Chart**.
11. In the side bar, click **Queries** , then right-click **Query1**, then click **Rename query**, then enter **Revenue_Chart_Query**.
12. Click the **Data** tab, expand **Sales (query) > Products**, and then drag **Product line to Categories (x-axis)**.
13. Expand **Sales**, and then drag **Revenue** to **Default measure (y-axis)**.
14. On the Application bar, click **Run options** , then click **Run Active Report**.

The results appear as follows:



The desired data is displayed, but we want to have separate charts to show data for Canada, Mexico, and the United States.

15. Close the browser tab.

Task 2. Add a Radio button group and Deck to select the country

1. From the **Toolbox** tab, the ACTIVE REPORT section, drag a **Radio button group** to the top table cell.
 2. In the Properties pane, under MISCELLANEOUS section, set **Name** to **Country_RBG**.
 3. It the Toolbar, click **Radio Buttons Definition** . And create these labels:
 - Canada
 - Mexico
 - United States
 4. Click **OK**.
- We are unable to filter a chart object, so we will need a Deck to hold a chart for each country.
5. From the Toolbox, drag a **Deck** to the right of the **Revenue_Chart** control.
 6. In the Properties pane, under MISCELLANEOUS, specify the **Name** property as **Country_Deck**.
 7. Click **Deck Cards Definition**.
 8. It the Toolbar, create three cards as follows:
 - Canada

- Mexico
 - United States
9. Click **OK**.
 10. Click the **Country_RBG**, then in the Toolbar, click **Create a new connection**.
 11. Under **Active Report Variable**, set the **Name** to **vCountry**.
 12. Click **Set Default Value(s)**, and set the default value to **Canada**.
 13. Click **OK**.
 14. Click **Connect**.

Task 3. Add the chart to the Deck, and create copies for each country

1. Drag **Revenue_Chart** into the drop zone of **Country_Deck**.
2. Select **Revenue_Chart**, then in the Properties pane, under MISCELLANEOUS, change **Name** to **Canada_Chart**.
Next, you will put a copy of the chart on each card in the Deck.
3. On the Toolbar, click **More** , and then click **Copy**.
4. From the drop-down list of **Country_Deck**, select **Mexico**.
5. Click the **Country_Deck** control's drop zone to select it, then press **Ctrl+V**.
6. In the Properties pane, under miscellaneoUS section, set **Name** to **Mexico_Chart**.
7. Repeat steps **4** through **6** to paste the Chart into the **United States** card.
 - Name: **UnitedStates_Chart**
8. On the Application bar, click **Run options** , then click **Run Active Report**.
9. Select each of the countries in the **Country_RBG** control.
Changing the selected country does not affect the appearance of the chart. This is because the charts are all based on the same query, and you are unable to filter charts in Active Reports.
You will define queries to filter out data that does not relate to the selected country.

9. Close the rendered report tab.

Task 4. Filter the query to collect data for only one country

1. On the side bar, click **Queries** , and then click **Revenue_Chart_Query**.
2. On the side bar, click **Data**.
3. Drag Retailers > **Retailer country** into the **Detail Filters** pane.
4. In the **Expression Definition** pane, edit the expression to be:

- [Sales (query)].[Retailers].[Retailer country]='Canada'
5. Click **Validate**, and then click **OK**.
 6. On the side panel, click **Queries**, then right-click **Revenue_Chart_Query**.
 7. Click **Rename query**.
 8. Type **Canada_Query**, and then click **OK**.

Task 5. Create separate queries for each country

1. In the Queries pane, click the **Queries**  folder.
2. Right-click **Canada_Query**, and select **Copy**.
3. Right-click the background of the Query explorer, then click **Paste**.
4. Right-click **Canada_Query1** and click **Rename query**.
5. Type **Mexico_Query**, and then click **OK**.
6. Repeat steps 2 to 5 to create a query named **UnitedStates_Query**.
7. Double-click **Mexico_Query**.
8. Double-click the expression in the **Detail Filters** pane.
9. In the **Expression Definition** pane, replace **Canada** with **Mexico**. The expression should now be:
 - [Sales (query)].[Retailers].[Retailer country]='Mexico'
10. Click **Validate**, and then click **OK**.
11. In the **Queries** tab, click **UnitedStates_Query**.
12. Double-click the expression in the **Detail Filters** pane.
13. In the **Expression Definition** pane, replace **Canada** with **United States**. The expression should now be:
 - [Sales (query)].[Retailers].[Retailer country]='United States'
14. Click **Validate**, and then click **OK**.

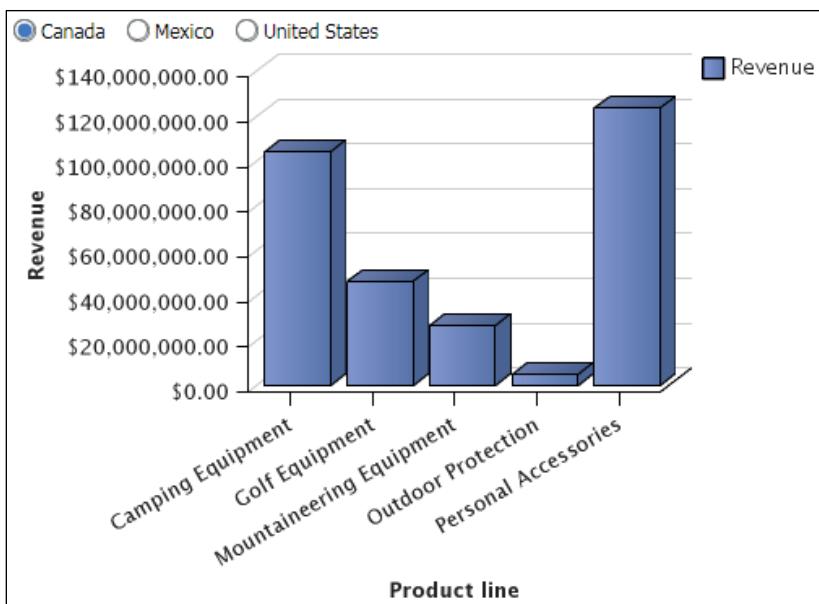
Task 6. Assign the appropriate query to each of the charts

1. In the side bar, click **Pages** , then under **Report pages**, click **Page1**.
2. From the **Country_Deck** control's drop-down list, select **Canada**.
3. Click the **Canada_Chart** control, and then in the Properties pane, under DATA, verify that the **Query** property is set to **Canada_Query**.
4. From the **Country_Deck** control's drop-down list, select **Mexico**.
5. Click the **Mexico_Chart** control, and then in the Properties pane, under DATA, set the **Query** property to **Mexico_Query**.
6. From the **Country_Deck** control's drop-down list, select **United States**.

- Click the **UnitedStates_Chart** control, and then in the Properties pane, under DATA, set the **Query** property to **UnitedStates_Query**.

Task 7. Run the report and verify that the country-specific charts display correctly

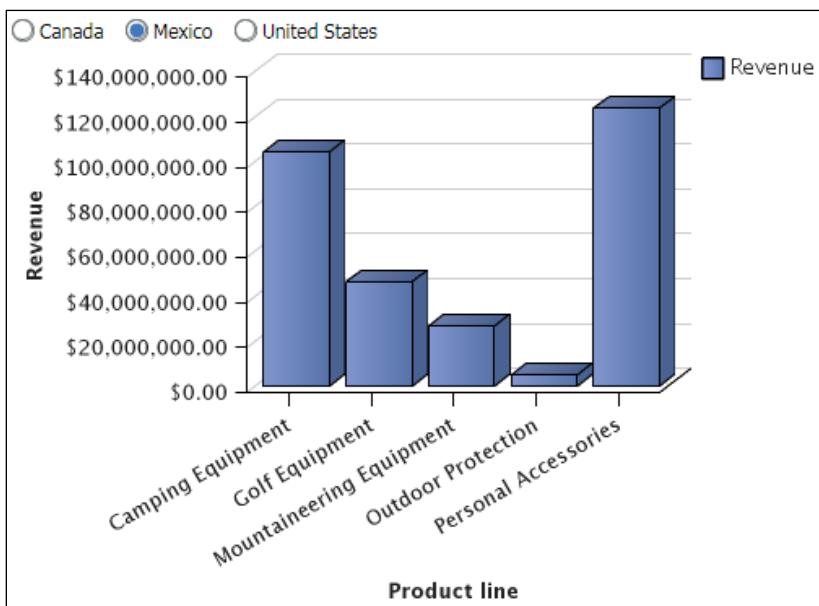
- On the Application bar, click **Run options** , then click **Run Active Report**. The results appear as follows:



The chart displays data for revenue from Canada.

- Select **Mexico**.

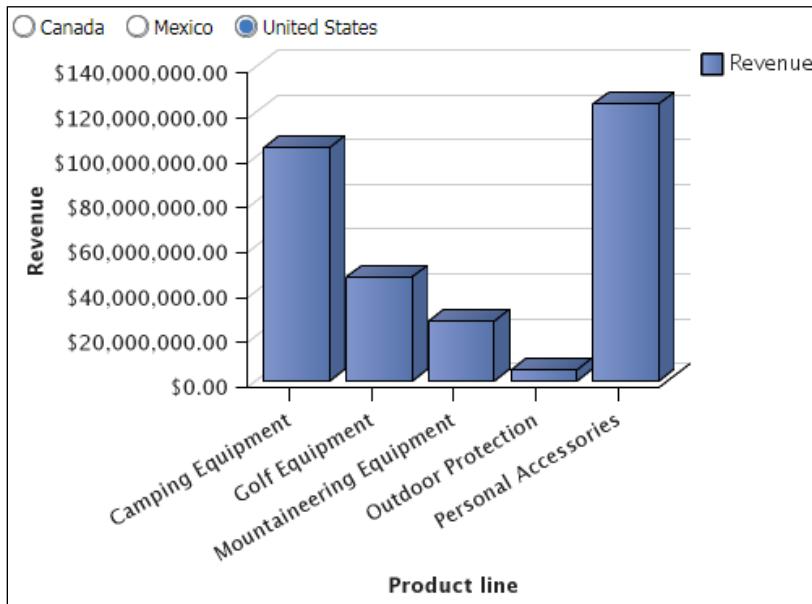
The results appear as follows:



The chart displays data for revenue from Mexico.

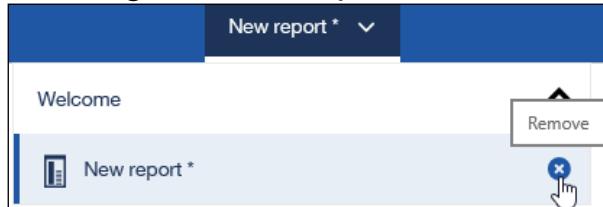
- Select **United States**.

The results appear as follows:



The chart displays data for revenue from the United States.

4. Close the rendered report tab.
5. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



6. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You used a Deck with separate queries to create an optimized report to display chart data for a subset of the available countries in your database without storing data that is not related to the available country selections.

Demonstration 3

Optimize a Data deck report for size

The screenshot shows a Data Deck report interface. On the left, there is a sidebar with dropdown menus for 'Australia' and 'Global Product Info'. Below these are two tables: one for '1 for 1 Sports shop' and another for 'Australia Revenue Data by Product Line'.

Global Product Info

Product line	Revenue
Camping Equipment	\$1,331,569.87
Mountaineering Equipment	\$792,886.58
Outdoor Protection	\$145,302.07
Personal Accessories	\$4,162,491.80
Overall - Total	6,432,250.32

Australia Revenue Data by Product Line

Revenue	2011	2012	2013
Camping Equipment	\$9,752,591.01	\$19,175,957.20	\$13,007,383.98
Golf Equipment	\$4,094,643.54	\$8,482,438.67	\$6,502,474.22
Mountaineering Equipment	\$2,691,279.15	\$5,861,253.12	\$5,380,587.79
Outdoor Protection	\$600,956.77	\$367,636.38	\$171,750.41
Personal Accessories	\$8,467,573.03	\$13,912,451.77	\$10,830,992.10

Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Demonstration 3: Optimize a Data deck report for size

Demonstration 3: Optimize a Data deck report for size

Purpose:

You have been asked to create a dashboard for mobile users containing both country-specific and generic revenue information. You will examine the impact on the size based on how you arrange controls and content in your report.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create a Data deck and Data drop-down list to display country-specific information

1. Using the **GO sales (query)** package, create a new **Blank active report**.
2. Click the **Toolbox**, then from under the ACTIVE REPORT section, drag a **Data drop-down list** to the page body.
 - Name: **Country_Data_DDL**
 - Query Name: **Country_Data_DDL_Query**
3. From the **Data**  tab, drag Retailers > **Retailer country** onto **Country_Data_DDL**.
4. From the **Toolbox** tab, drag a **Data deck** under the Data drop-down list:
 - Name: **Country_Data_Deck**
 - Query Name: **Country_Data_Deck_Query**
5. From the **Data**  tab, drag Retailers > **Retailer country** to the Deck Items drop zone of **Country_Data_Deck**.

Task 2. Create a connection between the Data drop-down list and the Data deck

1. Click the **Country_Data_DDL**, then in the Toolbar, click **Create a new connection**, and specify the following:
 - Country_Data_DDL (Retailer country) > Select > Country_Data_Deck (Retailer country)
 - New variable: **vCountry**
 - Default value: **Australia**

2. Click Connect.

Task 3. Add global information to the Deck card

1. From the **Toolbox** tab, drag a **Table** to the **Country_Data_Deck** drop zone.
2. Accept the default settings by clicking **OK**.
3. From the **Toolbox** tab, drag a **Text item** to the left table cell.
4. Type **Global Product Info**, and then click **OK**.
5. On the Toolbar, click **Font A**.
6. Set **Size** to **14pt**, **Weight** to **Bold**, **Foreground Color** to **Blue**, then click **OK**.
7. From the **Toolbox** tab, drag a **Data drop-down list** after the Text item:
 - Name: **Company_Data_DDL**
 - Query Name: **Company_Data_DDL_Query**
8. From the **Data**  tab, drag Retailers > **Company name** onto **Company_Data_DDL**.
9. From the **Toolbox** panel, drag a **List** after the **Company_Data_DDL** control:
 - Name: **Company_Revenue_List**
 - Query Name: **Company_Revenue_List_Query**
10. From the **Data** tab, drag the following query items to **Company_Revenue_List**:
 - **Products > Product line**
 - **Sales > Revenue**
11. Click **<Revenue>**, then from the Toolbar, click **Summarize** > Summarize in Output > **Total**.

The result appears as follows:



The screenshot shows a 'Deck card' interface. At the top, there is a header cell containing the text '<Retailer country>'. Below it is a table with a single row labeled 'Global Product Info'. This row contains a cell with the text '<Company name>' and another cell with a plus sign (+) inside a circle, indicating a dropdown menu. The main body of the card contains a table with the following data:

Product line	Revenue
<Product line>	<Revenue>
<Product line>	<Revenue>
<Product line>	<Revenue>
Overall - Total	<Total(Revenue)>

Task 4. Create a connection between the Data Drop-Down

List control and the list

1. Click the **Company_Data_DDL** control, then from the Toolbar, click **Create a new connection**, and specify the following:
 - Company_Data_DDL (Company name) > **Filter** > **Company_Revenue_List** (Copy Data Item from source)
 - New variable: **vCompany**
 - Default value: **1 for 1 Sports shop**
2. Click **Connect**.

Task 5. Add country-specific information to the Deck card

3. From the **Toolbox** tab, under the ACTIVE REPORT section, drag a **Variable text item** to the right table cell.
4. Select **vCountry**, and then click **OK**.
5. From the **Toolbox** tab, drag a **Text Item** to the right **<%vCountry%>**.
6. Type a space, type **Revenue Data by Product Line**, and then click **OK**.
7. Click the **Global Product Info** text item, and then on the Toolbar, click **Pick up style** .
8. Select the **<%vCountry%>** variable text item, then on the toolbar, click **Apply style** .
9. Select the **Revenue Data by Product Line** text item, then on the toolbar, click **Apply Style**.
10. From the **Toolbox**, drag a **Crosstab** to the right table cell under the Text item:
 - Name: **Country_Revenue_Crosstab**
 - Query Name: **Country_Rev_Crosstab_Query**
11. From the **Data**  tab, drag the following query items to the Crosstab:
 - Rows: Products > **Product line**
 - Columns: Time > **Year**
 - Measures: Sales > **Revenue**

Task 6. Filter the crosstab by a country parameter

1. Select the **Country_Revenue_Crosstab** control.
2. On the Toolbar, click **Filters** > **Edit Filters**.
3. Click **Add**, then click **Advanced**, and then click **OK**.

4. Under **Available Components**, expand **Sales (query)** and **Retailers**, and then drag **Retailer country** to the **Expression Definition** pane.
5. Edit the expression to the following:
 - **[Sales (query)].[Retailers].[Retailer country]=?pCountry?**
6. Click **OK**.
7. Click **OK**.

Task 7. Create a Master detail relationship between the country variable and the crosstab

1. Select the **Country_Revenue_Crosstab**.
2. On the Toolbar, click **More** > **Master Detail Relationships**.
3. Click **New Link**.
4. Click **pCountry** under **Parameters** to define the link, then click **OK**.

Task 8. Save the report, run it, and then save an MHT file to check the file size

1. On the Application bar, click **Save**.
 2. Save the report as **Dashboard_1** in **My content**.
 3. On the Application bar, click **Run options** , then click **Run Active Report**. Note that it may take a while for the report to display.
- The results appear as follows:

Global Product Info		Australia Revenue Data by Product Line			
Product line	Revenue	Revenue	2011	2012	2013
Camping Equipment	\$1,331,569.87	Camping Equipment	\$9,752,591.01	\$19,175,957.20	\$13,007,383.98
Mountaineering Equipment	\$792,886.58	Golf Equipment	\$4,094,643.54	\$8,482,438.67	\$6,502,474.22
Outdoor Protection	\$145,302.07	Mountaineering Equipment	\$2,691,279.15	\$5,861,253.12	\$5,380,587.79
Personal Accessories	\$4,162,491.80	Outdoor Protection	\$600,956.77	\$367,636.38	\$171,750.41
Overall - Summary	\$4,686,775,768.85	Personal Accessories	\$8,467,573.03	\$13,912,451.77	\$10,830,992.10

4. Close the browser tab.
5. On the Application bar, click **Run options** , then click **Run Active Report**.
6. From the Application bar click **Run options**, then click **Download Active Report**.
7. At the prompt to save **Dashboard_1.mht**, click the down arrow beside **Save**, and click **Save as**.
8. Save the report as **C:\Edcognos\B6098\Dashboard_1.mht**.
9. Click **Open folder**, and then take note of the size of **Dashboard_1.mht**.

The size of the file is a little large for a small dataset, and might not scale well.

You will now change the report to reduce the size of the MHT file.

10. Close **Windows Explorer**, and then close the browser tab.

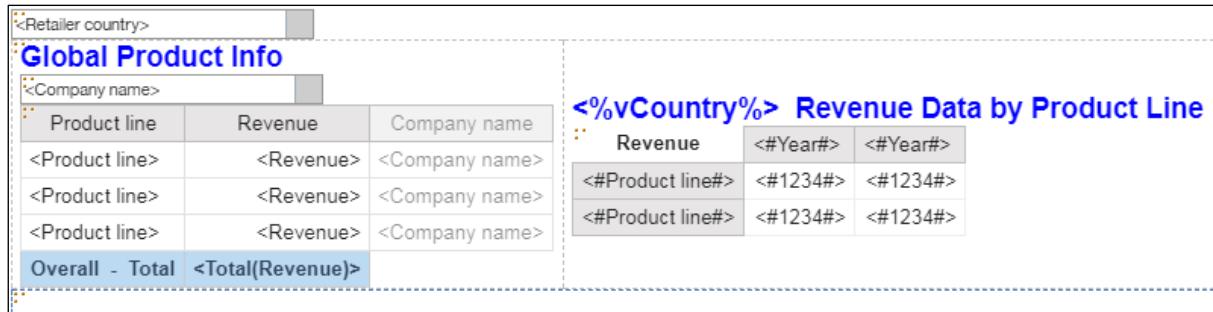
Task 9. Save a new copy of the report, and then move controls that are not related to country out of the Data deck

Because many of the controls in this report do not change as the country changes, we can move these controls out of the Data deck to avoid duplicating these controls for each card in the Data deck.

1. On the Application bar, click the down arrow beside **Save**  , and then click **Save as**.
2. Save the report as **Dashboard_2** in **My content**.
3. Select the Table, then drag it after the **Country_Data_DDL**, before the **Country_Data_Deck**.

Hint: To select the Table, you may need to click the **Company_Revenue_List**, then on the Toolbar, click **Select ancestor**  , then click **Table**.

The result appears as follows:



4. In the Properties pane, under **BOX**, double-click **Margin**.
5. For the **Top margin**, type **10**, then click **OK**.
6. Click **<%vCountry%>**, Ctrl+click the **Revenue Data by Product Line** Text item beside it, and Ctrl+click the **Country_Revenue_Crosstab**.
7. Drag the selection onto the **Country_Data_Deck**.
8. Click **Select Country_Data_Deck**  , then drag it to the right cell of the Table.

The result appears as follows:

The screenshot shows a report titled "Global Product Info" with a table of revenue data. The table has columns for "Product line", "Revenue", and "Company name". It includes rows for "Overall - Total" and "Total(Revenue)". To the right of the main table is a secondary table titled "<%vCountry%> Revenue Data by Product Line", which also displays revenue data for different product lines and years.

Task 10. Save the report, run it, and then save an MHT file to check the file size

1. On the Application bar, click **Save**.
2. On the Application bar, click **Run options** , then click **Run Active Report**.

The results appear as follows:

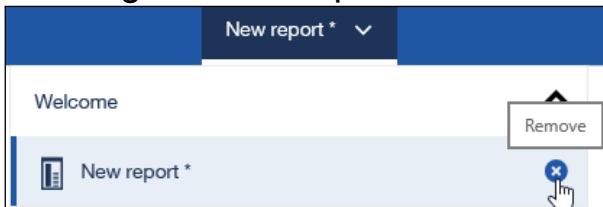
The screenshot shows the "Global Product Info" report for "Australia". It includes a dropdown menu for "Australia" and a table of revenue data for "1 for 1 Sports shop". To the right is a separate table titled "Australia Revenue Data by Product Line", which provides detailed revenue data for various product categories over three years (2011, 2012, 2013).

Product line	Revenue	2011	2012	2013
Camping Equipment	\$1,331,569.87	\$9,752,591.01	\$19,175,957.20	\$13,007,383.98
Mountaineering Equipment	\$792,886.58	\$4,094,643.54	\$8,482,438.67	\$6,502,474.22
Outdoor Protection	\$145,302.07	\$2,691,279.15	\$5,861,253.12	\$5,380,587.79
Personal Accessories	\$4,162,491.80	\$600,956.77	\$367,636.38	\$171,750.41
Overall - Total	6,432,250.32	\$8,467,573.03	\$13,912,451.77	\$10,830,992.10

The results appear the same as when the report was run previously. Next, you will generate an MHT file and compare the new file's size to the old one.

3. Close the browser tab.
 4. On the Application bar, click **Run options**, then click **Download Active Report**.
 5. At the prompt to save **Dashboard_2.mht**, click the down arrow beside **Save**, and click **Save as**.
 6. Save the report as **C:\Edcognos\B6098\Dashboard_2.mht**.
 7. Click **Open folder**, and then compare the size of **Dashboard_2.mht** to the size of **Dashboard_1.mht**.
- The new MHT file is significantly smaller, which will make it faster to download and display on a mobile device.
8. Close **Windows Explorer**.
 9. Close the rendered report tab.

10. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



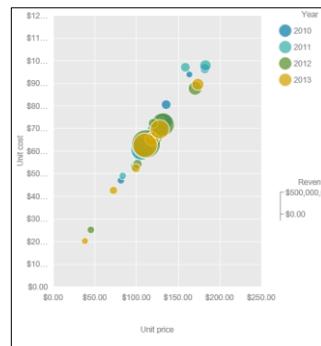
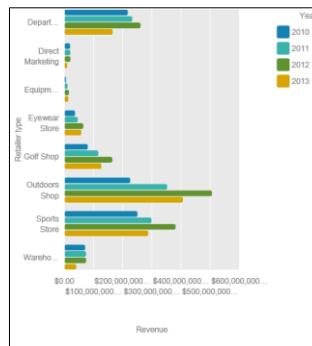
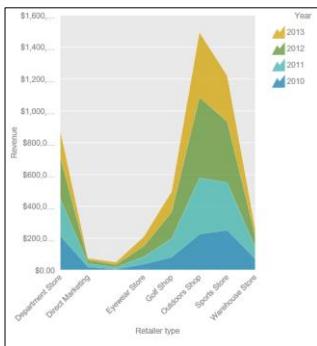
11. Leave the IBM Cognos Analytics portal open for the next demonstration.

Results:

You created a basic dashboard for mobile users, and explored changes that can optimize the size of the resulting MHT file.

Rapidly Adaptive Visualization Engine (RAVE)

- Interactive
- Animated
- Variety of chart types
- Customizable
- Import additional visualizations



Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Rapidly Adaptive Visualization Engine (RAVE)

To get more flexibility when designing Active Reports, report authors can incorporate RAVE visualizations into their reports to complement or enhance views of data already provided by standard chart controls.

MHT files generated for Active Reports contain an engine to generate RAVE visualizations, so visualizations in Active Reports can be rendered on the client side dynamically based on user selections, or they can be rendered on the server as a static image when the MHT file is first created.

RAVE visualizations can also be animated to dynamically highlight aspects of the data, in order to provide a more engaging experience to report consumers.

Traditional charts vs. RAVE visualizations

- Traditional charts:
 - stored as images (static)
 - deck or query used to filter content
- RAVE visualizations:
 - extensible
 - rendered on client or server (dynamic or static)
 - can be filtered
 - graphical animations

Traditional charts vs. RAVE visualizations

Both traditional charts and RAVE visualizations can be used in Active Reports, and each has its own characteristics and advantages.

Traditional charts are generated at the time the Active Report is first generated, so subsequent runs of the MHT will be faster.

By using a Deck, or assigning unique queries to different charts, content displayed to the user can be customized, even though traditional filters can be used.

RAVE visualizations can be customized and imported to the IBM Cognos Analytics server, allowing access to a dynamically growing library of visualizations.

RAVE visualizations can be generated on the client side in the reprot, allowing content to be dynamically updated for more interactivity. Alternately, visualizations can be rendered on the server side, creating a static image, similar to traditional charts.

Content in a RAVE visualization can be dynamically filtered at the MHT file's run time, and can also include graphical animations to enhance the report user's experience.

Connections and RAVE visualizations

- Active Report connections support RAVE visualization controls:
 - control -> filter/select -> visualization
 - visualization -> filter/select -> control
 - visualization -> filter/select -> visualization

Connections and RAVE visualizations

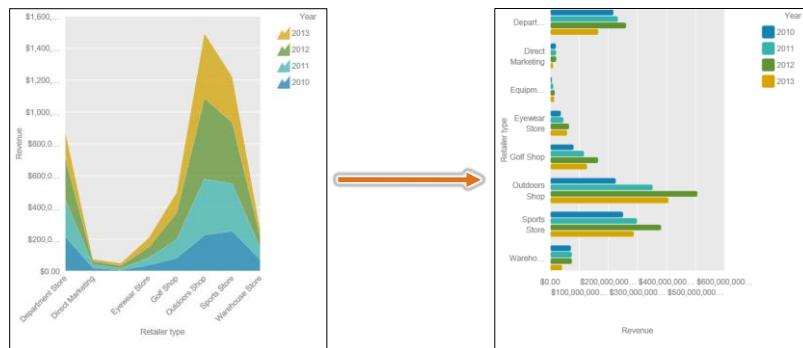
Because RAVE visualizations are also Active Report controls, their behavior can be controlled by creating a connection from an Active Report control to filter or select the Visualization control.

It is also possible for a connection to be set up from a RAVE visualization to another control, or even a second RAVE visualization.

For example, a report could be created where clicking on a bar in a RAVE bar chart visualization highlights the rows in a List object that contribute to the bar chart's data.

Converting visualizations in a report

- Convert between different types of visualizations.
- Data from the old visualization is mapped to the new visualization.



Converting visualizations in a report

As a report author works with visualizations in a report, there are times when a different visualization might provide better insights than the one currently in the report, using the same or similar underlying data.

By clicking a visualization in the report and using the On-demand toolbar, the report author can choose to convert the visualization type to another visualization in the library.

When IBM Cognos Analytics converts the visualization, it also attempts to map the data from the original visualization to data slots in the new visualization, so that the author does not need to add the same data again. The report author can then move data around or add more data to fine-tune the appearance of the new visualization.

Rendering RAVE visualizations

- Client-side
 - visualization rendered at run-time
 - full interactivity
 - slightly smaller file size
- Server-side
 - visualization rendered at file generation time
 - static images
 - slightly better performance

Rendering RAVE visualizations

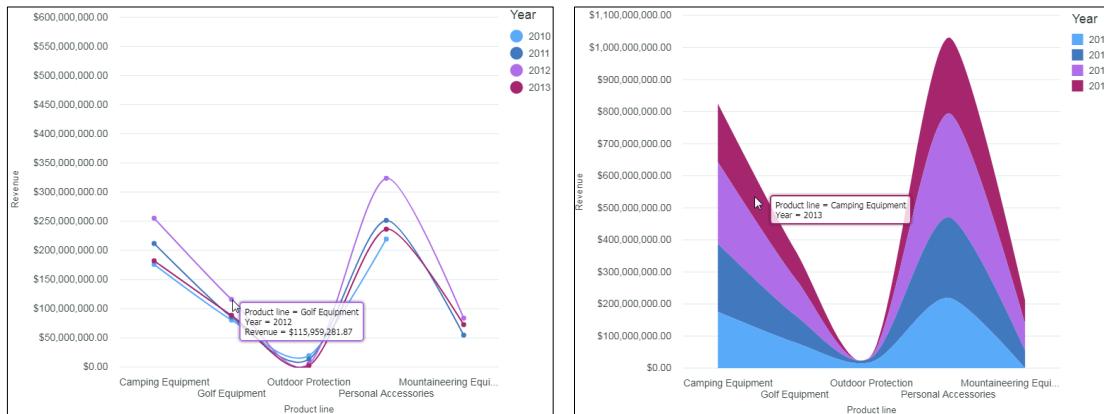
There are two options available for rendering visualizations when using RAVE visualizations in an Active Report, depending on the requirements of the report consumer.

If a RAVE visualization is set to use client-side rendering, then code in the report will dynamically render the visualization in the browser at run-time. This allows for full interactivity with the visualization, and can result in a slightly smaller file size.

If a RAVE visualization is set to use server-side rendering, then the visualization is rendered at the server when the report is first created, using static images, like traditional charts. Since code is not executed in this case, there may be a slight increase in performance.

Demonstration 4

Create an Active Report using RAVE visualizations



Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Demonstration 4: Create an Active Report using RAVE visualizations

Demonstration 4:

Create an Active Report using RAVE visualizations

Purpose:

You want to create an interactive Active Report using animated charts from RAVE visualizations. You will create a report displaying data using different visualizations to better engage the consumer.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO sales (query)

Namespace: Sales (query)

Task 1. Create an Active Report with a List

1. Using the **GO sales (query)** package, create a new **Blank active report**.
2. From the **Toolbox**, expand **ACTIVE REPORT**, then drag a **Data toggle button bar** object to the top table cell.
 - Name: **Region_Data_TBB**
 - Query Name: **Region_Data_TBB_Query**
3. From the **Data** tab, drag Retailers > Region to the **Region_Data_TBB** drop zone.
4. On the Application bar, click **Show properties**  to open the Properties pane.
5. In the Properties pane, under GENERAL, double-click **Button gap**, enter 5 and click **OK**.
6. From the **Toolbox** tab, drag a **Table** under the Data button bar, and click **OK**.
7. In the left Table cell, click **Add** , then click **List** 
 - Name: **Revenue_List**
 - Query Name: **Revenue_List_Query**
8. From the **Data** tab, drag the following to the List:
 - **Sales (query) > Retailers: Region, Retailer country**
 - **Sales: Revenue and Quantity**.
9. Click **<Revenue>**, and Ctrl-click **<Quantity>**, then from the Toolbar, click **Summarize > Default summary > Total**.

The result appears as follows:

<Region>	<Region>	<Region>	
Region	Retailer country	Revenue	Quantity
<Region>	<Retailer country>	<Revenue>	<Quantity>
<Region>	<Retailer country>	<Revenue>	<Quantity>
<Region>	<Retailer country>	<Revenue>	<Quantity>
Overall - Total		<Total(Revenue)>	<Total(Quantity)>



Task 2. Connect the Data toggle bar to the List

- Click the **Region_Data_TBB**, then from the Toolbar, click **Add a new connection**, and specify the following:
 - Region_Data_TBB (Region) > Select > Revenue_List (Region)
 - Create a new variable: **vRegion**
 - Click **Connect**.

The vRegion variable will be used later to filter the data displayed in the RAVE visualization, but here you want to continue to display all data in the List, but highlight the data in the List that is displayed in the RAVE visualizations.

- From the Application bar, click **Run options** , then click **Run Active Report**.
- Select **Americas** from the **Region_Data_TBB** control.

The result appears as follows:

Americas	Asia Pacific	Central Europe	Northern Europe	Southern Europe
Region	Retailer country	Revenue	Quantity	
Asia Pacific	Australia	\$109,299,969.14	2,000,781	
Southern Europe	Austria	\$143,709,405.32	2,742,824	
Central Europe	Belgium	\$113,031,617.63	2,124,791	
Americas	Brazil	\$138,276,679.05	2,591,989	
Americas	Canada	\$306,159,341.39	5,722,733	
Asia Pacific	China	\$317,244,910.76	6,110,945	
Northern Europe	Denmark	\$62,013,023.51	1,301,136	
Northern Europe	Finland	\$188,575,310.13	3,603,492	
Central Europe	France	\$286,569,502.15	5,529,613	
Central Europe	Germany	\$262,313,028.79	5,084,611	
Southern Europe	Italy	\$186,648,098.68	3,545,695	
Asia Pacific	Japan	\$357,446,606.95	6,787,127	
Asia Pacific	Korea	\$200,725,307.09	3,902,092	
Americas	Mexico	\$167,187,013.22	3,175,752	
Northern Europe	Netherlands	\$184,321,676.63	3,448,760	
Asia Pacific	Singapore	\$197,622,385.04	3,788,595	
Southern Europe	Spain	\$165,066,464.22	3,171,715	
Northern Europe	Sweden	\$95,411,435.54	1,681,811	
Central Europe	Switzerland	\$100,731,875.51	1,822,191	
Central Europe	United Kingdom	\$277,509,546.56	5,378,361	
Americas	United States	\$826,912,571.54	15,722,077	
Overall - Total		\$4,686,775,768.85	89,237,091	

- Close the rendered reports' tabs.

Task 3. Add a RAVE Visualization to the report.

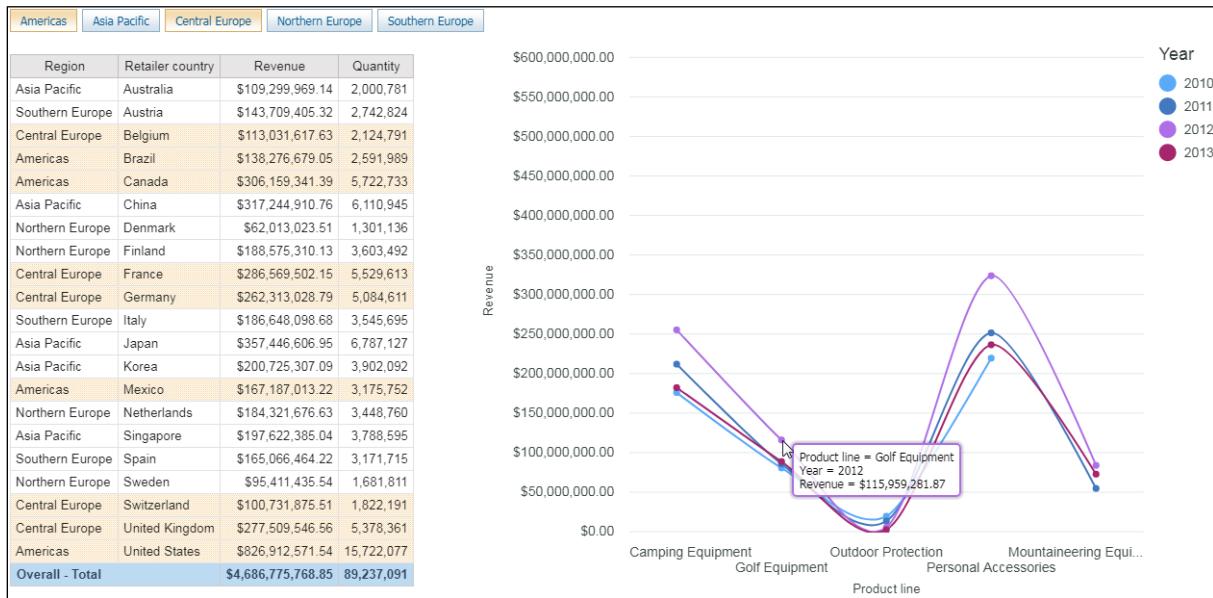
1. In the right table cell, click **Add** , then click **Visualization** .
2. From the top right corner, click the visualization filter, then click **New visualizations**.
3. In the categories on the left, click **Line**, then double-click **Smooth line**.
 - Visualization name: **Revenue_Visualization**
 - Data container name: **Revenue_Visualization_Data**
 - Query Name: **Revenue_Visualization_Query**

Note that we do not need to put the visualization inside a Deck, because the RAVE visualization engine dynamically updates the visualization content, rather than storing a static copy of all possible visuals.
4. From the **Data** tab, drag the following items to the specified drop zones:
 - Values > Value: Sales > **Revenue**
 - Categories > X axis: Products > **Product line**
 - Color: Time > **Year**
5. Click the Visualization, then in the Properties pane, under COMMON, set **Width** to **1000**, and Height to **600**.

Task 4. Connect the Data toggle button bar to the Visualization

1. Select the **Region_Data_TBB**, then in the Toolbar, click **Create a new connection**, and specify the following:
 - Region_Data_TBB (Region) > **Filter** > **Revenue_Visualization** (Region)
 - Reuse an existing variable: vRegion
 - Click **Connect**.
2. On the Application bar, click **Run options** , then click **Run Active Report**.
3. Select **Americas** and **Central Europe** from the **Region_Data_TBB** control.

The result appears as follows:

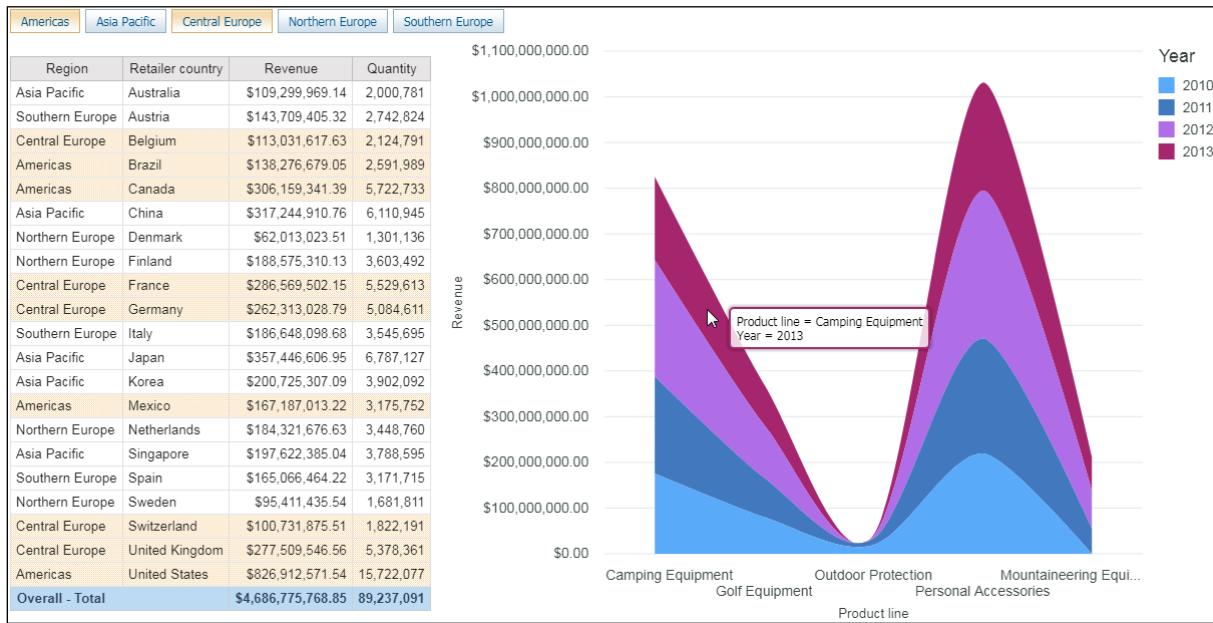


- Close the rendered report tab.

Task 5. Convert the line Visualization to an area Visualization

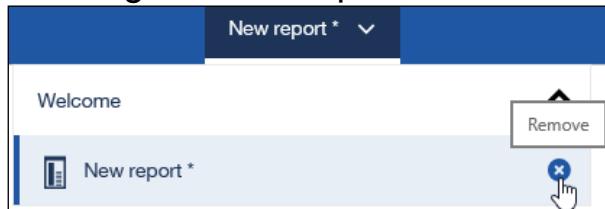
- Click the **Revenue_Visualization**, then on the Toolbar, click **More ... > Convert visualization**.
- In the categories list on the left, click **Area**, then double-click **Smooth area**. IBM Cognos Analytics populates the new visualization based on the data that was in the old visualization, so Revenue is added to the Values zone, Product line is added to the X categories zone, Region is added to the Extra Categories zone, and Year is added to the Color zone.
- In the **Properties** pane, under CLIENT BEHAVIOR, set **Animation** to **Transition**.
- On the Application bar, click **Run options** , then click **Run Active Report**.
- Select **Americas** and **Central Europe** from the **Region_Data_TBB** control.

The results appear as follows:



As you change selections, the colored sections of the area chart resize to reflect the new filtered data.

6. Close the rendered reports' tabs.
7. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



8. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You created a dynamic Active Report using RAVE visualizations to present an interactive experience for report users, and converted a visualization to provide a different graphical experience for the user.

Demonstration 5

Create a professional-looking Active Report (Optional)



Active Report charts, visualizations, and decks

© Copyright IBM Corporation 2018

Demonstration 5: Create a professional-looking Active Report (Optional)

Demonstration 5: Create a professional-looking Active Report (Optional)

Purpose:

You have a request to create a report that will be used by executives to present to their sales teams. The report will contain key metric information on Revenue, Planned revenue, Quantity and Gross profit for Australian sales in 2012. The report should be visually appealing.

Portal: <http://vclassbase/ibmcognos>

User/Password: brettonf/Education1

Package: GO data warehouse (query)

Namespace: Sales (query)

Note that this demonstration is fairly advanced, and time-consuming. It should be used only once the other material in this course have been completed, and can be safely skipped if time is short.

Task 1. Create and format an Active Report

1. Using the **GO data warehouse (query)** package, create a new **Blank active report**.
 Note that this is a different package from the one used in other demonstrations in this course.
2. In the page body, click **Add** , then click **Block** .
3. On the Application bar, click **Show properties**  to open the Properties pane.
4. In the Properties pane, under BOX, double-click **Padding**.
5. Set **Top padding** to **10px**, and **Bottom padding** to **10px**, then click **OK**.
6. In the Properties pane, under BOX, double-click **Margin**.
7. Set **Bottom margin** to **15px**.
8. Click **OK**.
9. In the **Properties** pane, under COLOR & BACKGROUND, double-click **Background effects**.
10. Select the **Fill** check box.
11. Under **Colors**, click the color box, and then click **Color**.
12. In the Basic Colors tab, click **White**, and then click **OK**.
13. Click **New** , and then click **Color**.
14. In the **Basic Colors** tab, double-click **Teal**.
15. Under **Angle**, type **270**, and then click **OK**.
16. In the Properties pane, under POSITIONING, double-click **Size & overflow**.

17. Set **Width** to **1000px**, and then set **Height** to **125px**.
18. Click **OK**.
19. In the Properties pane, under **FONT & TEXT**, set **Horizontal alignment** to **Center**.
20. From the Toolbox, expand **ACTIVE REPORT** section, and drag a **Data radio button group** into the Box:
 - Name: **ProductLine_Data_RBG**
 - Query name: **ProductLine_Data_RBG_Query**
21. From the **Data**  tab, drag **Sales and Marketing (query) > Sales (query) > Products > Product line** to the **ProductLine_Data_RBG** control's drop zone.

Task 2. Display key metrics

1. From the Toolbox, drag a **List** onto the Block, after **ProductLine_Data_RBG**:
 - Name: **Sales_List**
 - Query name: **Sales_List_Query**
2. In the Properties pane, under **BOX**, double-click **Margin**, set **Top margin** and **Bottom margin** to **15px**.
3. From the **Data**  tab, drag the following query items to the List:
 - Sales fact > **Revenue**, **Planned revenue**, **Quantity**, and **Gross profit**.
4. In the Properties pane, under **GENERAL**, change **Column Titles** to **Hide**.
5. Click **<Revenue>**, then on the Toolbar, click **Select ancestor** , then click **List columns body style** .
6. In the Toolbar, click **Border** .
7. Under **Style**, click **None**, then click **Apply All Borders** , then click **OK**.
8. In the Properties pane, under **POSITIONING**, double-click **Size & overflow**.
9. Set **Width** to **250px**, and then click **OK**.
10. In the Toolbar, click **Font**.
11. Set **Size** to **20pt**, **Weight** to **Bold**, and then click **Foreground Color**.
12. Double-click **Blue**, then click **OK**.
13. In the Properties pane, under **FONT & TEXT**, set **Horizontal alignment** to **Center**.

The results appear as follows:

<Revenue>	<Planned revenue>	<Quantity>	<Gross profit>
<Revenue>	<Planned revenue>	<Quantity>	<Gross profit>
<Revenue>	<Planned revenue>	<Quantity>	<Gross profit>

14. In the Properties pane, under DATA, double-click **Data format**.
15. Under **Format type**, select **Currency**, and define the following properties:
 - Number of decimal places: **0**
 - Scale: **-6**
 - Pattern: **\$### M**
16. Click **OK**.
17. Click **<Quantity>**, then on the Toolbar, click **Filters > Edit Filters**.
18. Click **Add**, click **Advanced**, and then click **OK**.
19. In the **Expression Definition** pane, define the following expression:
 - **[Sales (query)].[Retailers].[Retailer country]='Australia'**
20. Click **OK**.
21. Click **Add**, click **Advanced**, and then click **OK**.
22. In the **Expression Definition** pane, define the following expression:
 - **[Sales (query)].[Time].[Year]=2012**
23. Click **OK**.
24. Click **OK**, to close the Filters dialog.
25. In the Toolbar, click **Data format**
26. Under **Format type**, select **Number**.
27. Define the following properties:
 - Scale: **-3**
 - Pattern: **### K**
28. Click **OK**.
29. From the Toolbox, drag a **Table** into the Block, below the **Sales_List** control.
30. Set **Number of columns** to **4**, and then click **OK**.
31. In the Properties panel, under **FONT & TEXT**, set **Horizontal alignment** to **Center**.

32. In the Toolbar, click **Font**.
33. Set **Size** to **12pt**, **Weight** to **Bold**.
34. Click **Foreground Color**, then double-click **White**.
35. Click **OK** to close the Font dialog.
36. In each of the Table cells, click **Add** then click **Text item**, and type the following:
 - Column 1: **Revenue**
 - Column 2: **Planned Revenue**
 - Column 3: **Quantity**
 - Column 4: **Gross Profit**
37. Ctrl-click the 4 table cells, then in the Properties pane, under POSITIONING, double-click **Size & overflow**.
38. Under **Width**, type **250px**, and then click **OK**.
39. Click the **ProductLine_Data_RBG** control.
40. In the Toolbar, click **Create a new connection**, and specify the following:
 - **ProductLine_Data_RBG** (Product line) > **Filter** > **Sales_List** (Copy Data Item from Source)
 - Create a new variable: **vProductLine**
 - Default value: **Camping Equipment**
41. Click **Connect**.
42. On the Application bar, click **Run options** , then click **Run Active Report**.
43. Select **Personal Accessories**.

The result appears as follows:

<input type="radio"/> Camping Equipment	<input type="radio"/> Golf Equipment	<input type="radio"/> Mountaineering Equipment	<input type="radio"/> Outdoor Protection	<input checked="" type="radio"/> Personal Accessories
\$14 M	\$14 M	258 K	\$6 M	

44. Close the rendered report tab.

Task 3. Add the Data deck to the report

1. From the **Toolbox**, drag a **Data deck** below the Block:
 - Name: **ProductLine_Data_Deck**
 - Query name: **ProductLine_Data_Deck_Query**
2. From the **Data** panel, drag **Products > Product line** into the **ProductLine_Data_Deck** object's **Deck items** zone.

Now you will specify interaction between **ProductLine_Data_RBG** and the **ProductLine_Data_Deck** Data deck

3. Click the **ProductLine_Data_RBG**.
4. In the Toolbar, click **Create a new connection**, and specify the following:
 - ProductLine_Data_RBG (Product line) > Select > **ProductLine_Data_Deck** (Product line)
 - Reuse an existing variable: **vProductLine**
5. Click **Connect**.
6. From the **Toolbox** tab, drag a **Table** onto the **ProductLine_Data_Deck**.
7. Set **Number of rows** to **2**, uncheck **Maximize width**, and then click **OK**.
8. In the Properties panel, under POSITIONING, double-click **Size & overflow**, set **Width** to **1000px**.
9. Select the top 2 table cells.
10. In the Toolbar, click **Font** .
11. Set **Size** to **12pt**, then click **Foreground Color**.
12. Double-click **Teal**, then click **OK** to close the Font dialog.
13. In the Properties pane, under FONT & TEXT, set **Horizontal alignment** to **Center**.
14. In the Properties pane, under POSITIONING, double-click **Size & overflow**, set **Width** to **50%**.
15. In the top-left table cell, click **Add** , then click **Text item**.
16. Type **2012 Revenue by Retailer Type**, and then click **OK**.
17. In the top-right table cell, click **Add** , then click **Text item**.
18. Type **2012 Margin by Retailer Type**, and then click **OK**.

Task 4. Add revenue chart to the report

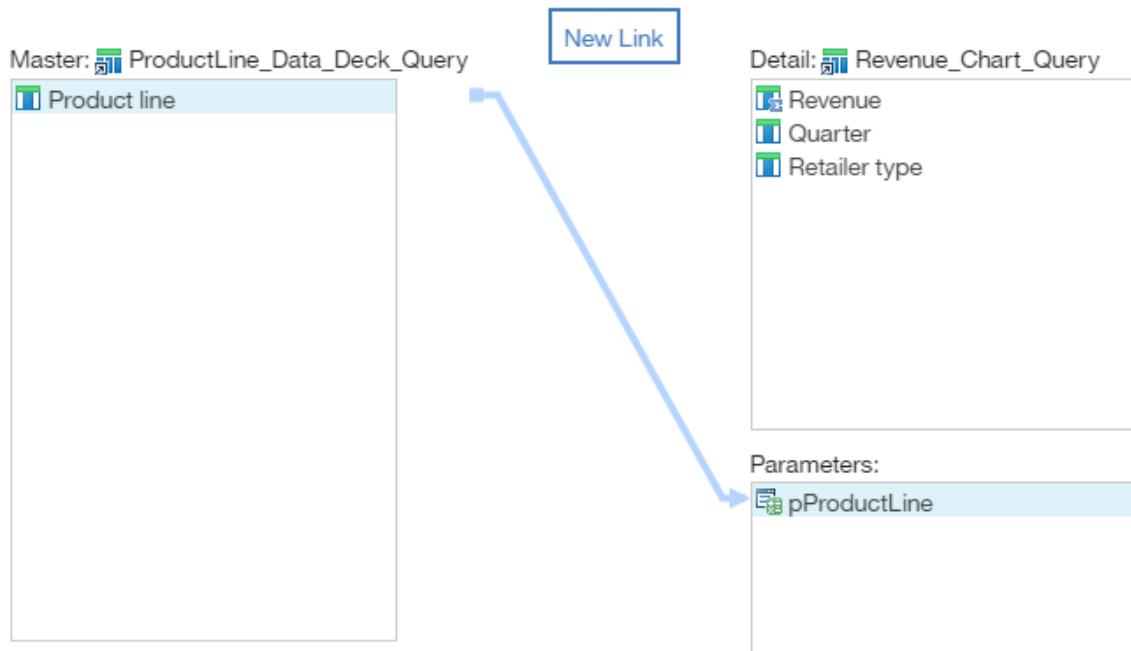
1. In the bottom-left Table cell, click **Add** , then click **Visualization** .
2. In the top right corner of the Visualization gallery dialog, click **Charts**.
3. In the **Find** box, type: **Cylinder**, then double-click the **Clustered Cylinder** chart.
4. In the Properties pane, under MISCELLANEOUS, set **Name** to **Revenue_Chart**.
5. In the Toolbar, click **More** , then click **Go to query**.
6. In the Properties pane, under MISCELLANEOUS, set **Name** to **Revenue_Chart_Query**.

7. From the side bar, click **Pages** , and then click **Page1**.
8. From the **Data** panel, drag **Sales fact > Revenue** to **Default measure (y-axis)**.
9. Expand the **Time** query subject, and then drag **Quarter** to **Categories (x axis)**.
10. Expand **Retailer type**, and then drag **Retailer type** to **Series (primary axis)**.
11. In the Properties pane, under POSITIONING, double-click **Size & overflow**.
12. Set **Width** to **500px** and **Height** to **400px**, and then click **OK**.
13. On Toolbar, click **Filters > Edit Filters**.
14. Click **Add**, click **Advanced**, and then click **OK**.
15. In the **Expression Definition** pane, define the following expression:
 - **[Sales (query)].[Retailers].[Retailer country]='Australia'**
16. Click **OK**.
17. Click **Add**, click **Advanced**, and then click **OK**.
18. In the **Expression Definition** pane, define the following expression:
 - **[Sales (query)].[Time].[Year]=2012**
19. Click **OK**.
20. Click **Add**, click **Advanced**, and then click **OK**.
21. In the **Expression Definition** pane, define the following expression:
 - **[Sales (query)].[Products].[Product line]=?pProductLine?**
22. Click **OK** to close the Detail filter expression dialog.
23. Click **OK** to close the Filters dialog.

Now you will link the query of the **ProductLine_Data_Deck** to specify the value of the **pProductLine** parameter; to be used in the filter expression.
24. On the Toolbar, click **More** , then click **Master Detail Relationships**.
25. Click **New Link**.
26. Under **Parameters**, click **pProductLine** to create a link between Product line from **ProductLine_Data_Deck_Query** and the parameter **pProductLine**.

The result appears as follows:

Master detail relationships



27. Click **OK**.

Task 5. Add margin Chart to the report

1. In the bottom-right Table cell, click **Add** , then click **Visualization** .
2. From the top right corner, click the visualization filter, then click **New visualization**.
3. Double-click the **Clustered Cylinder Bar** chart.
4. In the Properties pane, under MISCELLANEOUS, set **Name** to **Margin_Chart**.
5. In the Toolbar, click **More** , then click **Go to query**.
6. In the Properties pane, under MISCELLANEOUS, set **Name** to **Margin_Chart_Query**.
7. In the **Queries** tab, expand **Margin_Chart_Query**, then click **Margin_Chart**.
8. From the **Data** tab, drag **Gross margin** to **Default measure (x-axis)**.
9. Drag **Time > Quarter** to **Categories (y-axis)**.
10. Drag **Retailer type > Retailer type** to **Series (primary axis)**.
11. In the Properties pane, under POSITIONING, double-click **Size & overflow**.
12. Set **Width** to **500px** and **Height** to **400px**, and then click **OK**.
13. On the Toolbar, click **Filters**, and then click **Edit Filters**.
14. Click **Add**, click **Advanced**, and then click **OK**.
15. In the **Expression Definition** pane, define the following expression:
 - **[Sales (query)].[Retailers].[Retailer country]='Australia'**

16. Click **OK**.
17. Click **Add**, click **Advanced**, and then click **OK**.
18. In the Expression Definition pane, define the following expression:
 - **[Sales (query)].[Time].[Year]=2012**
19. Click **OK**.
20. Click **Add**, click **Advanced**, and then click **OK**.
21. In the **Expression Definition** pane, define the following expression:
 - **[Sales (query)].[Products].[Product line]=?pProdLine?**
22. Click **OK** to close the Detail filter expression dialog.
23. Click **OK** to close the Filters dialog.

Now you will link the query of the **ProductLine_Data_Deck** to specify the value of the **pProductLine** parameter; to be used in the filter expression.
24. On the Toolbar, click **More** , then click **Master Detail Relationships**.
25. Click **New Link**.
26. Under **Parameters**, click **pProductLine** to create a link between Product line from **ProductLine_Data_Deck_Query** and the parameter **pProductLine**.
27. Click **OK**.
28. Click **<Gross margin>**, then from the Properties pane, under DATA ITEM, set **Detail aggregation** to **Average**.

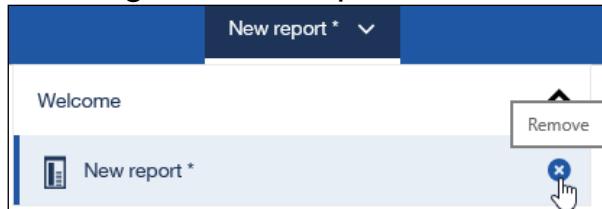
Task 5. Run the Active Report

1. On the Application bar, click **Run options** , then click **Run Active Report**.
2. Select **Personal Accessories**.

The result appears as follows:



3. Close the rendered report's tabs.
4. In the Application bar, click the report dropdown menu, then click **Remove**  to the right of New report to close it. Then click **OK** to confirm your action.



5. Leave the IBM Cognos Analytics portal open for the next exercise.

Results:

You created a professional-looking formatted Active Report that provides insight on sales in Australia for 2012.

Summary

- Describe characteristics of traditional charts in Active Reports
- Control data display using decks and data decks
- Use decks and data decks to display traditional charts
- Optimize decks for performance
- Describe characteristics of RAVE Visualizations