

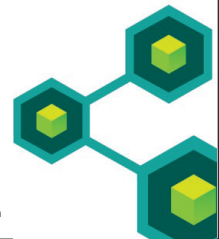


Explore Dynamic Query Mode

IBM Cognos BI 10.2.2

Business Analytics software

© 2015 IBM Corporation



Objectives

- At the end of this module, you should be able to:
 - explain Dynamic Query Mode (DQM) logging
 - explain IBM Cognos Dynamic Query Analyzer (DQA)



Available Dynamic Query Mode Logging

- XQE log files
- IPF logging
- QFS logging
- BAPI tracing (for SAP BW)
- QueryService logging

© 2015 IBM Corporation



Usually, this type of logging would be done under the guidance of customer support.

A general starting point for troubleshooting errors or connectivity issues is to review XQELog files. IPF logging is used mainly to enable XQE audit logging. QFS logging is used for tracing performance and communication between XQE and the BIBus. BAPI tracing for SAP BW is enabled in XQE.diagnosticlogging.xml.

QueryService logging is useful for troubleshooting report issues.

Describe XQE Log Files (1 of 2)

- located in ..\logs\XQE by default
- log files generated by default
 - xqelog-<date>-<timestamp>.xml
 - queriesNotPlanned
 - usage
 - requestDumps

© 2015 IBM Corporation



The xqelog files are at the root of the XQE directory and are date and time stamped. These are a good starting point for DQM issues. Log files are not cleaned up automatically, and will need to be removed manually or through some batch process implemented outside of IBM Cognos 10.

You can change the default XQE log output location in the xqe.config.xml file under the XQEConfiguration\General section. Uncomment `<!--logsFolder value="..\../logs"/-->` and specify the location in the value attribute. This can be helpful in a distributed environment, to define a central log location.

queriesNotPlanned is DQM information output for queries that could not be planned for one reason or another. For example, these logs may be generated if the IBM Cognos data source was missing for a report. The report spec query or queries cannot be planned in these cases. Another time you will see these logs generate is when a report is run and you see an XQE-PLN-0000 error. These files can indicate when DQM reports are failing.

DQM statistics are tracked in the usage files log. These files are generated when the BI server is stopped.

requestDumps are used to analyze data output for requests. This can be useful in troubleshooting data integrity issues between IBM Cognos 10 releases. For example, report data from release to release may return the same data differently. This logging would only be enabled under the guidance of customer support.

Describe XQE Log Files (2 of 2)

- start point for any errors or DQM issues
- contains IPF logging
- XQE logging can be configured in `xqe.diagnosticslogging.xml` located in `..\configuration`
 - can increase logging detail to the `xqelog` XML files in the `Diagnostics\Components` section
 - can enable or disable various XQE logging
- must restart Cognos BI service for changes to take effect

© 2015 IBM Corporation



When dealing with any DQM errors or issues, such as service or connectivity issues, the `xqelog-000000-0000000000.xml` log files are a good start. These logs can provide information to help guide you on where to focus your attention. For example, it may indicate an issue with the `MetadataService` component of DQM. You can set the `MetadataService` `eventGroup` item under the `Diagnostics\Components` section to trace, which will write more details about that component in the `xqelog-000000-0000000000.xml` log files. If there is no `level` attribute for a component, the default behavior is `none` (no logging).

In the `xqe.diagnosticslogging.xml` file, you can also enable and disable specific log files. For example, you can disable `queriesNotPlanned` or enable `queriesPlanned`. You can also turn off `xqelog-000000-0000000000.xml` logging by changing `<diagnostics enabled="true" appender="file" level="error">` to `<diagnostics enabled="false" appender="file" level="error">`.

IPF Logging for DQM

- enabled by renaming ipfXQEclientconfig.xml.sample to ipfclientconfig.xml in ..\configuration
- change is picked up after approximately 30 seconds
- for XQE, typically would be used to enable IPF audit logging since all IPF information is already written to the xqelog XML files

© 2015 IBM Corporation



The same concept about IPF logging applies here for the DQM component except that the ipfXQEclientconfig.xml logging was created for conformance with other components. This version of IPF logging is written to the xqelog-000000-0000000000.xml files (mentioned earlier). Enabling IPF logging here would be possible, but redundant.

Describe QFS Logging for DQM

- configured in qfs_config.xml in ..\configuration
- troubleshoot communication and performance between BIBus (C++) and DQM (Java)
- parameters of interest under the XQEQueryProvider section
 - TraceLogLevel (communication)
 - generates xqe prefixed log files in ..\logs
 - PerfLogLevel (performance)
 - generates xqePerf.csv file in ..\logs

© 2015 IBM Corporation



For TraceLogLevel, the xqe log file name will be xqeTrace.log when setting the value to 1 or 2. If set to 3 or higher, you will see an xqeTrace.log file and an XQERequest_00000000.log file. Value setting information is provided in this file.

Logging values for `<parameter name="TraceLogLevel" value="0"/>` are 0 Disable trace logging, 1 Service level trace logging, 2 Session level trace logging, 3 Request level trace logging, 4 Operation level trace logging, and 5 Detail trace logging.

The PerfLogLevel trace produces the xqePerf.csv file in the logs folder. This file indicates the object, the operation, a begin and end timestamp for each operation, and a column to show the time difference between the two.

There are other settings for tweaking performance and functionality also in this file, but use these under customer support guidance only. qfs_changes are supposed to be picked up automatically after approx. 30 seconds, but you may have to refresh the service.

Enable DQM BAPI Trace for SAP BW

- in ..\configuration, configure the SBWOPD section at the bottom of the xqe.diagnosticslogging.xml file change
 <eventGroup name="DataQuery"/>
to
 <eventGroup name="DataQuery" level="info"/>
- information written to XQElog XML files

© 2015 IBM Corporation



Add the level="info" comment in xqe.diagnosticslogging.xml, and then stop and start the Query Service on the Dispatcher where the file change was made, to pick up the changed configuration.

What is IBM Cognos DQA?

- IBM Cognos Dynamic Query Analyzer
- ships with IBM Cognos 10
- focuses on profiled runtree logs
- if report failed in planning you can open the last stage of planning log

© 2015 IBM Corporation



IBM Cognos Dynamic Query Analyzer (DQA) is a utility that provides a graphical user interface for the execution tree logs (runtreeLog.xml) that are produced by Dynamic Query Mode queries.

You can enable global execution trace logging for all DQM reports through the IBM Cognos Administration console, or you can use IBM Cognos DQA to run individual reports, which will automatically generate an execution trace for just that report, and can be immediately analyzed in IBM Cognos DQA. For more information, refer to the IBM Cognos *Dynamic Query Analyzer User Guide 10.2.2*.

What Can You Do with IBM Cognos DQA?








- debug reports
- run and display reports
- read logs locally or remotely
- display properties for each individual node
- display MDX and SQL and copy the syntax
- display performance times

© 2015 IBM Corporation



IBM Cognos DQA allows a report administrator to easily identify the individual pieces of a Dynamic Query Mode query. This overview helps to simplify the troubleshooting of Dynamic Query Mode query performance.

Review the IBM Cognos DQA Graph Legend

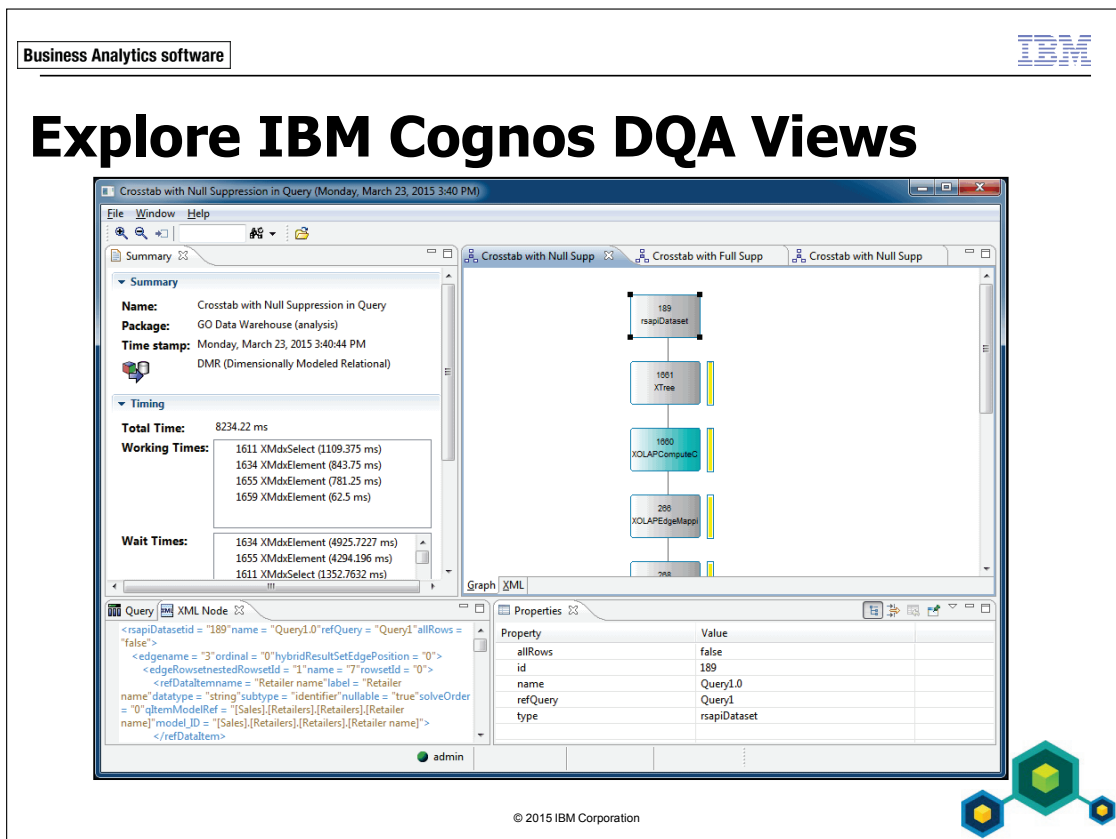
Node	Description
	An operation that occurred when the report was run.
	An attribute of an operation, such as the data being processed. By default, most of these nodes are suppressed in the graph. (To display all navigate to Window\ Preferences, and select Visualization\Node filtering.)
	A collapsed node. To display the hidden nodes, double-click the node, or right-click it and click Show Subtree.
	A node that with hidden children due to the Node filtering settings. (To display hidden nodes, right-click the node and click Expand Filtered.)
	The node that is currently selected. The properties of this node are displayed in the Properties view. Click a node to select it.
	This node has subqueries that can be opened in another graph. (To display subquery graphs, right-click the node and click Open sub queries. Some subqueries do not have an associated node in the parent trace and can be opened using this option.)
	The vertical bar represents timing information for the node. The color red represents the time for the report spent in the node. The color yellow represents the proportion of time spent in the children of the node. The color gray represents the time spent outside the node and its children. If the node is selected, the times are also displayed in the Properties view.

© 2015 IBM Corporation



When you open query log files in IBM Cognos DQA, you can view and analyze the data in a number of ways.


The graph that is displayed when a query log file is opened shows a series of linked nodes. Each node represents an operation that occurred when the report was run or represents an attribute of an operation (such as the data being processed). The meaning of the different node representations is defined in the table on the slide.




The Summary view tab provides information about the active graph. There are four sections on this tab:


- **Summary section:** displays the name, package, and timestamp of the query. It also indicates whether the data source type is relational, OLAP, or DMR (Dimensionally Modeled Relational).
- **Timing section:** displays timing information, the working times and waiting times for each node that has timing information attached to it, in descending time order. If you click a node, the corresponding node in the graph is selected. Timing information for a query is not displayed if all the timing information is contained in subqueries.
- **Analysis section:** displays information if the query planning trace option is enabled.
- **Node Shapes and Colors section:** provides the meaning of node shapes and colors.

The Query view tab shows the MDX or SQL query used in this report.

You can link MDX commands in the Query view with nodes in the graph by using the Link MDX to graph  icon. If you select an MDX command in the Query view and click the Link MDX to graph icon, the nodes in the graph where the MDX commands are executed are selected.

If the query is a SQL query, you can run the report again by clicking the Execute Sql query  icon.

The Properties view tab displays the properties for the currently selected graph node.

The Navigation view tab displays the graph as a tree structure. To open the Navigation view for the active graph, click File\Show in Navigation. You can have multiple Navigation views open, one for each graph. To navigate quickly between a Navigation view and the corresponding graph, click the Link with editor  icon. Clicking an item in the Navigation view selects the corresponding node in the graph, and vice versa.

Explore Aggregate Advisor

- in IBM Cognos Dynamic Query Analyzer, Aggregate Advisor is used with dynamic cubes
- can use default logging settings
- optimize aggregates
- resolve performance issues

© 2015 IBM Corporation



In IBM Cognos Dynamic Query Analyzer, you can launch the Aggregate Advisor, to analyze dynamic cubes and suggest aggregates that, when implemented, will improve cube performance. While tweaking a model is required when using in-database aggregates, for in-memory aggregates the Aggregate Advisor allows performance problems to be resolved without re-authoring or re-modeling.

Default logging is sufficient to enable logs for use by the Aggregate Advisor, as Aggregate Advisor can analyze the underlying model in a dynamic cube data source and recommend which aggregates to create. Aggregate Advisor runs on the query service and can reference a workload log file. Administrators should be aware that to consider information from workload logs when making recommendations, the workload log file must be enabled on the dynamic cube.

When enabled, the workload log file captures the information that represents user workload usage such as running reports or performing interactive analysis. This log file allows Aggregate Advisor to suggest aggregates (in-database or in-memory) that correspond directly to the reports referenced in the log file.

Consider running Aggregate Advisor if non-trivial changes are made to the model, if there are significant data changes, if query performance is not meeting expectations, or if workload characteristics change significantly.

For more information on using the Aggregate Advisor, refer to the IBM Cognos *Dynamic Query Analyzer User Guide 10.2.2*, in the section Getting started with Cognos IBM Cognos Dynamic Query Analyzer: Optimize dynamic cube performance with Aggregate Advisor.

Summary

- At the end of this module, you should be able to:
 - explain Dynamic Query Mode (DQM) logging
 - explain IBM Cognos Dynamic Query Analyzer (DQA)



Workshop 1: Enable DQM, Publish a Package, and Review the Logs Created

The following workshop will explain how to configure a relational data source for Dynamic Query Mode, publish a package in DQM, enable global trace logging, and review the logs that are created. When you are done, you will disable global trace logging.

To do this you will:

- Ensure that the query service is enabled in IBM Cognos Configuration.
- Create a dynamic query mode enabled project and connection to IBM DB2. Create a project named DB2 DQM Model, enabling Dynamic Query Mode; use admin/Education1 credentials.
- Create a data source connection named GOSALESDW(DB2) based on an IBM DB2 connection, and enable the JDBC connection to the GS_DB database, with GOSALESDW/Education1 credentials, and test the connection for compatible mode.
- Configure the JDBC connection to VCLASSBASE:50000 for the GS_DB database, and then test for dynamic query mode compatibility.
- For the GOSALESDW(DB2) data source, import the following tables: GO_TIME_DIM, SLS_PRODUCT_DIM, SLS_SALES_FACT, and then test the GO_TIME_DIM definition.
- Publish a package named GOSALESDW(DB2).
- In IBM Cognos Administration, ensure that the data source connection for GOSALESDW(DB2), is running in both modes (DQM and Compatible) on the dispatcher.

- In IBM Cognos Administration, select the query execution trace check box to enable the service.
 - Create, save, and run a simple report based on the GOSALESDW(DB2) package.
 - Review the logs generated by DQM at C:\Program Files\IBM\cognos\c10_64full\logs\XQE.
- If necessary, modify the <reportPath></reportPath> tags in the manifest.xml file to refer to the full path of the report.

For more information about where to work and the workshop results, refer to the Tasks and Results section that follows. If you need more information to complete a task, refer to earlier demos for detailed steps.

Workshop 1: Tasks and Results

At the beginning of this workshop, the IBM Cognos Full:9315 dispatcher is running.

Task 1. Ensure the Dynamic Query Mode service is enabled.

- From the **Start** menu, navigate to **All Programs\IBM Cognos 10 - 64 Full**, and then click **IBM Cognos Configuration**.
- In the **Explorer** pane, click **IBM Cognos Services**, and then ensure **Query service enabled?** is set to **True**.

If the value was set to False, change it to True, save the configuration, and then restart the IBM Cognos service.

- Close **IBM Cognos Configuration**.

Task 2. Create a dynamic query mode enabled project and connection to IBM DB2.

The following instructions use the sample Great Outdoors Warehouse IBM DB2 database called GS_DB to allow you to create a relational data source connection.

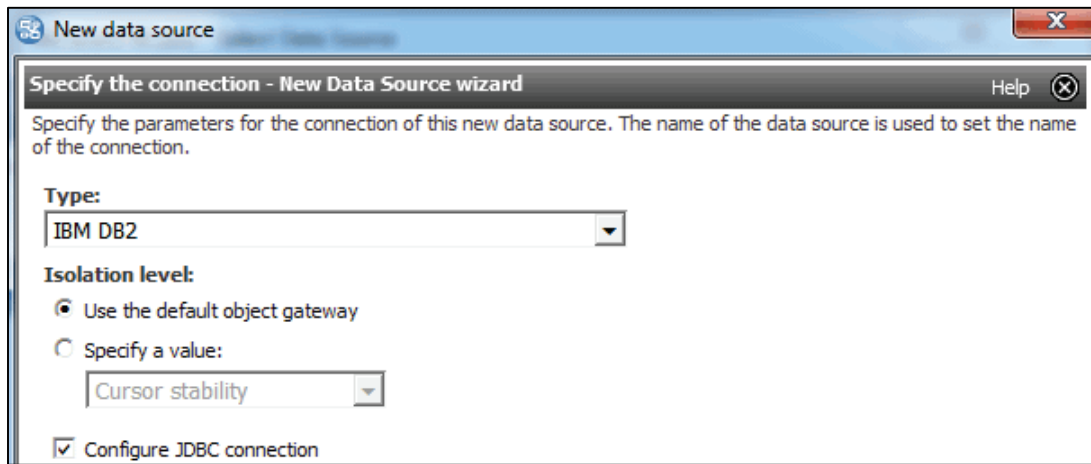
For an IBM DB2 data source, the IBM Cognos Framework Manager machine requires no additional software if all of your Framework Manager projects use the Dynamic Query Mode. If any of your projects use the Compatible Query Mode, however, an IBM DB2 client must be installed and configured on the IBM Cognos BI Framework Manager machine. This has been configured in the environment for this course.

Although you may already be familiar with creating a Dynamic Query Mode-enabled project, this workshop will be looking deeper into the process and the results.

- From the **Start** menu, navigate to **All Programs\IBM Cognos 10 FM**, and then click **IBM Cognos Framework Manager**.
- Click **Create a new project**, in the **Project name** box, type **DB2 DQM Model**, and ensure that the **Use Dynamic Query Mode** check box is selected.
- Notice the path of the project location, and then click **OK**.
- Click **OK** if a message is displayed to create the directory, and then login to the **LDAP_Dev** namespace with **admin/Education1** credentials.
- In the **Select Language** dialog box, ensure that **English** is selected, and then click **OK**.

- On the **Select Metadata Source** page, ensure that **Data Sources** is selected, click **Next**, and then click **New** to create a new data source connection.
- In the **New Data Source** wizard, click **Next**, in the **Name** box, type **GOSALESDW(DB2)**, and then click **Next**.
- Under **Type**, click **IBM DB2**.

Notice the Configure JDBC connection check box. Ensure that this box is selected, so that information can be provided to connect through the JDBC driver which is required for Dynamic Query Mode.




- Click **Next**.

In the next steps, the information provided is based on how the IBM DB2 clients on the Framework Manager machine and the IBM Cognos BI servers were configured and how security is implemented for IBM DB2. Connection information and sign on information should be provided by the database administrator.

- In the **DB2 database name** box, type **GS_DB**, in the **Signon\Signons** section, click the **Password** check box to select it, in the **User ID** box, type **GOSALESDW**, and then in the **Password** and **Confirm password** boxes, type **Education1**.

- Click **Test the connection**, and then click **Test**.

Notice that the Query Mode is Compatible. This means that Dynamic Query Mode has not been configured for this data source at this point.


	...> Name	Type / Query Mode	Status
	http://vclassbase:9315/p2pd	IBM DB2 / Compatible	Succeeded

- On the **View the results** page click **Close**, on the **Test the connection** page click **Close**, and then on the **Specify the IBM DB2 connection string** page, click **Next**.

You will now configure the JDBC connection information.

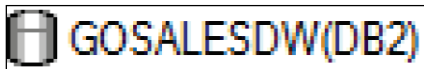
- In the **Server name** box, type **VCLASSBASE** (the name of the server hosting the database), in the **Port number** box, type **50000** (the port number of the database), and then in the **Database name** box, type **GS_DB**.
- In the **Testing** section click **Test the connection**, and then click **Test**.

Notice that the Query Mode is Dynamic.

	...> Name	Type / Query Mode	Status
	http://vclassbase:9315/p2pd	IBM DB2 (JDBC) / Dynamic	Succeeded

- On the **View the results** page click **Close**, on the **Test the connection** page click **Close**, click **Finish**, and then in the **Finish** dialog box, click **Close**.

The new GOSALESDW(DB2) data source that appears in the list is configured to query using either query mode.

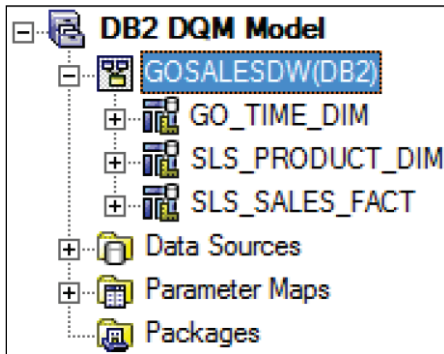


You will import metadata and test query subjects for this data source.

- Click **GOSALESDW(DB2)** to select it in the list of data sources, click **Next**, expand **GOSALESDW**, expand **Tables**, and then select the following tables:
 - GO_TIME_DIM**
 - SLS_PRODUCT_DIM**
 - SLS_SALES_FACT**
- Click **Next**, click **Import**, and then click **Finish**.

- In the **Project Viewer**, expand **GOSALESDW(DB2)**.

The query subjects appear as child objects as follows.



- Double-click **GO_TIME_DIM** to open its definition, and then click the **Test** tab.

If the Query Mode property of the project is set to Dynamic when testing a query subject, the test query will run in Dynamic Query Mode. If the Query Mode property is set to Compatible, however, there is an option to use the Dynamic Query Mode on the Test tab in the lower left corner, provided that the query subject is for a data source supported by the Dynamic Query Mode. In this case, the property for this project was set to Dynamic Query Mode, and therefore you do not see this check box option.

- At the lower right, click **Test Sample**.

Framework Manager sends the test query through the IBM Cognos 10 gateway to one of the IBM Cognos BI servers, which, in turn, queries the reporting database. The data retrieved by the test query appear in the Test results pane.

A section of the result appears as follows:

Query Subject Definition - GO_TIME_DIM					
SQL Calculations Filters Determinants Test Query Information					
Test results					
DAY_KEY	DAY_DATE	MONTH_KEY	CURRENT_MONTH	MONTH_NUMB	
20100000		201000	0	0	
20100101	Jan 1, 2010 12:00:00 AM	201001	1	1	
20100102	Jan 2, 2010 12:00:00 AM	201001	1	1	

- Click **OK** to close the **Query Subject Definition** window.

You should test all your model objects against the Dynamic Query Mode to ensure that SQL generation is as expected for your requirements. If you are building an OLAP over Relational model, this includes foundation objects such as Data Source and Model Query Subjects as well as Regular and Measure Dimensions. OLAP over Relational model replaces the term DMR model, for DQM.

When you have finished building the model, you can create and publish a package that uses the Dynamic Query Mode.

Task 3. Create and publish a package that uses the dynamic query mode, and then verify the package properties.

- In the **Project Viewer**, right-click **Packages**, click **Create**, and then click **Package**.
- In the **Name** box, type **GOSALESDW(DB2)**, click **Next**, and then click **Finish**.
- At the prompt to open the **Publish Package Wizard**, click **Yes**.

- Change the **Folder location in the Content Store** to the root of **Public Folders** (if necessary), in the **Publish Wizard** click the **Enable model versioning** check box to clear it, click **Next**, and then on the **Add Security** page click **Next**.

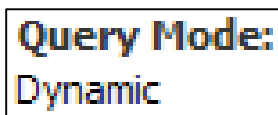
On the Publish Wizard - Options page, if the Query Mode property of the project is set to Compatible and if the package contained supported DQM data sources, then the Use Dynamic Query Mode check box option would be displayed. You have already defined the property of this project as Dynamic Query Mode, so this option does not appear.

- Click **Publish**, and then click **Finish**.

The package is now available in IBM Cognos 10 and will use the Dynamic Query Mode for reports written against this package. In IBM Cognos Connection, the query mode used by the package can be verified in the package properties.

- Close **Framework Manager**, saving your changes.
- Launch **Internet Explorer**, go to **http://vclassbase:88/C10Full**, log on to the **LDAP_Dev** namespace with **admin/Education1** credentials, and then launch **IBM Cognos Connection**.
- In the **Actions** column for the **GOSALESDW(DB2)** package that you published, click **Set properties**.

Notice that the Query Mode is Dynamic.







- Click **OK**.

Task 4. In IBM Cognos Administration, enable the Query execution trace to generate XQE log files.

In this task you will enable global tracing. Be aware that in a busy production environment, enabling global tracing will result in all Dynamic Query reports being traced as they are executed, which may have an impact on the system performance. In the next workshop, you will generate log files on a report by report basis, a strategy which would be better for a busy production environment.

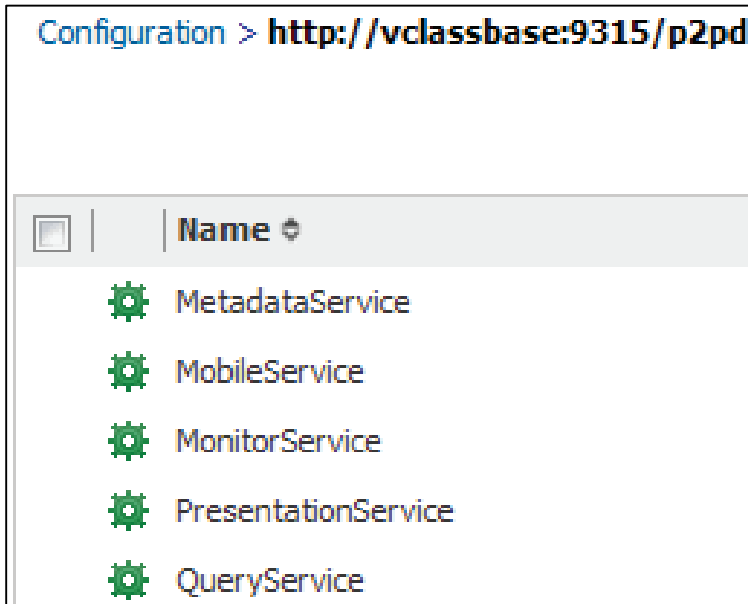
- Launch **IBM Cognos Administration**, and then click the **Configuration** tab.
On the Configuration tab, there are four locations pertaining to the Dynamic Query Service:
 - Data Source Connections for configuring data sources including supported Dynamic Query Mode data sources and dynamic cubes
 - Content Administration for scheduling Query service administration tasks
 - Dispatchers and Services for configuring the QueryService service
 - Query Service Caching to immediately perform cache tasks
- On the **Configuration** tab\ **Data Source Connections** pane, in the **Actions** column for the **GOSALESDW(DB2)** data source connection, click **More**.
- Click **View connections**, in the **Actions** column click **Test the connection**, and then click **Test**.

There are two query modes listed for this dispatcher.

	...> Name	Type / Query Mode	Status
	 http://vclassbase:9315/p2pd	IBM DB2 / Compatible	Succeeded
	 http://vclassbase:9315/p2pd	IBM DB2 (JDBC) / Dynamic	Succeeded

- Click **Close**, and then on the **Test the connection** page click **Close**.
- In the left pane, click the **Dispatchers and Services** link, and then click **http://vclassbase:9315/p2pd**.

- Click the **Next Page** button, and notice that the **QueryService** appears in the list.



- In the **Actions** column for the **QueryService**, click **Set properties**, on the **Settings** tab, in the **Value** column, click the **Enable query execution trace** check box to select it.

The trace configuration change will be picked up automatically within 15 seconds.

Enabling the query execution trace will write information such as the native MDX to a run tree log in the ..\logs\XQE directory. Profiler information is also written to capture execution and waiting time metrics for query constructs.

- In the **Value** column, notice the **Enable query planning trace** check box.

Enabling the query planning trace setting will write information related to the transformation of the query to the plan tree log within the ..\logs\XQE directory. This trace is useful when attempting to determine what decisions were made by the Dynamic Query Mode to build the execution plan, however resultant log files are large and may impact overall query performance. Use only for reports that fail before the profile is complete.

Do not enable the query planning trace in this workshop.

- Click **OK**, and then wait 15 seconds before proceeding.

Task 5. Create and run a simple report based on a DQM data source.

- Launch **IBM Cognos Connection**, click the **GOSALESDW(DB2)** package, and then from the **Launch** menu, click **Report Studio**.
- Click **Create New**, and then double-click **List**.
- From the **Source** tab, populate the list with the following items from the package:
 - **GO_TIME_DIM: DAY_DATE**
 - **SLS_PRODUCT_DIM: PRODUCT_KEY**
 - **SLS_SALES_FACT: QUANTITY**

You could use any items from the package for your report; the specific content of the items selected above is not important to this workshop. You can drag and drop items in the list, or double-click items on the Source tab to add them to the list.

- Save the report as **DQM Report**, and then run the report.

A section of the report will appear similar to the following:

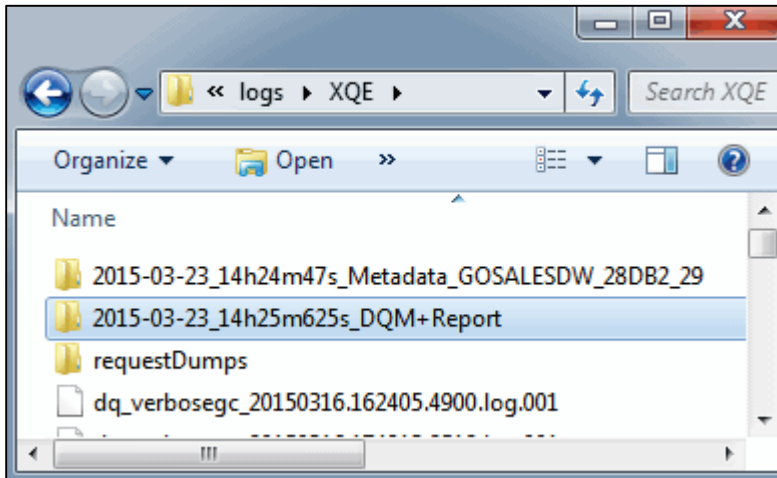
DAY_DATE	PRODUCT_KEY	QUANTITY
Jan 12, 2010 12:00:00 AM	30001	51,522
Jan 12, 2010 12:00:00 AM	30002	24,182
Jan 12, 2010 12:00:00 AM	30003	11,265

- Close all open windows.

Task 6. Review the logs generated by DQM.

- In **Windows Explorer**, navigate to **C:\Program Files\IBM\cognos\c10_64full\logs\XQE**.

This is where the QueryService logs are located. Notice that there is a folder named <date>_<timestamp>_<report name> that was generated for the report that you ran.



- Double-click the **DQM Report** folder to open it.

The following files are available:

- manifest.xml
- profilingLog-0.xml
- runtreeLog.xml

Some report executions require executing sub-queries. Sub-queries execution trace is stored under a separate directory named subqueries, within the main report directory, and contains the same logging elements as the main report, runtreeLog.xml and profilingLog-#.xml.

If you had enabled the planning trace, you would also see many planningLog_pass_###.xml files.

- Open the **manifest.xml**, **profilingLog-0.xml**, and **runtreeLog.xml** files in **Internet Explorer** to familiarize yourself with the information that has been captured.

The manifest.xml file appears similar to the following:

```
<?xml version="1.0" encoding="UTF-8"?>
- <xqetraceInfo>
  <date>2015-03-23T15:11:32-04:00</date>
  <reportName>DQM Report</reportName>
  <reportPath>/content/package[@name='GOSALESDW(DB2)']/report[@name='DQM Report']</reportPath>
  <packageName>GOSALESDW(DB2)</packageName>
  <modelPath>/content/package[@name='GOSALESDW(DB2)']/model[@name='model']</modelPath>
  <logVersion>1.2.0.v10.1</logVersion>
  <requestId>C4wCw8sv8C5sM8h2G4jICydsGywddqvM2wGqC48I</requestId>
  <operationName>execute</operationName>
  <subRequestId>C4wCw8sv8C5sM8h2G4jICydsGywddqvM2wGqC48I</subRequestId>
  <version>UNKNOWN</version>
  <expectedNumberOfPasses>549</expectedNumberOfPasses>
</xqetraceInfo>
```

- If the <reportPath></reportPath> tags contain a value of UNKNOWN, open the manifest.xml file in Notepad, modify the value within the tags to appear as follows, and then save and close the file:

```
<reportPath>/content/package/[[@name='GOSALESDW(DB2)']]/
report[@name='DQM Report']</reportPath>
```

Task 7. Disable the query execution trace.

- Launch **Internet Explorer**, go to **http://vclassbase:88/C10Full**, log on to the **LDAP_Dev** namespace with **admin/Education1** credentials, launch **IBM Cognos Administration**, and then click the **Configuration** tab.
- In the left pane, click the **Dispatchers and Services** link, and then click **http://vclassbase:9315/p2pd**.
- Click the **Next Page** button, in the **Actions** column for the **QueryService**, click **Set properties**, on the **Settings** tab, in the **Value** column, click the **Enable query execution trace** check box to clear it.

The trace configuration change will be picked up automatically within 15 seconds.

- Click **OK**, log off **Admin Person**, and then close the browser.

Workshop 2: Use IBM Cognos Dynamic Query Analyzer to Analyze a Query

IBM Cognos Dynamic Query Analyzer (DQA) is a tool that provides graphical representations for the query logs produced by Dynamic Query mode queries. This workshop will introduce you to using IBM Cognos Dynamic Query Analyzer to analyze results of the logs generated by the queries.

It is assumed that you have generated log files for use with IBM Cognos Dynamic Query Analyzer, as described in Workshop 1. Please complete Workshop 1 before proceeding with this workshop. This will help you become familiar with analyzing log files in IBM Dynamic Query Analyzer, before you assess differences in report variations.

In this workshop scenario, with no global tracing enabled, you will run a report directly in IBM Cognos Dynamic Query Analyzer, which will generate log files that you can immediately analyze. This method of generating log files will have less of an impact on the production environment than the global tracing method.

To do this workshop you will:

- Launch IBM Cognos DQA. If you encounter any "null pointer" errors, close and reopen IBM Cognos DQA. If you encounter any login-related messages, dismiss them. Review the settings in Window\Preferences to familiarize yourself with the configuration of the environment. If necessary, modify the settings to reference the correct IBM Cognos server and the logs directory URL. After you have validated the settings, close and reopen DQA (logging on if prompted).
- In IBM Cognos DQA, refresh the content store display (if necessary), and explore the GOSALESDW(DB2)\DQM Report\DQM Report (<Date>, <Time>)\ Profile 0 log. Refer to the DQA Graph Legend page in this module, as you review the objects in the graph, to assist in your interpretation of the graph.
- In a browser session, login to the portal at <http://vclassbase:88/C10Full>, with LDAP_Dev/admin/Education1 credentials.

- Launch Report Studio with the Public Folders\Samples_DQ\Models\GO Data Warehouse (analysis) package.
- Create a crosstab report with the Sales and Marketing (analysis)\Sales source (Measures: Sales fact\Quantity, Rows: Products dimension\Products hierarchy\Product level, Columns: Retailers dimension\Retailers hierarchy\Retailer name level, and Order method dimension\Order method hierarchy\Order method type level, nested under <#Retailer name#>). Apply full suppression to rows and columns. Save the report as Crosstab with Full Suppression.
- In the same report, change the suppression to nulls only, for rows and columns, and save the report as Crosstab with Null Suppression.
- In the same report, remove the suppression from the rows and columns, apply suppression within the query, and save the report as Crosstab with Null Suppression in query.
- Run one of the three reports, clearing the cache after it has run. Repeat this for the other two reports.
- In IBM Cognos DQA, review the Profile 0 logs generated by each of the three report executions. Explore the runtreen graph, XML node information, Properties of the nodes, and timings. Compare results of the three report variations.

You have compared three different variations of a report, using IBM Cognos Dynamic Query Analyzer, and can use the results to ensure that the most efficient report is available for users. Which report is the most efficient, based on your findings?

For more information about where to work and the workshop results, refer to the Tasks and Results section that follows. If you need more information to complete a task, refer to earlier demos for detailed steps.

Workshop 2: Tasks and Results

At the beginning of this workshop, the IBM Cognos Full:9315 dispatcher is running, and log files are available in the `..\logs\XQE` directory.

Task 1. Start IBM Cognos Dynamic Query Analyzer and review the configuration.

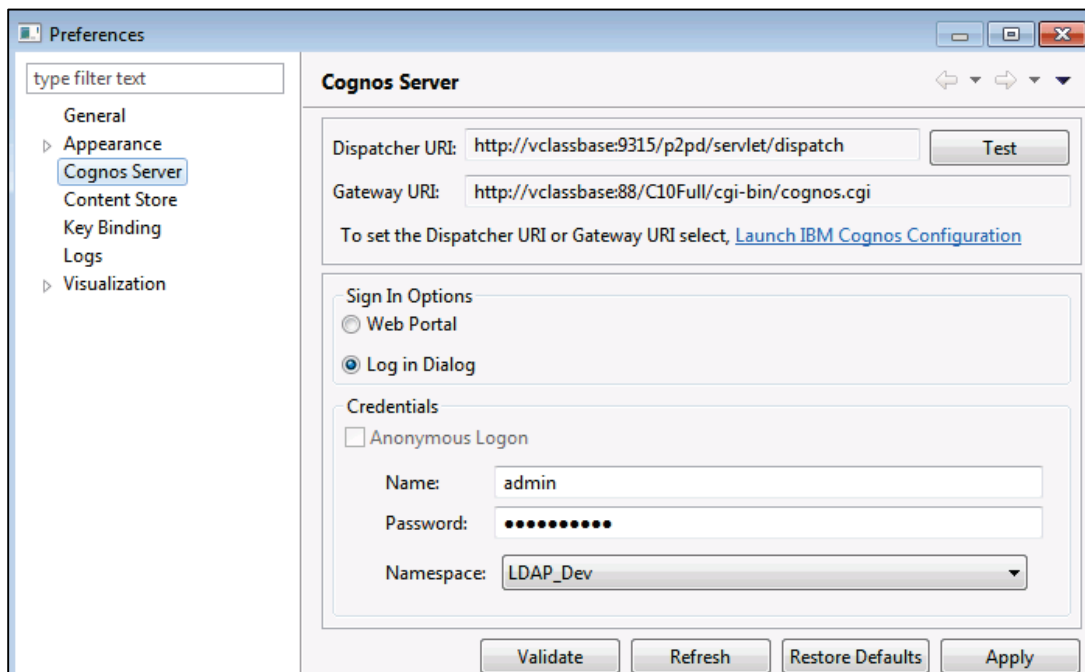
- From the **Start** menu, navigate to **All Programs\IBM Cognos 10 - 64 Full**, and then click **IBM Cognos Dynamic Query Analyzer**.

If you receive any "null pointer" errors (either standalone or at the top of what appears to be a login dialog box), dismiss the errors and then close and reopen IBM Cognos Dynamic Query Analyzer before proceeding.

- If you receive any other login-related errors, dismiss them and leave IBM Cognos Dynamic Query Analyzer open.
- From the **Window** menu, click **Preferences**, and then in the pane on the left, click **Cognos Server**.

This property has the settings for the host and port of the IBM Cognos server which contains the reports to be analyzed. This has been configured by the administrator who set up the environment.

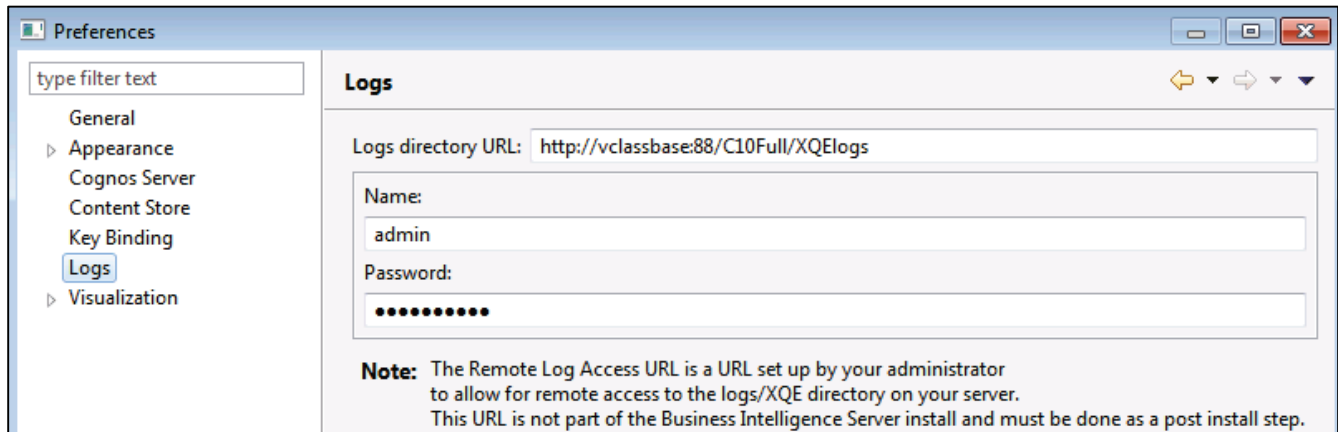
- If necessary, ensure that the settings appear as shown below (the value in the Password box is Education1), click **Validate**, click **OK**, and then click **Apply**.



- On the left, click **Logs**.

The Logs directory URL points to a virtual directory (XQElogs) on the Web server, which references the XQE directory where the log files are saved on the server. This property has been configured by the administrator who set up the environment.

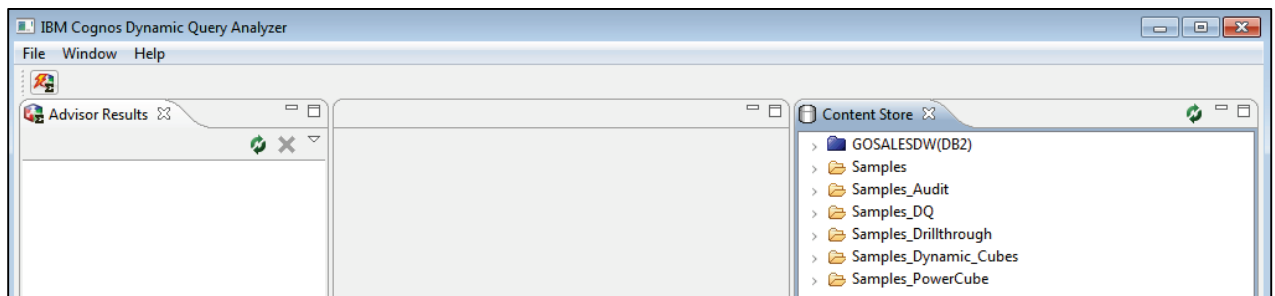
If necessary, ensure that the settings appear as shown below (the value in the Password box is Education1), and then click Apply:



- Click **OK** to close the **Preferences** dialog box.
- To ensure that you are in the correct view for this workshop, from the **Window** menu, click **Show View**, click **Navigation\Content Store**, and then click **OK**.

Dismiss any error messages that may appear, and then close and reopen DQA. If you are prompted to log on at this point, specify the LDAP_Dev namespace, a user ID of admin, and a password of Education1. Also specify that you want to save the selected namespace, user ID, and password.

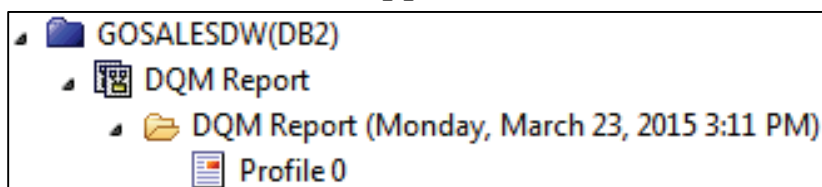
The results appear similar to the following:



Task 2. Review the results of a report that was run, in IBM Cognos Dynamic Query Analyzer.

- On the Content Store tab, if necessary, click the **Refresh** button, expand the **GOSALESDW(DB2)** entry (click the entry, then click > to the left of the entry), and then expand the **DQM Report\DQM Report (<Date>, <Time>)** entries.

A section of the result appears as follows:



- Double-click the **Profile 0** entry, and then in the right pane, click the **Summary** tab.

The analysis will load, and the runtime log is displayed in a graphical view.


The results in the Summary tab appear similar to the following:

Summary

Name: DQM Report

Package: GOSALESDW(DB2)

Time stamp: Monday, March 23, 2015 3:11:32 PM


 Relational


Timing


Total Time: 1240.968 ms

Working Times: 365 XValueDecoration (15.625 ms)

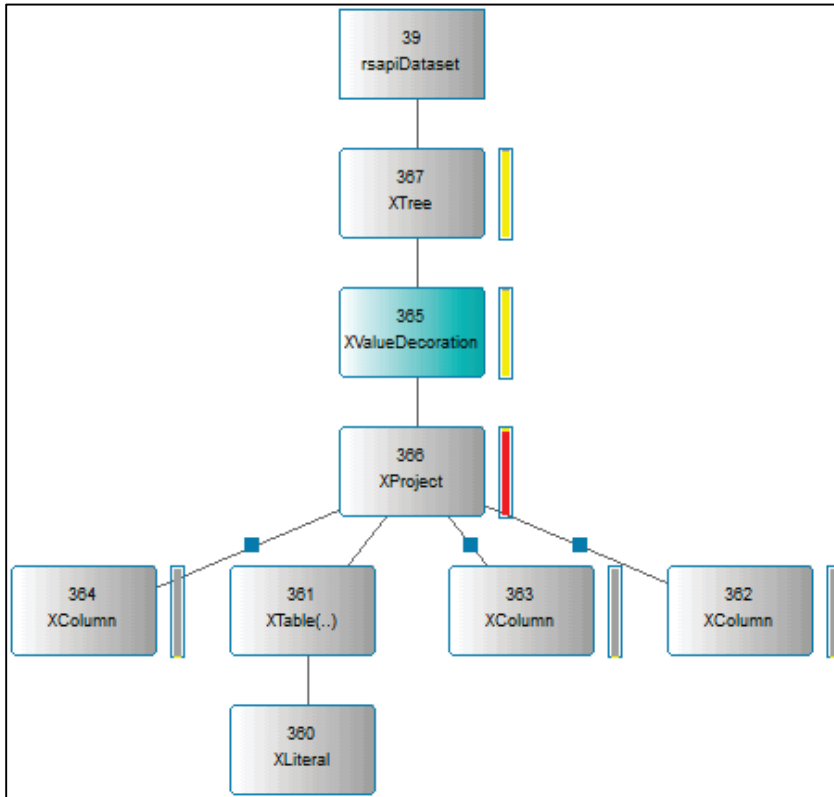
Wait Times: 366 XProject (1230.4697 ms)
365 XValueDecoration (5.126742 ms)
362 XColumn (0.006425 ms)
363 XColumn (0.003911 ms)
364 XColumn (0.003632 ms)

 **Total time in node children (working and waiting)**

 **Overall report time not in node**

 **Total time in node (working and waiting)**

The results in the middle pane appear similar to the following:



- Maximize the window, and adjust the width of the **Summary** tab layout to allow you to see the information.
- On the **Summary** tab, expand the **Node Shapes and Colors** section, to assist you in interpreting the results.
- Review the **Timing** section, and note the total time in node children and total time in node color annotations, and summary of working times and wait times.
- Refer to the **DQA Graph Legend** page earlier in the content of this module, as you review the objects in the graph, to assist in your interpretation of the graph.

Nodes can be SQL execution nodes, MDX execution nodes, local processing nodes, decoration nodes, and many more. Results flow from the bottom of the run tree (leaf nodes) to the top (XTree node) where the result is represented in RSAPI (Resultset API) format and can be sent to the report service for rendering.

- Click the **XTree** node object in the graph, to select it.

- Review the properties by clicking the **Properties** tab, and then select the other objects, one by one, to familiarize yourself with the properties and the information that is displayed for each one.
- Notice the **XColumn** objects, and compare the timing results of each column.
- Close **IBM Cognos Dynamic Query Analyzer**, and any other open windows.

Task 3. Create report variations.

You want to create reports with variations, so that you can compare the results in IBM Cognos Dynamic Query Analyzer. To do this, you will create three variations of a report. Typically, a report would be created, ran, and then the log files would be analyzed to determine if efficiencies could be found. Then, the updated report would be run and the log files analyzed to determine the difference in timing. In the interest of time you will create three reports, save them, run them to generate the log files, and then in the next task, you will open the log files, and compare results of all three.

- Launch **Internet Explorer**, go to **http://vclassbase:88/C10Full**, and then log on to the **LDAP_Dev** namespace with **admin/Education1** credentials.
- Launch **IBM Cognos Connection**, on the toolbar click **New Folder**, in the **Name** box type **Suppression Reports**, ensure that the location is set to **Public Folders**, and then click **Finish**.

You will save the reports to this public folder, rather than a private folder, so that you can access them from the Content Store pane in IBM Cognos Dynamic Query Analyzer. You will proceed to open the package to use in your reports.


- Navigate to **Public Folders\Samples_ DQ\Models\GO Data Warehouse (analysis)**, and then from the **Launch** menu, click **Report Studio**.
- Click **Create New**, and then double-click **Crosstab**.
- On the **Source** tab, expand **Sales and Marketing (analysis)**, and then expand **Sales**.

You will build a report from items in this namespace.

- Populate the crosstab by dragging the following source items to the crosstab in the report layout:
 - **Measures** drop zone: Sales fact**Quantity**
 - **Rows** drop zone: **Products** dimension**Products** hierarchy**Product** level
 - **Columns** drop zone:
 - **Retailers** dimension**Retailers** hierarchy**Retailer name** level
 - **Order method** dimension**Order method** hierarchy**Order method type** level, nested under <#Retailer name#>

The crosstab appears as follows:

Quantity	<#Retailer name#>		<#Retailer name#>	
	<#Order method type#>	<#Order method type#>	<#Order method type#>	<#Order method type#>
<#Product#>	<#1234#>	<#1234#>	<#1234#>	<#1234#>
<#Product#>	<#1234#>	<#1234#>	<#1234#>	<#1234#>



- On the toolbar, click the **Suppress**  arrow, and then click **Suppression Options**.
- In the **Suppression** section, click **Rows and columns**, in the **Suppress the following** section, ensure that all check boxes are selected, and then click **OK**.
- From the **File** menu, click **Save As**, name the report **Crosstab with Full Suppression**, and then save the report in **Public Folders\Suppression Reports**.

What if the author was to decide that it is not full suppression that was required, but merely suppression of null (missing) values? You will make the changes to the report and save the new variation to analyze later.

- On the toolbar, click the **Suppress** arrow, and then click **Suppression Options**.
- In the **Suppress the following** section, ensure that only the **Missing values** check box is selected, and then click **OK**.

- From the **File** menu, click **Save As**, name the report **Crosstab with Null Suppression**, and then save the report in **Public Folders\Suppression Reports**.

What if the author was to decide that suppression of null (missing) values could be run in the query directly? You will make the changes to the report and save the new variation to analyze later.

- On the toolbar, click the **Suppress** arrow, and then click **No Suppression**.
- On the **Explorer** bar, to the left of the report layout, click **Query Explorer** , and then click **Query1**.
- From the **Data Items** pane, drag **Quantity** to the **Detail Filters** pane.
- In the **Detail Filter Expression** pane, in the **Expression Definition** pane, click the cursor at the end of the expression, type **is not null**, and then click **Validate** .

The [Quantity] is not null expression has no errors.

- Click **OK**.
- From the **File** menu, click **Save As**, name the report **Crosstab with Null Suppression in Query**, and then save the report in **Public Folders\Suppression Reports**.
- Close **Report Studio**.

Task 4. Run the reports in IBM Cognos Dynamic Query Analyzer to generate log files for analysis.

- Launch **IBM Cognos Dynamic Query Analyzer**, if prompted log in with **LDAP_Dev/admin/Education1** credentials, and then from the **File** menu, click **Close All Results**, if it is available, to ensure that all previous results are closed.

If you encounter any `NullPointerException` errors, close IBM Cognos Dynamic Query Analyzer and reopen it.

- Click the **Content Store** tab, click **OK** to accept the Login settings (if necessary), and then expand the **Suppression Reports** folder to see the reports that you created.

- Double-click the **Crosstab with Full Suppression** report, click **OK** to close the **Login to Cognos Server** message, and then log in to the portal with **LDAP_Dev/admin/Education1** credentials.

Before you run the first report, you want to clear the query service cache.

- Return to the browser window where **Admin Person** is logged into **IBM Cognos Connection**, and then from the **Launch** menu, click **IBM Cognos Administration**.
- On the **Configuration** tab\ **Query Service Caching** pane, in the **Server Group(s)** column, click the **Group 64** check box to select it, and then click **Clear cache**.
- When the cache has succeeded in clearing, click **Close**.
- Switch to the **IBM Cognos Query Analyzer** window, in the **Content Store** tab, right-click the **Crosstab with Full Suppression** entry, and then click **Run Report**.

The report executes, and displays the results on the report tab in the middle.

- In the **Open Logs** dialog box, click **No**, as you will open the log later.

You want to clear the cache before running a different report, to ensure that results are not skewed by cached information.

- Return to the browser window, where **Admin Person** is logged into **IBM Cognos Administration**, and repeat the previous steps to clear the **Query Service** cache.
- Switch to the **IBM Cognos Dynamic Query Analyzer** window, and repeat the previous steps to run the **Crosstab with Null Suppression** report and clear the query service cache.
- Switch to **IBM Cognos Dynamic Query Analyzer**, and repeat the previous steps to run the **Crosstab with Null Suppression in Query** report and clear the query service cache. It may take a while for this report to finish running.

By running the reports in IBM Cognos Dynamic Query Analyzer, even though query execution is not enabled in your environment, logs have been generated, for which you can analyze results.

Task 5. Review the logs from different report variations.

- In **IBM Cognos Dynamic Query Analyzer**, from the **File** menu, click **Close All Results**.
- From the **Window** menu, click **Analyze Logs**, from the **File** menu, click **Open log**, and then click **From URL**.

You can use this option if a virtual directory has been configured for the XQE folder. In this environment, all of the dynamic query logs are stored here.

- Click the **Crosstab with Full Suppression (<Date>, <Time>)** entry, click the **>** at the left of the entry to expand the list, and then in the expanded list, click the **Profile 0** check box to select it.


You only want to load the Profile 0 file for comparison analysis at this time. Other files that you may see in the log files are Fetch Cell Values and LoadLevels.

- Expand the **Crosstab with Null Suppression (<Date>, <Time>)** entry, and then in the expanded list, click the **Profile 0** check box to select it.
- Expand the **Crosstab with Null Suppression in Query (<Date>, <Time>)** entry, in the expanded list, click the **Profile 0** check box to select it, and then click **OK**.
- Click the **Crosstab with Full Suppression (<Date>, <Time>)** tab, to review the results.
- From the **Window** menu, click **Show View**, expand **Graph Details**, click **XML Node**, and then click **OK**.

As you explore each node, you can refer to this tab to view the XML associated with it.

- In the **XML Node** tab, click in the blank pane, and then in the **Crosstab with Full Suppression** pane, select the node objects, one by one, to familiarize yourself with the properties and the information that is displayed for each one.

Refer to the DQA Graph Legend page earlier in the content of this module, as you review the objects in the graph, to assist in your interpretation of the graph.

- For expandable nodes, double-click the node or click the  icon to expand and explore more nodes and properties in the runtime.

In this runtime, the MDX query (green nodes) are sent to an MDX data source for execution. The results returned will go through some decoration nodes (decoration is an internal process that allows the mode to distinguish between different parts of a query result), then a node that will flatten the result (MDX results are multidimensional by nature) and finally another node will process the flattened result.

The XMDXSelect node is the node which will display the pieces of the actual MDX query used to satisfy the report request. The scale icon beside node is used as a visual representation of the performance of the node.

- Click the **XMDXSelect** node to display the properties.

A section of the result appears similar to the following:

Property	Value
cellProperties	CELL_ORDINAL, FORMAT_STRING, VALUE
cubeName	go_data_warehouse
id	1609
type	XMDxSelect
Wait time in node	6509.2734 ms
Wait time in node and children	83137.14 ms
Working time in node	1671.875 ms
Working time in node and children	5531.25 ms

The properties pane of this node reveals the wait time spent in the node and the wait time in the node and children. In this case, the time spent in the node itself was 1671.875 ms and the cumulative time for the node and its children was 5531.25 ms.

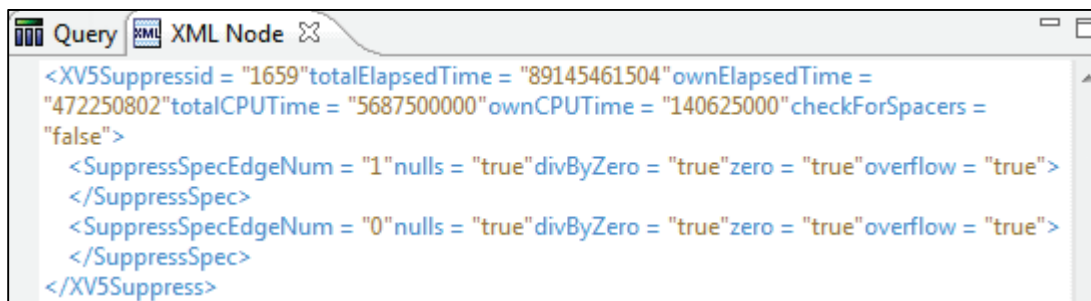
Your result times may be different, depending on what is happening in your environment. Compare each of the results of your findings against each other, in your analysis in this workshop.

- Click the **XV5Suppress** node near the top of the tree, and observe the properties.

Property	Value
checkForSpacers	false
id	1659
type	XV5Suppress
Wait time in node	331.62582 ms
Wait time in node and children	83457.96 ms
Working time in node	140.62498 ms
Working time in node and children	5687.5 ms

The XV5Suppress node is evoked by the application of the Suppress\Rows and Columns on the report.

- View the contents of the **XML Node** tab, while the **XV5Suppress** node is selected.
- Results appear similar to the following:



```

<XV5Suppressid = "1659"totalElapsedTime = "89145461504"ownElapsedTime =
"472250802"totalCPUTime = "5687500000"ownCPUTime = "140625000"checkForSpacers =
"false">
  <SuppressSpecEdgeNum = "1"nulls = "true"divByZero = "true"zero = "true"overflow = "true">
  </SuppressSpec>
  <SuppressSpecEdgeNum = "0"nulls = "true"divByZero = "true"zero = "true"overflow = "true">
  </SuppressSpec>
</XV5Suppress>

```

In this particular run, the time spent in the node itself was 140.62498 ms and the cumulative time for the node and its children was 5687.5 ms.

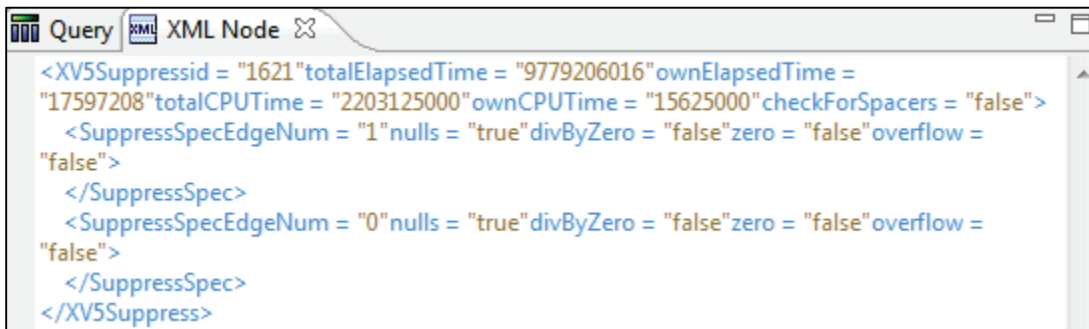
Notice the code for the suppression, and how all suppression values are true. Based on the fact that there are two EdgeNum entries and nulls, divByZero and overflow are all set to true, the properties confirm that a user applied zero, divide by zero, overflow and null suppression on both the rows and columns.

At this point, it would be time to ask the report author whether or not they actually need the `divByZero`, zero and overflow suppression, or whether or not they just wanted to suppress nulls. Another good question to ask would be whether suppression was actually needed on both the rows and columns.

The report author could come back with the statement that only null suppression is required on both columns and rows. You have made this change to the report and saved it as **Crosstab with Null Suppression (<Date>, <Time>)** on the same package, and you ran the report to generate log files.

- In **IBM Cognos Dynamic Query Analyzer**, click the **Crosstab with Null Suppression (<Date>, <Time>)** tab, and review the objects in the runtreen.
- Click the **XV5Suppress** node, and review the **XML Node** tab.

Results similar to the following appear:



In this particular run, the time spent in the node itself was 15.625 ms and the cumulative time for the node and its children was 2203.125 ms. The timing has changed significantly from the previous report design execution.

Notice the code for the suppression, and how not all suppression values are true, only those for null values. Based on the fact that there are two `EdgeNum` entries, the properties confirm that a user null suppression on both the rows and the columns.

For this report run only nulls are being suppressed on both the row and column edge of the crosstab. Since the requirement is to only suppress nulls, the visual null suppression can be replaced by the Null suppression on the actual query. At this point, it would be time to ask the report author to make this change to the report design and query. You have made this change to the report and saved it as **Crosstab with Null Suppression in Query (<Date>, <Time>)** on the same package, and you ran the report to generate log files.

- In **IBM Cognos Dynamic Query Analyzer**, click the **Crosstab with Null Suppression in Query (<Date>, <Time>)** tab, and review the objects in the runtime.
- In the **Graph** tab, notice that there is no XV5Suppress node, due to visual suppression being removed from the report, and running the suppression in the query.
- On the **Summary** tab, notice the **Timing** section.

The results appear similar to the following:

Timing	
Total Time:	8234.22 ms
Working Times:	1611 XMdxSelect (1109.375 ms) 1634 XMdxElement (843.75 ms) 1655 XMdxElement (781.25 ms) 1659 XMdxElement (62.5 ms)
Wait Times:	1634 XMdxElement (4925.7227 ms) 1655 XMdxElement (4294.196 ms) 1611 XMdxSelect (1352.7632 ms) 1659 XMdxElement (629.484 ms)

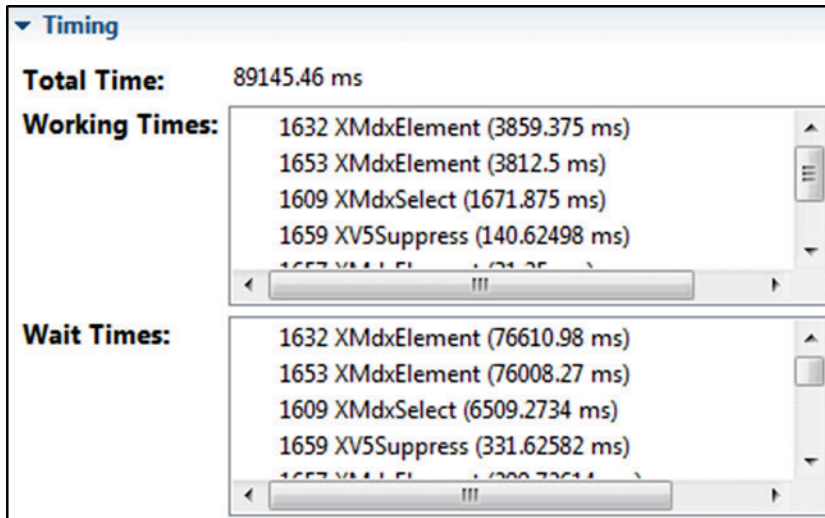
- In **IBM Cognos Dynamic Query Analyzer**, click the **Crosstab with Null Suppression (<Date>, <Time>)** tab, and review the **Timing** section results.

The results appear similar to the following:

Timing	
Total Time:	9779.206 ms
Working Times:	1571 XMdxSelect (1171.875 ms) 1594 XMdxElement (984.375 ms) 1615 XMdxElement (796.875 ms) 1619 XMdxElement (171.87502 ms) 1621 XV5Suppress (15.625 ms)
Wait Times:	1571 XMdxSelect (4134.761 ms) 1594 XMdxElement (3456.9849 ms) 1615 XMdxElement (2972.5955 ms) 1619 XMdxElement (487.07047 ms)

- Compare the results with the timing results for the initial report, **Crosstab with Full Suppression (<Date>, <Time>)**.

The results appear similar to the following:



You have compared three different variations of a report, using IBM Cognos Dynamic Query Analyzer, and the results have revealed that the Crosstab with Null Suppression in Query report is the most efficient report for users.

- Close **IBM Cognos Dynamic Query Analyzer**, log out from **IBM Cognos Administration** and then close the web browser
- From the **Taskbar**, launch **Services**, and then stop the **IBM Cognos 10 Full:9315** service for the next module demo.

