#SHAREorg

SHARE

Using the IMS Universal Drivers and QMF to Access Your IMS Data - Hands-on Lab

Suzie Wendler
Kenny Blackman
IBM

Wednesday, March 12, 2014:12:15 PM - 01:15 PM
#14782

SHARE in Anaheim

Using the IMS Universal Drivers and QMF to Access Your IMS Data Hands-on Lab

Overview

QMF for Workstation is an Eclipse-based, rich client desktop Java application, that uses JDBC to connect to data sources to provide querying, reporting and Business Intelligence (BI) solution development and execution capabilities.

This hands-on lab covers how to use QMF for Workstation to access IMS DB using the IMS Universal Driver.

QMF can be used

- Allow users to graphically construct ad-hoc IMS queries
- Create reports and dashboards that draw directly from IMS data
- Roll out web-based graphical content that blends IMS data with relational and multi-dimensional data sources

The lab exercises cover the following topics:

1. Installing the IMS driver

2. Creating a personal repository.

3. Working with queries.

4. Developing reports using QMF forms.

5. Defining virtual data sources.

Exercise 1: Configuring the IMS JDBC Driver

QMF for Workstation uses JDBC drivers to connect to data sources. The product does not include the actual JDBC driver files. Administrators must define the location of the JDBC driver files.

How to get the IMS Universal Drivers

The IMS Universal Drivers are shipped with IMS.

The IMS distribution libraries (DLIBs) contain the master copy of elements in IMS and can be used to restore SYSMODs in the target library or to rebuild a target environment. These data sets are maintained by SMP/E.

The IMS.ADFSJHFS: ADFSJHFS contains the type-2 and type-4 Universal driver Java class libraries used for IMS DB access through the JDBC and DLI for Java interfaces.

The TLIB data sets are the IMS SMP/E target libraries (SYSLIBs), and are the libraries that

are used to run and use IMS.

The following data sets that reside in a UNIX System Services (USS) file system are also maintained by the SMP/E APPLY processing:
SDFSJCPS
SDFSJTOL
SDFSIC4J
SDFSJCIC
SDFSJCPI
SDFSJHFS
SDFSJRAR
SDFSJSAM

The IMS HFS data sets contain
SDFSJCIC: Maps to PathPrefix/usr/lpp/ims/imsnn/imsjava/classic/cics/IBM/
SDFSJHFS: Maps to PathPrefix/usr/lpp/ims/imsnn/imsjava/IBM/
SDFSJSAM: Maps to PathPrefix/usr/lpp/ims/imsnn/imsjava/ivp/IBM/
SDFSJRAR: Maps to PathPrefix/usr/lpp/ims/imsnn/imsjava/IBM/
SDFSJCPI: Maps to PathPrefix/usr/lpp/ims/imsnn/imsjava/classic/IBM/
SDFSJTOL: Maps to PathPrefix/usr/lpp/ims/imsnn/imsjava/classic/dlimodel/IBM/
SDFSJCPS: Maps to PathPrefix/usr/lpp/ims/imsnn/imsjava/classic/classic/ivp/IBM/
SDFSIC4J: Maps to PathPrefix/usr/lpp/ims/imsnn/ico/IBM/

The IMS Universal JDBC driver (**imsudb.jar**) is used to make SQL calls with the JDBC API and can be download as a binary file from HFS path: PathPrefix/usr/lpp/ims/ims**nn**/imsjava/IBM/ where **nn** is the IMS version you have installed..

For this lab the imsudb.jar has already been downloaded and can be found at

C:\share\anaheim2014\lab

Creating the IMS Universal Driver JDBC driver configuration file

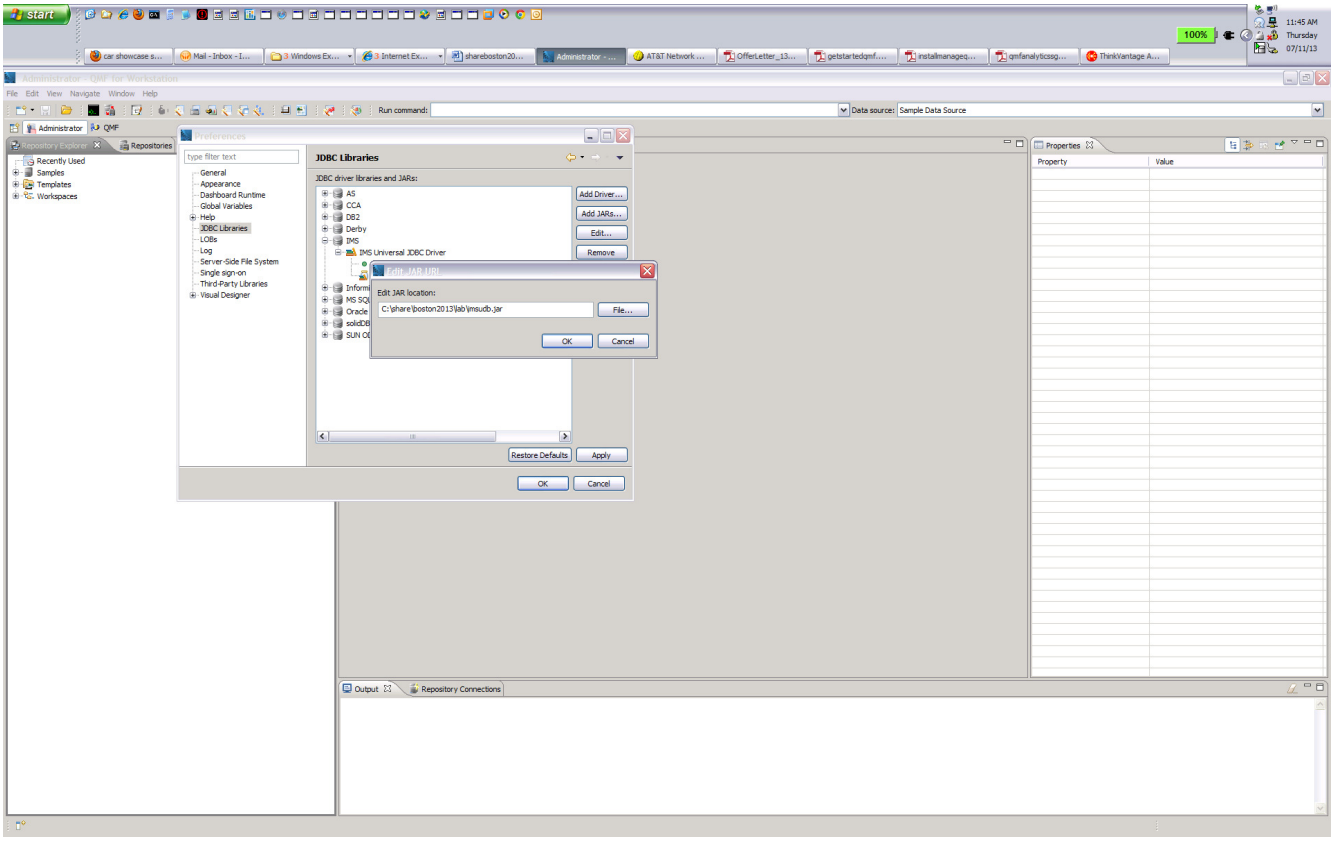
Launch QMF for Workstation via the shortcut on the desktop or via the Windows Start Menu. You'll find the application link under 'DB2 Query Management Facility'.

Open the Administrator perspective if you have not already done so. To open the Administrative perspective go to the menu pane and select:
Window > Open perspective > Other > Administrator.

Select **Preference**s from the **View** menu to open the Preferences window. Select **JDBC Libraries**. The JDBC Libraries page opens. QMF supplies pre-populated libraries named for specific databases including IMS. To specify where to find the associated

JAR files:

a. Expand the library that contains the IMS JDBC driver class name to which you want to add JAR file location information.
b. Select **Add JARS**. The Add JARS to [libraryname] window opens.
c. Search for and select the IMS JDBC driver file that you want to add.
d. Select **Open.** The location of the IMS JDBC driver JAR file is saved in the JDBC library that you have selected. Select **OK.**



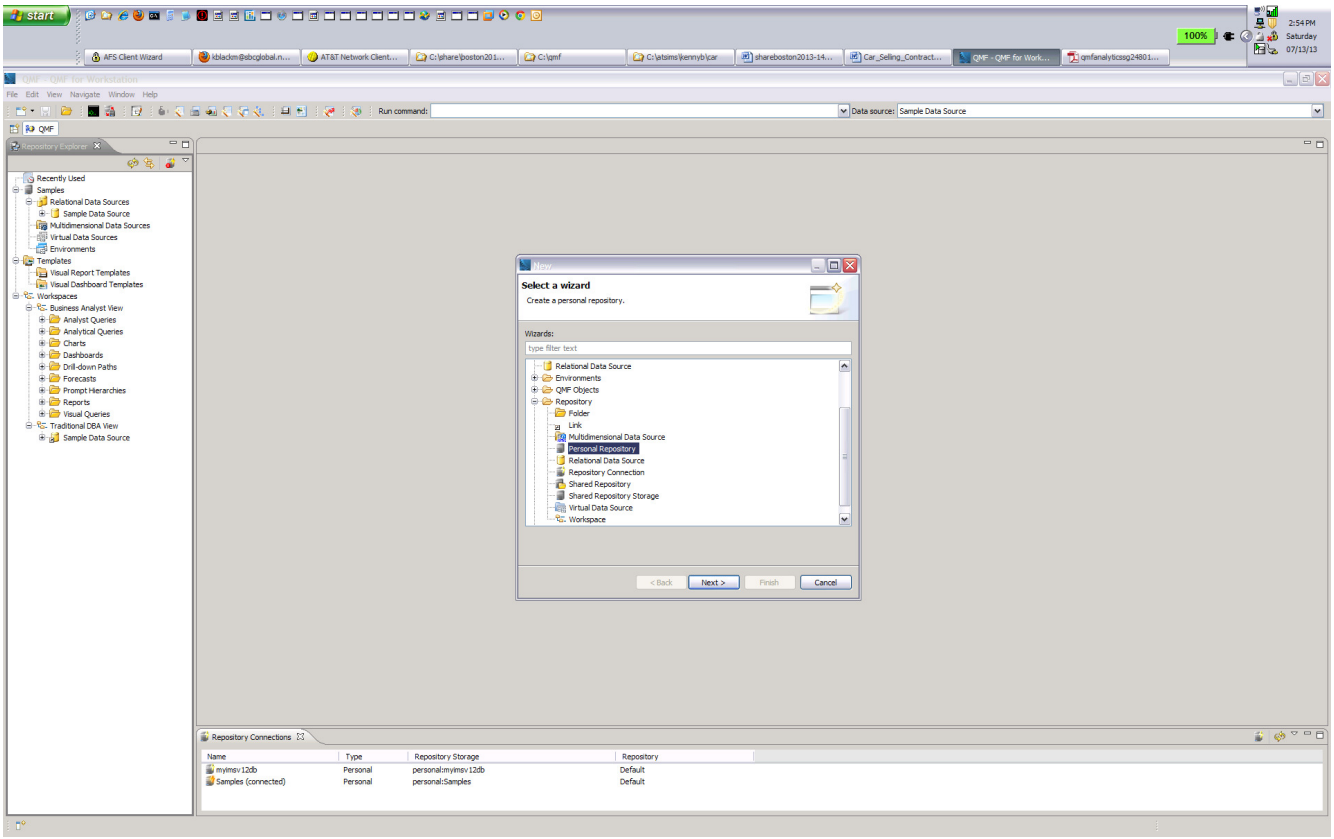
This completes the IMS JDBC Driver installation.

Exercise 2: Create a personal repository

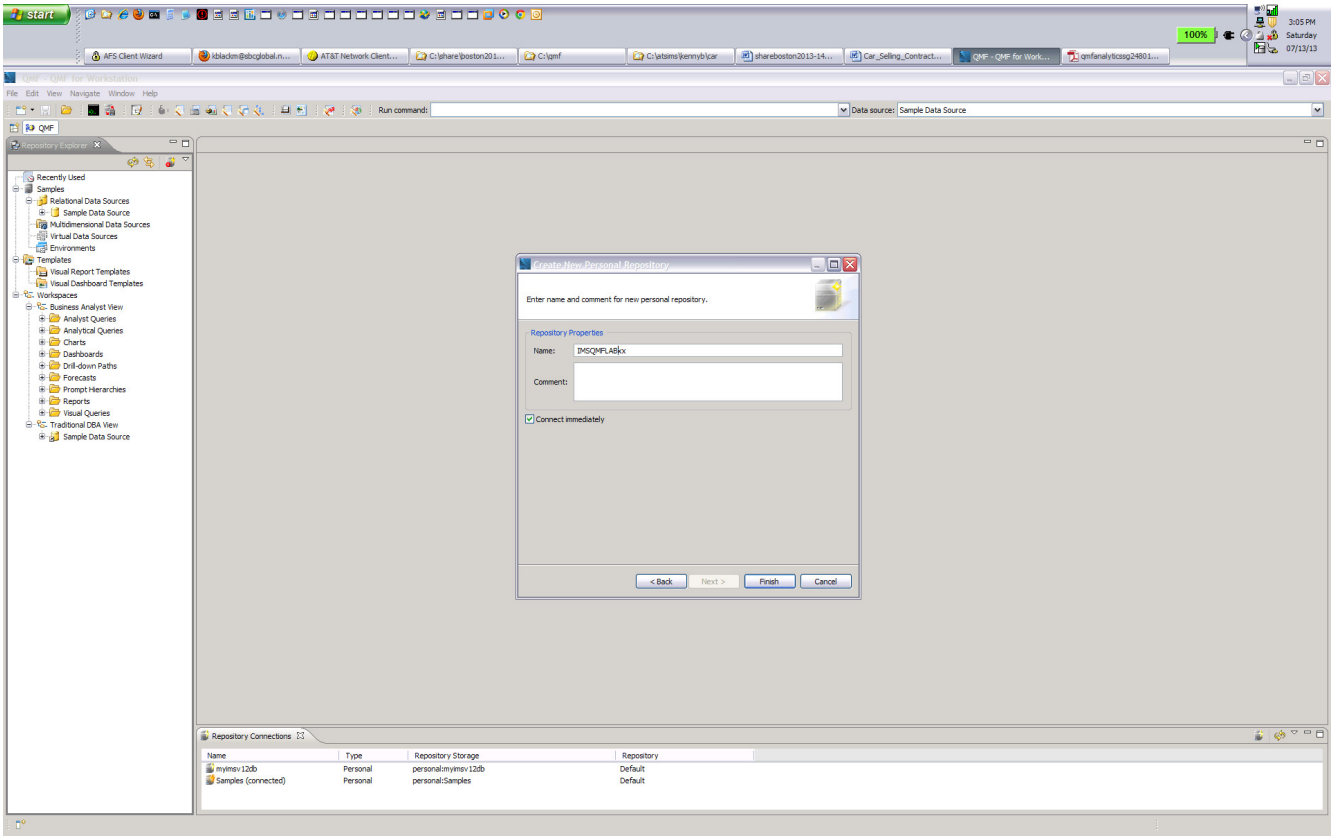
A personal repository is a set of database tables that stores QMF for Workstation objects, such as queries and dashboards, as well as database connection information, QMF configuration information, and application data.

To set up a personal repository, follow these steps:

1. Select **File ->✹ New -> other** and then expand **Repository.** Choose **Personal Repository.**



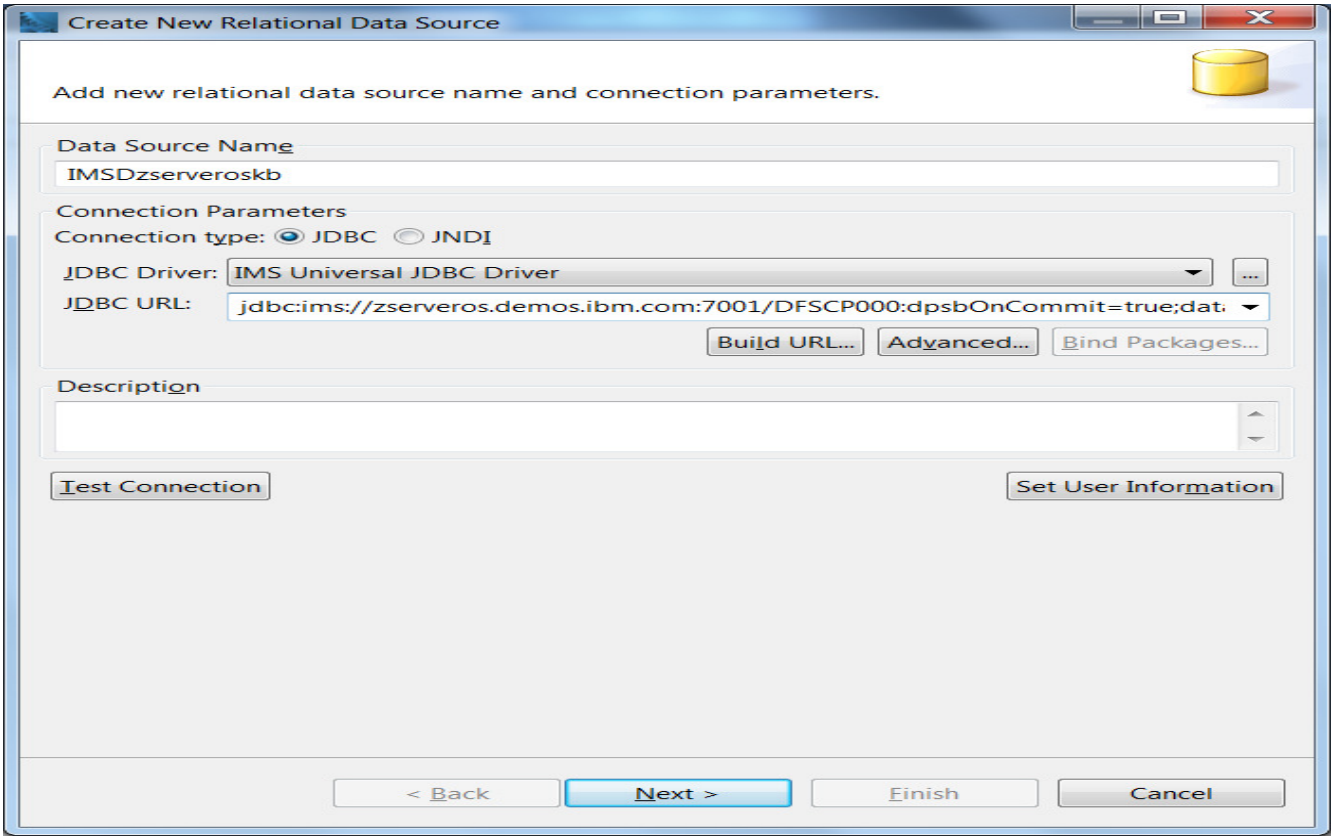
Select **NEXT** and enter a name. For example **IMSQMFLAB**xx where xx is your team number.



Click **FINISH.**

The I**MSQMFLAB**xx Personal Repository connection information needs to be updated to connect to a repository:

Expand **IMSQMFLAB**xx. Right click **Relational Data Source - > NEW - > Relational Data Source**



In the **Data Source Name** field enter **IMSDzseveros***nn* where *nn* is your team number.

Set the connection type button to **JDBC.**

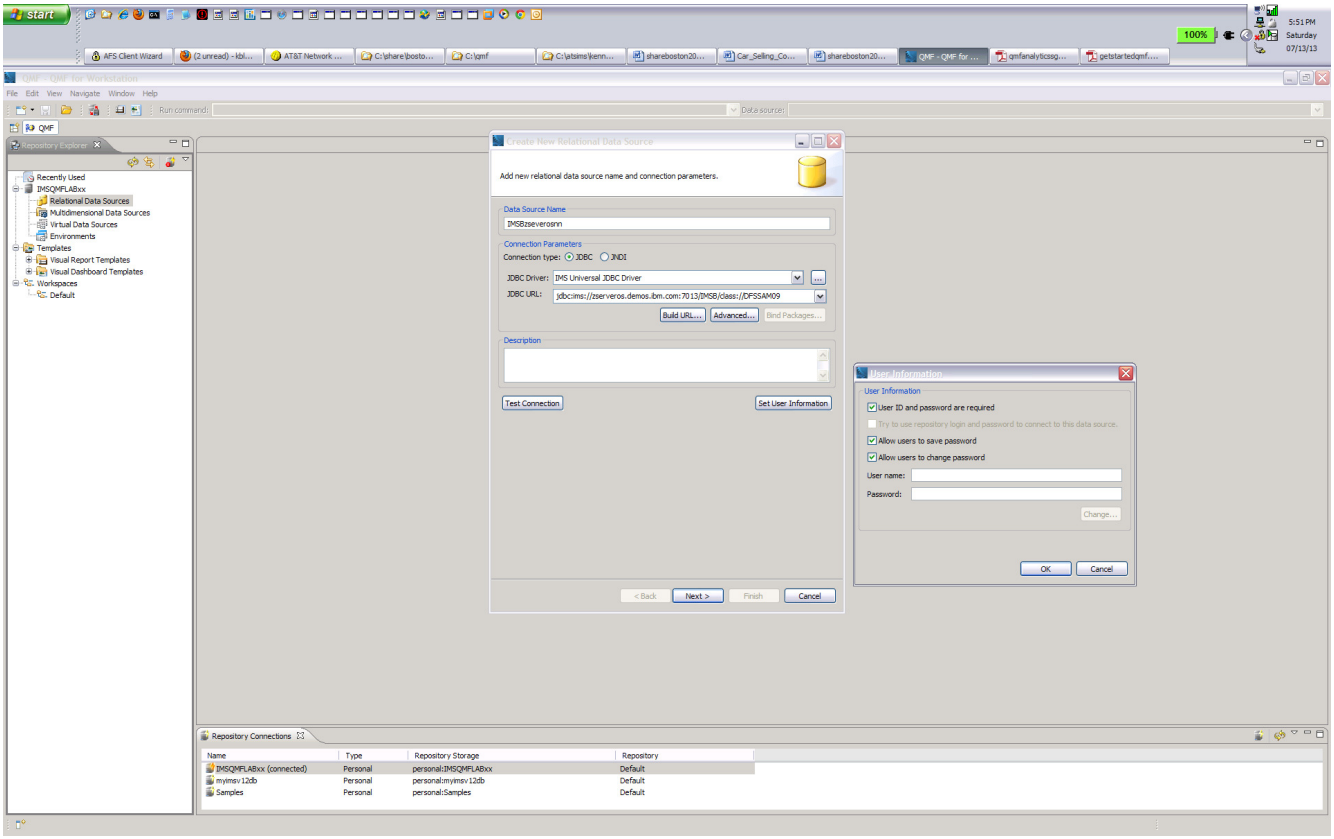
In the JDBC Driver drop down list select **IMS Universal JDBC Driver.**

Enter the following for the JDBC URL to access the IMS 12 Catalog metadata:

```
jdbc:ims://zserveros.demos.ibm.com:7001/DFSCP000:dpsbOnCommit=true;datastoreName=IMSD;
```

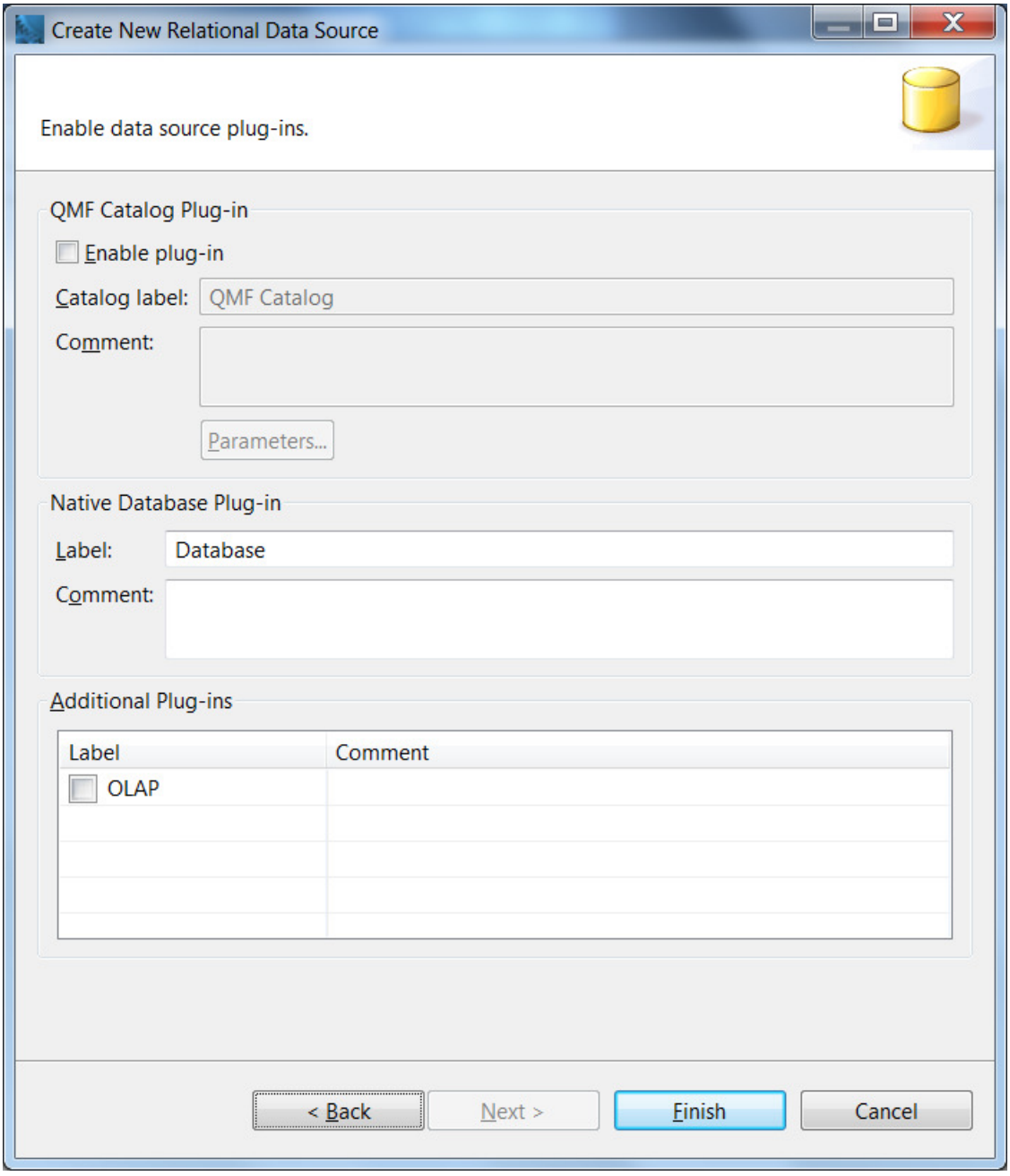
Select **NEXT.**

(DFSCP000 is the PSB for the IMS catalog on an IMS 13 environment).



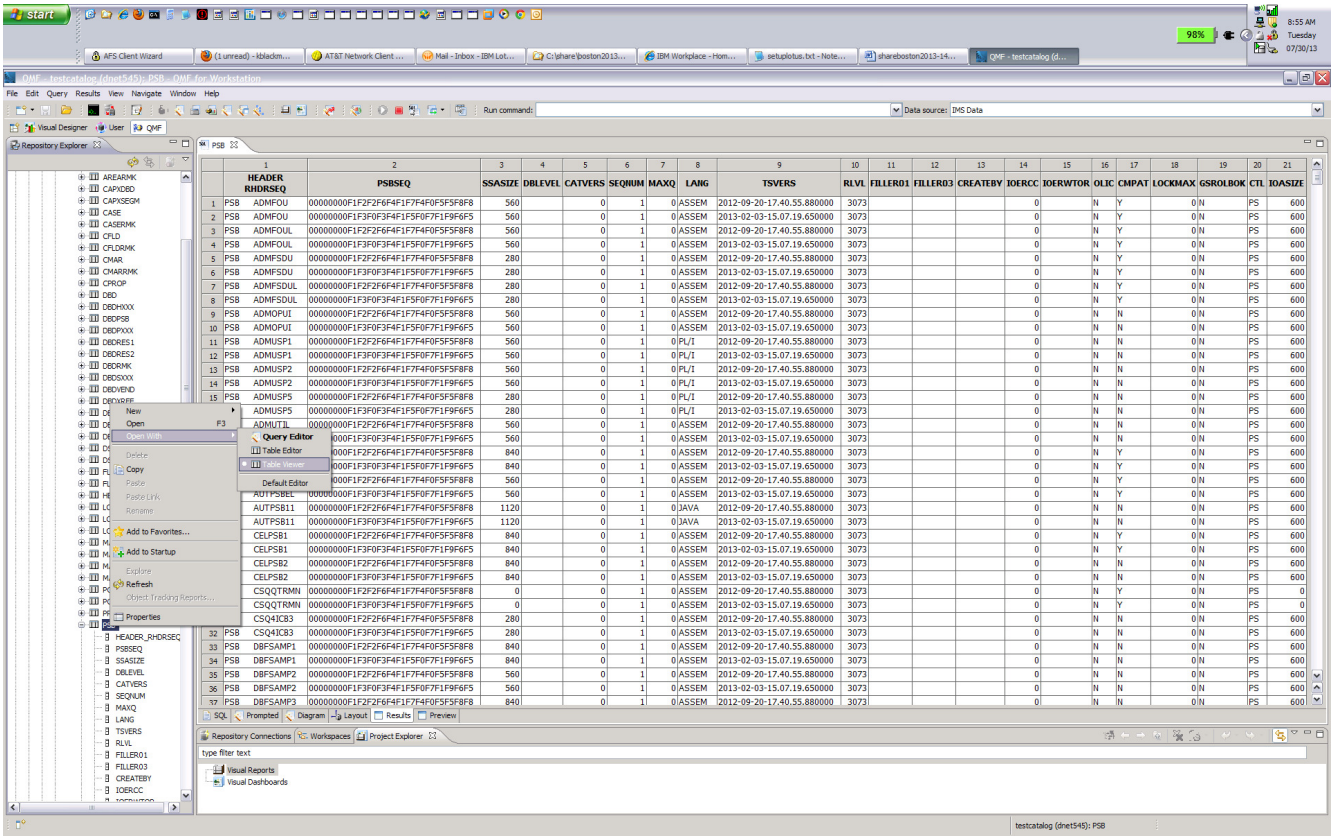
In the **userid** field enter **IMPOT**xx where xx is your team number. In the **password** field enter **ibm08pot**. Click **OK.**

The following screen may appear – Click Finish



This should allow you access to the IMS 13 system with a catalog.

Select PSB table and right click then use Open with Table Viewer to get the following output.



Create another Data Source connection, this time for access to PSB **DFSIVP37.**
Right click **Relational Data Source NEW -> Relational Data Source**

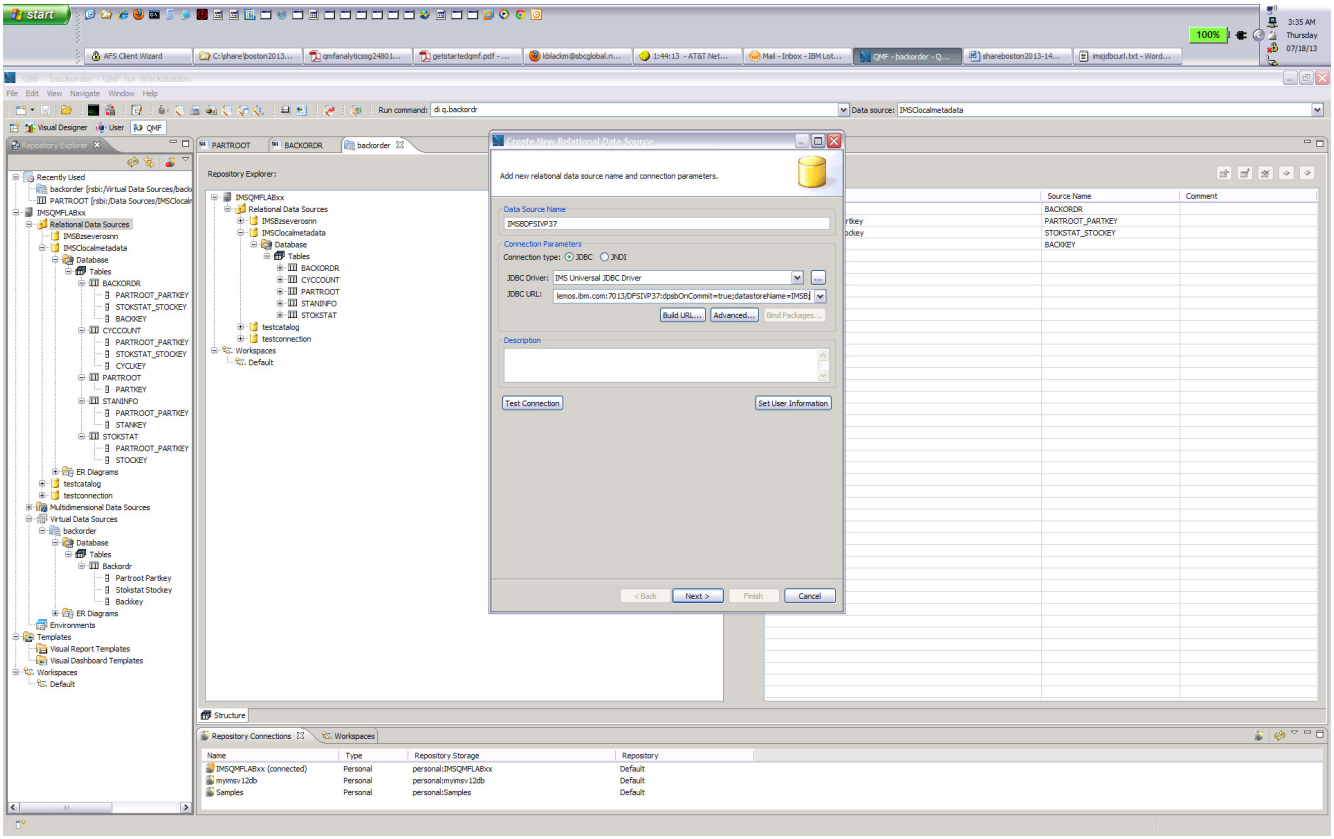
In the Data Source Name field enter **IMSDDFSIVP37**

Set the connection type button to **JDBC.** In the JDBC Driver drop done list select **IMS Universal JDBC Driver.** Enter the following for the JDBC URL to access the DFSIVP37 local file metadata:

```
jdbc:ims://zserveros.demos.ibm.com:7001/class://dfsivp37.DFSIVP37DatabaseView:dbVie
wLocation=C:/share/IMS Universal Drivers Metadata;fetchSize=0;
```

Select **NEXT.**

(Note that in this case, we are using the metadata that is stored in a local file). The IMS system we are going to is an IMS 13 system for this test.



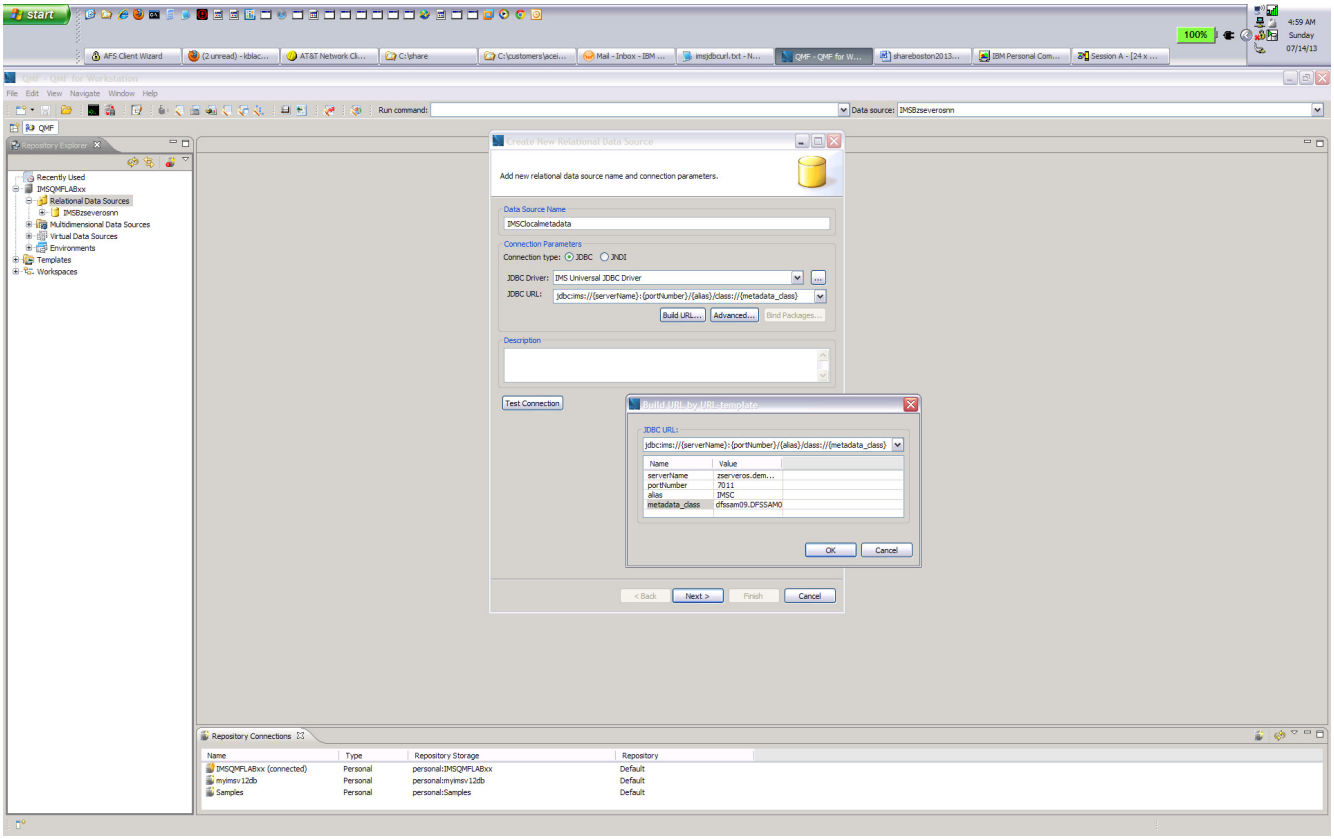
In the userid field enter I**MPOT***xx* where *xx* is your team number
In the password field enter **ibm08pot.** Click **OK.**

Create another **IMSQMFLAB***xx* Personal Repository connection. The URL for this connection also uses a local metadata file:

Right click **Relational Data Source NEW -> Relational Data Source** .
In the **Data Source Name** field enter **IMSBlocalmetadata.**

Set the connection type button to **JDBC.** In the **JDBC Driver** drop down list select **IMS Universal JDBC Driver.** In the **JDBC URL** drop down list enter the following:

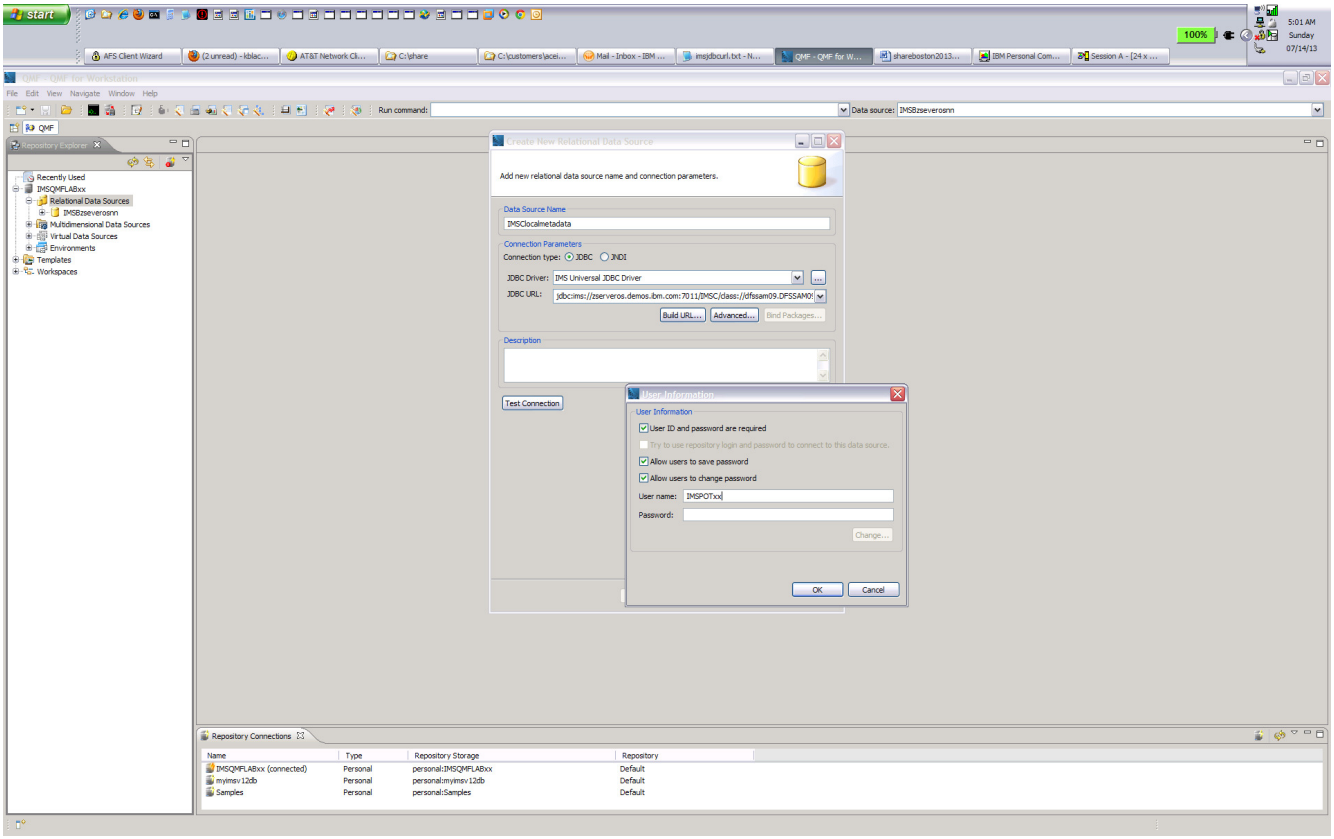
```
jdbc:ims://zserveros.demos.ibm.com:7013/class://dfssam09.DFSSAM09DatabaseView:dbViewLocation
=C:/share/IMS Universal Drivers Metadata;fetchSize=0;
```



Select **OK** to close the template.

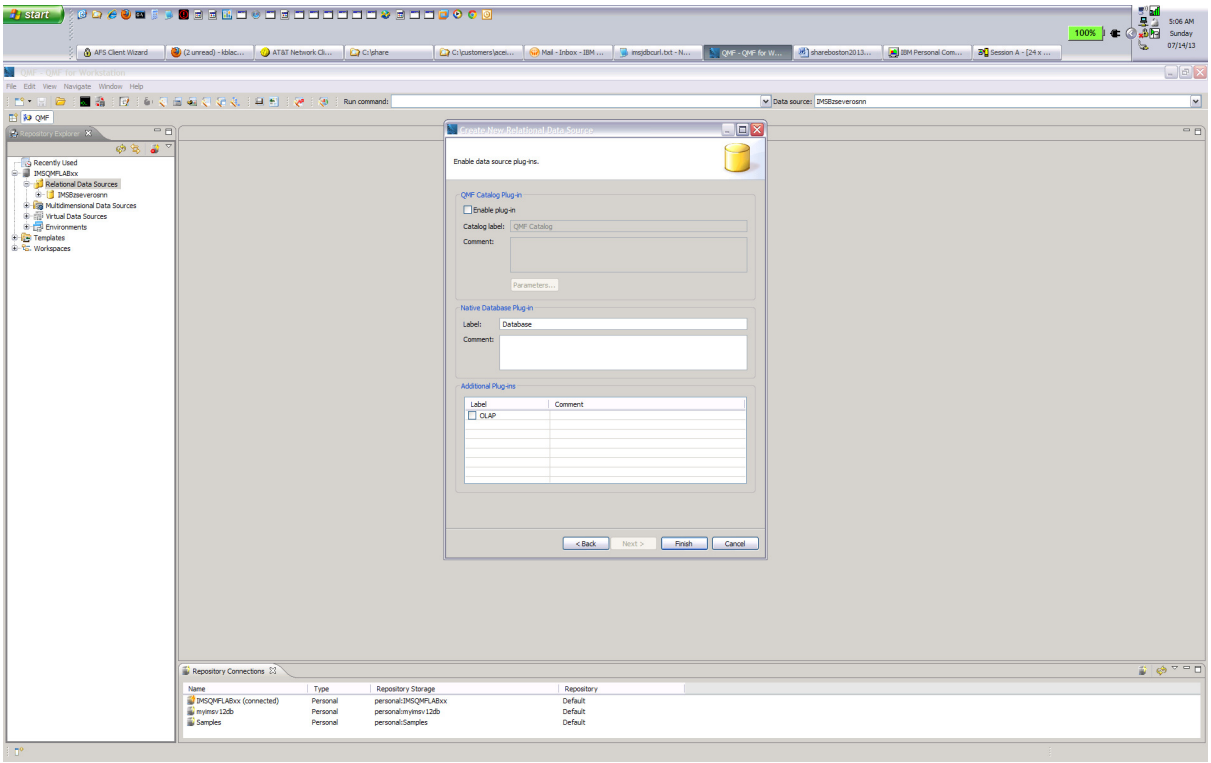
Select **NEXT.**

In the **userid** field enter **IMPOT***xx* where *xx* is your team number
In the **password** field enter **ibm08pot.**



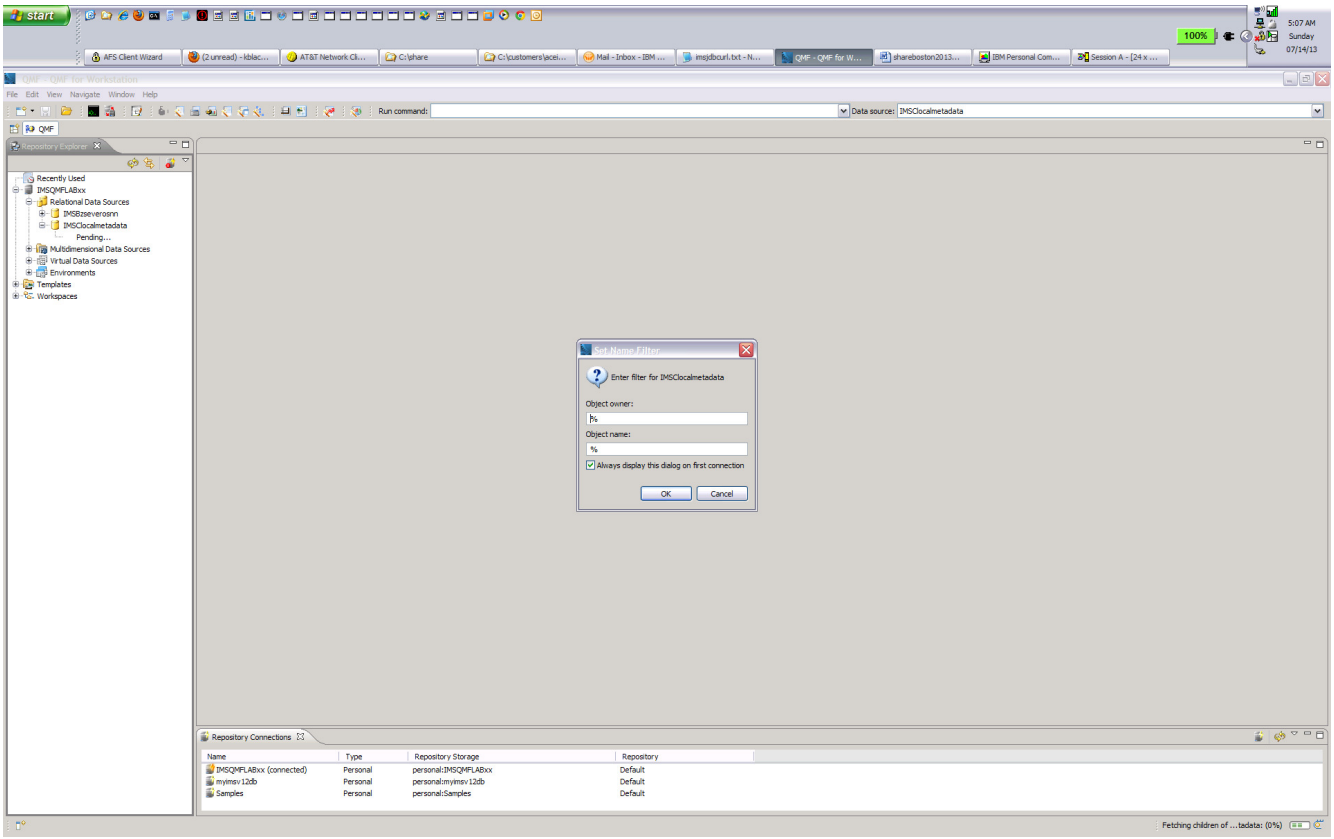
Select **OK** to close the template.

Select **NEXT** to get following screen:



Select **FINISH.**

Select **IMSBlocalmetadata** and select **OK** for the following screen:



In the **Repository Explorer** you will see the tables in the IMS PARTS DB.
This completes access setup to IMS.

Exercise 3 – Working with queries

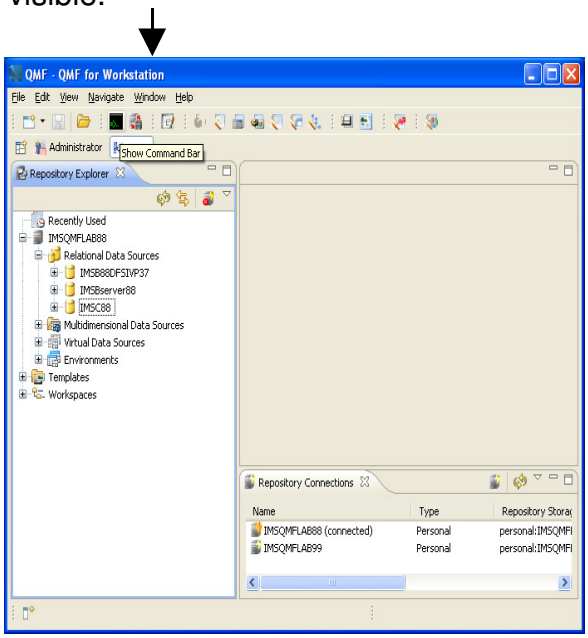
This section introduces you to the query development facilities in QMF for Workstation.

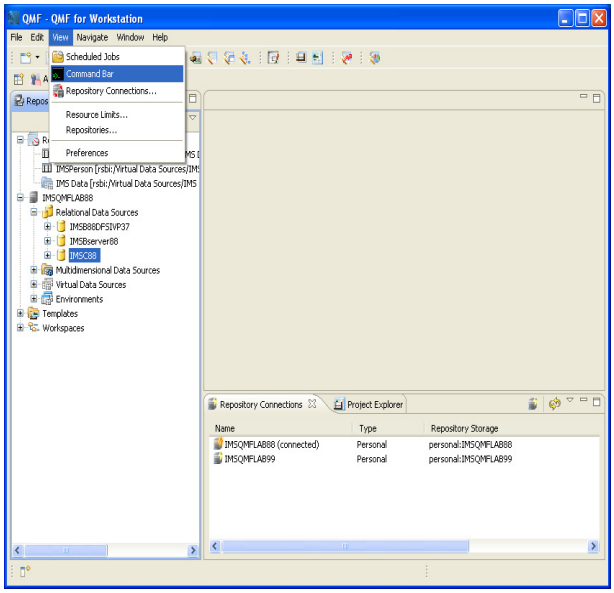
There are a number of ways to create a new query, including:

- Browsing through your database structure and double-clicking on a given table.
- Using the File->New->Query menu item or 'New Query' toolbar button.
- Using the QMF command bar to directly display a given table with a default query.
- Using the 'Draw Query' toolbar button.
- Clicking on tables that have been arranged in an arbitrary folder structure in your QMF workspace.

In our case, we will start by using the QMF command bar.

1. Click on the 'Show Command Bar' toolbar button (fourth from the left). The QMF command bar visibility is toggled. Click on the button such that the command bar is visible.

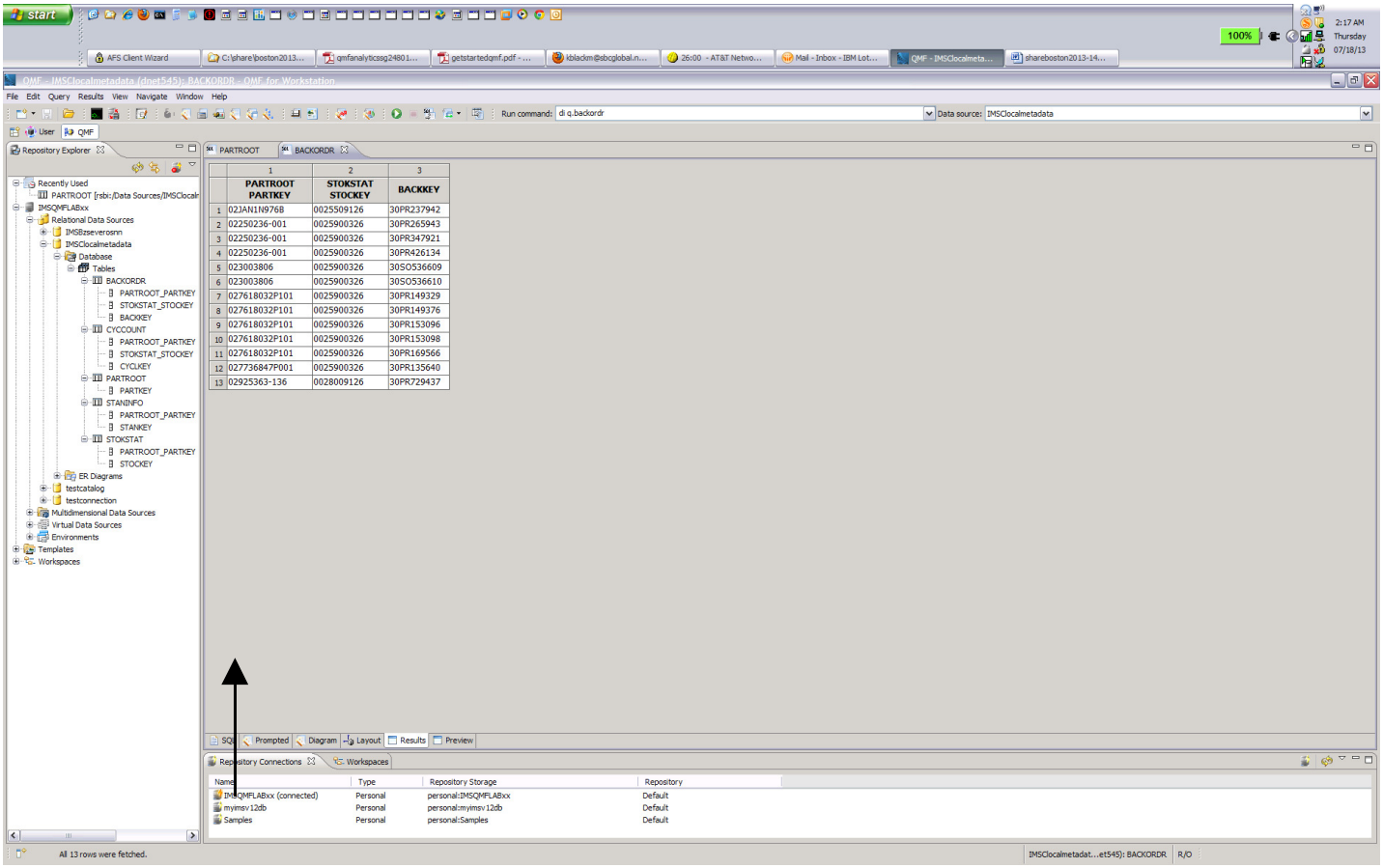




Alternatively, you can click on **View** -> **Command Bar**

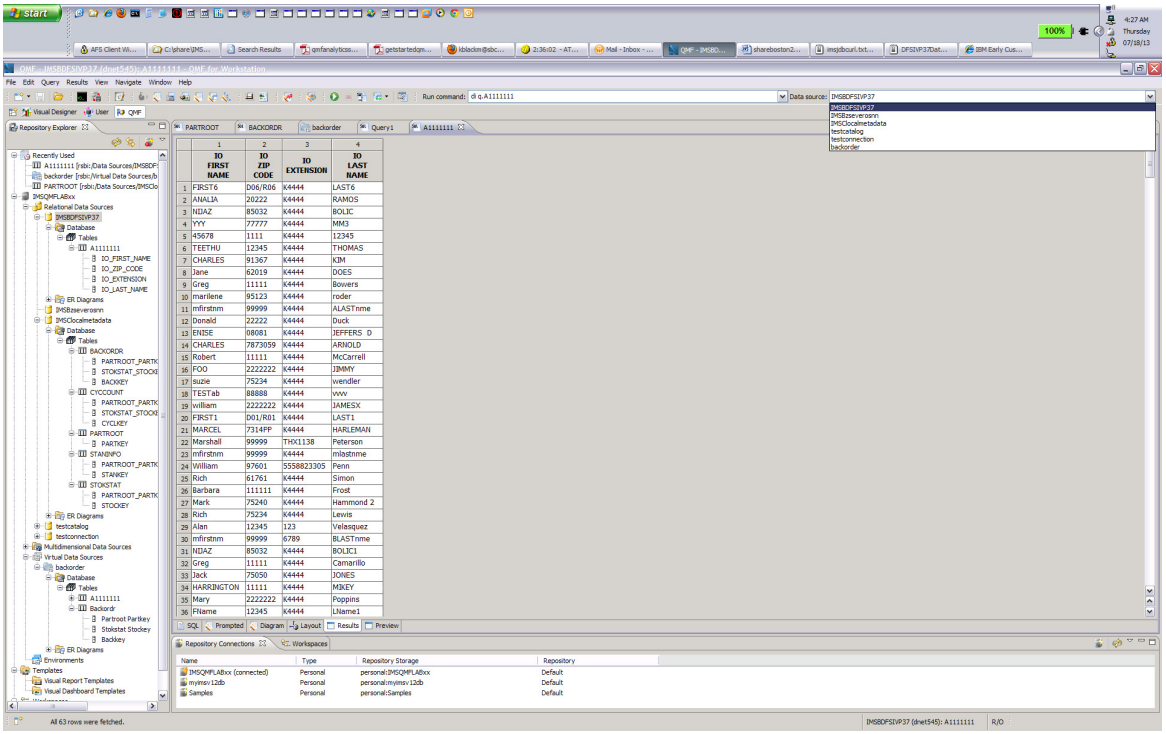
The command bar accepts QMF procedure commands. You only need enough letters from the command's name to allow QMF for Workstation to distinguish it from other procedure commands. We will use the DISPLAY command but only need enter **'di'** since DISPLAY is the only command that starts with 'di'.

2. Enter **di q.backordr** into the **Run** command field and on the right in the **Data Source** field, choose the one that you last created which has the table, e.g., IMSBlocalmetadata. Press enter. QMF for Workstation creates a default query and runs it.



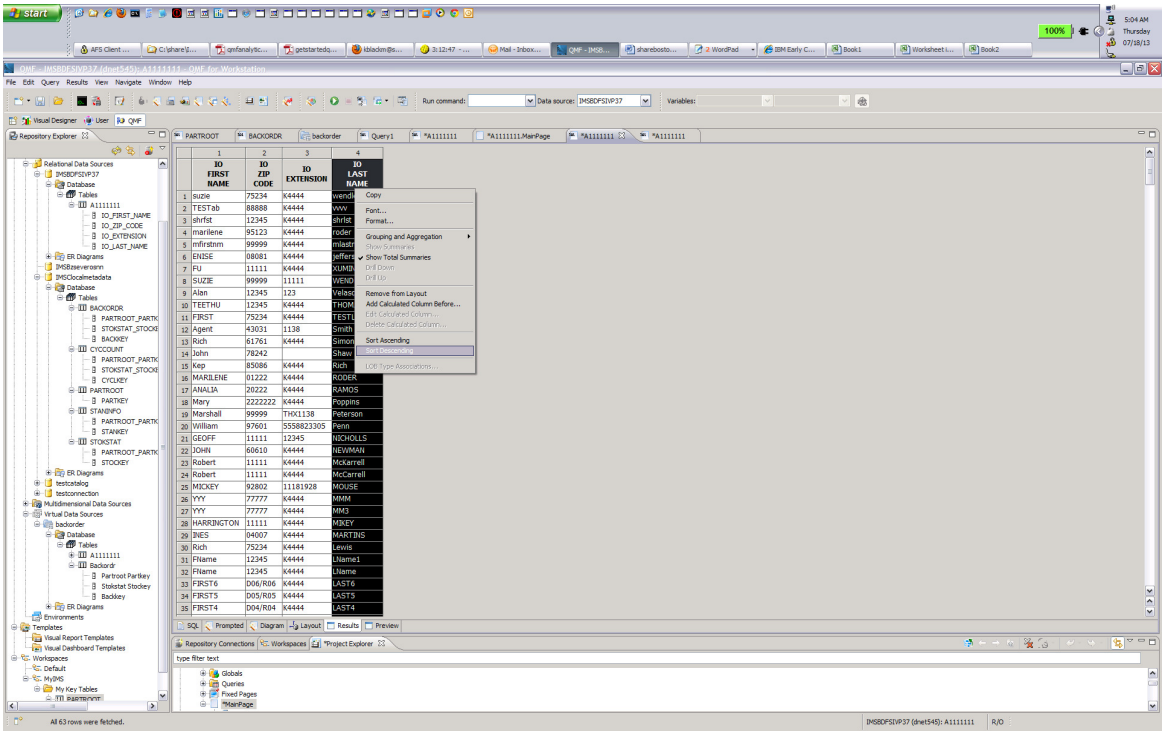
2. Review the SQL that was generated by clicking on the SQL tab toward the bottom.

3. Try another query. Enter **di q.A1111111** into the **Run command** field and in the Data source: drop down box select **IMSDDFSIVP37** then press enter. QMF for Workstation creates a default query and runs it.



4. Sort the Results.

Right click on the **IO LAST NAME** column and select either **Sort Ascending** or Sort **Descending** to view the results in a different order.

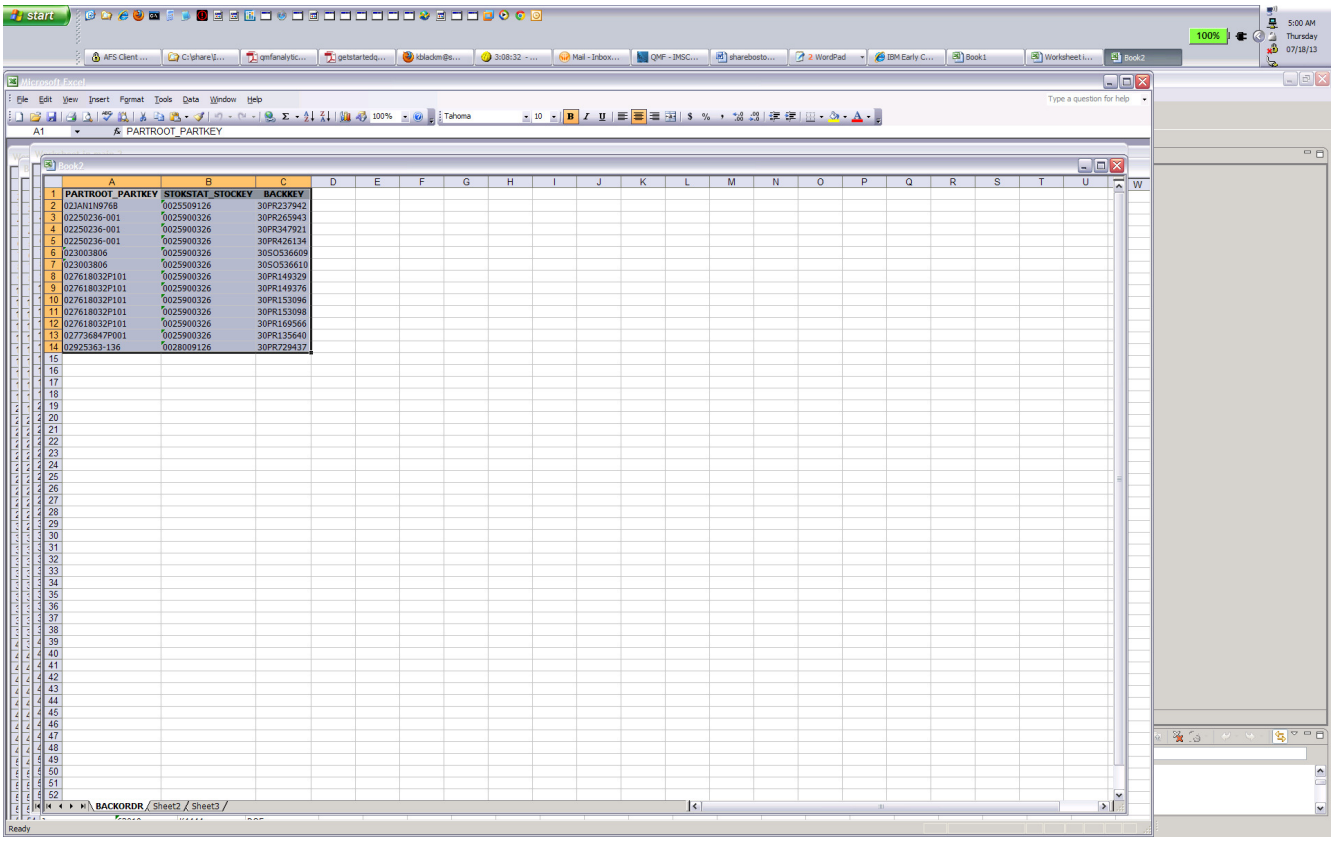


5. Close the query by clicking on the **X** in the query tab.

Transferring Data to Microsoft Excel

(this can be done back at your shop but cannot be done in this lab because Microsoft Excel is not installed)

QMF for Workstation is capable of directly transferring query results to Microsoft Excel. When running QMF for Workstation on a desktop with Excel installed, data can be immediately exported to Excel by pressing Ctrl+B or selecting the Results->Display Excel Sheet menu item.



A screen capture of the outcome of exporting a query result set to Microsoft Excel using QMF for Workstation's 'Display Excel Sheet' function. The export is performed 'live' without the need to create and open an external file.

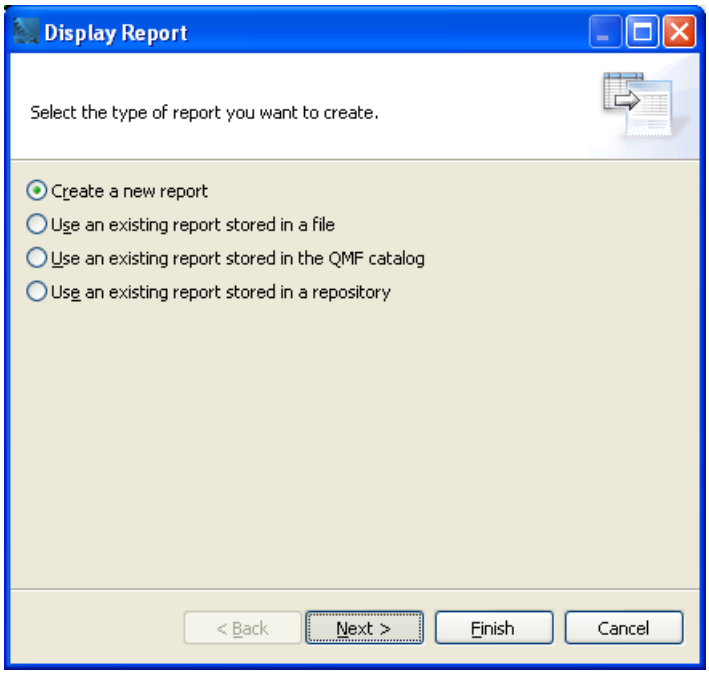
Exercise 4 – Developing Reports using QMF Forms

In this exercise, we will apply a QMF form to the result set produced from the query that was created in the prior exercise.

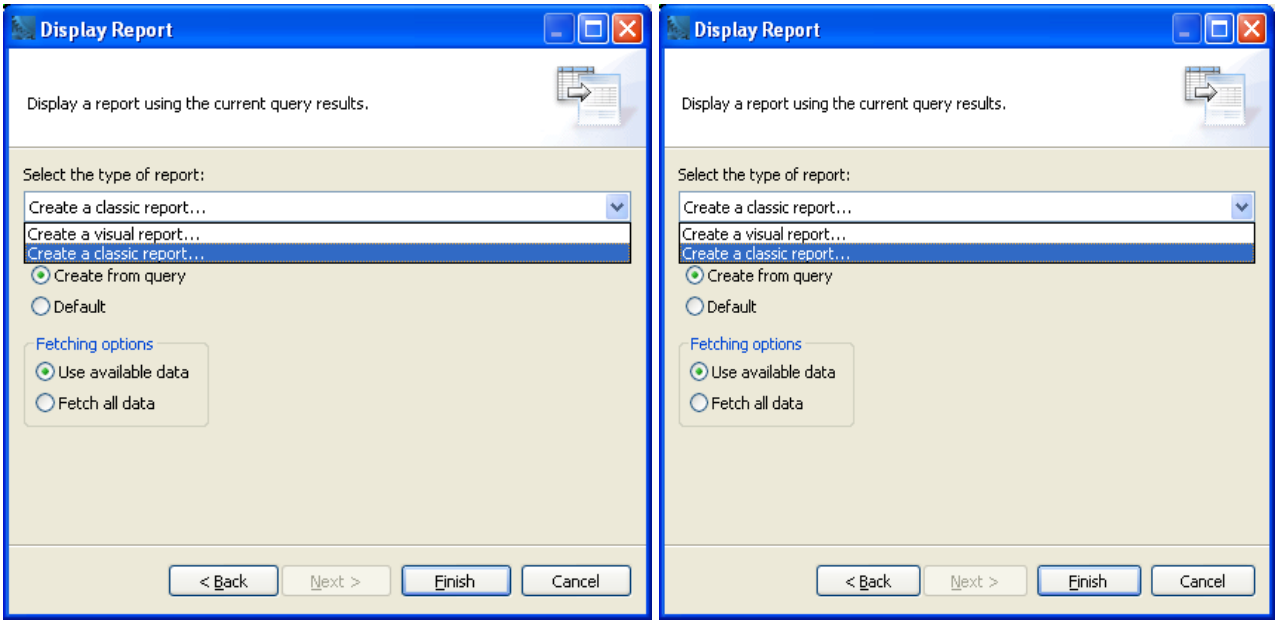
1. Click on the **backorder** tab to display the result set that you created earlier.

2. Click on the '**Display a report**' toolbar button, or alternatively, click on **Results -> Display Report** tabs.

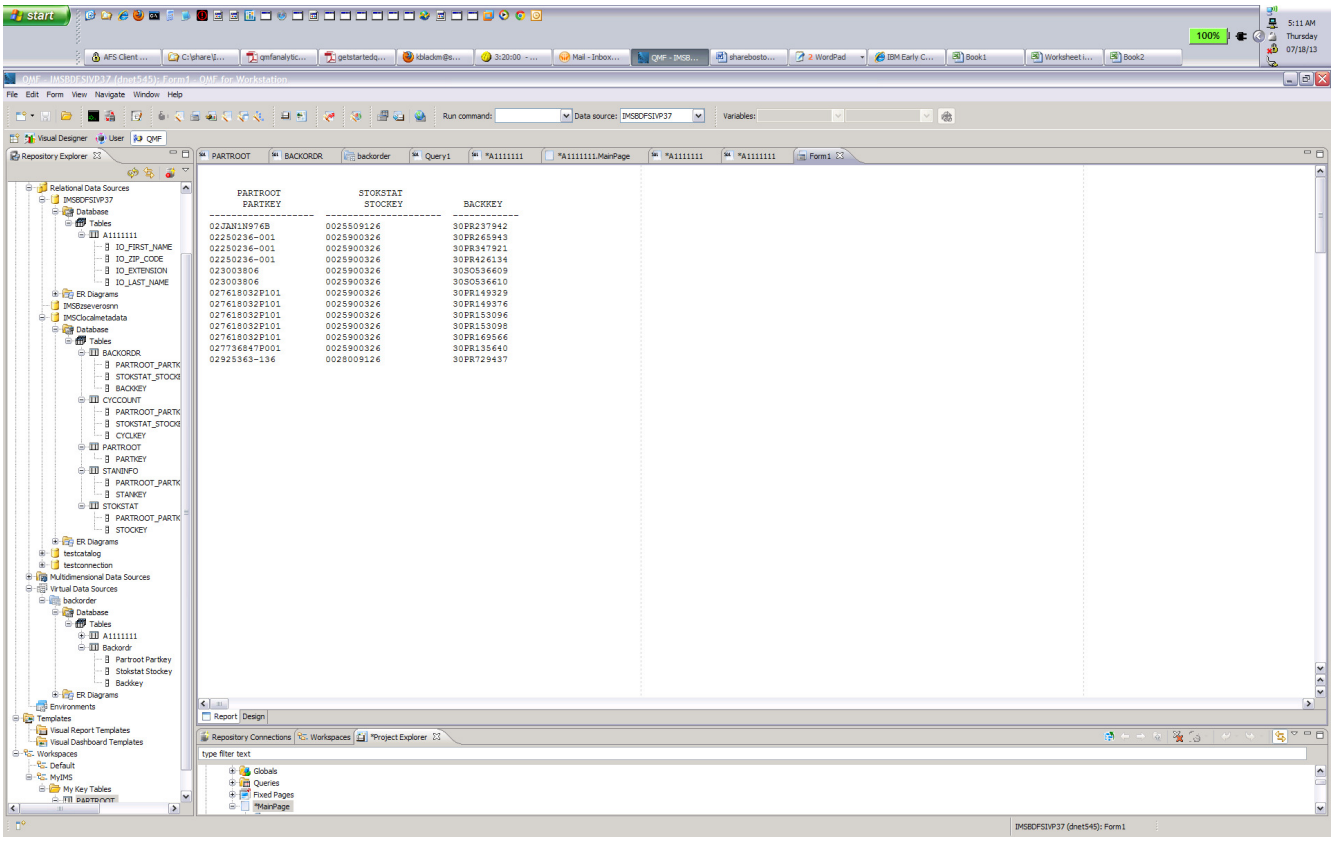
3. The **Display Report** wizard allows users to either create a new report or apply an existing report to a given query result set. Leave 'Create a new report' selected and click on the **Next** button.



4. Select **Create a classic report** from the report type combo box. Note the remaining options in the dialog – one can create a default report format or derive the report from the format already contained in the query result set. Ensure that 'Create from query' and the 'use available data is selected then click on the finish button.



5. The report is created and executed. Note that the report format matches the layout of the query.



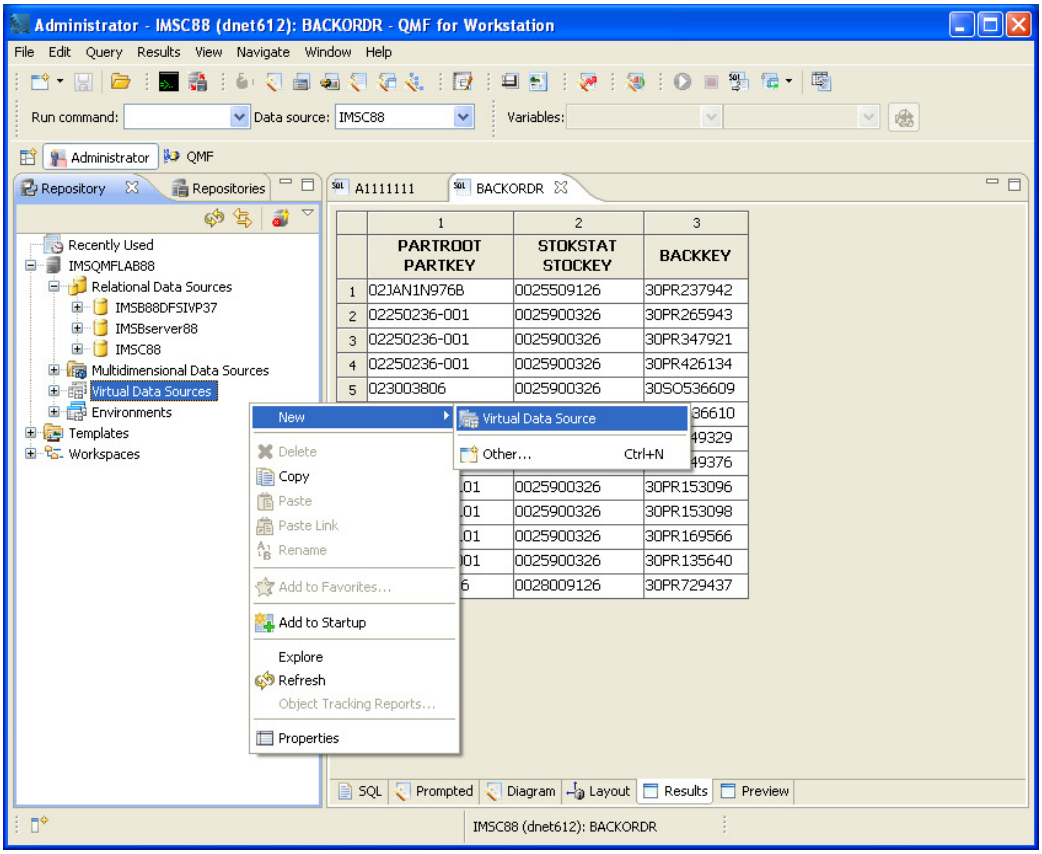
Exercise 5 – Defining virtual data sources to simplify database schemas for non-technical users

Virtual data sources allow QMF administrators to define simplified data schemas that make it easier for end users to work with your enterprise data. Traditionally, QMF users have been required to understand the explicit data schemas in your data sources since they work directly with the tables and views when building queries and forms. With virtual data sources, you can now define a level of simplification between the underlying data sources and the end users. This has two distinct advantages:

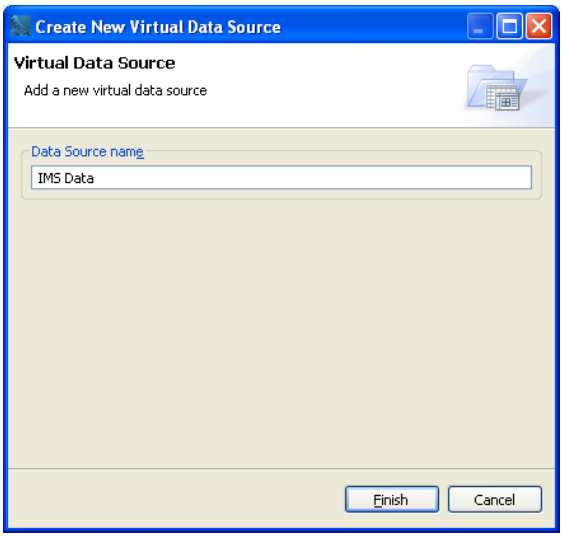
- Users are shielded from the complexities of the underlying data sources and only see relevant columns that pertain to their job function.
- A metadata layer allows changes to the underlying data schema without necessarily altering the virtual schema used by queries, reports and dashboards. This can be used to isolate BI content from database changes.

1. Select the Administrator Perspective. This displays the Repository Explorer.

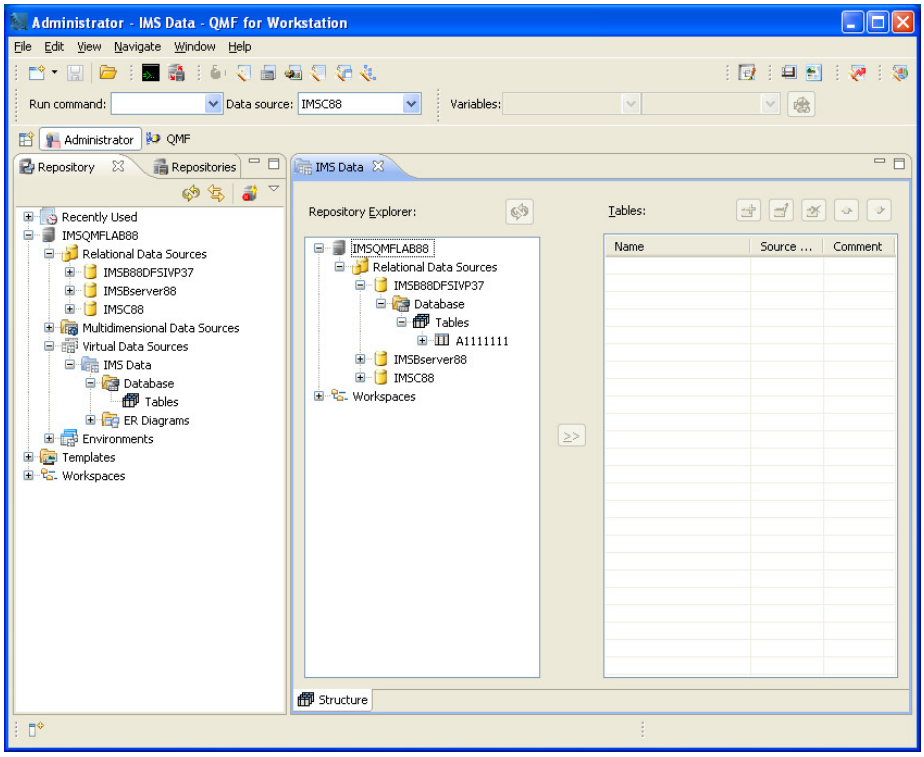
2. Expand **IMSQMFLAB***xx* and right-click on **Virtual Data Sources**. Select **New->Virtual Data Source**



3. Enter **IMS Data** for the data source name and click on the **Finish** button.



4. Expand the new **IMS Data** virtual data source and note that it appears much like any other data source in the repository explorer, complete with a tables tree item.



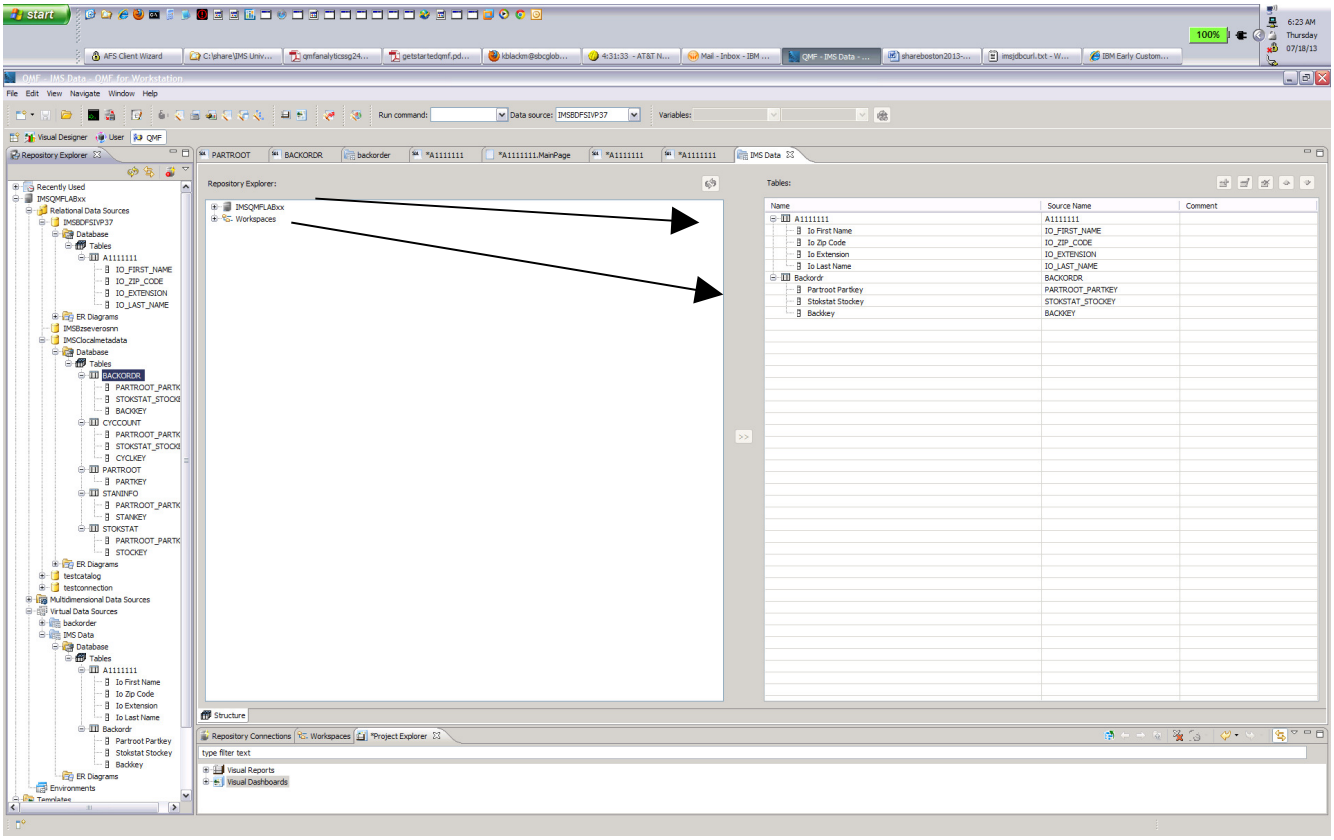
Virtual data source tables can be added using two key approaches:

- Adding tables from real data sources into the virtual data sources. Once added, the table name and columns can be renamed and specific columns can be removed from the virtual data source copy.

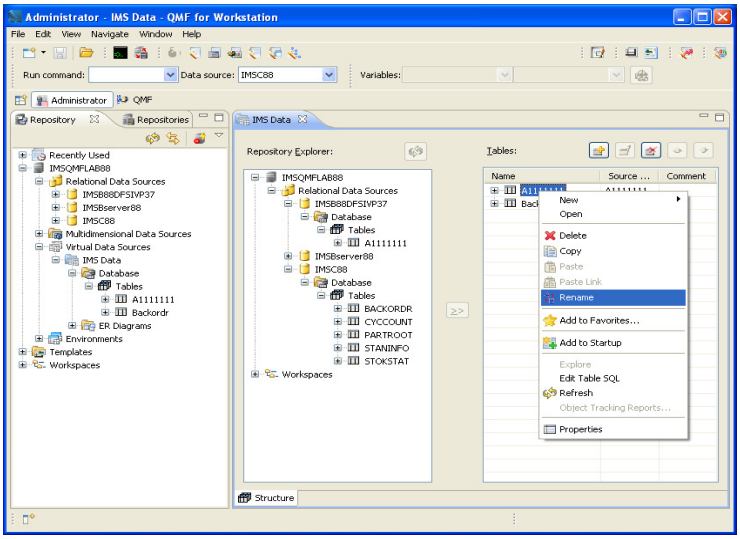
- Adding saved QMF queries into the virtual data source's table collection. The query

will appear as a regular table. As above, the table name and columns can be renamed and specific query columns can be removed.

5. We will copy a table from an existing data source. Locate the **DFSIVP37** `A1111111` table in the **IMSDDFSIVP37** Data Source, listed under **IMSQMFLABxx->Relational Data Sources** tree item. When located, select the table and **hold left mouse key to drag** to the table folder in **Virtual Data Sources**. Repeat to copy table from IMSBLocalmetadata data source.



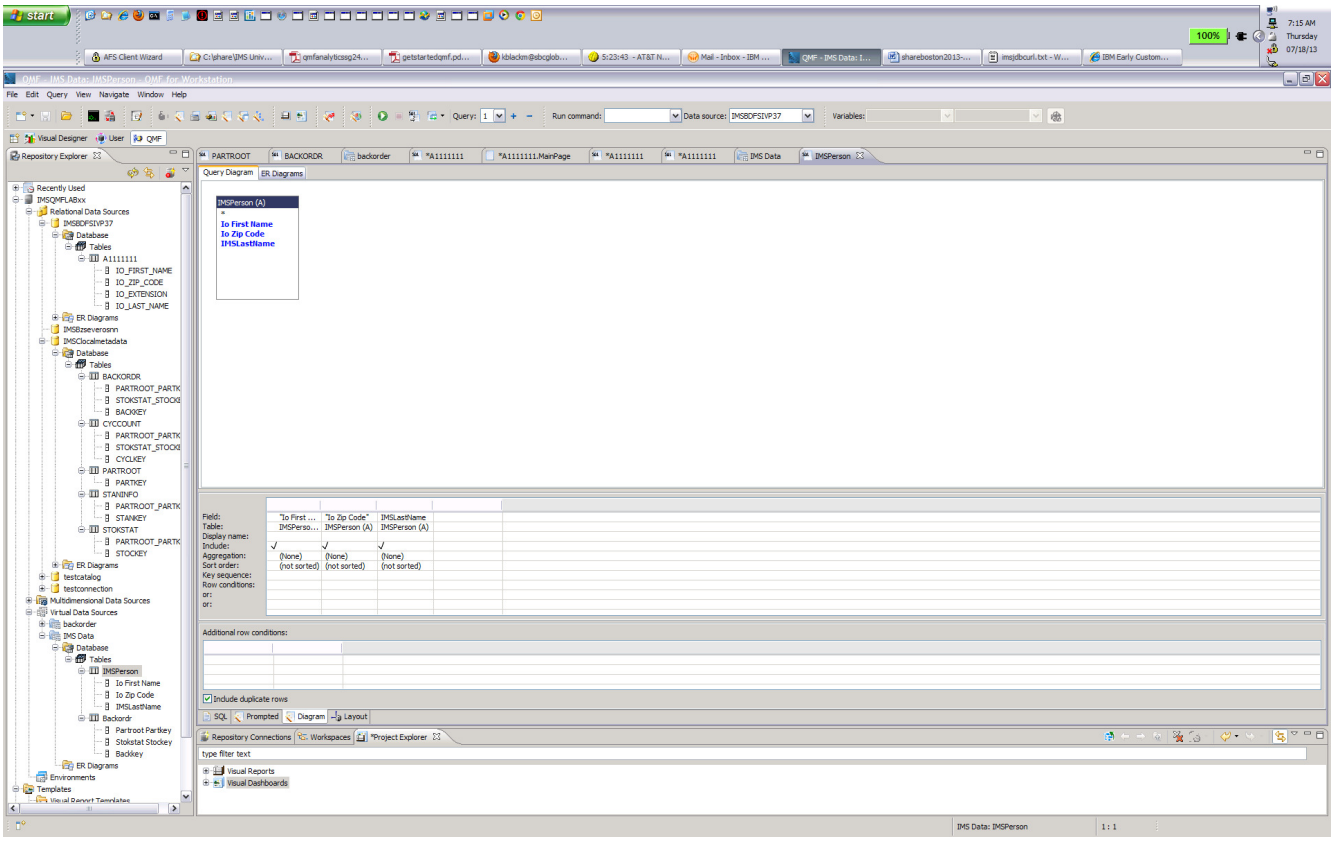
6. Right-click on the `A1111111` table and select **rename**. Type **IMSPerson** for the table name.

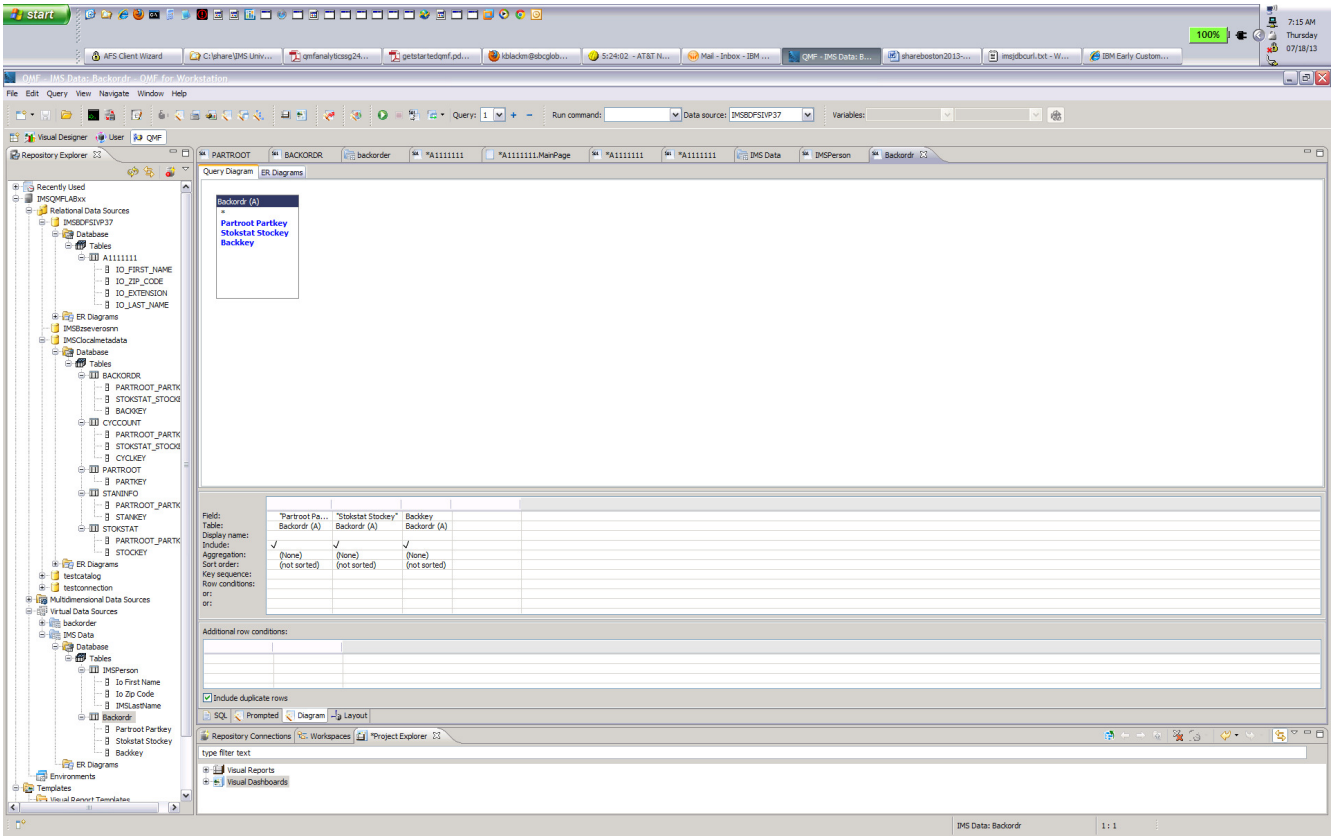


7. Expand the table to view the columns and right-click on the **io-lastname** column. Rename it to **IMSLastName**.

8. Right-click on the **io-extension** column and select **delete** to remove it from the virtual table. Any number of columns can be removed from a virtual data source table.

9. Double-click on the tabs for each of the tables to view the schemas.





This concludes the hands on lab. Thank you for taking the time to complete this set of exercises.