

EXCEL @ SQL REPORTING

Presentation Files Documentation

[SQL Saturday #107](#)

April 21, 2012

YES Prep Public Schools
North Central Campus
13703 Aldine-Westfield
Houston, TX 77039

[Google Maps](#)

Presented by:

[Michael D. Newby](#)

Recursive Creativity LLC



Disclaimers

All files for this presentation are in folders. I compress the folder in its entirety and that is what gets posted.

Some folders contain some useful links to web sites. Links to useful web sites will be listed below in the Table of Contents.

Unless otherwise specified below, all database queries use the [AdventureWorks 2008](#) database. All MDX (OLAP) queries use AdventureWorks DW 2008 (Adventure Works) cube. The data warehouse (DW), cube or Multidimensional Data Model (UDM), files are an option to install when installing the AdventureWorks database. I never upgraded to SQL Server 2008 R2 so I still use [AdventureWorks 2008 SR4](#). However, the queries should still work with R2.

If possible, I try to create files for Excel 2003, 2007 and 2010. However, some options such as the OLAP PivotTable Extensions for Excel are not supported for Excel 2003. SQL.REQUEST also gives error messages when the file is opened using Excel 2010. However, the functionality still works. I hope to spend more time trying to resolve any related issues with Excel 2010 before SQL Saturday.

TABLE OF CONTENTS

ADO and ADOMD

Dashboard Queries

[AdventureWorks2008 - Database overview](#)

MS Query

[Beyond Excel: Parameterized Query](#)

OLAP PivotTable Extensions

[OLAP PivotTable Extensions for Excel](#)

[MDX Function Reference](#)

[MDXpert.com](#)

[Excel Functions Supported by MDX](#)

[VBA Functions Supported by MDX](#)

Recordset

[Filter Property \(ADO\)](#)

[ADO Recordset Object](#)

SQL.REQUEST

[SQL.REQUEST for Excel 2003](#)

Tables - Pivot and Data

ADO and ADOMD

Most of the specifics here are discussed in the PowerPoint presentation or demonstrated in VBA code in files accompanying this presentation.

Use Single Cell for SQL Query.xls

Use Single Cell for SQL Query.xlm

The above files show one method not to use a SQL statement in VBA. While convenient and easy, it does not work 100% of the time! Something that is critical during month-end close or other instances where time is critical.

DASHBOARD QUERIES

Dashboard Queries - CASE, COALESCE, PIVOT.xls

Dashboard Queries - CASE, COALESCE, PIVOT.xlsm

The above files return the results from three different SQL Server stored procedures (SP). Each SP uses one of the T-SQL statements listed in the file name: CASE, COALESCE, and PIVOT.

A short summary of the use of each:

CASE

This is a bit antiquated. Once you examine the code and compare it to the other methods you will see very little use for in this scenario. It still does have practical uses but not for the purpose of this demonstration.....summarizing data for dashboards.

COALESCE

This is useful for creating summaries of data. COALESCE would be used in conjunction with PIVOT when the number of columns would be impractical to code or would be dynamic.....unknown amount or subject to change.

PIVOT

This is useful for creating summaries of data. PIVOT would be used when the number of columns is limited or fixed. For example, as used in the files above, the number of months in a year.

Dashboard Queries - STUFF, FOR XML PATH with Recordset.xls

Dashboard Queries - STUFF, FOR XML PATH with Recordset.xlsm

The above files provide example of querying for a list of products and generating a comma separated list of those products in one cell in Excel. The examples also demonstrate some of the various uses of [RecordSet](#) and [RecordSet.Filter](#) using Visual Basic for Applications (VBA).

In short, RecordSet.Filter replaces the WHERE clause in SQL. Why wouldn't you just change the SQL upstream (considered a best practice)? What if your company had limited resources or required a change order (Sarbanes Oxley) that took too long and was an arduous process, etc. Creating a SELECT * query that end users could then filter would save time and money.

MS QUERY

The data used for this demonstration are stock prices, dividends and other pertinent information on daily trades for the major US stock exchanges (AMEX, NASDAQ and NYSE) downloaded from [InfoChimps](#).

MS Query - Stocks.xlsm

MS Query - Stocks - NASDAQ.xlsm

MS Query - Stocks - NYSE.xlsm

The link above in the Table of Contents shows how to set up a parameterized MS Query file. However, the presumption is that all the values in each field are known so they can be entered manually.

The examples provided have a form that has fields tied to the parameterized query fields. Stored procedures populate the fields in the form with data from the underlying database. Thus, end users do not have to know the contents of each field but rather make choices from drop-down menus.

MS Query.txt

This file is a simple query used during the demonstration to avoid typing and save time.

OLAP PIVOTTABLE EXTENSIONS

AdventureWorksDW2008 - OLAP Extensions.xlsm

Please note that the PivotTable OLAP Extensions must be installed to create

The above file has numerous tabs and I have used a variation of this file in other [presentations](#).

Blank	This tab has been left blank on purpose. It is used for demo purposes if needed.
PivotTable	This sheet contains the original PivotTable that summarizes AdventureWork's Sales by Product.
PivotTable-OLAPExtensions	This sheet contains the PivotTable that summarizes AdventureWork's Sales by Product but has some OLAP Extensions added. Two equations are added for purposes of this demo.
PivotTable-OLAPExtensions-Copy	This sheet has the previous PivotTable converted to equations for a more professional reporting appearance. It also has some buttons added to facilitate viewing various levels of detail.
AdventureWorks-Old	This is the original PivotTable converted to equations (Free Form) for presentation purposes.
AdventureWorks-Old (RV)	This is the previous Free Form PivotTable range valued (RV) so no equations remain.
AdventureWorks-New	This is the original PivotTable converted to equations (Free Form) for presentation purposes.
AdventureWorks-New (RV)	This is the previous Free Form PivotTable range valued (RV) so no equations remain.
Macros	Generally used for lists and other housekeeping and VBA purposes.

The four files above with OLD and NEW in their names were presented at the [Houston SQL Server User Group](#) in November, 2011. The presentation, [Excel Reporting from Cubes \(SSAS\)](#),

[Overcoming some Design Limitations Post-Implementation](#), and all related materials are posted on my website.

By examining the equations in the OLD and NEW tabs (with the (RV)) you will see that the OLD method relied on changing every equation in the free form PivotTable. However, the NEW method uses the Report Filter and the equations in the dynamic named range on the macros sheet. Thus, there is no need to change all the equations in the free form PivotTable.

The gist of the presentation on this topic was that if you need to distribute files with data that is contained in the PivotTable, omit it from the list that is tied to the Report Filter, hide and password protect the macros sheet and users will not have access to that data.

The same can be done for a regular PivotTable. It does not just apply to the free form PivotTable.

RECORDSET

The data used for this demonstration are stock prices, dividends and other pertinent information on daily trades for the major US stock exchanges (AMEX, NASDAQ and NYSE) downloaded from [InfoChimps](#).

Stocks - Loop - Recordset.xlsm

Using the Record.Filter command, this file demonstrates how to filter stock data (prices or dividends) for any given year alphabetically.....one sheet for each letter of the alphabet.

Stocks - Loop - SQL Query.xlsm

Using LIKE, this file demonstrates how to filter stock data (prices or dividends) for any given year alphabetically.....one sheet for each letter of the alphabet.

There is a folder that contains two text files with some SQL and VBA. These files are embedded in the PowerPoint file accompanying this presentation. They contain code useful for a somewhat outdated use of the Recordset.Filter.

Older versions of Excel had row limitations and this code was ideal for putting that data in Excel. Newer versions of Excel have considerable more but at some point the file size and performance become issues. If there is a need for a reporting solution for very large data sets, SQL Server Reporting Services (SSRS) was designed for that purpose.

Excel 2003 versions of this were not provided due to the size of a file once data was extracted.

SQL.REQUEST

The two folders, Boston and Chicago, contain Excel data files used by the two main files listed below.

There are two Excel add-ins available for SQL.REQUEST. SQL.REQUEST was originally available in Excel 2002 but the add-in and associated files can be downloaded or copied and used in Excel 2003, 2007 and 2010.

XLODBC.zip This compressed files contains the original files associated with the Excel 2002 add-in for ODBC connectivity and can be downloaded from [Microsoft downloads](#).

Once downloaded and installed, the folder can be copied or moved anywhere. After copying or moving just register the add-in with Excel as you would any other add-in for the version of Excel you are using.

FixXL5Modules.zip The above add-in is not supported beyond Excel 2003 but still works natively with Excel 2007 but not Excel 2010. There is much documentation and a few files available that supposedly fix the issues with getting SQL.REQUEST to work with Excel 2010. However, none of them work.

I got SQL.REQUEST to work with Excel 2010 briefly but went about my business doing other tasks. The error messages reoccurred upon opening a SQL.REQUEST file to make more changes. I did not pay attention to all that I did to know if it was something I did. I added a sheet for using SQL.REQUEST with SQL Server and maybe setting up the ODBC for that may have been the issue?

As of writing this document I have not had time to look into it further.

I deleted an old OLEDB connection for SQL Server I had and changed to using SQL Native Client, which OLEDB is now a part of. I will test on an older computer when time permits.

SQL Request.xls
SQL Request.xlsm
SQL Request.xlsx

The above are the two main files for using SQL.REQUEST with the difference versions of Excel.

There are three tabs in each file. One tab, appropriately named Excel, demonstrates SQL.REQUEST using Excel files. The second tab, SQL, demonstrates using SQL.REQUEST with SQL Server (AdventureWorks 2008). The last tab contains some lists used by data validation on the first two tabs.

Directions:

1. Set any ODBC necessary using Data Sources (ODBC) found under Administrative Tools. Excel files are already set up under User DSN tab. However, SQL Server will need to be added
2. Download compressed file (.zip) to desktop (anywhere you can REMEMBER)
3. Unzip folder to any location.

Equations are setup to look for Excel files being queried starting in folder where the SQL Request.xls* files are located.

4. Register the Add-ins above in Excel
 - a. Excel 2003: Tools, Add-ins, Browse.....
 - b. Excel 2007: Office button (upper left), Excel Options, Add-ins, Manage: Go, Browse.....(I use, Documents\Development\Add-ins\Excel)
5. Open file: SQL Request.xls
6. I had to do the following in Excel 2003 but not Excel 2007. I do not know the behavior would have been different if I had first opened the file in Excel 2007 and then 2003.

The update links should work but you also need to "Enable Macros".

Firs time doing this the following behavior occurred:

Single cell

- a. Enter =SQL.REQUEST(\$B\$6,,, \$B\$7)
- b. Press Enter
- c. You will be prompted for Add-in, browse to it and double-click

Array (multiple cells)

- a. Highlight cells with cursor
- b. Enter =SQL.REQUEST(\$B\$6,,, \$B\$7) where B7 = Array1, B8 = Array2 and B9 = Array3

- c. Press Ctrl+Shift+Enter (enter as Array formula)

TABLES – PIVOT and DATA

BillingData Dashboard.xlsm

This is a file I have also used in other [presentations](#). The data used by this file is not AdventureWorks but rather some billing data I created. The purpose of the above file and related data is create a dashboard to highlight invoice metrics for a professional services company and the impact on cash flow.

Scripts to create the database and related tables, views, stored procedures, data connections, etc. are available at the presentations link to my website above paragraph. I have also included the SQL Server Integrated Services package for loading the .CSV files containing the billing data.

This file uses both PivotTables and DataTables to return data from SQL Server. Each table type are Excel objects and VBA code in modules exist showing how to refresh each object programmatically, at the click of a button.

VBA - Refresh PivotTables.txt

VBA - Refresh QueryTables.txt

The above files contain code for refreshing the PivotTable and QueryTable objects. There is a short discussion on code for refreshing QueryTables using forms.