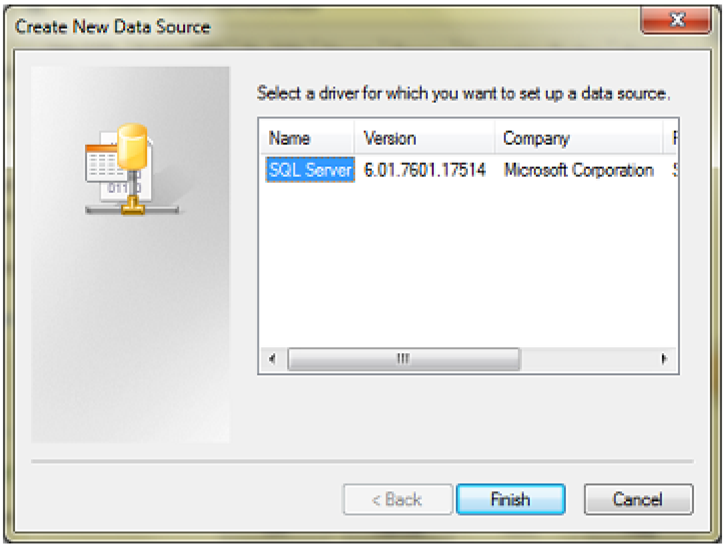
Configuring an Excel ODBC Data Source in Windows 7

A very neat little trick to compare tables of data in Microsoft Excel to reveal differences, is to highlight the data (all rows and columns containing data), and Define a name for the data, then use an ODBC Driver for Excel to subsequently query the tables as if they were a database.

Each tab in the Excel Workbook can contain a different table of (similar) data and the tables can be linked in Microsoft Query Editor so that SQL queries can be performed against the linked tables to reveal all rows of data where the same information exists in two separate columns of two separate tables, or even more useful (in my experience) the rows of data where the data that exists in one column of one table doesn't exist in the other.

In Windows 7, the Data Sources (ODBC) management console will only show drivers for SQL Server, and not the list of different drivers seen in previous versions of Windows.



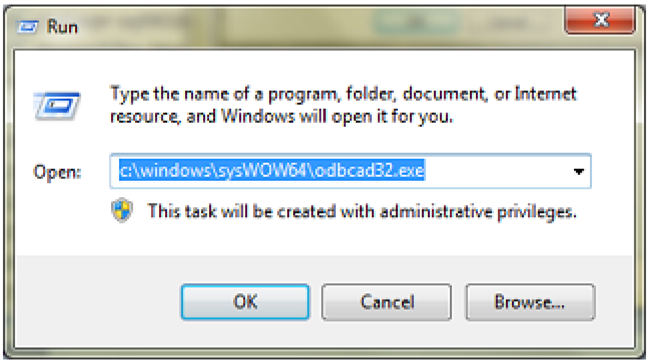
Create New Data Source

Select a driver for which you want to set up a data source.

Name	Version	Company
SQL Server	6.01.7601.17514	Microsoft Corporation

In order to create an Excel Data Source, you need to close the

64-bit version of the management console and run the following instead..

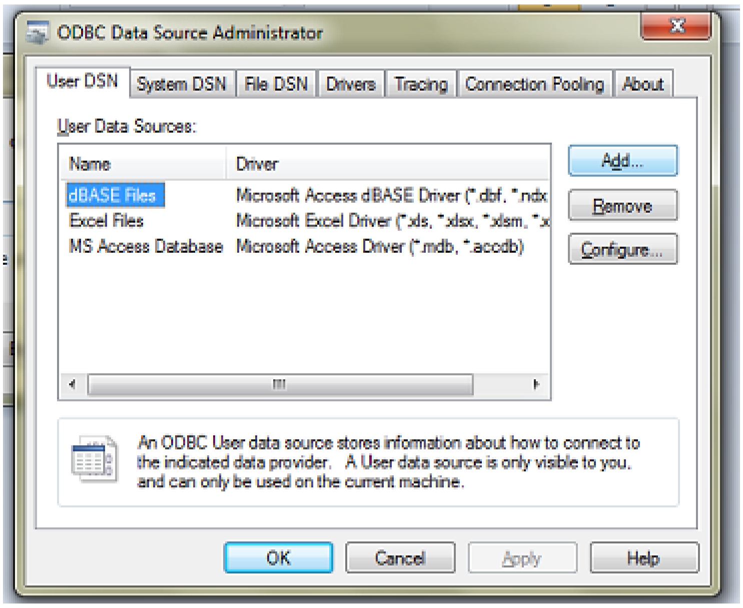


Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.

Open: c:\windows\sysWOW64\odbcad32.exe

This task will be created with administrative privileges.

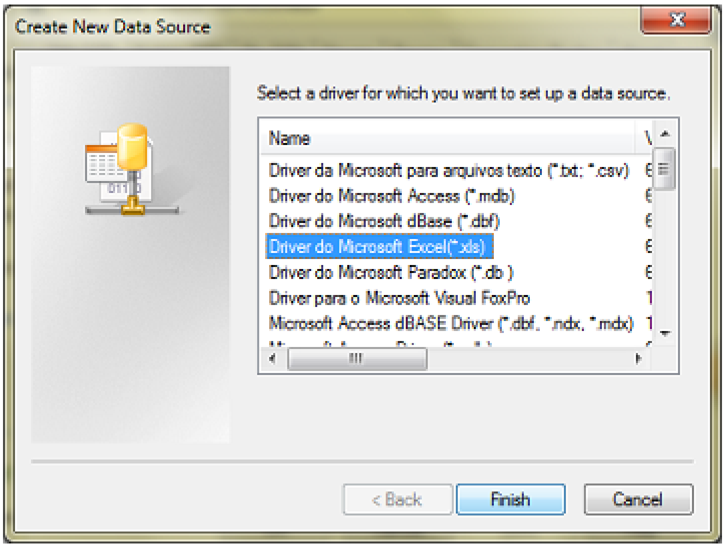
In the 32-bit odbcad32.exe (shown below), upon clicking Add, you’ll see the long list from which you can choose Excel Driver and point it at your spreadsheet that has the defined data within it.



User Data Sources:

Name	Driver
dBASE Files	Microsoft Access dBASE Driver (*.dbf, *.ndx
Excel Files	Microsoft Excel Driver (*.xls, *.xlsx, *.xlsm, *.x
MS Access Database	Microsoft Access Driver (*.mdb, *.accdb)

An ODBC User data source stores information about how to connect to the indicated data provider. A User data source is only visible to you, and can only be used on the current machine.



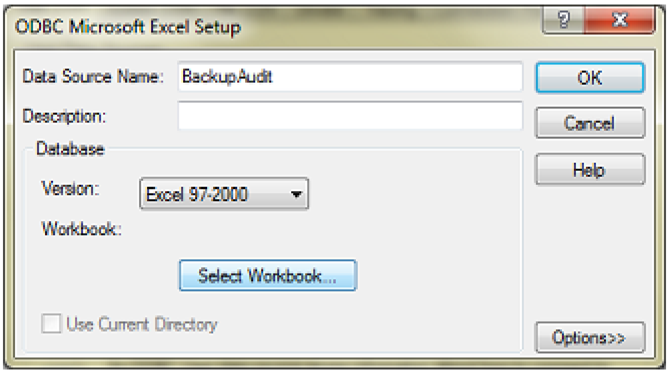
Create New Data Source

Select a driver for which you want to set up a data source.

Name

Driver da Microsoft para arquivos texto (*.txt; *.csv)
Driver do Microsoft Access (*.mdb)
Driver do Microsoft dBase (*.dbf)
Driver do Microsoft Excel(*.xls)
Driver do Microsoft Paradox (*.db)
Driver para o Microsoft Visual FoxPro
Microsoft Access dBASE Driver (*.dbf, *.ndx, *.mdx)

Note, if you intend to use this method to query the data in multiple Excel tables using the Query Analyzer, you will have to save a copy of the workbook in the older Excel 97-2000 .xls format before you create an ODBC Data Source for it.



ODBC Microsoft Excel Setup

Data Source Name: BackupAudit

Description:

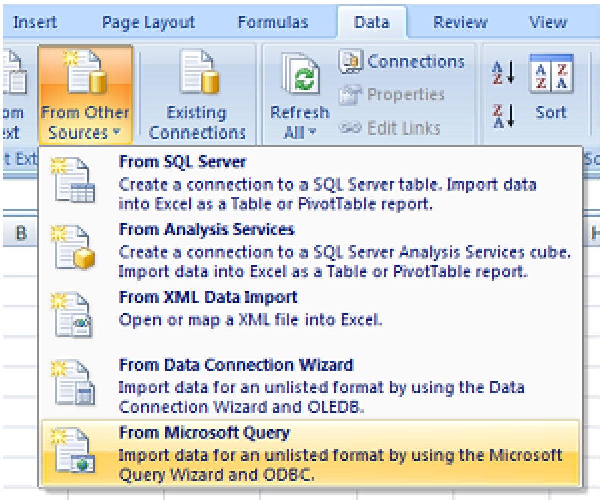
Database

Version: Excel 97-2000

Workbook:

□ Use Current Directory

Upon returning to Excel, Create a new tab in the same workbook or create a new Workbook if you like, and Open Query Analyzer as shown…



From SQL Server
Create a connection to a SQL Server table. Import data into Excel as a Table or PivotTable report.

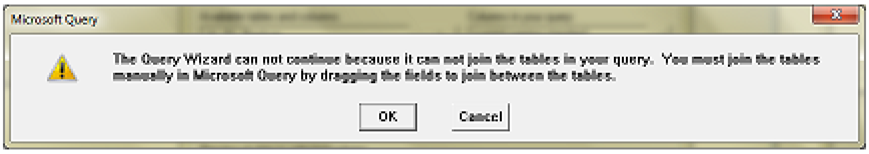
From Analysis Services
Create a connection to a SQL Server Analysis Services cube. Import data into Excel as a Table or PivotTable report.

From XML Data Import
Open or map a XML file into Excel.

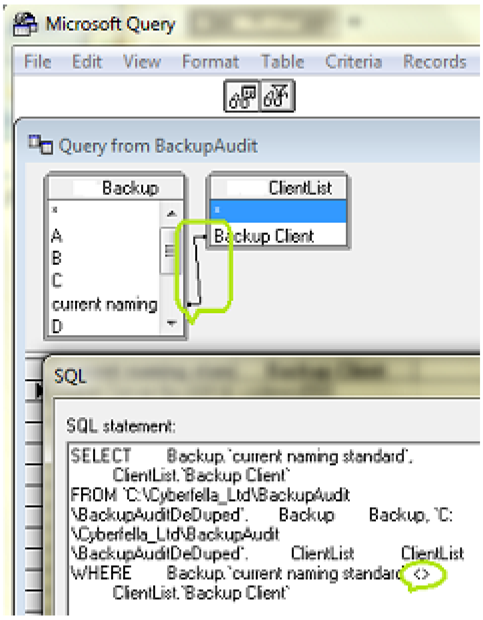
From Data Connection Wizard
Import data for an unlisted format by using the Data Connection Wizard and OLEDB.

From Microsoft Query
Import data for an unlisted format by using the Microsoft Query Wizard and ODBC.

Choose the Data Source Name (DSN) created using the 32 bit ODBC dialog, choose what tables (Definitions) you wish to include in your query and manually join the tables containing the similar data, then edit the SQL statement as required (usually just changing = for <> produces the sort of results I'm looking for, i.e. differences between the two columns, not similarities.



The Query Wizard can not continue because it can not join the tables in your query. You must join the tables manually in Microsoft Query by dragging the fields to join between the tables.



SQL statement:

```
SELECT      Backup.`current naming standard`,
        ClientList.`Backup Client`
FROM `C:\Cyberfella_Ltd\BackupAudit
\BackupAuditDeDuped`.    Backup     Backup, `C:
\Cyberfella_Ltd\BackupAudit
\BackupAuditDeDuped`.     ClientList     ClientList
WHERE      Backup.`current naming standard` <>
        ClientList.`Backup Client`
```

My personal preference when it comes to comparing columns of data is to export the columns to separate text files, *WinSCP* the text files onto a Linux/UNIX machine, then use *cat* | *sort* | *uniq* on each file, then *comm (not diff)* to perform the comparison and show entries in one but not the other of the two files being compared. I’ll endeavour to cover this method in another post to compliment this Windows 7 oriented post.

Donate