

Connect to MySQL or Microsoft SQL Server using R

1 Introduction

Connecting to a MySQL database or Microsoft SQL Server from the R environment can be extremely useful. It allows a research direct access to the data without having to first export it from a database and then import it from a CSV file or entering it directly into R.

Database connectivity using R requires a bit more work than just installing a library into R. Depending on the database (e.g. PostgreSQL, MySQL, SQL Server, etc.) it also requires adding some sort of connectivity driver like ODBC.

This example will focus on connecting to SQL Server and MySQL and uses Microsoft Windows XP. For MySQL the connector/ODBC is available for download at <http://dev.mysql.com/downloads/connector/odbc>. For SQL Server the ODBC driver is already available (tested on Windows XP and higher and is available as a native client. After the driver is installed the connection for either database is roughly the same.

Once the driver is installed you need to let Windows know the name of the connection. This is done by going to Control Panel \gg Administrative Tools \gg Data Sources (ODBC). This will bring up a dialog box and will list all of the current data sources. To add a new one click on “Add” and this will bring up another window that will look similar to Figure 1. Then select the database you wish to connect.

2 MySQL

In this example to connect to MySQL one would select ‘MySQL ODBC 5.1 Driver’ (or current version). Another dialog box as seen in Figure 2 will open.

There are several fields that need to be completed. The ‘Data Source Name’ field is important as it is what will be used in the R code this can be named whatever the researcher would like. The ‘TCP/IP Server’ will be the IP address or hostname of the database server and is supplied by the database administrator. This

can also be 'localhost' if the database is on the same computer that you're running R. 'User' and 'Password' are simply the username and password to access the database. 'Database' is the name of the database on the server you wish to connect. At that point you can test the connection to make sure it works and then click 'OK'.

3 Microsoft SQL Server

To use SQL Server one would select 'SQL Server Native Client 10.0' (or current version). Likewise, another dialog box as seen in Figure 4 will open for a SQL Server connection.

This process requires four steps. The first is shown in Figure 4 is to define the 'Name' which can, once again, be anything the research would like. 'Server' is the IP address or hostname of the database. Clicking 'Next' then asks for the 'Login ID' and 'Password' which is, once again, the username and password to access the database and is shown in Figure 3. Clicking 'Next' will then ask for the name of the database to connect as seen in Figure 5. Clicking 'Next' one last time brings up the window as seen in Figure 6. Unless the database has special configuration setting the 'Finish' button can be selected.

4 Using R

At this point R can be used to connect to the database using the RODBC library available in R. As far as SQL Server or MySQL the SQL syntax is roughly the same. Once the query is written and the data is pulled from the database R can be used as normal.

```
library(RODBC);  
channel <- odbcConnect("ODBC_NAME", uid="username", pwd="password");  
  
p <- sqlQuery(channel, "  
SELECT * FROM myTable  
");  
  
close(channel);
```

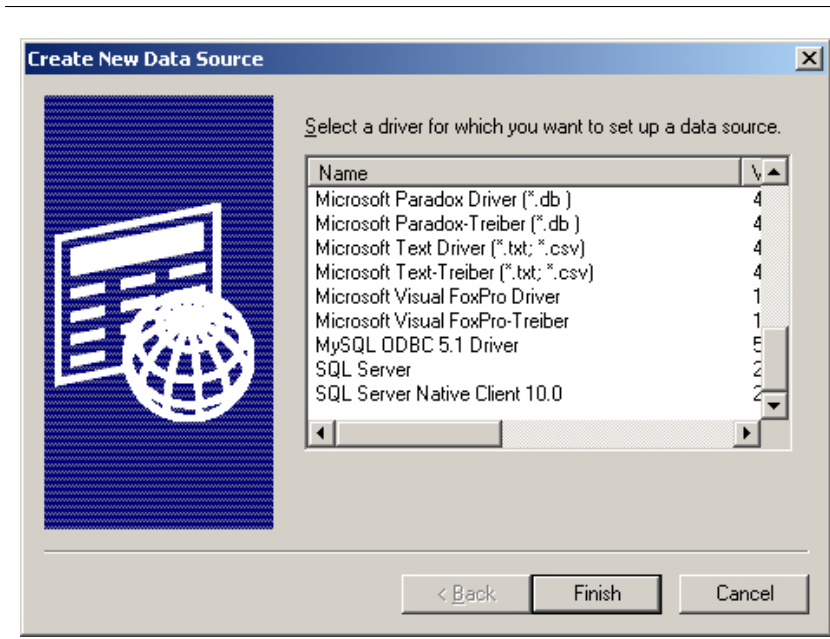


Figure 1: Create New Data Source Dialog Box

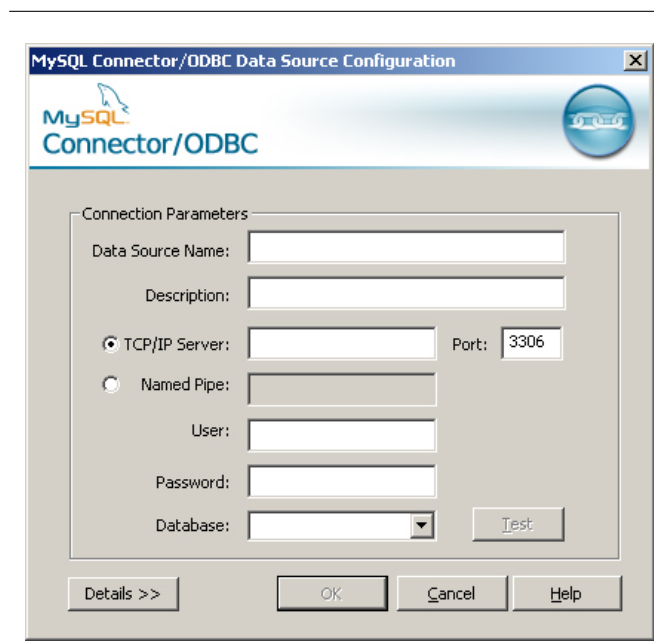


Figure 2: MySQL Create New Data Source

The screenshot shows a Windows-style dialog box titled "Create a New Data Source to SQL Server". On the left is a sidebar with the Microsoft SQL Server 2008 logo. The main area contains the question "How should SQL Server verify the authenticity of the login ID?". There are two radio button options: "With Integrated Windows authentication." (unselected) and "With SQL Server authentication using a login ID and password entered by the user." (selected). Below the second option are text boxes for "Login ID:" containing "wes" and "Password:" containing "xxxxxx". A checkbox labeled "Connect to SQL Server to obtain default settings for the additional configuration options." is checked. At the bottom are buttons for "< Back", "Next >", "Cancel", and "Help".

Figure 3: MySQL Confirm Username and Password

The screenshot shows the same dialog box as Figure 3, but at a different step. The text reads: "This wizard will help you create an ODBC data source that you can use to connect to SQL Server." followed by "What name do you want to use to refer to the data source?". There is a text box labeled "Name:". Below this is the question "How do you want to describe the data source?" with a text box labeled "Description:". Then, "Which SQL Server do you want to connect to?" is followed by a dropdown menu labeled "Server:". At the bottom are buttons for "Finish", "Next >", "Cancel", and "Help".

Figure 4: SQL Server Enter Connection Name and IP Address

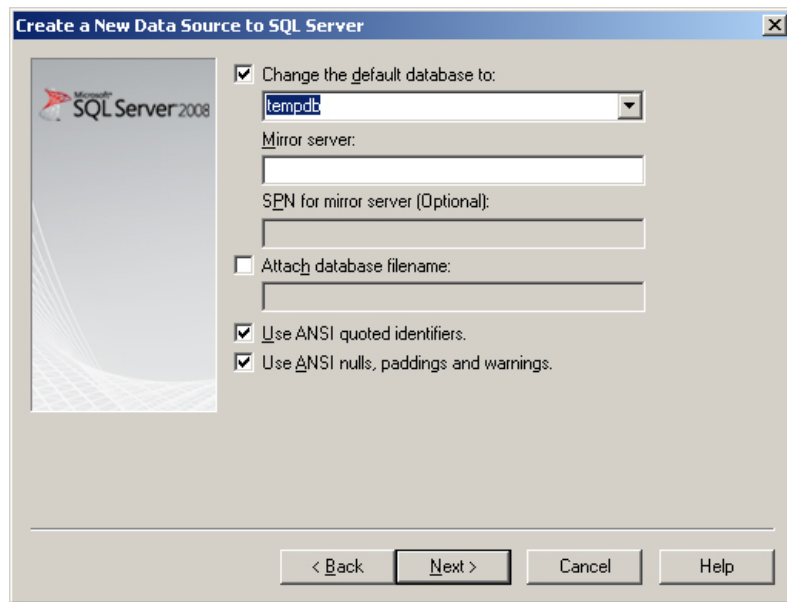


Figure 5: SQL Server Select Database Name

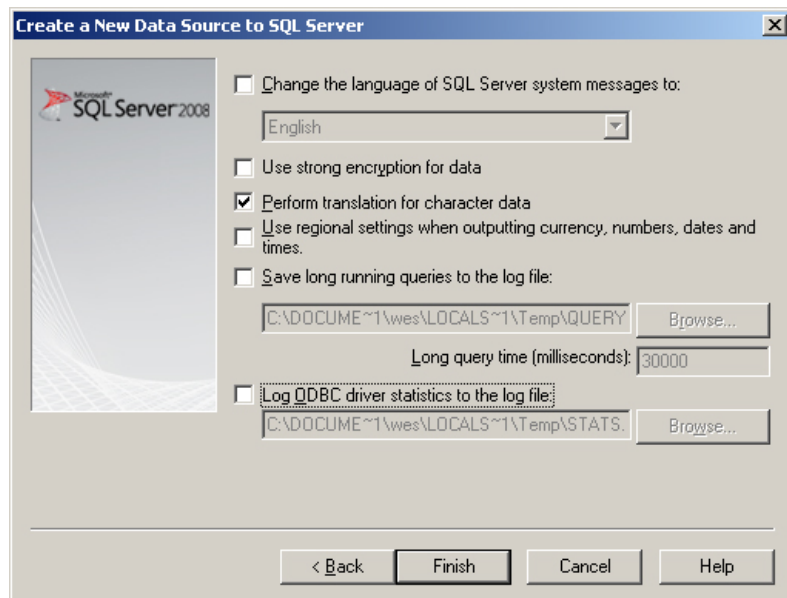


Figure 6: SQL Server Confirmation