Database Update App

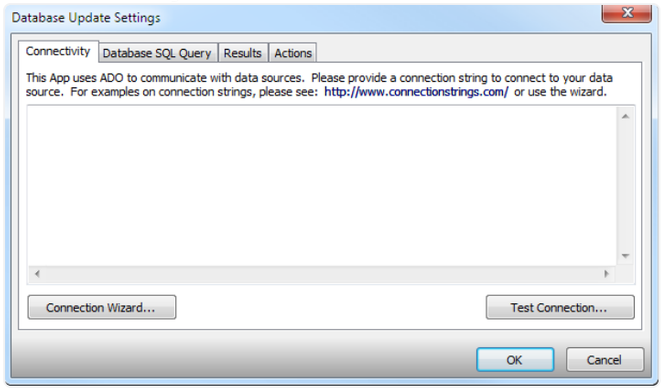
Purpose
This page app allows you to import data from a data source such as a database or spreadsheet into Acquire.

The app will process an SQL query and save the results to either a CSV file, an XML file or directly to Acquire Variables. Upon completion of this the app can ask Acquire to load a different page.
This app does ***not*** display anything on the Acquire page.

Operation
The app will execute the given SQL query about 200 milliseconds after the page starts in Acquire. This allows for disk activity to calm down. Shortly after this the results will be created and stored as required.

Configuration
The configuration window as four tabs follows:

Connectivity



This tab deals with how to connect and communicate with the data source. The app uses ADO (ActiveX Data Objects) to communicate with the data source and it does this using the OLE DB interface. *(The OLE DB interface replaced the ODBC interface, which is now considered obsolete)*.

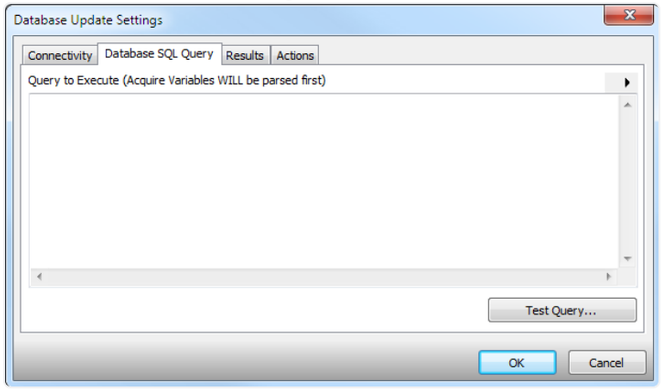
To enable connection, a suitable connection string must be supplied. There are numerous websites that will help with this. However, most connections should easily be made using the *connection wizard* (see below).

Pressing **Test Connection** will allow you to test if the connection string is correct and that it can connect to the data source.

as:light

NOTE: *Some data sources may require additional components to be installed on the computer, but this depends on the data source used.*

Database SQL Query



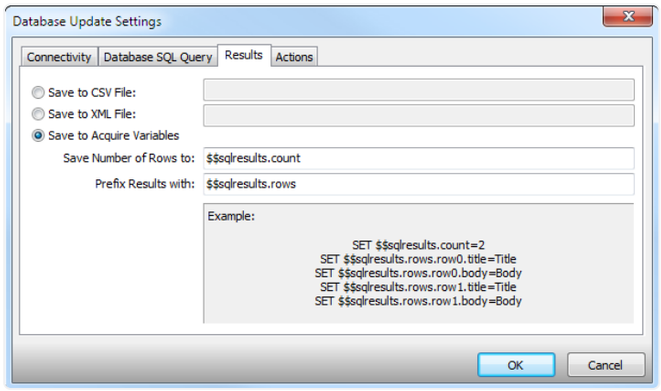
This tab allows you to specify a database query to execute. You may use Acquire Variables within this query.

Note: *Care must be taken to ensure these values will not cause the syntax to become invalid.*

The small arrow in the top right of the window contains a menu which will insert some simple examples for fetching data from some sources (eg: Excel.)
Pressing the **Test Query...** button will allow you to see if your syntax is correct.

Note: *During this test, Acquire Variables are NOT processed.*

Results



This tab allows you to choose how the results are stored.
The options are:

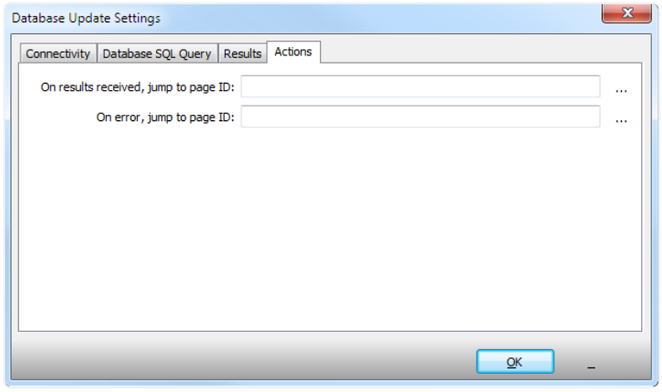
- Save to a CSV file: Enter the name of the file. The first line will be the list of field names.

  See *Appendix B.*
- Save to an XML file.: Enter the name of the file. This file format is compatible with the Bar and Line graph page apps.

  See *Appendix C.*
- Save to Acquire Variables for greater flexibility. Enter a variable name to save the number of rows to. Enter a variable prefix to save the results to.

  See *Appendix D.*

Actions

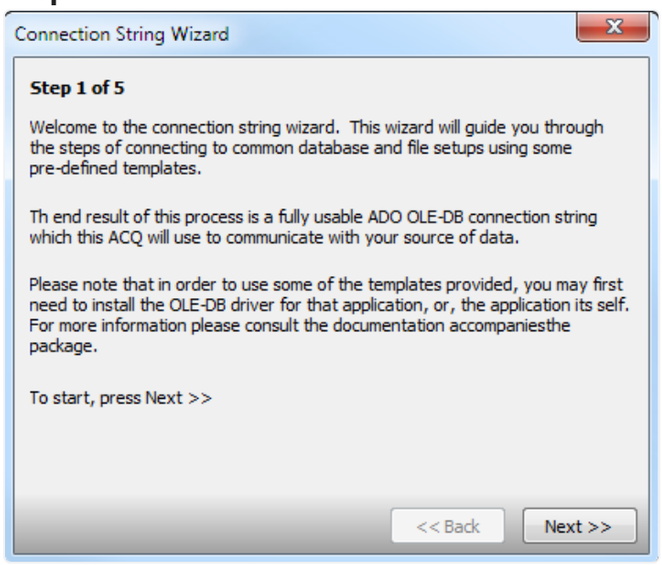


This tab allows you to choose what happens when the query executes successfully or when an error occurs. Enter the ID of the page to jump to.
The error message will be stored in the Acquire log file. But you may wish to display a friendly message on screen.

The Connection Wizard

The connection wizard provides a simple way of connecting to common sources of data. When pressed, the five-stage wizard will guide you through the process of creating the connection string. At any time, closing the wizard will cancel the process.

Step 1



Press Next to continue.

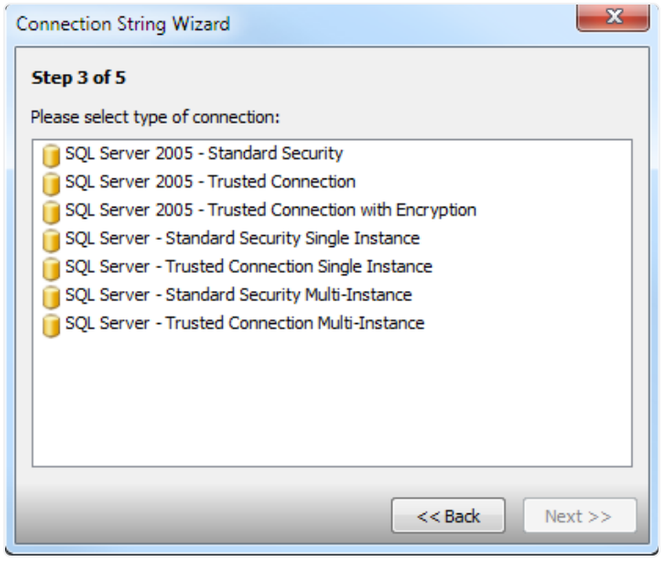
Step 2

This step allows you to choose where you would like to import your data from.

Select the source you require and press Next

Step 3

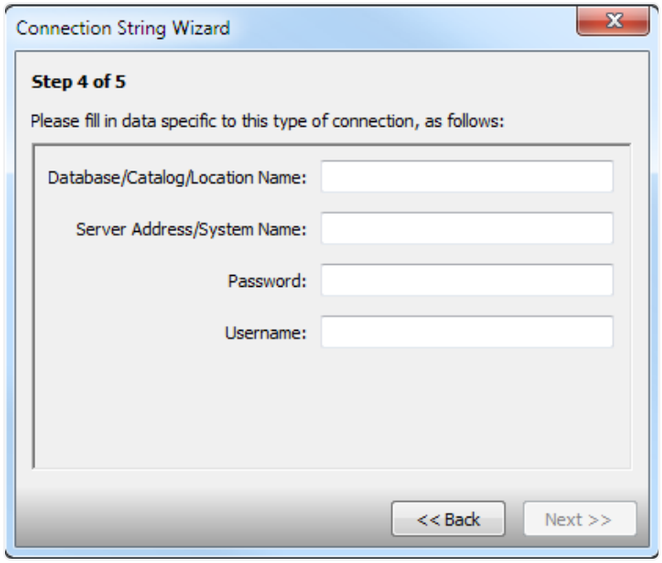
This step allows you to choose the kind of connection you wish to make to the source you have chosen.
The options here are specific to each of the sources.



Pick the type of connection you require and press Next.

Step 4

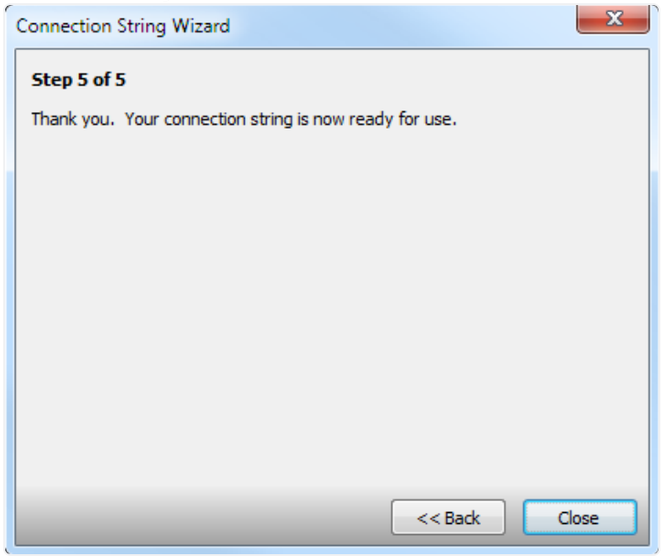
This step asks you for information specific to this type of connection.
If you are unsure what information to provide in these boxes please consult the administrator of the system you are connecting to.



When all information has been filled in (all boxes are mandatory) press Next.

Step 5

This is the final step. The wizard now has enough information to build up the connection string for the connection you wish to make.



Press Close to set the connection string for this app.

Appendix A – Sample Data

These result are used in the examples in the following appendices.

ID	Username	Password
1	Tim	f9in
2	Andrew	34ny
3	Joan	fnnnww
4	Angela	fib
5	Kim	kgu8e

6	Sam	fighe

Appendix B – Sample CSV File Output

This is a sample of the output from the app in CSV file mode.
It contains the results shown in Appendix A.
The first line contains the field names and subsequent lines contain the results.

"ID","Username","Password"
"1","Tim","f9in"
"2","Andrew","34ny"
"3","Joan","fnnnww"
"4","Angela","fib"
"5","Kim","kgu8e"
"6","Sam","fighe"

Appendix C – Sample XML File Output

This is a sample of the output from the app in XML file mode.
It contains the results shown in Appendix A.


```
<RecordSet>
  <Record>
      <ID>1</ID>
      <Username>Tim</Username>
      <Password>f9in</Password>
  </Record>
  <Record>
      <ID>2</ID>
      <Username>Andrew</Username>
      <Password>34ny</Password>
  </Record>
  <Record>
      <ID>3</ID>
      <Username>Joan</Username>
      <Password>fnnnww</Password>
  </Record>
  <Record>
      <ID>4</ID>
      <Username>Angela</Username>
      <Password>fib</Password>
  </Record>
  <Record>
      <ID>5</ID>
      <Username>Kim</Username>
      <Password>kgu8e</Password>
  </Record>
  <Record>
```

```
        <ID>6</ID>
        <Username>Sam</Username>
        <Password>fighe</Password>
    </Record>
</RecordSet>
```

Appendix D – Sample Acquire Variables Output

This is a sample of the output from the app in Acquire Variables mode.
It uses the results shown in Appendix A.

For this output, the configuration was as follows:
*S*ave Number of Rows: $$sqlresults.count
Prefix Results with: $$sqlresults.rows

```
$$sqlresults.count=0
$$sqlresults.rows.row0.ID=1
$$sqlresults.rows.row0.Username=Tim
$$sqlresults.rows.row0.Password=f9in
$$sqlresults.rows.row1.ID=2
$$sqlresults.rows.row1.Username=Andrew
$$sqlresults.rows.row1.Password=34ny
$$sqlresults.rows.row2.ID=3
$$sqlresults.rows.row2.Username=Joan
$$sqlresults.rows.row2.Password=fnnnww
$$sqlresults.rows.row3.ID=4
$$sqlresults.rows.row3.Username=Angela
$$sqlresults.rows.row3.Password=fib
$$sqlresults.rows.row4.ID=5
$$sqlresults.rows.row4.Username=Kim
$$sqlresults.rows.row4.Password=kgu8e
$$sqlresults.rows.row5.ID=6
$$sqlresults.rows.row5.Username=Sam
$$sqlresults.rows.row5.Password=fighe
$$sqlresults.count=6
```