SQL Server as Backend DB for Look2 Inventory v3.0 12/26/22

By default, the Look 2 Inventory Software connects to backend databases in Microsoft Access format. This document goes over the steps to convert the Look2_Inventory_DATA.ACCDB and Look2_Inventory_CONFIG.ACCDB files to instead run in SQL Server (or the free version of SQL Express). An overview of the process is as follows:

1. Install SQL Server / SQL Express (and if desired, SQL Server Management Studio - SSMS)
2. Install SQL Server Migration Assistant (SSMA)
3. Use SSMA to convert the existing Look2 backend database file(s) into SQL
4. Re-Link the Look2 front-end GUI file to instead point to SQL

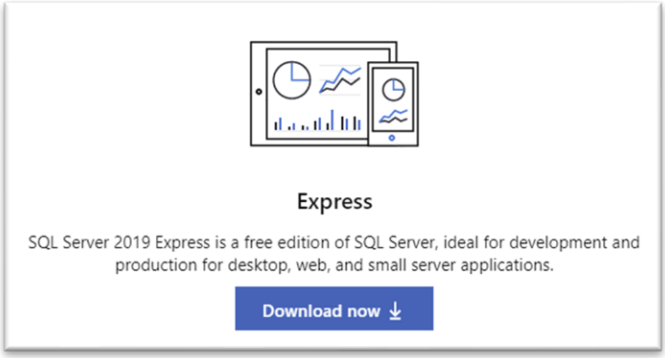
It is assumed a qualified IT person will be handling the installation and maintenance of SQL Server, and following this conversion process. Therefore, this document only offers a quick overview as a means of outlining some key points.

1. Install SQL Server and Tools

If you haven't installed SQL Server (or Express) and its associated tools, follow these steps.

1.1. SQL Express (or SQL Standard)

You can download 2019 SQL Express for FREE from Microsoft directly. As of this writing, go to: https://www.microsoft.com/en-us/sql-server/sql-server-downloads



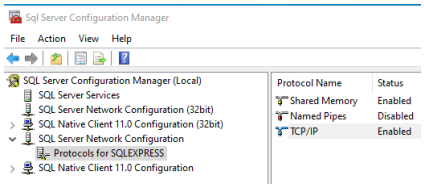
Be sure to use the Download Now button for Express. This is really SQL Server that has some limitations applied to it, and it runs on Windows 10, Windows Server 2016 and Windows Server 2019. This option only works if you're OK with the limitations placed on this version. Some items to consider include:

Feature	Express	Web	Standard
Max DB Size	**10 GB**	524 PB	524 PB
CPU Usage	**1 socket or 4 cores**	4 sockets or 16 cores	4 sockets or 16 cores
Smart Backups, Encrypted Backups	**No**	No	Yes
SQL Profiler	**No**	No	Yes
Install on Compressed Drive	**No**	No	Yes

This is not an exhaustive list, but given the fact that the Inventory Software DATA file size is tiny (< 50 Meg) and does not use extensive CPU on the data side (queries only when taking a physical reading), SQL Express may indeed work for your situation.

1.2. Networking? Ensure Connectivity

In order to get at the data other than a local machine, there are many settings that may need to be set appropriately for a networked installation to work. These vary by installation setup. Among the items to consider include:



- Using SQL Server Configuration Manager, enable TCP on its Network Services
- Using SSMS, right-click the top server (in left-side tree) and then under Connections make sure "Allow remote connections to this server" is checked
- Using SSMS, give permissions for the networked user
- Still having connection issues? Try disabling the firewall (at least at first) to make sure it's not blocking. Eventually, consider turning firewall back on, but opening up specific ports (e.g., 1433)

Getting "authentication" working for the connections to the SQL Server database tables varies by installation, and requires a knowledgeable IT person. It has been tested with SQL Server running on the same PC as the GUI file (so all "local"), and also from a networked PC using SQL authentication (it was not part of domain, so neither Windows Authentication nor other has been tested). It is suggested for future connectivity consistency that a common user/password be created (if so desired): *User=SQLAccess, pass=Look2Access, set as an Admin*

1.3. SQL Server Management Studio

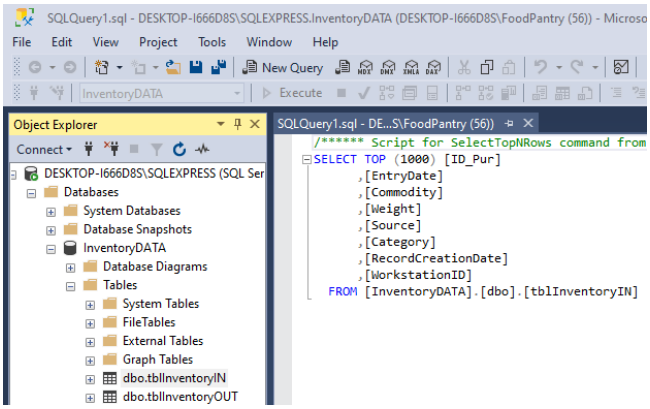
It is suggested you install SQL Server Management Studio.

While not absolutely necessary to run the software, it is the tool that helps you look at the database files and perform future upgrades, etc.

As of this writing, go to Microsoft's site:

https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver15

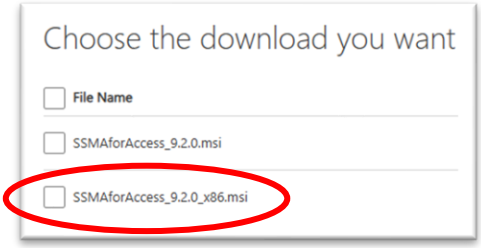
The example to the right shows the tblInventorIN table with a query to look at the top 1000 records.



2. Install SQL Server Migration Assistant

To convert the Access tables in the ACCDB file, you must install the appropriate version of SQL Server Migration Assistant for Access (SSMA).

It must match the bit-version of Access that is installed on the machine you'll be using SSMA on. For example, if you're running it on the same machine that you run the Inventory software on, you would download the 32-bit (x86) version (example circled to the right) if running Look2_Inventory_GUIx32, or the 64-bit version if using the GUI without the x32 in its name.

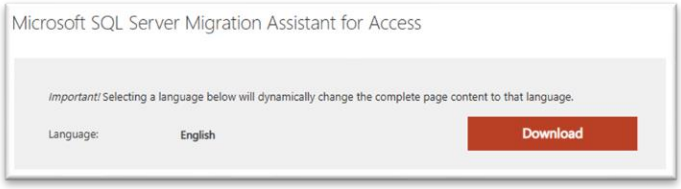


It was also tested on a machine running SQL Server which did NOT have MS Office installed. Instead, the FREE MS Access 2016 Runtime 64-bit version (which installed the conversion resources which SSMA uses) was downloaded and installed with the associated 64-bit version of SSMA (the one without the x86 in its name). The conversion was thus done directly on the machine running SQL Server, avoiding the "authorization connection" issues over a network.

As of this writing, an overview of the tool is available on Microsoft's site:
https://docs.microsoft.com/en-us/sql/ssma/sql-server-migration-assistant?view=sql-server-ver15

The download is available from a link within that page. For convenience, here's one as of December 2022:

https://www.microsoft.com/en-us/download/details.aspx?id=54255

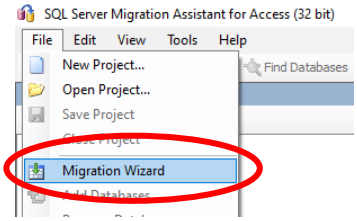


An overview of using this tool for purposes of converting the backend file(s) is given in the next section.

3. Migrate ACCDB File(s) to SQL

When launching SSMA, the Migration Wizard starts.
(If it doesn't auto-start, launch it under the File menu).

An overview window appears. Click "Next" to start the wizard.

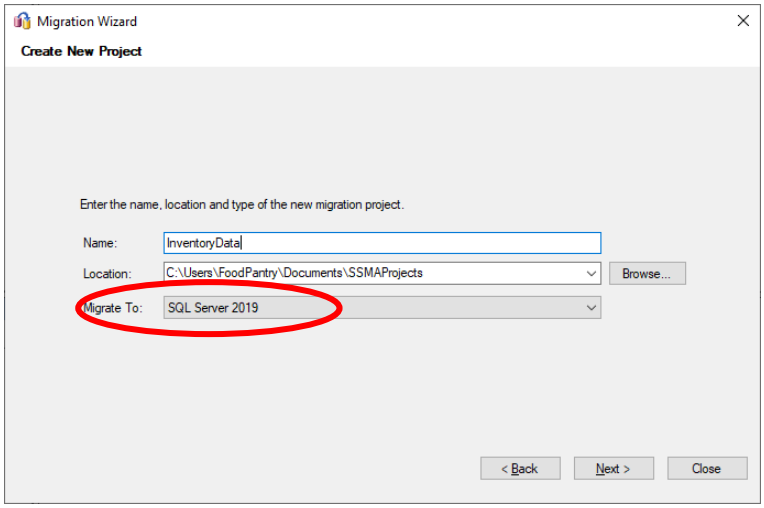


3.1. Project Name, Target

Give the project a name. *InventoryData* is what's shown to the right. However, ultimately this is not important, as it's only the name of the translation project for SSMA, so it can be anything you want.

The location is ultimately not critical, since it's also only used for saving the migration project.

Most critical is picking the version of SQL Server you'll be migrating to (SQL Server 2019 in this example).



3.2. Select DATA File

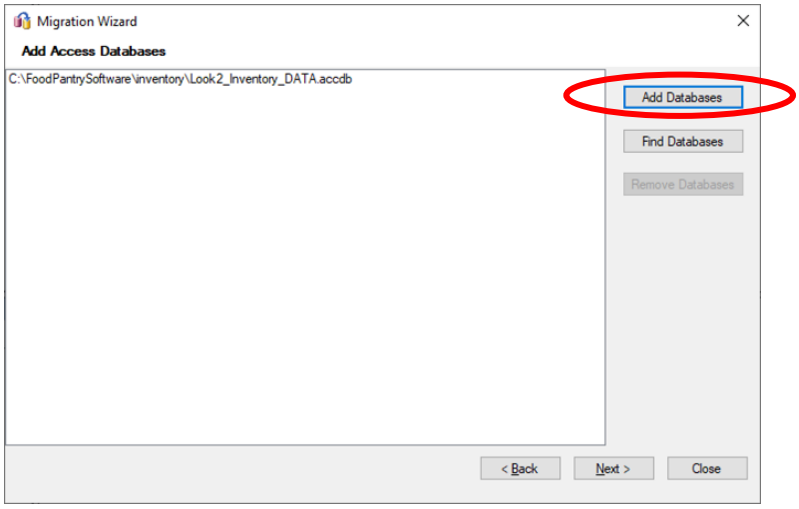
In this step of the Wizard, you will be selecting the back-end Look2 Inventory DATA file for migration.

NOTE: If you haven't yet installed the Inventory Software, please follow the steps outlined in the Inventory Quick Setup Guide: [Inventory_Software_Quick_Setup_Guide.pdf (look2consulting.com)]

You can either directly navigate to the Look2 Inventory installation directory if available from your SSMA machine, or you can simply copy the file over if it's on another PC, using it just for conversion purposes.

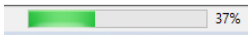
Click on the Add Databases button and navigate to the *Look2_Inventory_DATA.accdb* file.

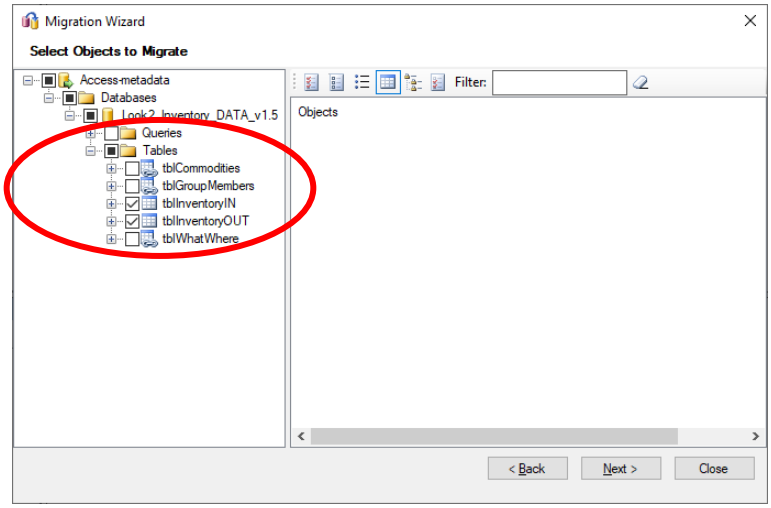
After it's selected and appears in the Migration Wizard list (as shown to the right), click the Next button.



3.3. Select Tables

When the Select Objects to Migrate of the Wizard appears, SSMA analyzes the database file. It takes a minute or so (depends on CPU speed) for it to finish. A progress bar is shown in the bottom right of the larger SSMA window. Please wait until the list is populated.





Click on the tiny + control to expand the Data file and then also the + to expand the Tables listing.

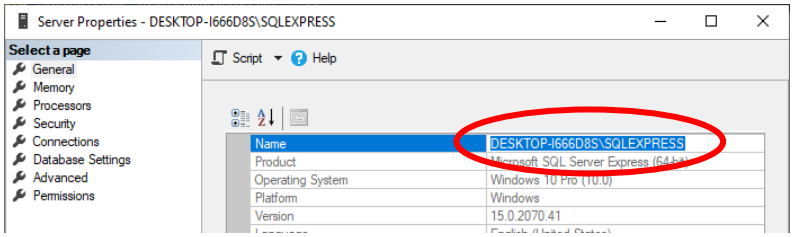
Uncheck the 3 config tables (only IN and OUT should be checked, as shown).

Note: if tables don't show up you may not have SSMA matching the bit-version of Access on that machine, or there may be some other error reported. Another thing to try is saving the ACCDB as an MDB file (shouldn't have to … but out of dozens of testing tries, this happened once for some unknown reason).

3.4. Connect to SQL Server

The next step is to connect to SQL Server.

Enter the SQL Server name.
Tip: to ensure you have the correct spelling, you can get it from properties window within SQL Management Studio



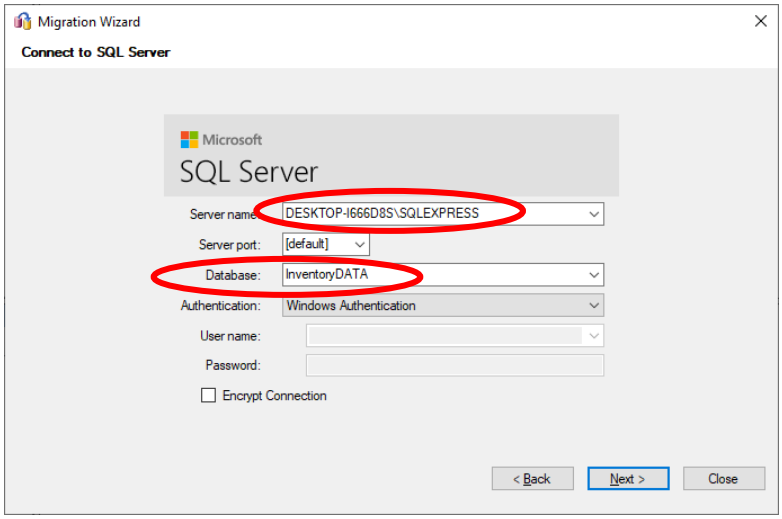
Enter as the database name: *InventoryDATA*

Enter any Authentication as necessary for your installation. If you simply set up SQL to use Windows Authentication, just leave this as-is.

Make sure Encrypt Connection is **unchecked**.

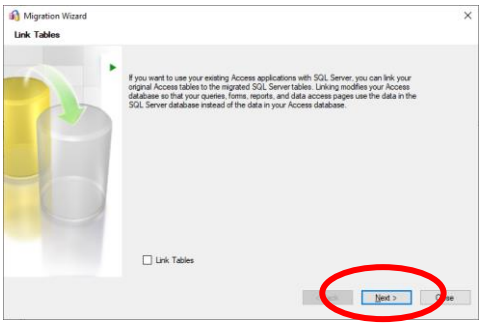
When you click Next, a message appears stating the DB doesn't yet exist. Click Yes to create it.

The Wizard attempts to connect to the SQL Server. It may take a few seconds, so please be patient. If the connection fails, make sure you have the server name correct, and permissions over the network.



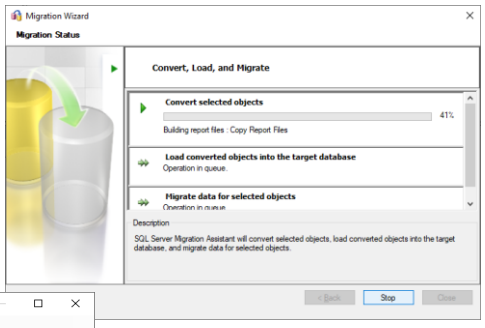
3.5. "No" to Link Tables

Make sure the Link Tables is **NOT** checked in the next step of the Wizard and click Next.

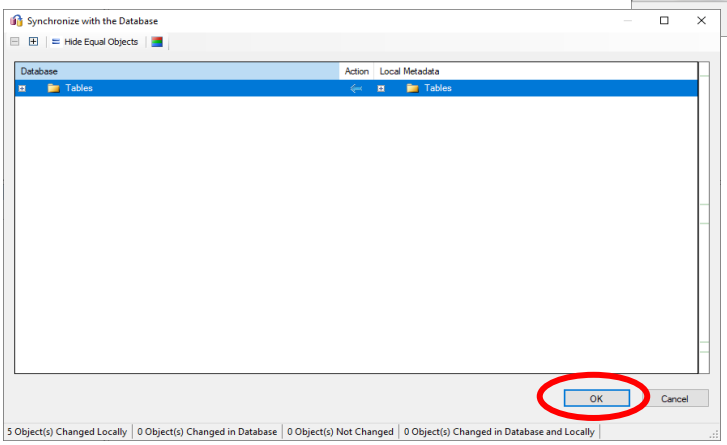


3.6. Conversion Preparation

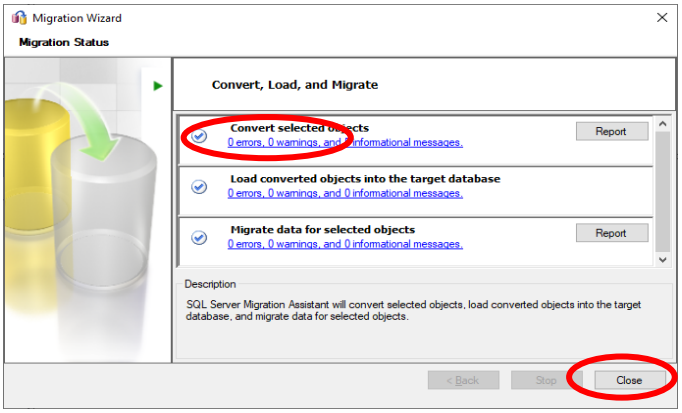
SSMA analyzes the database tables and prepares the conversion. Progress bars are shown in the Migration Status window. Just let it run through its conversion process (for an empty DB file it takes a few seconds ... for a large existing DB file it may take up to a minute).



A Synchronize with the Database window will appear. Click OK to initiate the migration.



Finally, the Migration Status appears and should show 0 errors and 0 warnings *(if it doesn't, you will have to troubleshoot what went wrong and re-attempt the conversion).*



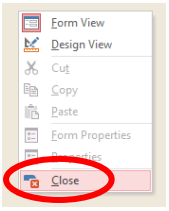
Click Close to complete the Migration Wizard.

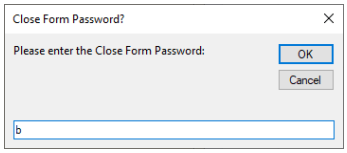
You may then exit out of SSMA. Your DATA file has been converted and is running in SQLExpress.

4. Link SQL Tables in the Access GUI file

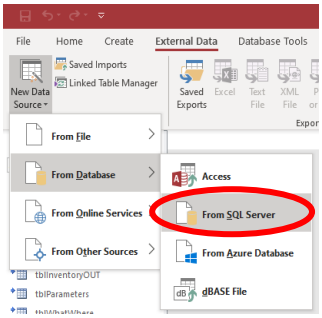
The next major step is to replace the links in the MS Access GUI file to point instead to the new SQL tables.

Follow these steps:

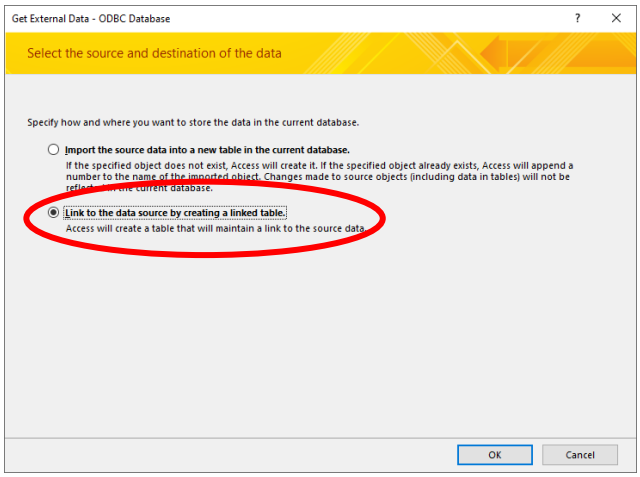


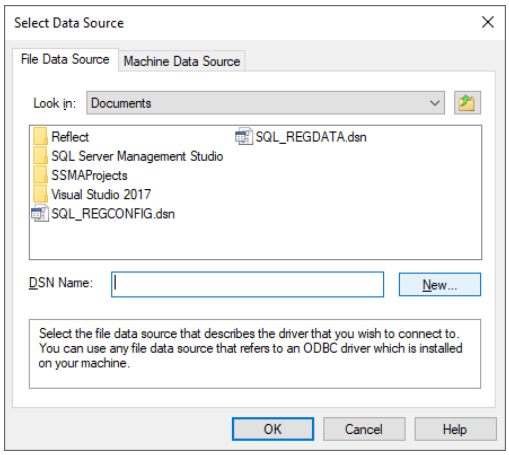


1. Launch the GUI file (starts up the Inventory software).
2. Right-click anywhere on a background and choose Close from the pop-up menu. It will ask for a password. The default is "b".
3. Go to the Access External Data toolbar and choose
   New Data Source
   From Database
   From SQL Server



4. In the Get External Data window, select
   Link to the data source by creating a linked table and click OK.



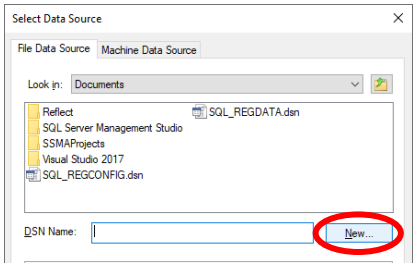


The Select Data Source dialog appears.

The first time you ever link to the SQL tables, you need to create a new File Data Source from scratch. These steps are outlined in Section 4.1

For any subsequent re-linking of these tables (for example, during a product update whereby a newer revision of the GUI file gets redistributed) you can select the DSN you're now creating. So, if you already have the DSN defined, you can skip Section 4.1 and jump to Section 4.2 step 5.

4.1. Creating a New SQL Data Source

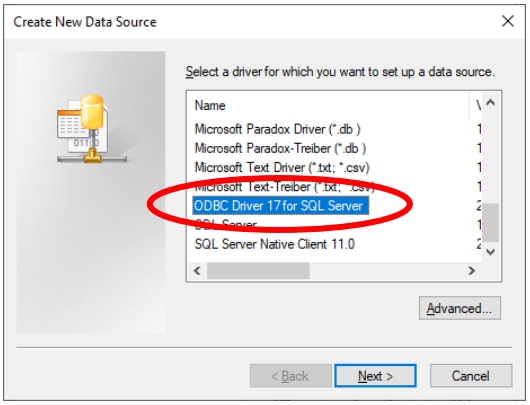


To build a new connection to SQL, click on the New button *(skip to Section 4.2 step 5 if you are reconnecting an updated GUI to existing SQL tables).*

a. The Create New Data Source Wizard appears, walking you through a series of steps to create the new DSN.

   The first step within this sub-process is to select the ODBC Driver for SQL Server (towards the bottom of the list), then click Next.

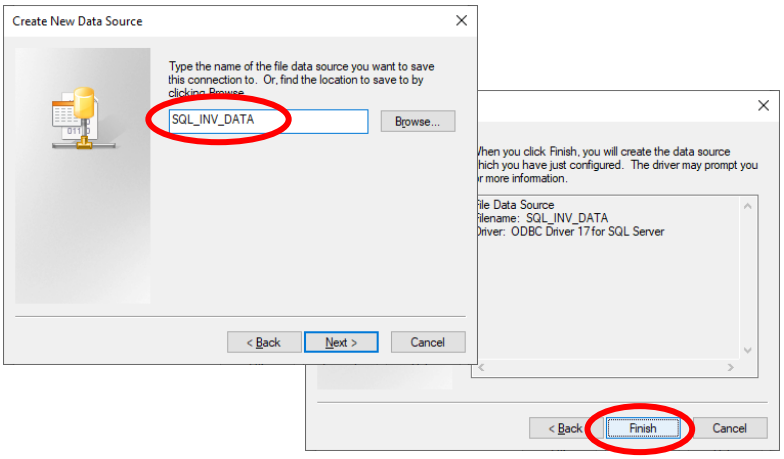
   (IMPORTANT NOTE: It must be via this connection for the Inventory Software to properly link)



b. Give the data source a name. To keep with the conventions of this document (and other Inventory documentation) it is recommended you use “SQL_INV_DATA”.

Then click Next.

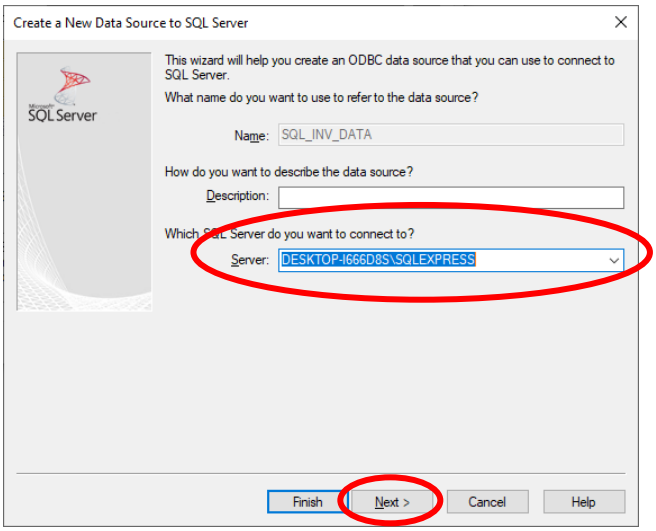
The Wizard shows you a summary. Simply click Finish.



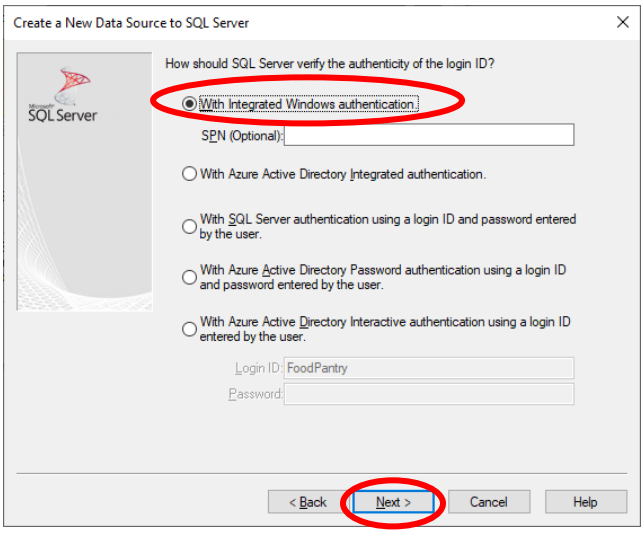
c. A SQL Connection Wizard starts and asks for the specifics of the SQL Server to connect to.

Your instance of SQL Server may not appear in the dropdown (broadcasting of its name is typically turned off by default … you’d have to specifically enable SQL Server Browser for it to automatically appear in this list).

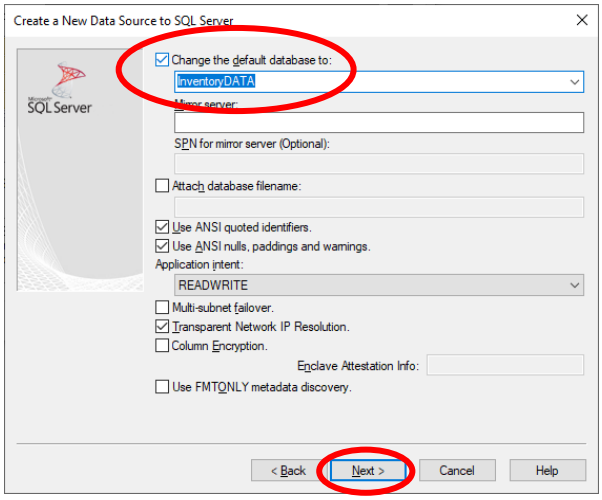
Simply type in the name (including the domain name) to the server and click Next.



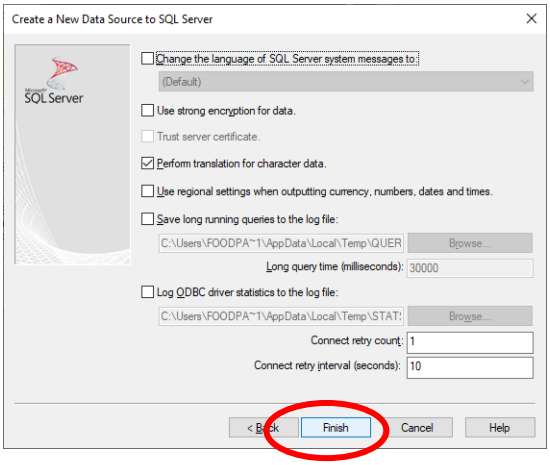
d. The next page shows the “With Integrated Windows authentication” already selected (which is what’s used in this document … though yours may be different), so just click Next.



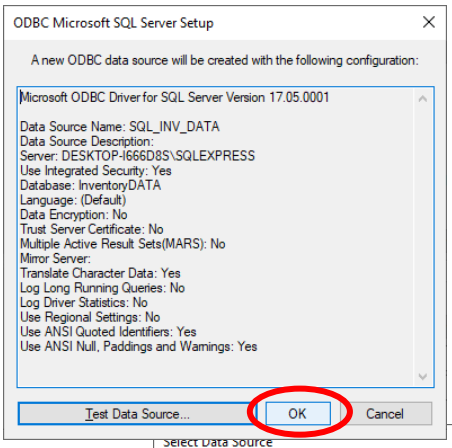
e. The next screen in the Wizard asks for the default database to connect to.
Check the top box and select *InventoryDATA* (the name used in Section 3.4). Then click Next.



f. The final page can be left as-is. So just click Finish.



g. Press OK at the summary of the ODBC SQL Server Setup.

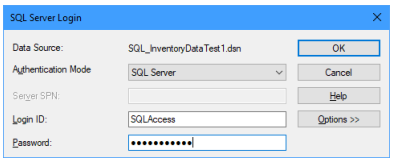


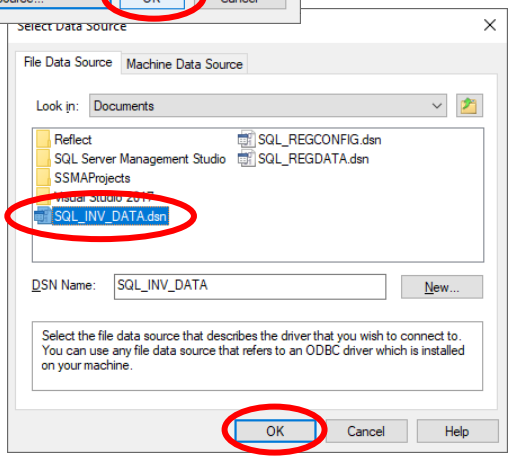
4.2. Continuation of Linking to Data Source

With a DNS fully defined, it appears in the File Data Source list.

5. Select the DNS from the list (*SQL_INV_DATA*) and click OK.

6. If your prompted for a password, enter it (e.g., *User=SQLAccess, pass=Look2Access*)



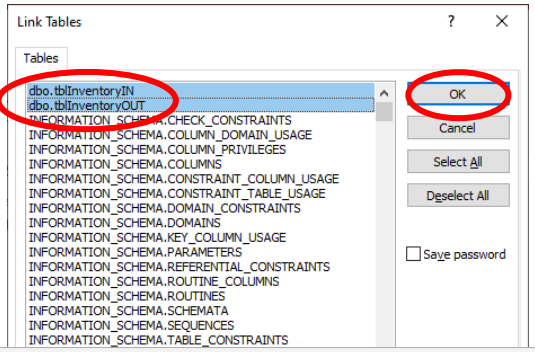


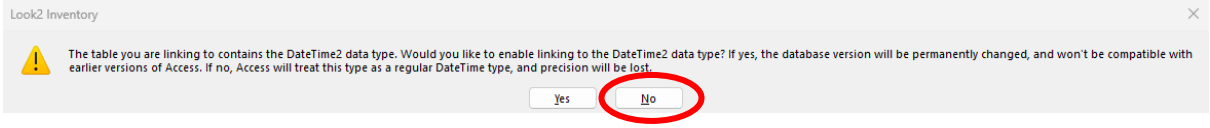
7. Now that Access is told which SQL database to link to, it queries it for all of its tables. The vast majority of them are for SQL's internal use.

   Select only the top 2:
   dbo.tblInventoryIN
   and
   dbo.tblInventoryOUT

   Then, with only those 2 highlighted, click OK.

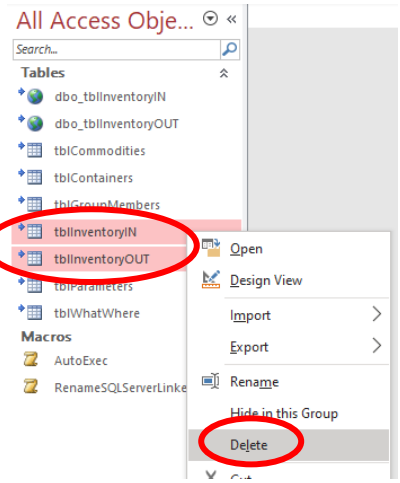
   If you are prompted about DateTime2, select NO.





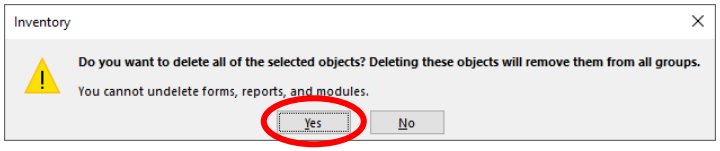
5. Remove Access Links, Rename New Ones

With the SQL Tables now linked (note the 2 new dbo. SQL Table Links at the top of the list), the next step is to remove the links to the ones in the Access DATA file.

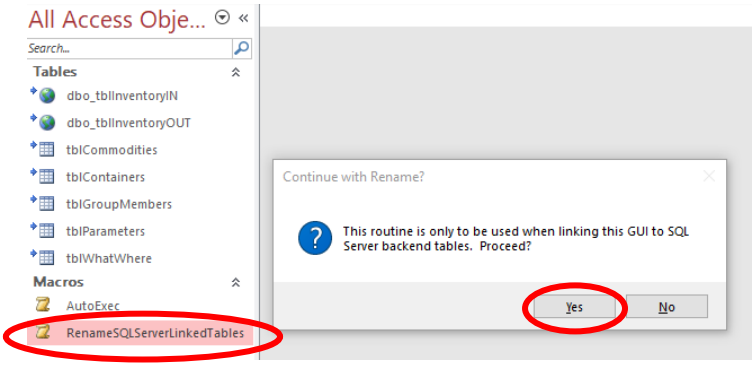


Select the 2 "old" Access table links to tblInventoryIN and tblInventoryOUT, then right-click and choose Delete from the pop-up menu.

You will be asked to confirm the deletion of these links. Choose Yes.

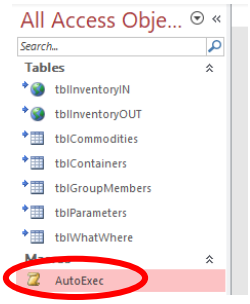


The final step is to rename the linked SQL tables. While you can do this manually, a macro has been created to remove the "dbo_" in front of each one.



Simply double-click on the *RenameSQLServerLinkedTables* macro and choose Yes from the pop-up window.

Access doesn't show the updated names unless you move the mouse over the list. So, after giving the macro a few seconds to run, please move the cursor over the names and you should see the "dbo_" removed from those 2 SQL tables.



6. Relaunch GUI Software

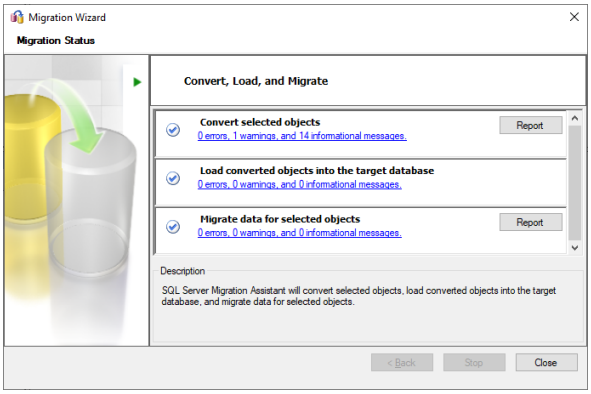
When the tables have been renamed to just be *tblInventoryIN* and *tblInventoryOUT*, you can re-launch the software without exiting/restarting.

Simply double-click the AutoExec Macro.

The Inventory software will then start up, this time connected to SQL Server for its backend DATA tables.

7. CONFIG and Multi-Station Comments

Here are some final thoughts on this process.



- You can also go through this Migration and Relinking process for all tables in the CONFIG file. You will get an SSMA warning that the field 'Group' in table tblWhatWhere "*...has a name that might cause problems for the Access application to function correctly against SQL Server*". This is OK, as the Look2 GUI programming correctly works with this column in the DB.
- If you have more than one workstation pointing to the same SQL tables, note that the application does NOT automatically refresh when data is changed from a different station (e.g., the main IN/OUT tables do not refresh to show new entries on their own accord). They refresh when: another reading occurs, you press the Refresh button, click on the Print icon, or open the History window. Also, if sharing CONFIG, it is recommended to exit out of the other workstation's GUI application before the Admin makes changes, so those changes can be processed upon relaunching of that second workstation (or, post updates, open and close Settings on workstation 2 to refresh).