.com Solutions Inc.

FmPro Migrator - FileMaker to Access Migration Procedure

FmPro Migrator - FileMaker to Access Migration Procedure

Overview

Overview - FileMaker to Access Conversion

This document provides an explanation of the steps required to perform a FileMaker Pro to Microsoft Access Conversion project using FmPro Migrator. This migration process includes the migration of Tables/Fields, data, Relationships, Value Lists, FileMaker layouts to Forms/Reports and FileMaker Scripts to Visual Basic code.

Revision 5
FmPro Migrator 9.37
8/28/2021

[Revision Notes: This update includes: Updated ODBC driver installation, GUI screenshots, replacement of the Access2013.accdc file with a new Access2016.zip file and screenshots, added info using SSMA and SQL Server for images, updated the info to show importing UTF8 data using the FileMaker ODBC driver (including improvements to the xferData.pl scripts, and updates for terminology and the use of advanced features in FileMaker 17+ versions - which previously required FileMaker Pro Advanced.]

Pre-Migration Preparation

Pre-Migration Preparation

The following steps are recommended prior to starting a FileMaker Pro to SQL database migration project with FmPro Migrator.

The first part of this process generally involves transferring the data into a SQL database. And for that part of the process you may need to delete the Unstored Calc, Global and Summary fields to make that part of the process more manageable.

But if you plan to convert the graphical interface and scripts later on, you would want to try to preserve as many of those fields as possible. Some databases only handle a limited number of columns (Microsoft Access is limited to 255 fields), so some deletion of fields may still be required.

You could transfer the data as one part of the process using a limited number of records, then use that smaller copy of the database having all of the fields in order to perform the GUI conversion (including layouts and scripts).

FileMaker Pro 2,3,4 - File Preparation

There is no ODBC driver for FileMaker Pro 2,3,4 files, and the ODBC driver is required for transferring data from FileMaker Pro into other databases.

These older FileMaker Pro database files should be upgraded to FileMaker Pro 6 prior to performing the migration.
The software download page for the licensed version of FmPro Migrator includes a FREE download link for FileMaker Pro 6 in order to upgrade older files.
1) Select the Help -> Check for Updates... menu item within FmPro Migrator.
The FmPro Migrator licensed software download page will be displayed in your web browser.
2) Download and install the FileMaker Pro 6 Trial Software for your computer operating system.
3) Launch the FileMaker Pro 6 software.
4) Drag & Drop the FileMaker Pro 2,3,4 files onto the FileMaker Pro 6 software window.
Each file will be converted to a new file having the .fp5 file extension used by FileMaker Pro 6.
5) Proceed to the next step - FileMaker Pro 5,6 File Preparation.

FileMaker Pro 5,6 - File Preparation

1) Make a copy of each of your FileMaker Pro files. Changes will need to be made within each file and you always want to be able to return to your original copy in case a problem occurs.
Select File -> Save a Copy As... -> compacted copy (smaller)

2) Rename each saved filename so that it contains no spaces or special characters. The filename gets used as a table name when it is used with the ODBC driver, and spaces and special characters are not permitted. Therefore a filename which was originally:

Asset Management.fp5
should be changed to:
Asset_Management.fp5

If you allowed FileMaker Pro to name the file when you saved a copy, you might have ended up with a filename like this:
Asset Management Copy.fp5
which should be changed to:
Asset_Management.fp5

It is confusing to be dealing with lots of files named ??? Copy.fp5, so just remove the word "Copy" which was appended by FileMaker Pro when you saved the file.

3) Copy all of your files into a convenient location, like a new folder so that you can keep them together. You might name this folder: Modified_Files in order to keep them separate from the original files and the empty copies of the files which will be created later. In fact, you will probably want to create a top-level folder to keep track of all of the files/folders which will be created during the migration project.
So you could create a folder named: FM_Migration_Project
This top-level folder can then serve as the Output Directory which you will select within FmPro Migrator.

Notice that each of the folders doesn't contain spaces. Under some (rare) circumstances FmPro Migrator may need to run migration scripts via the command line and this process only works if there are no spaces within any of the the folder names leading to the Output Directory.

Open each .fp5 file saved into the Modified_Files folder and make the following changes.

4) Delete all ScriptMaker Scripts.
Select the menu:
Scripts -> ScriptMaker...

5) Delete all Relationships.
Select the menu:
File -> Define Relationships...
Note: Relationships are only converted to SQL databases for FileMaker Pro 7+ file versions. You can potentially upgrade each file into an .fp7 file if you need this feature.

6) Delete all Passwords.
Select the menu:
File -> Access Privileges -> Passwords
Make sure that the remaining un-password protected access provides Full Access to the entire file, with all file options checked within the dialog box.

7) Delete all fields having the following types:
Global
Summary
Unstored Calculation

Select the menu:
File -> Define Fields...

It is necessary to delete these fields from large database files having more than a few thousand records because the amount of time required for FileMaker Pro to perform the calculations may exceed the ODBC driver timeout interval. If this problem occurs, no records will be transferred from FileMaker Pro to the destination database.

Note: You may leave the Stored Calculation fields, as the data may be valuable within the destination database.

It will take some work to delete recursive field dependencies. You could make one pass thru the fields from top to bottom to delete all of the fields which can be deleted, and then make another pass thru the fields list from the bottom up to the top of the list.

8) Once all of the unneeded objects have been deleted from each file - make an empty copy of each file. You could store the empty copy of each file within a folder named Empty_Files.
This empty copy of the database files will be used for Step 1 of the migration process where you will Drag & Drop the empty files onto the Step 1 icon within FmPro Migrator. This will cause FmPro Migrator to read the structure of each file to start the migration process.

Select the menu:
File -> Save a Copy As... -> clone (no records)

This empty copy of each file needs to have exactly the same name as the compacted copy you saved previously within the Original_Files folder.
This empty copy of the file will be used by FmPro Migrator for gathering the field info from each file.
Saving an empty copy cleans up the internal structure of the file and reduces the size of the file which needs to be read by FmPro Migrator and speeds up the process compared to reading a full copy of the file.

FileMaker Pro .fp7/.fmp12 - File Preparation

1) Use FileMaker Pro to export a DDR XML file from each .fp7 or .fmp12 file.
Starting with FileMaker Pro 17, the Advanced tools are included at no extra cost and are enabled by checking the "Use Advanced Tools" checkbox in the Preferences dialog. Previous FileMaker

versions required a separate purchase of FileMaker Pro Advanced. For the purposes of brevity, this manual will use assume that you are using FileMaker Pro 17+ versions.

This step is only required if you want FmPro Migrator to export relationships, value lists, layouts from FileMaker Pro into the destination SQL database.

2) Make a copy of each of your FileMaker Pro files. Changes will need to be made within each file and you always want to be able to return to your original copy in case a problem occurs.
Select File -> Save a Copy As... -> compacted copy (smaller)

Saving a compacted copy of the file cleans up the internal structure of the file and reduces the size of the file by removing the indexes within the file.

If you allowed FileMaker Pro to name the file when you saved a copy, you might have ended up with a filename like this:
Asset Management Copy.fp7/.fmp12
which should be changed to:
Asset Management.fp7/.fmp12

It can be confusing to deal with lots of files named ??? Copy.fp7/.fmp12, so just remove the word "Copy" which was appended by FileMaker Pro when you saved the file.

3) Copy all of your files into a convenient location, like a new folder so that you can keep them together. You might name this folder: Modified_Files in order to keep them separate from the original copy of the files. In fact, you will probably want to create a top-level folder to keep track of all of the files/folders which will be created during the migration project.
So you could create a folder named: FM_Migration_Project
This top-level folder can then serve as the Output Directory which you will select within FmPro Migrator.

Notice that each of the folders doesn't contain spaces. Under some (rare) circumstances FmPro Migrator may need to run migration scripts via the command line and this process only works if there are no spaces within any of the the folder names leading to the Output Directory.

Open each .fp7/.fmp12 file saved into the Modified_Files folder and make the following changes.

4) Delete all Script Workspace Scripts.
Select the menu:
Scripts -> Script Workspace...

5) Delete all Relationships and non-base table TOs on the RelationshipGraph
Select the menu:
File -> Define/Manage Database...

Click the Relationships tab.

Note: If it is necessary to change the names of TOs to match base table names, while maintaining the relationship structure of the file, then the DDR XML file should be re-exported after making these changes, but before actually deleting the relationships.

Since the DDR XML file has already been exported, none of the relationships are actually needed for the actual data transfer part of the migration process. The FileMaker ODBC driver needs to see TOs on the RelationshipGraph which exactly match the name of each base table listed on the Tables tab.

If there are a lot of TOs and relationships on the RelationshipGraph it may be easier to delete all of the objects on the RelationshipGraph and then simply recreate the TOs from the existing base tables within the file. This can be done by clicking the Add Table button and selecting the base table name for each base table.

6) Define a Primary Key for each table.
Each table must have a primary key if you want to create relationships for the table in the SQL database or if you want to migrate repeating fields data.

FmPro Migrator identifies the Primary Key for each table by looking for the Unique and Not Empty field validation options being set for any field within the table. If there are primary key fields which don't have these attributes set, they should be set now.

Note: If you are using FileMaker Pro 17+ or FileMaker Pro Advanced, you will be able to copy and paste the table definitions from FileMaker Pro into FmPro Migrator via the Clipboard. FmPro Migrator will then be able to gather all of the attributes for each field. These attributes are also obtained from the file when using AppleScript on macOS.

If the migrated tables don't actually have any field which can serve as a primary key, then a new field should be added as an Auto-Enter Serial Number field within the table. Then each record can be renumbered in Browse mode starting with 1 by clicking in the new field and selecting the Records -> Replace field Contents... menu. Before renumbering the records, select the Records -> Show All Records menu.

Once again, if it is necessary to re-define primary keys and the associated relationships within the database, the DDR XML file should be exported again after making these changes, but before deleting the relationships from within the file.

7) Verify Primary Key and Foreign Key Data Types
It is possible (but not recommended) to create relationships within FileMaker Pro databases between Text and Numeric field types. SQL databases will usually not accept differing data types when creating relationships. The field types can be corrected within the Fields tab of the

Define/Manage Database dialog.

8) Verify Admin Account Access to File
Select the menu:
File -> Manage Accounts & Privileges

In order to transfer data from FileMaker Pro using the ODBC driver, it is necessary for FmPro Migrator to log into the FileMaker Pro database using a password having [Full Access] including ODBC access privileges within the file. The built-in Admin account already has [Full Access] privileges within the file, but these privileges could have been changed after the file was originally created. FileMaker 2,3,4,5,6 files may not even have an Admin account with [Full Access] privileges by default, depending upon how the user accounts were created within the original file.

9) Delete all fields having the following types:
Global
Summary
Unstored Calculation

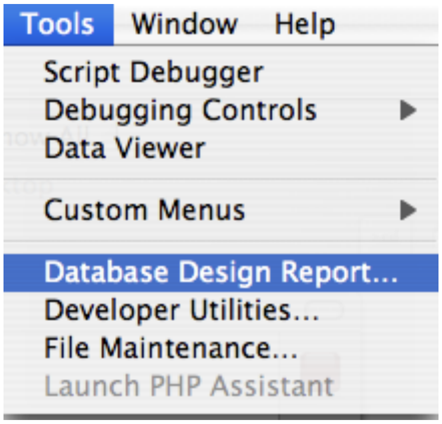
Select the menu:
File -> Define/Manage Database...

It is necessary to delete these fields from large database files having more than a few thousand records because the amount of time required for FileMaker Pro to perform the calculations may exceed the ODBC driver timeout interval. If this problem occurs, no records will be transferred from FileMaker Pro to the destination database. The deletion of Global fields is not mandatory, but since Global fields don't exist within SQL databases they should be removed in order to reduce the field count.

Note: You may leave the Stored Calculation fields, as the data may be valuable within the destination database.

It will take some work to delete recursive field dependencies. You could make one pass thru the fields from top to bottom to delete all of the fields which can be deleted, and then make another pass thru the fields list from the bottom up to the top of the list.

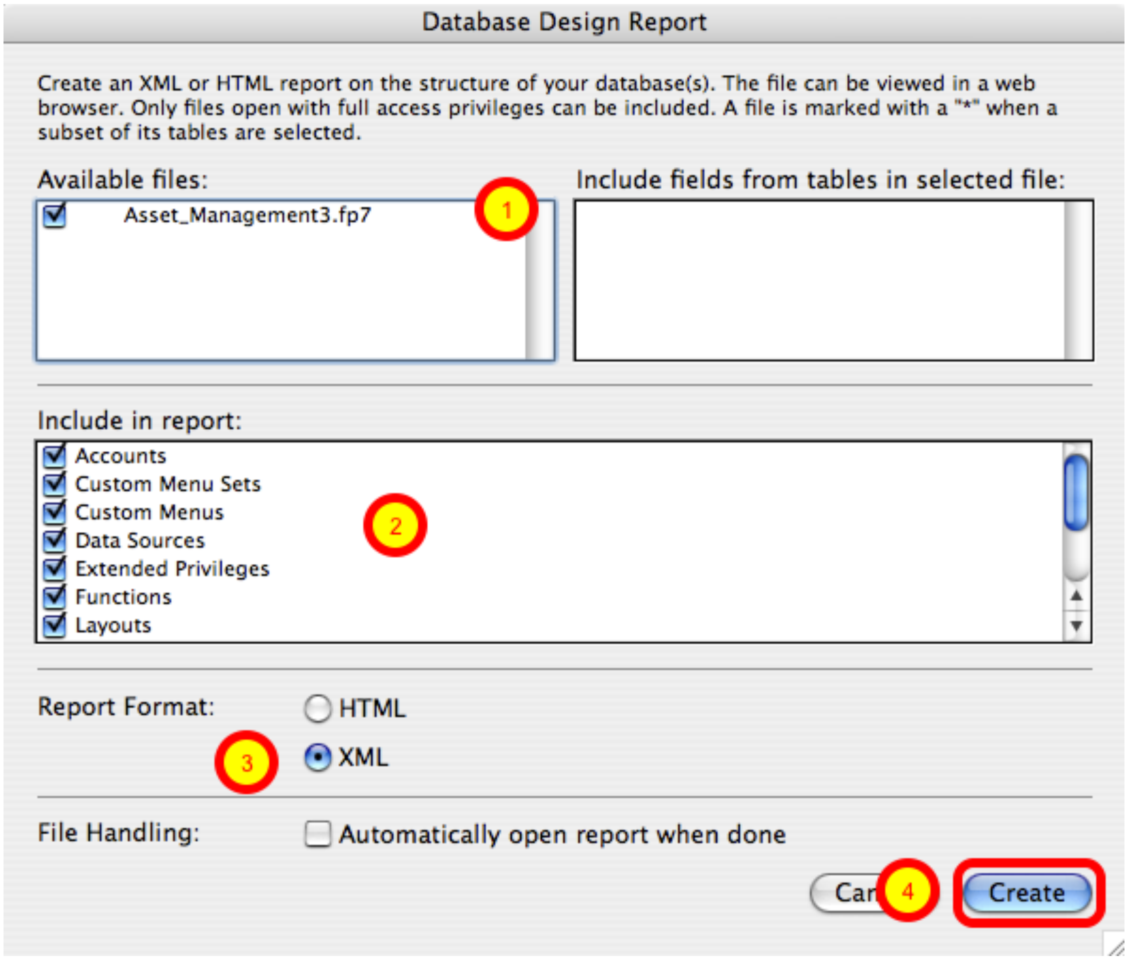
Exporting DDR XML File



Using FileMaker Pro, select Database Design Report... from the Tools menu.

Note: Export the DDR file for the database to be migrated before removing Relationships and Table Occurrences from the Relationship Graph.
This DDR file will be used for importing Relationships, TOs, Value Lists, Custom Functions and Layouts into FmPro Migrator.

Exporting DDR XML File - Selecting Output Options



Database Design Report

Create an XML or HTML report on the structure of your database(s). The file can be viewed in a web browser. Only files open with full access privileges can be included. A file is marked with a "*" when a subset of its tables are selected.

Available files:

☑ Asset_Management3.fp7

Include fields from tables in selected file:

Include in report:

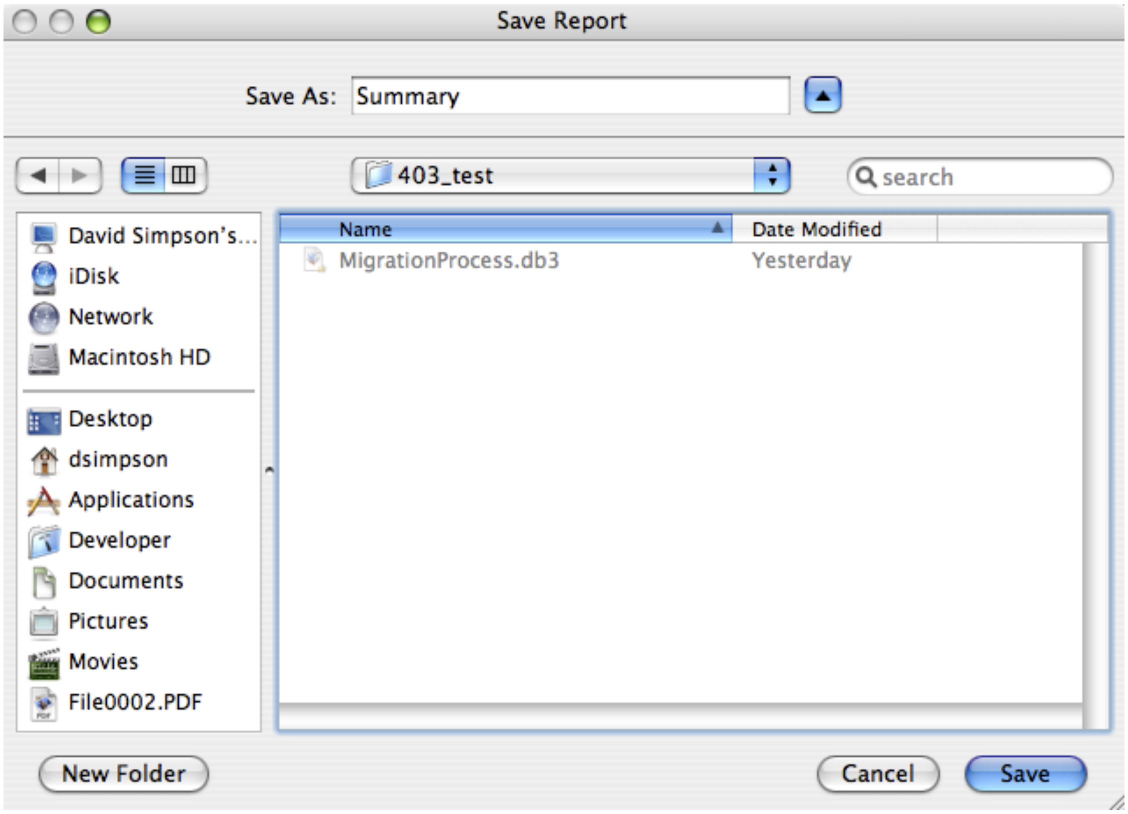
- ☑ Accounts
- ☑ Custom Menu Sets
- ☑ Custom Menus
- ☑ Data Sources
- ☑ Extended Privileges
- ☑ Functions
- ☑ Layouts

Report Format: ○ HTML ● XML

File Handling: □ Automatically open report when done

Within the DDR Export dialog, make sure that the database file is (1) checked for export, (2) along with all objects, (3) XML instead of HTML report format, then click the (4) Create button.

Exporting DDR XML File - Selecting Output Directory



Save Report

Save As: Summary

403_test

Name	Date Modified
MigrationProcess.db3	Yesterday

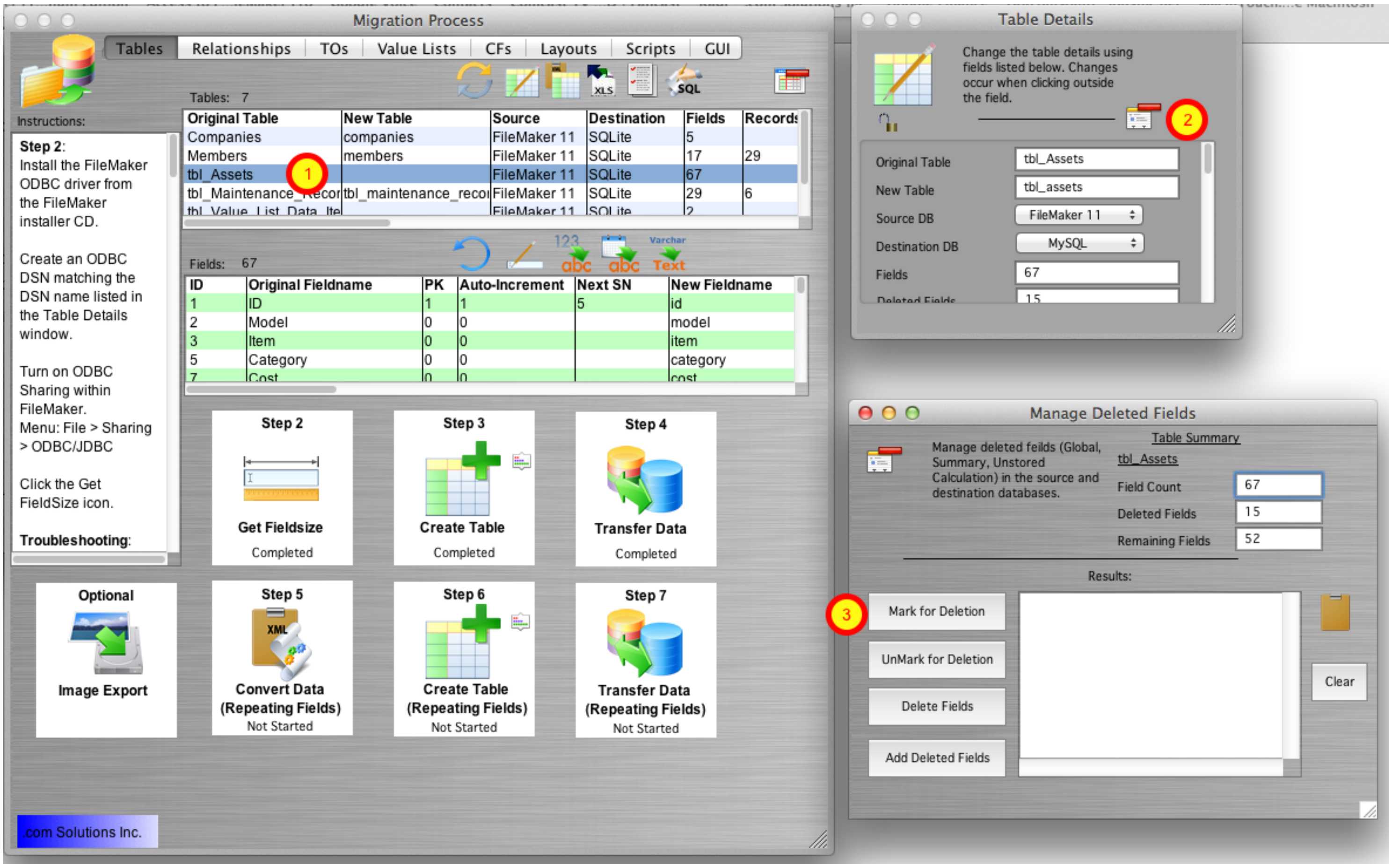
Select the output directory, then click the Save button.

Pre-Migration Task - Delete Unstored Calc Fields

Prior to transferring data from a FileMaker Pro database into a SQL database is is usually necessary to delete all Unstored Calculation and Summary fields from the FileMaker Pro database tables. This step is usually necessary when transferring more than a few thousand records because the amount of time required for FileMaker Pro to calculate these values may cause an ODBC timeout, thus preventing the data from being transferred at all. This step should be completed before Step 2 - Get Fieldsize is run.

FmPro Migrator includes an automated field deletion feature which can help with this task.

Open Manage Deleted Fields Window



Instructions:

Step 2:
Install the FileMaker ODBC driver from the FileMaker installer CD.

Create an ODBC DSN matching the DSN name listed in the Table Details window.

Turn on ODBC Sharing within FileMaker.
Menu: File > Sharing > ODBC/JDBC

Click the Get FieldSize icon.

Troubleshooting:

Tables: 7

Original Table	New Table	Source	Destination	Fields	Records
Companies	companies	FileMaker 11	SQLite	5	
Members	members	FileMaker 11	SQLite	17	29
tbl_Assets		FileMaker 11	SQLite	67	
tbl_Maintenance_Recor	tbl_maintenance_recor	FileMaker 11	SQLite	29	6
tbl_Value_List_Data_Ite		FileMaker 11	SQLite	2	

Fields: 67

ID	Original Fieldname	PK	Auto-Increment	Next SN	New Fieldname
1	ID	1	1	5	id
2	Model	0	0		model
3	Item	0	0		item
5	Category	0	0		category
7	Cost	0	0		cost

Table Details

Change the table details using fields listed below. Changes occur when clicking outside the field.

Original Table: tbl_Assets
New Table: tbl_assets
Source DB: FileMaker 11
Destination DB: MySQL
Fields: 67
Deleted Fields: 15

Manage Deleted Fields

Manage deleted feilds (Global, Summary, Unstored Calculation) in the source and destination databases.

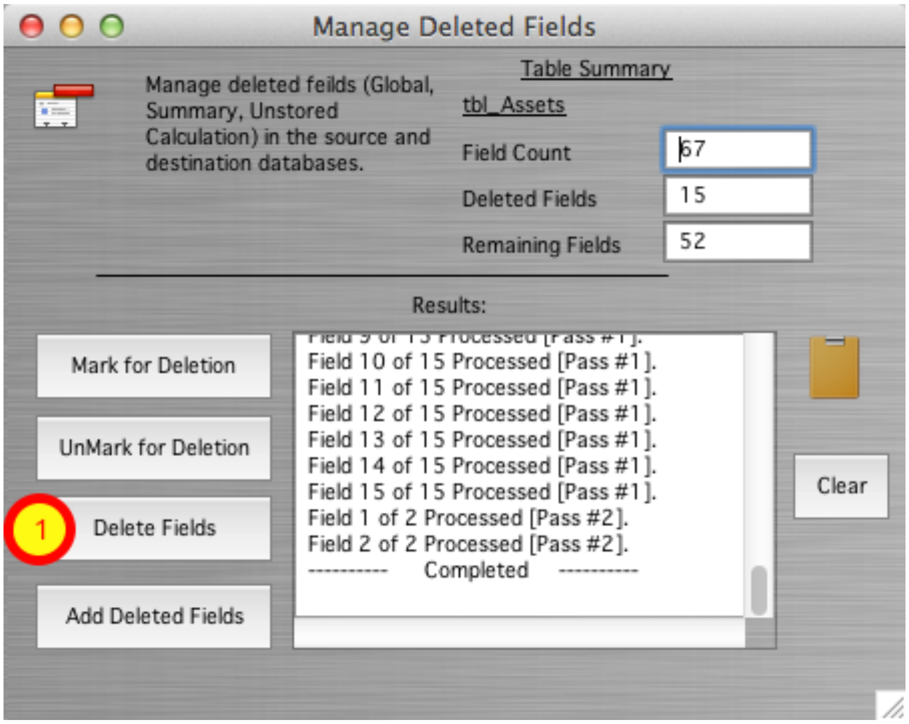
Table Summary
tbl_Assets
Field Count: 67
Deleted Fields: 15
Remaining Fields: 52

Results:

Once the FileMaker table definitions have been imported via the clipboard, the list of tables will be displayed within the Migration Process window.
1) Double-click on the table name, the Table Details window will be displayed. Click on the Manage Deleted Fields button near the top of the Table Details window, the Manage Deleted Fields window will open.
3) Click on the Mark for Deletion Button, all of the Unstored Calculation, Summary and Global fields in the selected table will be marked for deletion.

Note: Global fields are also marked for deletion in order to reduce the number of fields of data to speed up the data transfer process. For the example table shown in this screenshot, there were 15 fields marked for deletion. At this point, the fields have not yet been deleted, they have just been marked for deletion.

Delete Fields



Manage deleted feilds (Global, Summary, Unstored Calculation) in the source and destination databases.

Table Summary

tbl_Assets

Field Count 67

Deleted Fields 15

Remaining Fields 52

Results:

```
Field 9 of 15 Processed [Pass #1].
Field 10 of 15 Processed [Pass #1].
Field 11 of 15 Processed [Pass #1].
Field 12 of 15 Processed [Pass #1].
Field 13 of 15 Processed [Pass #1].
Field 14 of 15 Processed [Pass #1].
Field 15 of 15 Processed [Pass #1].
Field 1 of 2 Processed [Pass #2].
Field 2 of 2 Processed [Pass #2].
----------     Completed     ----------
```

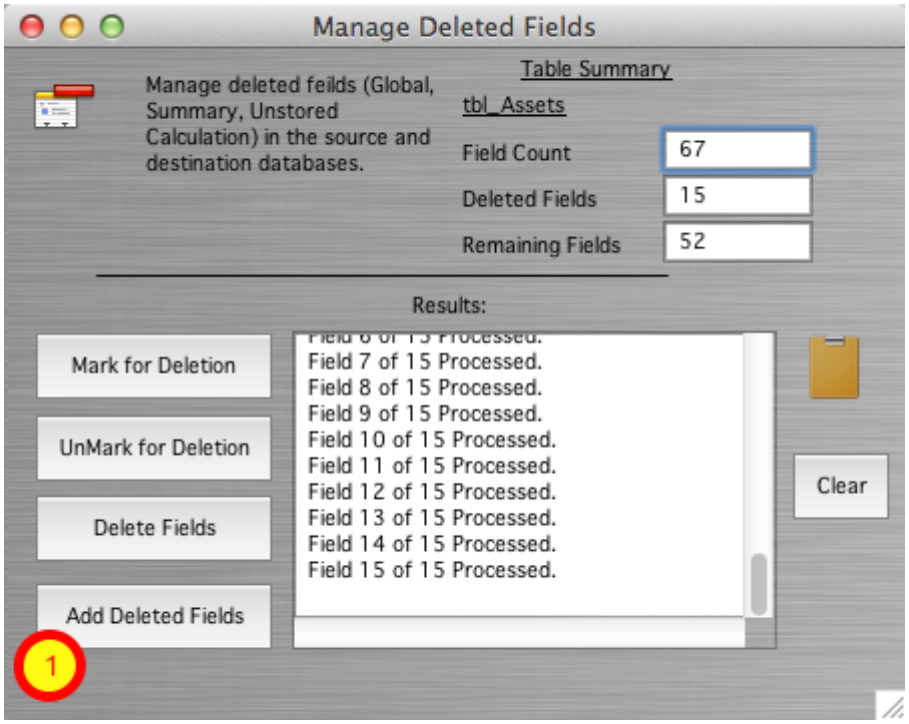
1) Click the Delete Fields button. FmPro Migrator will first print a list of the ALTER TABLE SQL commands in the Results window, then the ALTER TABLE commands will be sent directly to the FileMaker database thru the ODBC connection. An example of these commands is listed below.

Note: It is common for some DROP COLUMN commands to fail as a result of Outstanding References by other fields. In order to resolve this issue, FmPro Migrator logs the failed DROP COLUMN commands and then re-sends those commands to the FileMaker Pro database again, in reverse order during Pass #2 processing. If PASS #2 still results in DROP COLUMN command failures, the remaining list of undeleted fields will be listed at the end of the processing.

Click on the Clipboard icon to copy the commands any other software for review purposes. It will be helpful to have these field names available when deleting the fields manually. It is necessary to manually delete any fields which were not deleted automatically in order to proceed with Step #2 - Get Fieldsize.

```
ALTER TABLE "tbl_Assets" DROP COLUMN "Total Cost"
ALTER TABLE "tbl_Assets" DROP COLUMN "Total Book Value"
ALTER TABLE "tbl_Assets" DROP COLUMN "Total Depreciation"
ALTER TABLE "tbl_Assets" DROP COLUMN "Template Information Global"
ALTER TABLE "tbl_Assets" DROP COLUMN "HiliteLibrary"
ALTER TABLE "tbl_Assets" DROP COLUMN "HiliteSortedBy"
ALTER TABLE "tbl_Assets" DROP COLUMN "HiliteCategory"
ALTER TABLE "tbl_Assets" DROP COLUMN "HiliteAssignedTo"
ALTER TABLE "tbl_Assets" DROP COLUMN "HiliteItem"
ALTER TABLE "tbl_Assets" DROP COLUMN "HiliteModel"
ALTER TABLE "tbl_Assets" DROP COLUMN "HiliteSerialNumber"
ALTER TABLE "tbl_Assets" DROP COLUMN "HiliteLocation"
ALTER TABLE "tbl_Assets" DROP COLUMN "SC_Full_URL_ReadWrite"
ALTER TABLE "tbl_Assets" DROP COLUMN "SC_Full_URL_ReadOnly"
ALTER TABLE "tbl_Assets" DROP COLUMN "SC_URL_Prefix"
Field 1 of 15 Processed [Pass #1].
Field 2 of 15 Processed [Pass #1].
Field 3 of 15 Processed [Pass #1].
Field 4 of 15 Processed [Pass #1].
Field 5 of 15 Processed [Pass #1].
ALTER TABLE "tbl_Assets" DROP COLUMN "HiliteLibrary"
[FileMaker][FileMaker]  (13): Outstanding references
Field 6 of 15 Processed [Pass #1].
ALTER TABLE "tbl_Assets" DROP COLUMN "HiliteSortedBy"
[FileMaker][FileMaker]  (13): Outstanding references
Field 7 of 15 Processed [Pass #1].
Field 8 of 15 Processed [Pass #1].
Field 9 of 15 Processed [Pass #1].
Field 10 of 15 Processed [Pass #1].
Field 11 of 15 Processed [Pass #1].
Field 12 of 15 Processed [Pass #1].
Field 13 of 15 Processed [Pass #1].
Field 14 of 15 Processed [Pass #1].
Field 15 of 15 Processed [Pass #1].
Field 1 of 2 Processed [Pass #2].
Field 2 of 2 Processed [Pass #2].
----------     Completed     ----------
```

Adding Deleted Fields to the SQL Database [PHP & LiveCode Conversions]



Manage Deleted Fields

Manage deleted feilds (Global, Summary, Unstored Calculation) in the source and destination databases.

Table Summary

tbl_Assets

Field Count: 67

Deleted Fields: 15

Remaining Fields: 52

Results:

```
Field 6 of 15 Processed.
Field 7 of 15 Processed.
Field 8 of 15 Processed.
Field 9 of 15 Processed.
Field 10 of 15 Processed.
Field 11 of 15 Processed.
Field 12 of 15 Processed.
Field 13 of 15 Processed.
Field 14 of 15 Processed.
Field 15 of 15 Processed.
```

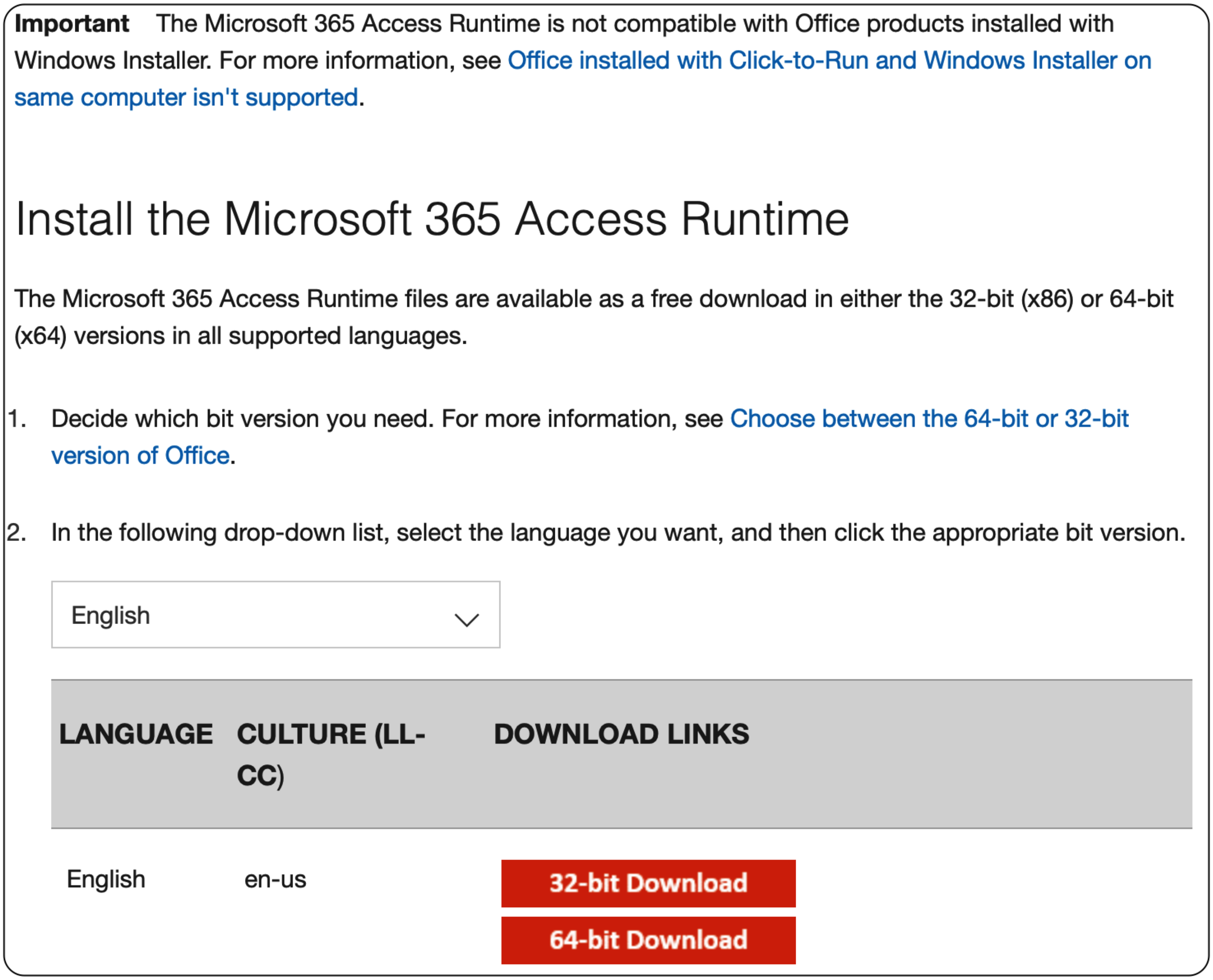
After the table has been created in the SQL database, and the data transferred to the SQL database, the deleted columns can then be added to the SQL database table.
Note: This step is only needed if a PHP Conversion project is being generated, in order to insure that the database columns referenced in the CakePHP Model files exist within the database table.

```
ALTER TABLE tbl_assets ADD total_cost DOUBLE NULL;
ALTER TABLE tbl_assets ADD total_book_value DOUBLE NULL;
ALTER TABLE tbl_assets ADD total_depreciation DOUBLE NULL;
ALTER TABLE tbl_assets ADD template_information_global TEXT NULL;
ALTER TABLE tbl_assets ADD hilitelibrary BLOB NULL;
ALTER TABLE tbl_assets ADD hilitesortedby TEXT NULL;
ALTER TABLE tbl_assets ADD hilitecategory BLOB NULL;
ALTER TABLE tbl_assets ADD hiliteassignedto BLOB NULL;
ALTER TABLE tbl_assets ADD hiliteitem BLOB NULL;
ALTER TABLE tbl_assets ADD hilitemodel BLOB NULL;
ALTER TABLE tbl_assets ADD hiliteserialnumber BLOB NULL;
ALTER TABLE tbl_assets ADD hilitelocation BLOB NULL;
ALTER TABLE tbl_assets ADD sc_full_url_readwrite TEXT NULL;
ALTER TABLE tbl_assets ADD sc_full_url_readonly TEXT NULL;
ALTER TABLE tbl_assets ADD sc_url_prefix TEXT NULL;
Field 1 of 15 Processed.
Field 2 of 15 Processed.
Field 3 of 15 Processed.
Field 4 of 15 Processed.
Field 5 of 15 Processed.
Field 6 of 15 Processed.
Field 7 of 15 Processed.
Field 8 of 15 Processed.
Field 9 of 15 Processed.
Field 10 of 15 Processed.
Field 11 of 15 Processed.
Field 12 of 15 Processed.
Field 13 of 15 Processed.
Field 14 of 15 Processed.
Field 15 of 15 Processed.
```

Office 365 Compatibility Problem

Microsoft Office 365 installs with a Click 2 Run installer, this sandboxed installation is incompatible with the installation of the ACEODBC.dll file required to provide support for reading or writing Access .accdb database files by external applications via ODBC.
When an error occurs while connecting to an Access .accdb database file, FmPro Migrator checks the registry to verify whether Office 365 has been installed. This verification checks for the ClickToRunStore registry value within the Office installation. Then the appropriate error message is displayed.

Solution #1 - Install Microsoft 365 Access Runtime



Important The Microsoft 365 Access Runtime is not compatible with Office products installed with Windows Installer. For more information, see Office installed with Click-to-Run and Windows Installer on same computer isn't supported.

Install the Microsoft 365 Access Runtime

The Microsoft 365 Access Runtime files are available as a free download in either the 32-bit (x86) or 64-bit (x64) versions in all supported languages.

1. Decide which bit version you need. For more information, see Choose between the 64-bit or 32-bit version of Office.

2. In the following drop-down list, select the language you want, and then click the appropriate bit version.

English

LANGUAGE	CULTURE (LL-CC)	DOWNLOAD LINKS
English	en-us	32-bit Download 64-bit Download

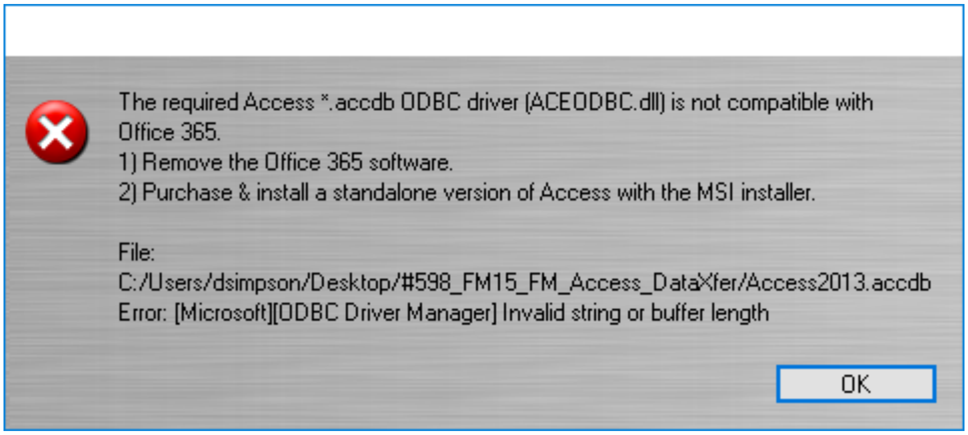
In August 2021, Microsoft released the Microsoft 365 Runtime, which is designed to solve this problem for OLEDB and ODBC connections to Access database files. Before ordering a new MSI installer version of Access, you may try installing this runtime.

But after installing the runtime, you will probably need to uninstall the runtime - which will leave the ODBC driver in place. Testing after installing the runtime caused problems with Access which prevented seeing any of the tables/fields or data within the database since the runtime was being used instead of the full copy of the Access database application.

The URL for this Microsoft 365 Runtime is currently located as follows:

https://support.microsoft.com/en-us/office/download-and-install-microsoft-365-access-runtime-185c5a

FmPro Migrator Office 365 Error Dialog - Importing .accdb Files

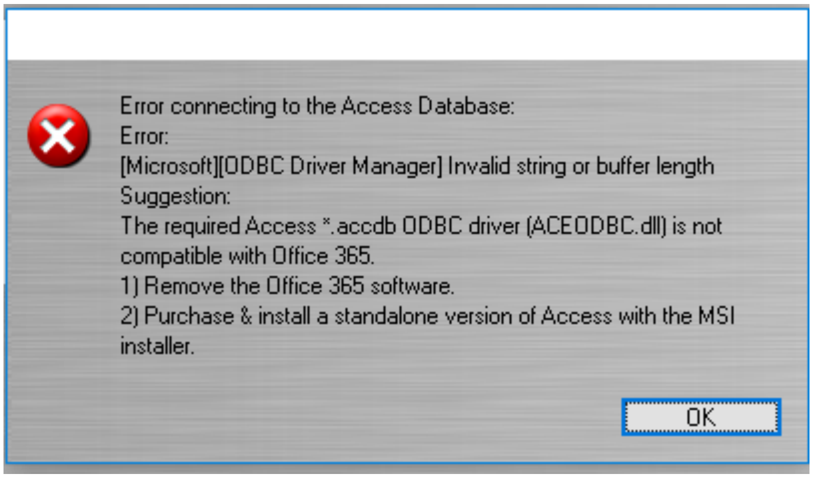


The required Access *.accdb ODBC driver (ACEODBC.dll) is not compatible with Office 365.
1) Remove the Office 365 software.
2) Purchase & install a standalone version of Access with the MSI installer.

File:
C:/Users/dsimpson/Desktop/#598_FM15_FM_Access_DataXfer/Access2013.accdb
Error: [Microsoft][ODBC Driver Manager] Invalid string or buffer length

FmPro Migrator needs to make a File DSN connection to an .accdb file to read information about tables and fields within the database file. If the process fails due to having Office 365 installed, this error dialog will be displayed.

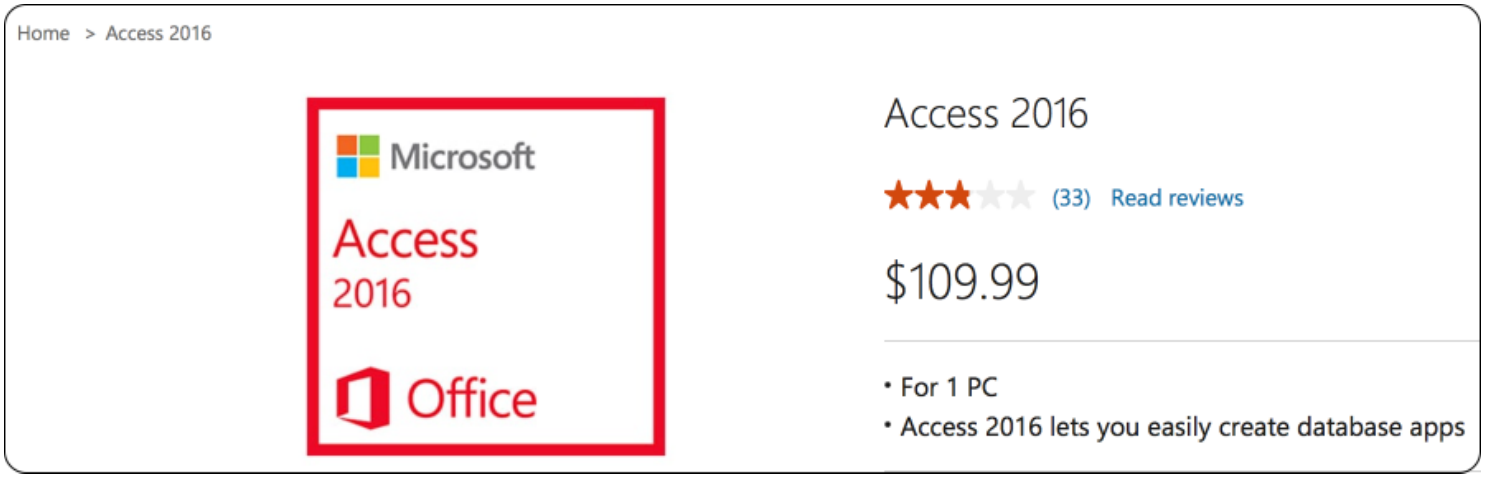
FmPro Migrator Office 365 Error Dialog - Exporting to .accdb Files



Error connecting to the Access Database:
Error:
[Microsoft][ODBC Driver Manager] Invalid string or buffer length
Suggestion:
The required Access *.accdb ODBC driver (ACEODBC.dll) is not compatible with Office 365.
1) Remove the Office 365 software.
2) Purchase & install a standalone version of Access with the MSI installer.

When creating a table or transferring data into an Access .accdb file this dialog will be displayed if Office 365 is installed.

Solution #2: Buy Access Standalone Version or Bundled Microsoft Office Pro Version


Home > Access 2016

Access 2016

★★★☆☆ (33) Read reviews

$109.99

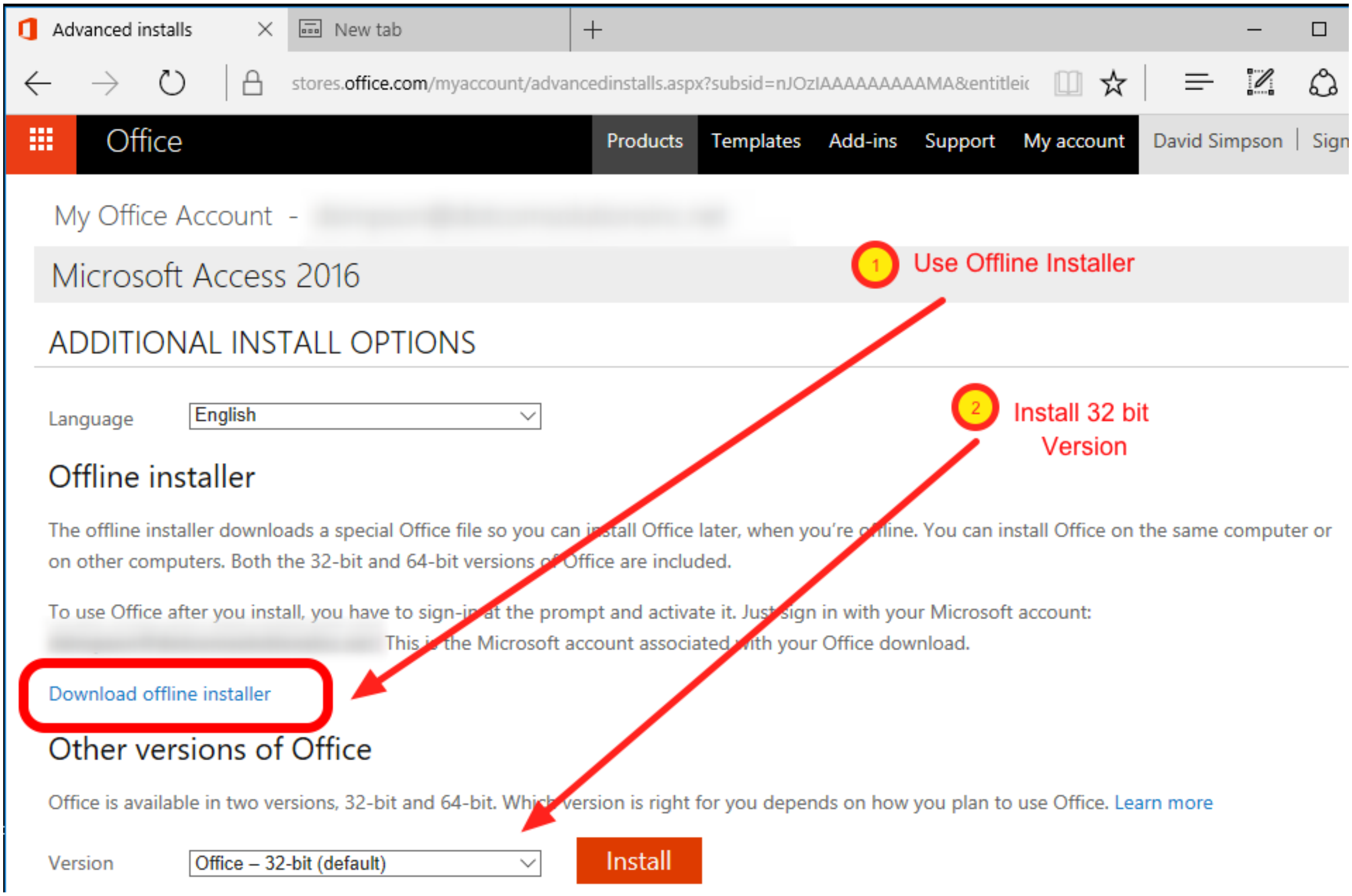
- For 1 PC
- Access 2016 lets you easily create database apps

The only solution to this problem, according to Microsoft is to purchase a new standalone copy of Microsoft Access, or buy the regular Office Pro bundle which includes Access. Using Microsoft Volume Licensing will also avoid Office 365, and will install Access onto your computer using the standard MSI installer.

You must avoid having any components of Office 365 installed on your computer when you perform the installation of the standalone version of Access. It may be necessary to get a special tool from Microsoft in order to remove every last part of the Office 365 software (or use a 3rd party removal tool), then restart Windows. One free removal tool from Microsoft is:

o15-ctrremove.diagcab

Access 2016 Advanced Install Options - Offline Installer



My Office Account - [redacted]

Microsoft Access 2016

ADDITIONAL INSTALL OPTIONS

Language: English

Offline installer

The offline installer downloads a special Office file so you can install Office later, when you're offline. You can install Office on the same computer or on other computers. Both the 32-bit and 64-bit versions of Office are included.

To use Office after you install, you have to sign-in at the prompt and activate it. Just sign in with your Microsoft account: [redacted] This is the Microsoft account associated with your Office download.

Download offline installer

Other versions of Office

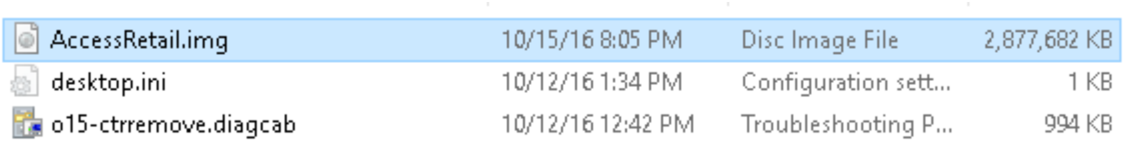
Office is available in two versions, 32-bit and 64-bit. Which version is right for you depends on how you plan to use Office. Learn more

Version: Office – 32-bit (default)

By default, Microsoft will use the ClickToRun installer. You should download the offline installer.

This screenshot shows installing a 32-bit version. FmPro Migrator is available in both 32-bit and 64-bit versions which are both included with your purchase. You should match the version of FileMaker Pro which you have isntalled. For instance, if the copy of FileMaker Pro is 64-bit, then all of the rest of the software should also be 64-bit.

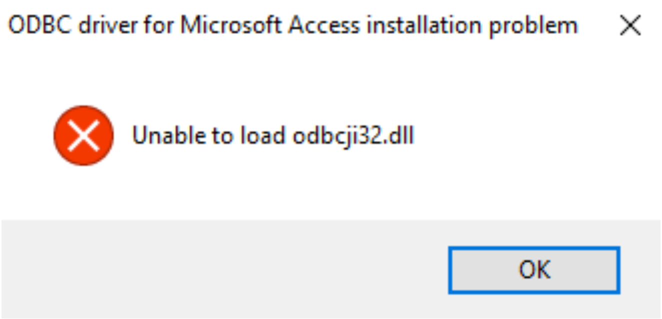
AccessRetail.img Disk Image File Downloaded



AccessRetail.img	10/15/16 8:05 PM	Disc Image File	2,877,682 KB
desktop.ini	10/12/16 1:34 PM	Configuration sett...	1 KB
o15-ctrremove.diagcab	10/12/16 12:42 PM	Troubleshooting P...	994 KB

The AccessRetail.img disk image file should be found in your downloads folder. This screenshot shows the Access 2016 disk image file.

Access ODBC Driver Error - Unable to load odbcji32.dll



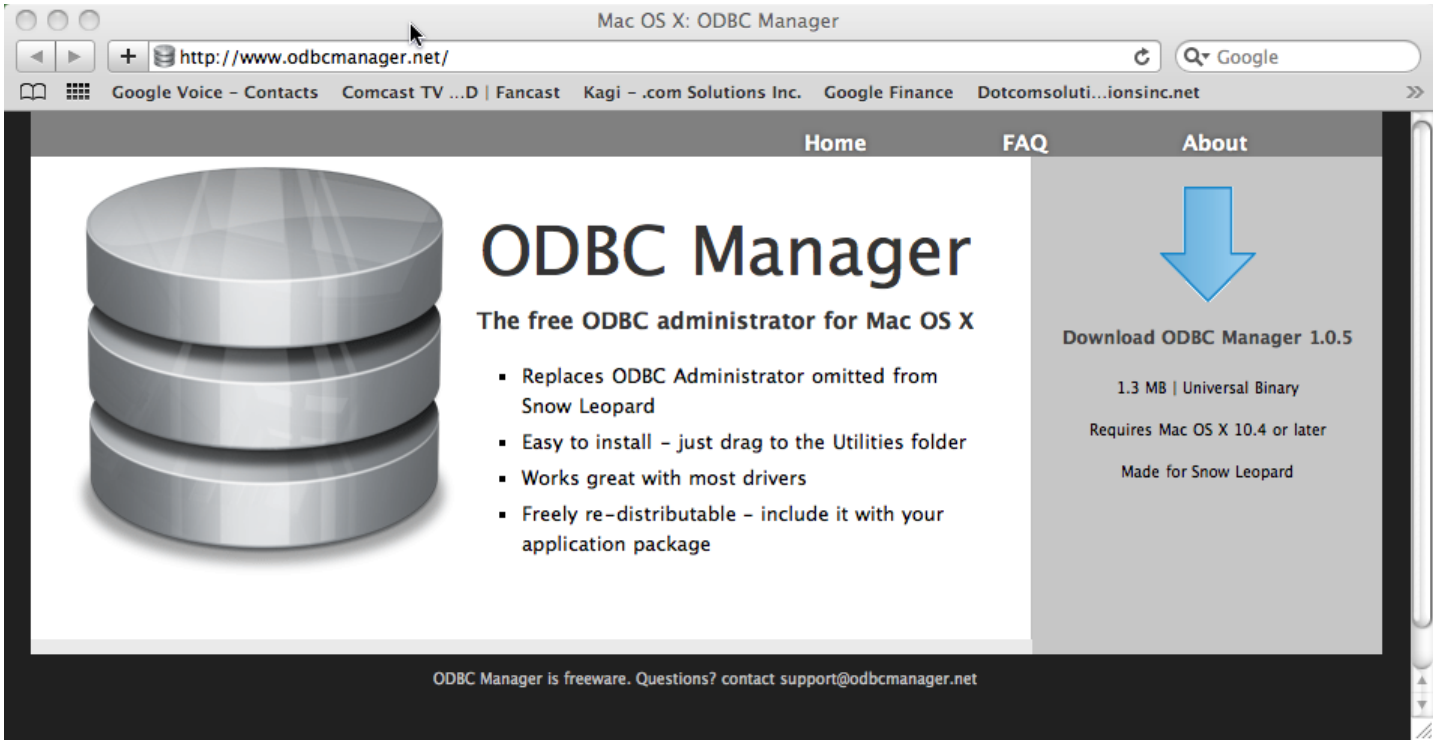
ODBC driver for Microsoft Access installation problem

Unable to load odbcji32.dll

Solution:
1) Download Access Database Engine 2010 from:
https://www.microsoft.com/en-us/download/confirmation.aspx?id=13255
2) Run the AccessDatabaseEngine.exe executable to install or repair the Access Database Engine files.

ODBC Driver Installation

Download ODBC Manager



Mac OS X: ODBC Manager

http://www.odbcmanager.net/

ODBC Manager

The free ODBC administrator for Mac OS X

- Replaces ODBC Administrator omitted from Snow Leopard
- Easy to install – just drag to the Utilities folder
- Works great with most drivers
- Freely re-distributable – include it with your application package

Download ODBC Manager 1.0.5

1.3 MB | Universal Binary

Requires Mac OS X 10.4 or later

Made for Snow Leopard

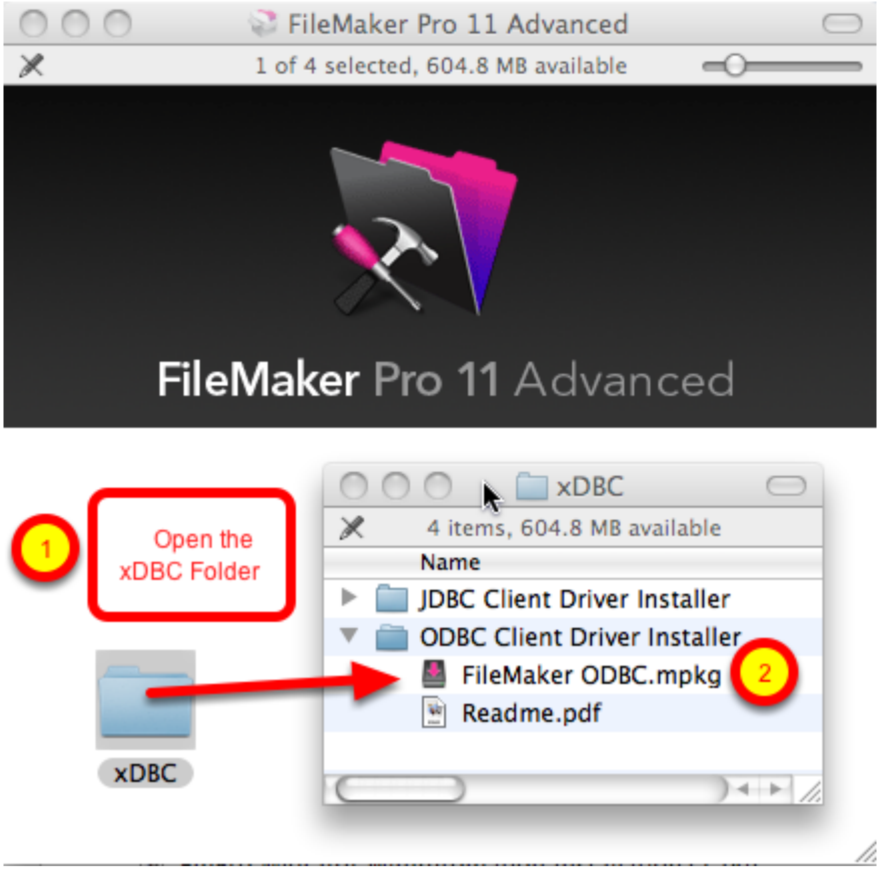
ODBC Manager is freeware. Questions? contact support@odbcmanager.net

Google
Google Voice – Contacts
Comcast TV ...D | Fancast
Kagi – .com Solutions Inc.
Google Finance
Dotcomsoluti...ionsinc.net
Home
FAQ
About

Download and install the free ODBC Administrator for macOS from www.odbcmanager.net

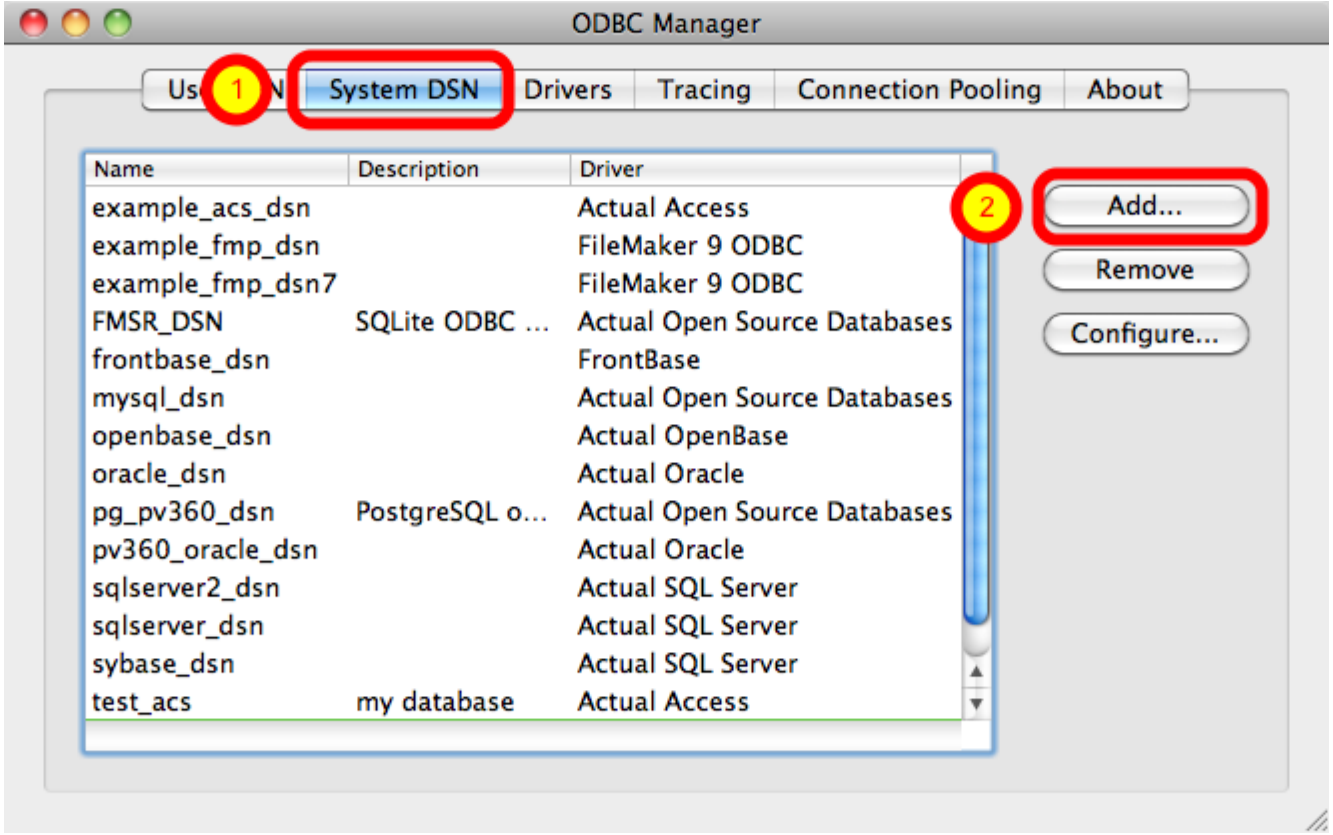
You will use the ODBC Administrator to create and manage your ODBC DSNs on macOS.

Install the FileMaker ODBC Driver


FileMaker Pro 11 Advanced
1 of 4 selected, 604.8 MB available
FileMaker Pro 11 Advanced
1
Open the xDBC Folder
xDBC
xDBC
4 items, 604.8 MB available
Name
JDBC Client Driver Installer
ODBC Client Driver Installer
FileMaker ODBC.mpkg
2
Readme.pdf

(1) Open the xDBC folder on the FileMaker install CD (or installer image DMG file).
(2) Double-click the FileMaker ODBC.mpkg installer to install the FileMaker ODBC driver. Follow the insular prompts.

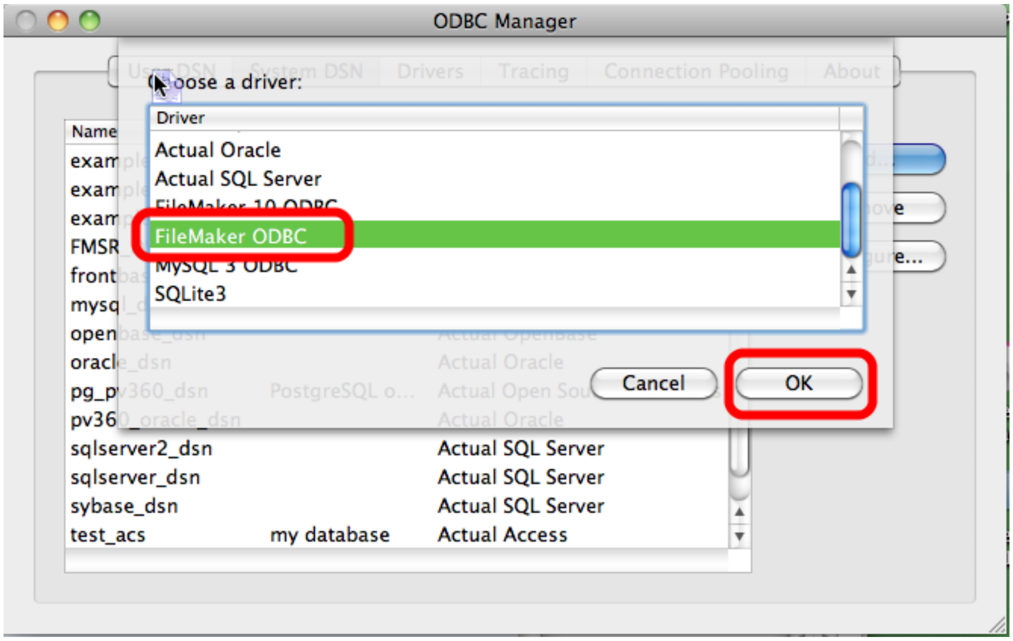
Add a System DSN



Name	Description	Driver
example_acs_dsn		Actual Access
example_fmp_dsn		FileMaker 9 ODBC
example_fmp_dsn7		FileMaker 9 ODBC
FMSR_DSN	SQLite ODBC ...	Actual Open Source Databases
frontbase_dsn		FrontBase
mysql_dsn		Actual Open Source Databases
openbase_dsn		Actual OpenBase
oracle_dsn		Actual Oracle
pg_pv360_dsn	PostgreSQL o...	Actual Open Source Databases
pv360_oracle_dsn		Actual Oracle
sqlserver2_dsn		Actual SQL Server
sqlserver_dsn		Actual SQL Server
sybase_dsn		Actual SQL Server
test_acs	my database	Actual Access

(1) Click the System DSN tab, then (2) click the Add button.

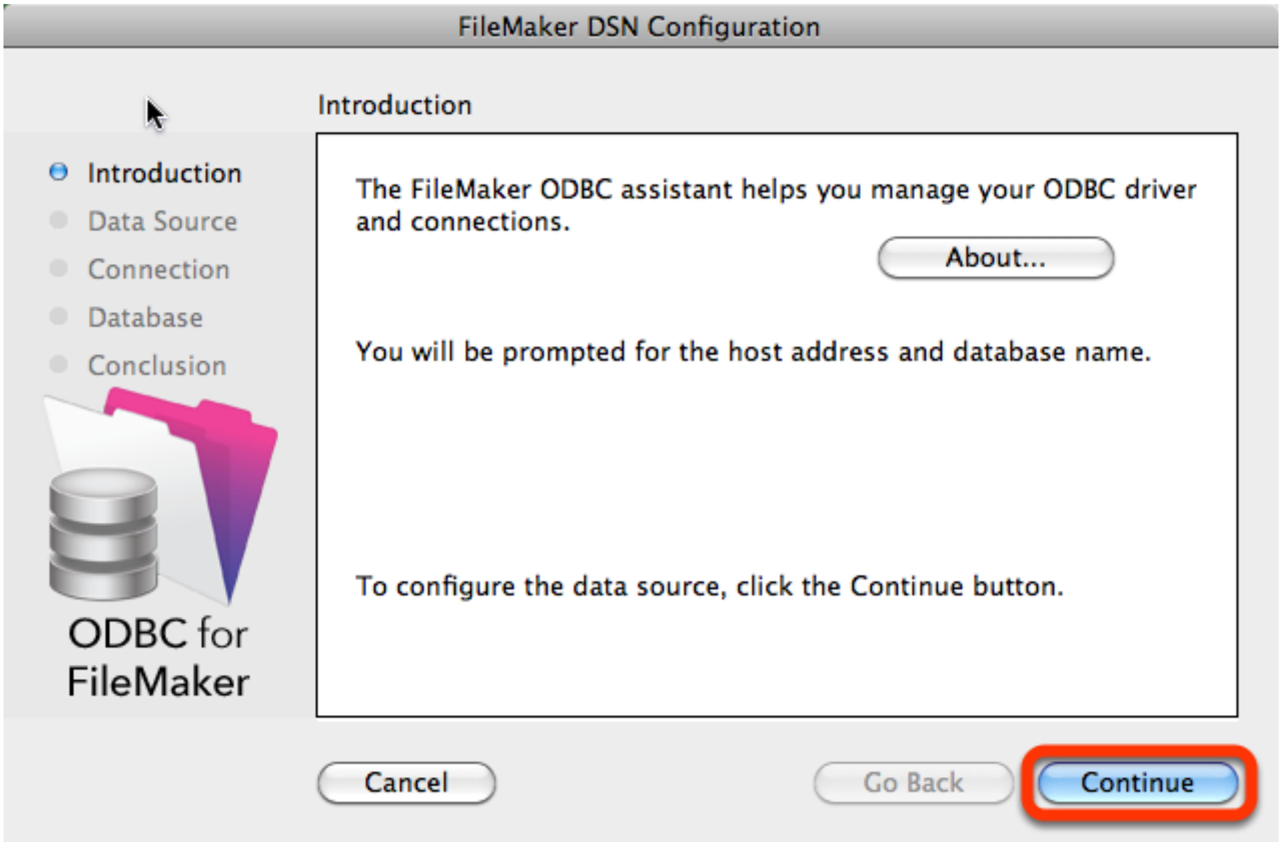
Choose the FileMaker ODBC Driver



Driver
Actual Oracle
Actual SQL Server
FileMaker 10 ODBC
FileMaker ODBC
MySQL 3 ODBC
SQLite3

(1) Select the FileMaker ODBC driver from the list, (2) click the Ok button.

FileMaker ODBC DSN Configuration - Page 1



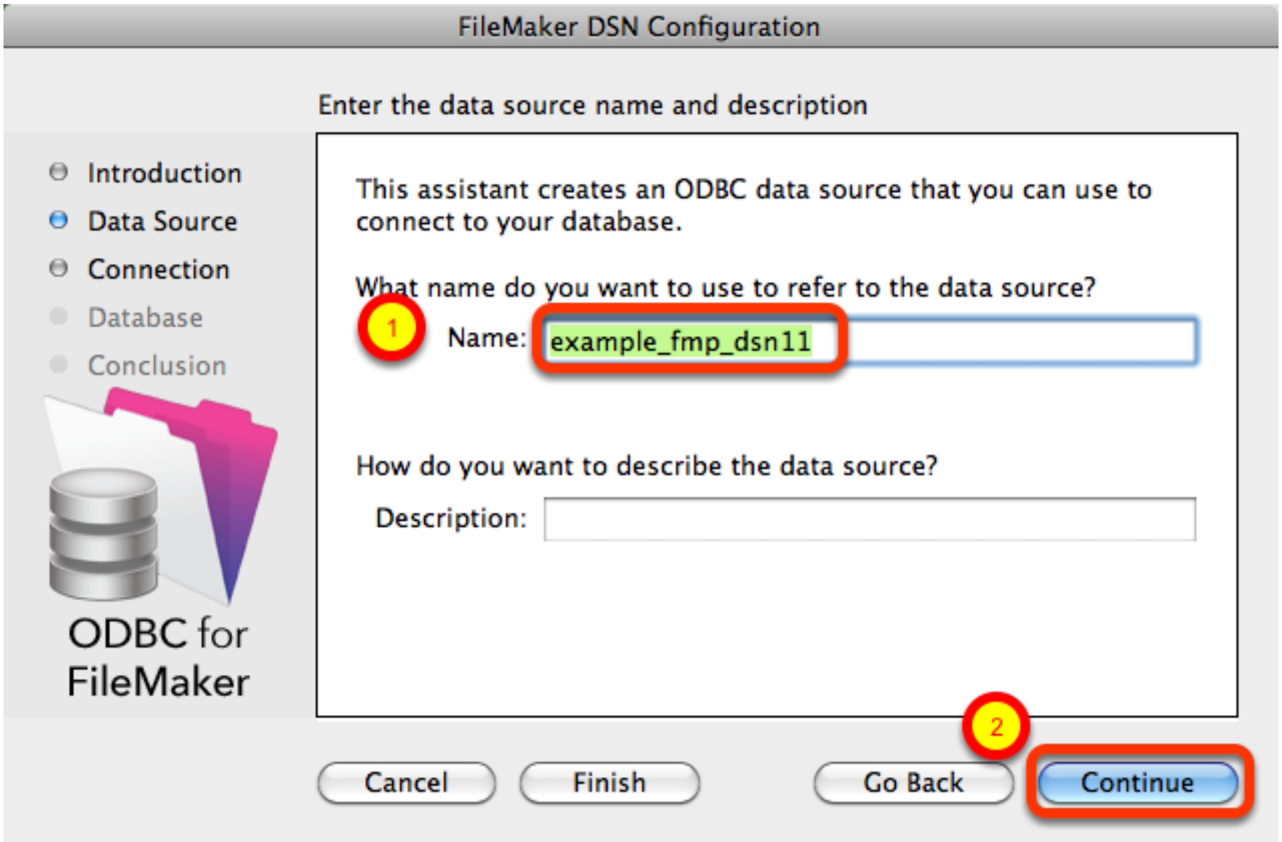
The FileMaker ODBC assistant helps you manage your ODBC driver and connections.

You will be prompted for the host address and database name.

To configure the data source, click the Continue button.

Click the Continue button.

FileMaker ODBC DSN Configuration - Page 2



This assistant creates an ODBC data source that you can use to connect to your database.

What name do you want to use to refer to the data source?

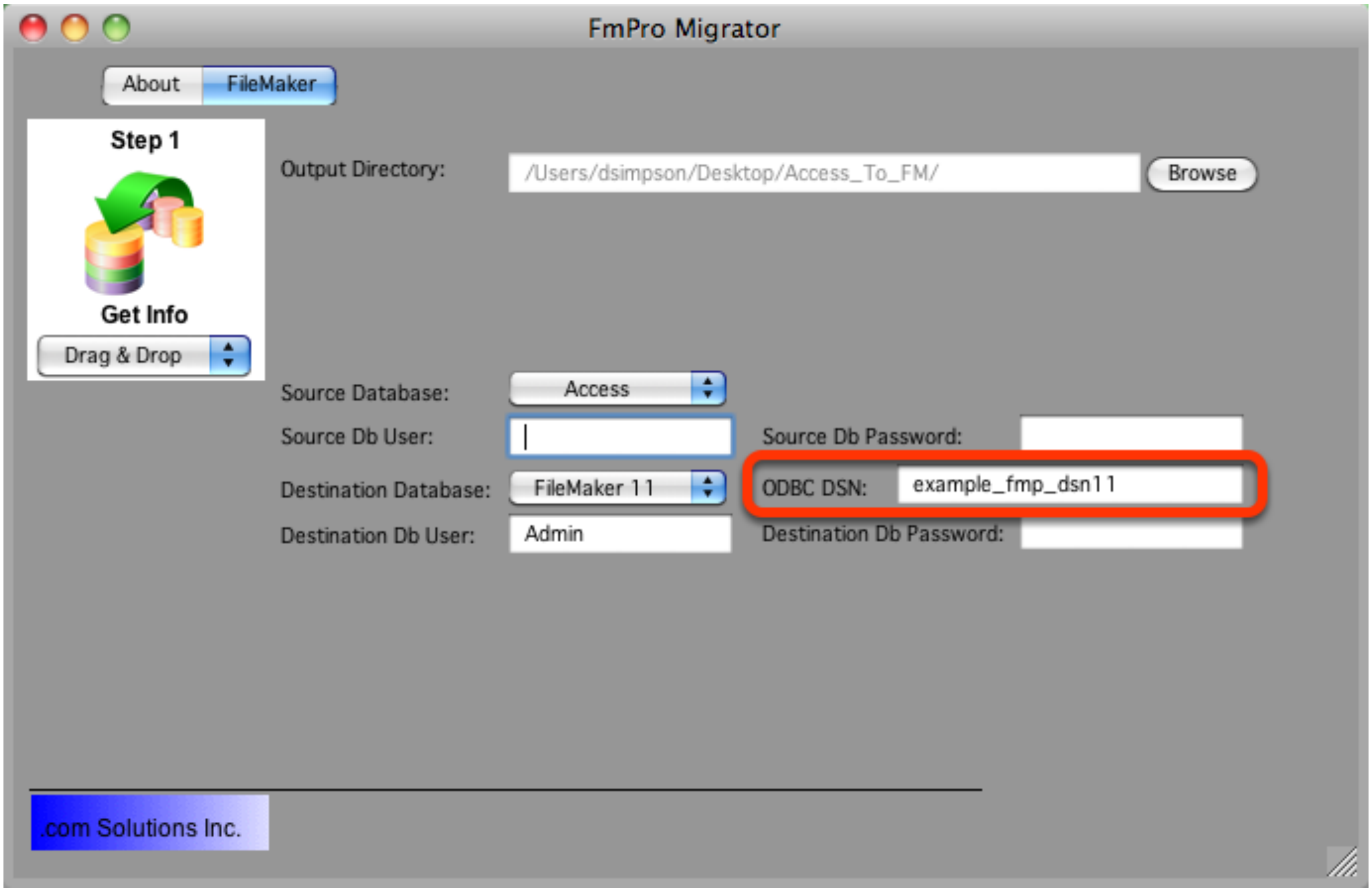
Name: example_fmp_dsn11

How do you want to describe the data source?

Description:

(1) Enter a name for the FileMaker ODBC DSN. The name used here needs to be exactly the same as the ODBC DSN Name you enter into FmPro Migrator.
(2) Click the Continue button.
Note: You can pick a different name than you see here, just make sure that you enter exactly the same name into FmPro Migrator.

FileMaker ODBC DSN Configuration - ODBC DSN Field in FmPro Migrator



Output Directory: /Users/dsimpson/Desktop/Access_To_FM/

Source Database: Access

Source Db User:

Source Db Password:

Destination Database: FileMaker 11

ODBC DSN: example_fmp_dsn11

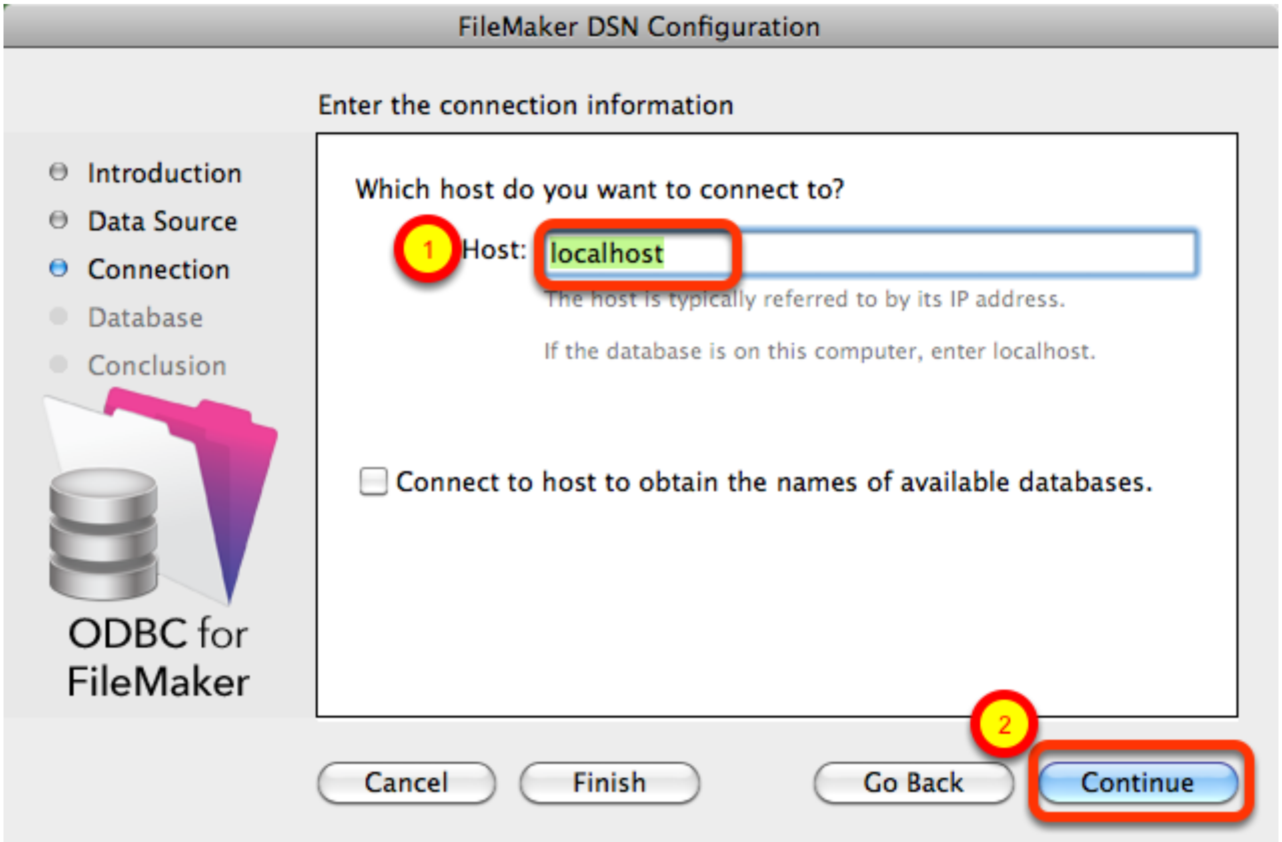
Destination Db User: Admin

Destination Db Password:

In this screenshot, the FileMaker 11 database is the destination database, therefore the FileMaker ODBC DSN name is entered in the Destination Database ODBC DSN field.

If the FileMaker database was the source database, and the data was being copied to a different database server, then the FileMaker ODBC DSN would be entered into the ODBC DSN field for the Source Database (not shown in this screenshot).

FileMaker ODBC DSN Configuration - Page 3



FileMaker DSN Configuration

Enter the connection information

Which host do you want to connect to?

Host: localhost

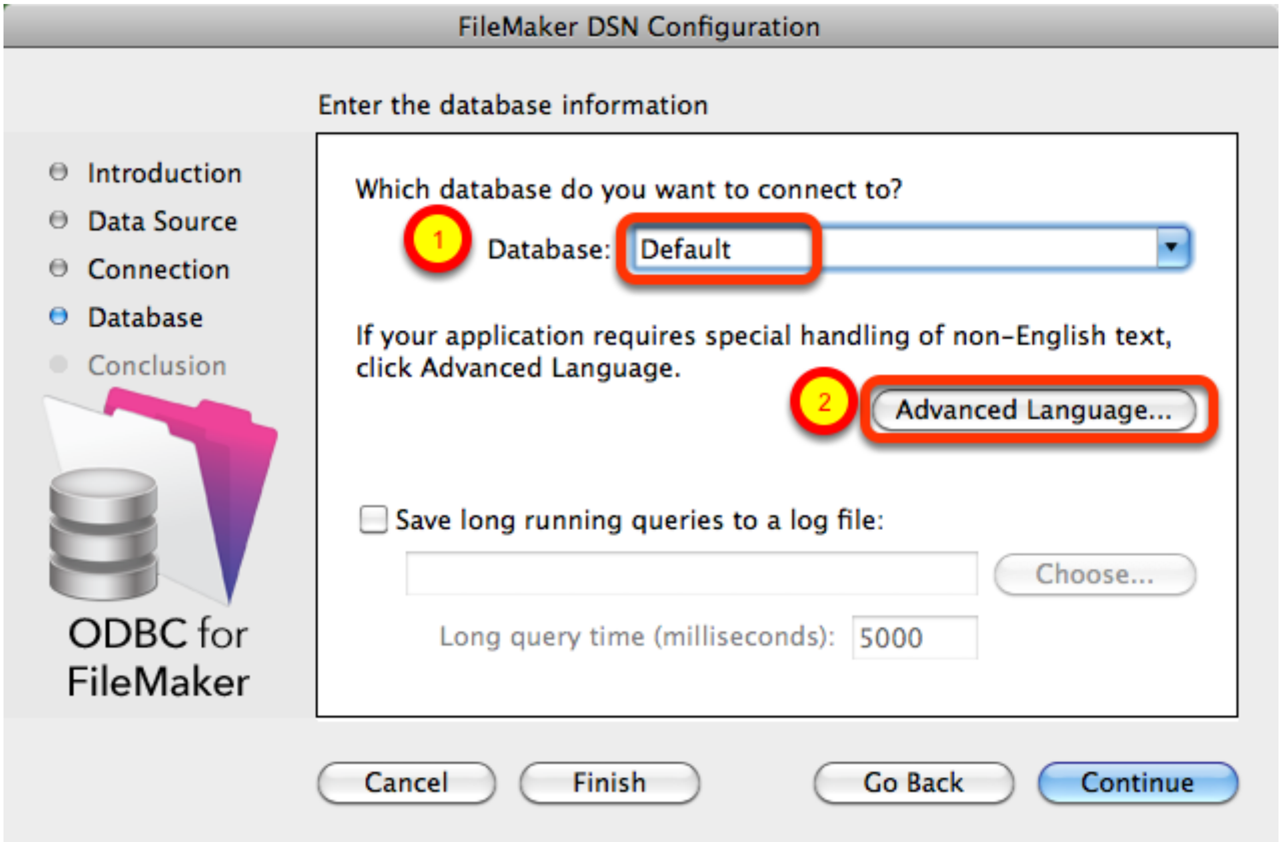
The host is typically referred to by its IP address.

If the database is on this computer, enter localhost.

□ Connect to host to obtain the names of available databases.

(1) Enter the hostname for the FileMaker database (default is localhost).
(2) Click the Continue button.
For a file which has been opened on your local desktop machine, leave the default setting of localhost.
For a hosted file hosted by FileMaker Server Advanced, enter the IP Address or hostname of the FileMaker Server Advanced computer.

FileMaker ODBC DSN Configuration - Page 4



FileMaker DSN Configuration

Enter the database information

Which database do you want to connect to?

1 Database: Default

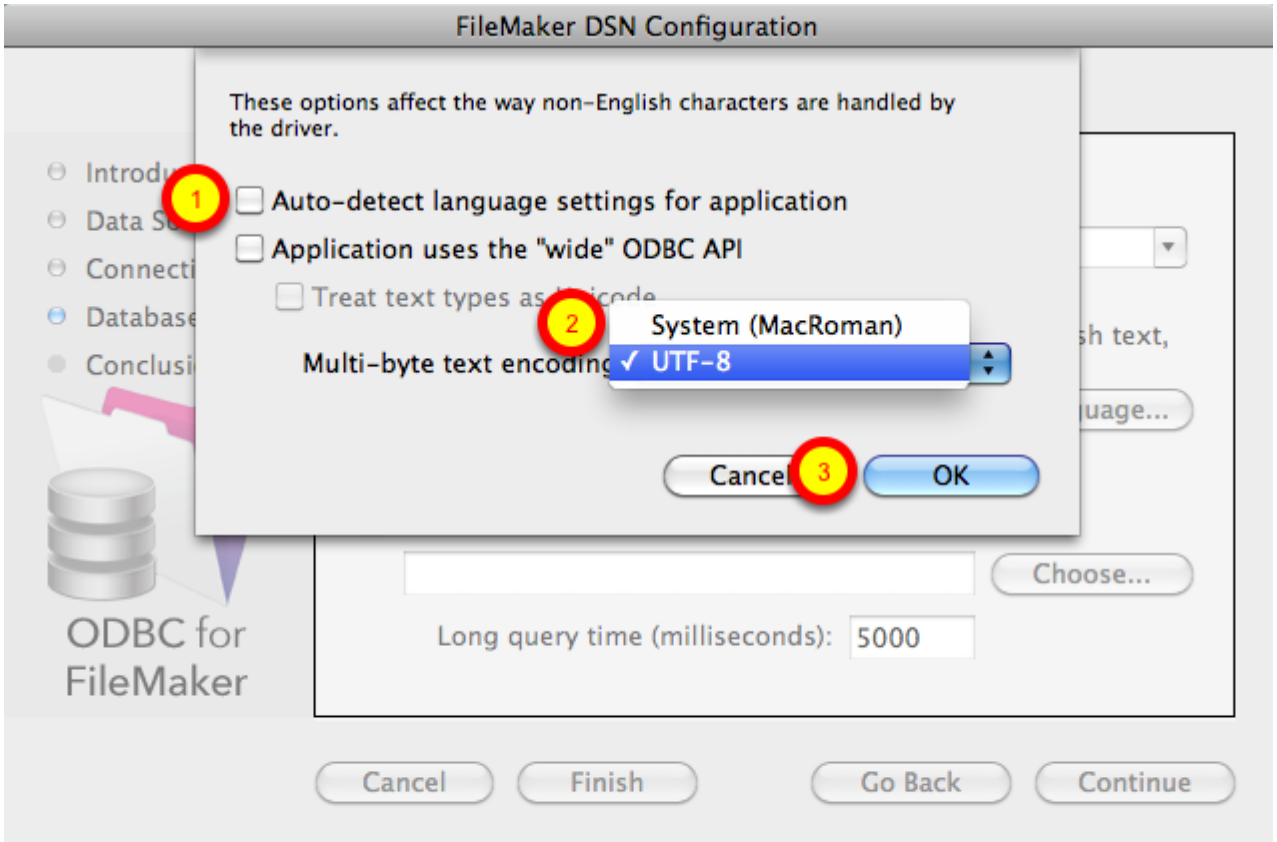
If your application requires special handling of non-English text, click Advanced Language.

☐ Save long running queries to a log file:

Long query time (milliseconds): 5000

(1) For most locally hosted database files, enter the text "Default" into the Database field, instead of selecting a database from the list. This way you can use any single database file without re-selecting the database name each time you open a new file.
To use this feature, you must only have one FileMaker database file open at a time, or an error will occur when attempting to read info from the database.
(2) Click the Advanced Language button.

FileMaker ODBC DSN Configuration - Page 5



FileMaker DSN Configuration

These options affect the way non-English characters are handled by the driver.

☐ Auto-detect language settings for application

☐ Application uses the "wide" ODBC API

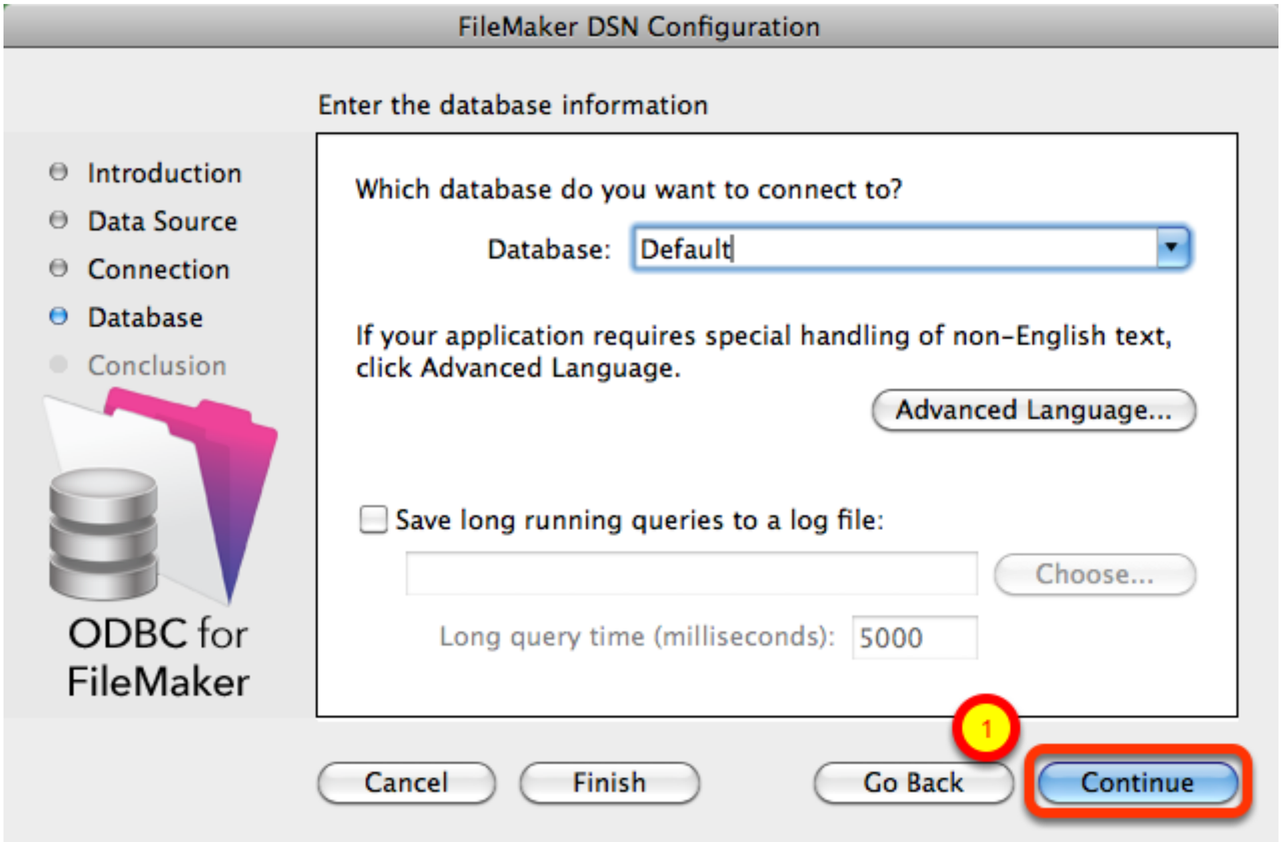
☐ Treat text types as Unicode

Multi-byte text encoding: UTF-8

Long query time (milliseconds): 5000

(1) Un-check the Auto-detect language settings checkbox.
(2) Select UTF-8 from the Multi-byte text encoding menu.
(3) Click the Ok button.

FileMaker ODBC DSN Configuration - Page 4



Enter the database information

Which database do you want to connect to?

Database: Default

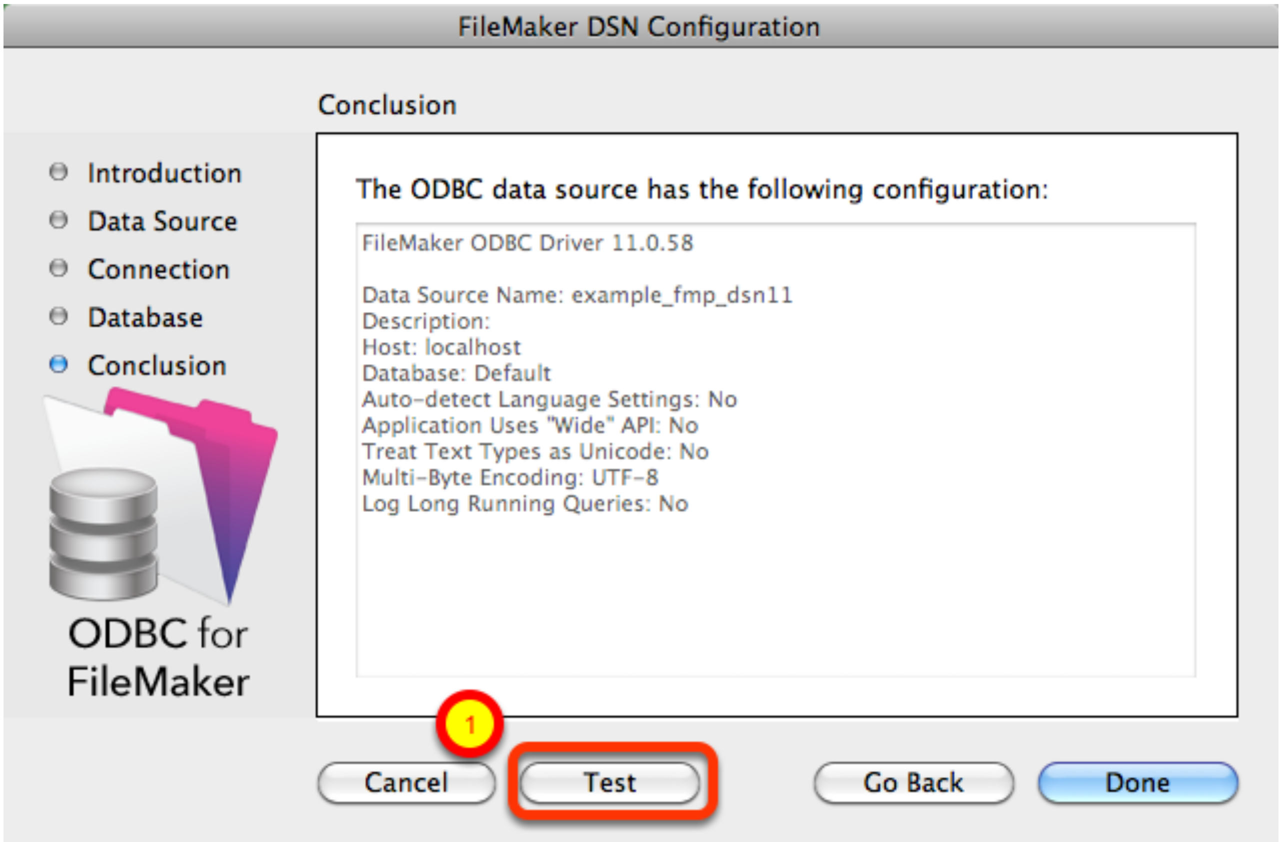
If your application requires special handling of non-English text, click Advanced Language.

□ Save long running queries to a log file:

Long query time (milliseconds): 5000

(1) Click the Continue button.

FileMaker ODBC DSN Configuration - Page 5

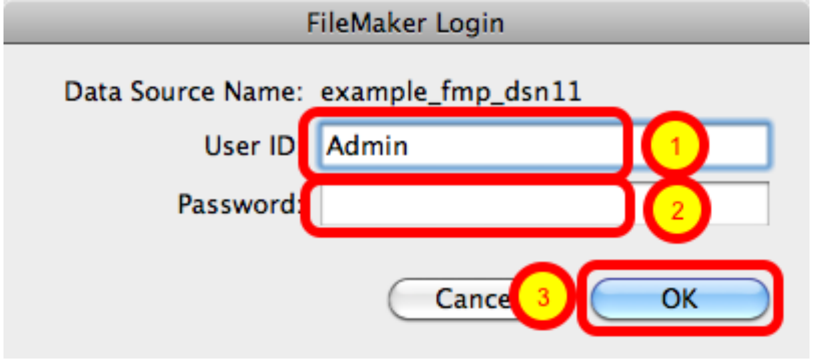


The ODBC data source has the following configuration:

```
FileMaker ODBC Driver 11.0.58

Data Source Name: example_fmp_dsn11
Description:
Host: localhost
Database: Default
Auto-detect Language Settings: No
Application Uses "Wide" API: No
Treat Text Types as Unicode: No
Multi-Byte Encoding: UTF-8
Log Long Running Queries: No
```

(1) Click the Test button.
If the test fails, use the following steps to enable ODBC sharing for the FileMaker database.
Note: Each FileMaker database needs to have ODBC sharing enabled individually.

FileMaker ODBC DSN Configuration - Page 5 - ODBC DSN Test



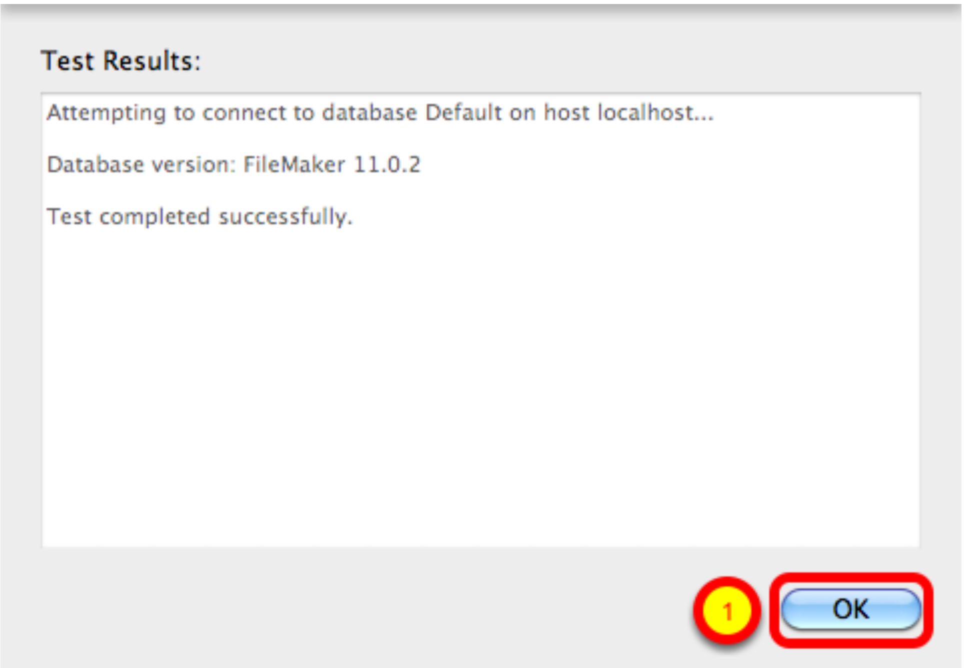
Data Source Name: example_fmp_dsn11

User ID: Admin

Password:

Enter the (1) Username, (2) Password, then (3) click the Ok button to test the ODBC DSN.

FileMaker ODBC DSN Configuration - Page 5 - ODBC DSN Test



Test Results:

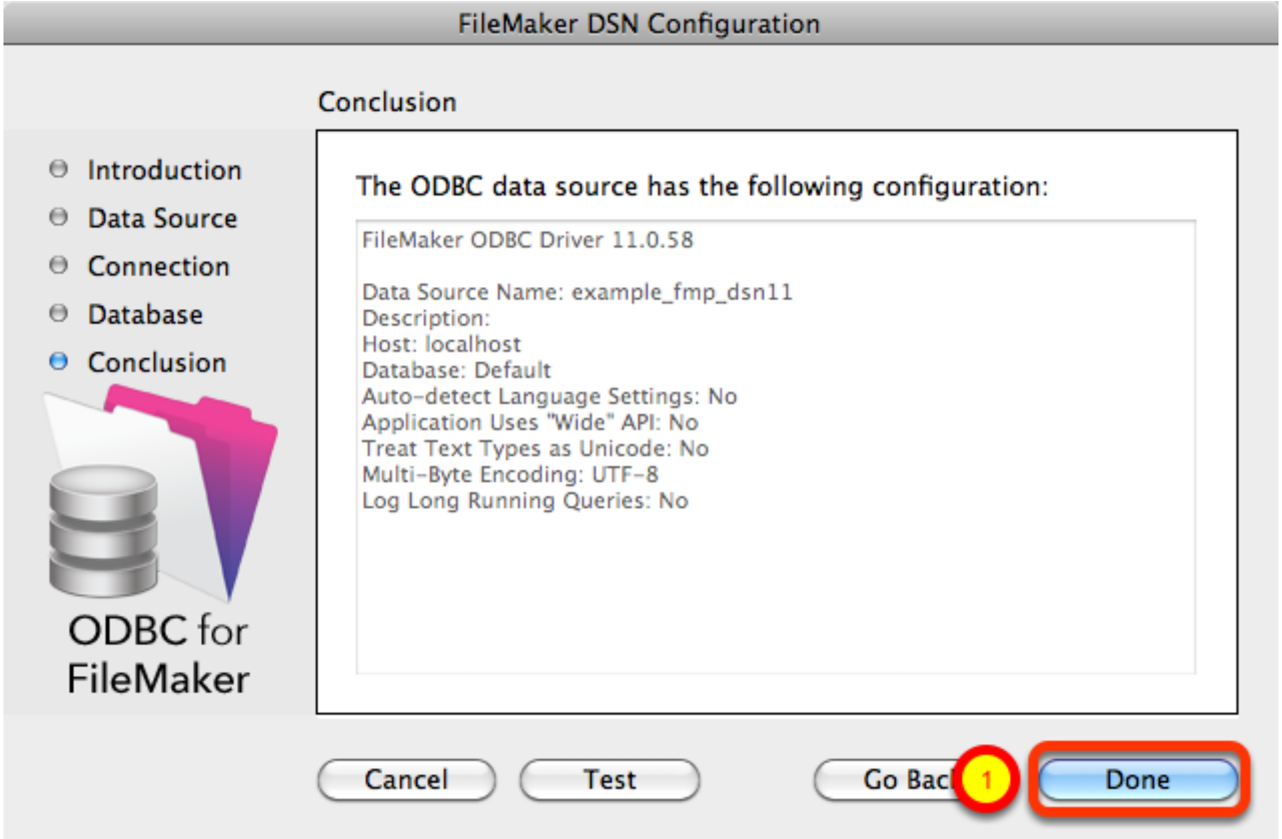
Attempting to connect to database Default on host localhost...

Database version: FileMaker 11.0.2

Test completed successfully.

(1) Click the Ok button.
If the test passes, this dialog will be displayed, showing that the ODBC Driver Manger has used this ODBC DSN to successfully connect to the FileMaker database.

If the test fails:
1) Verify that the FileMaker database file is open.
2) Verify that FileMaker ODBC sharing is enabled for the account entered in the connection test dialog prompt.
4) Make sure that there aren't two different copies of FileMaker running at the same time. Both older and current versions of FileMaker use TCP/IP Port# 2399 for ODBC connectivity. If this type of conflict occurs, close both versions of the FileMaker application, and only open the one copy you want to serve the database file being migrated.
5) Try rebooting the computer (after clicking Ok to save changes and close this ODBC dialog). Then test the DSN again after rebooting.

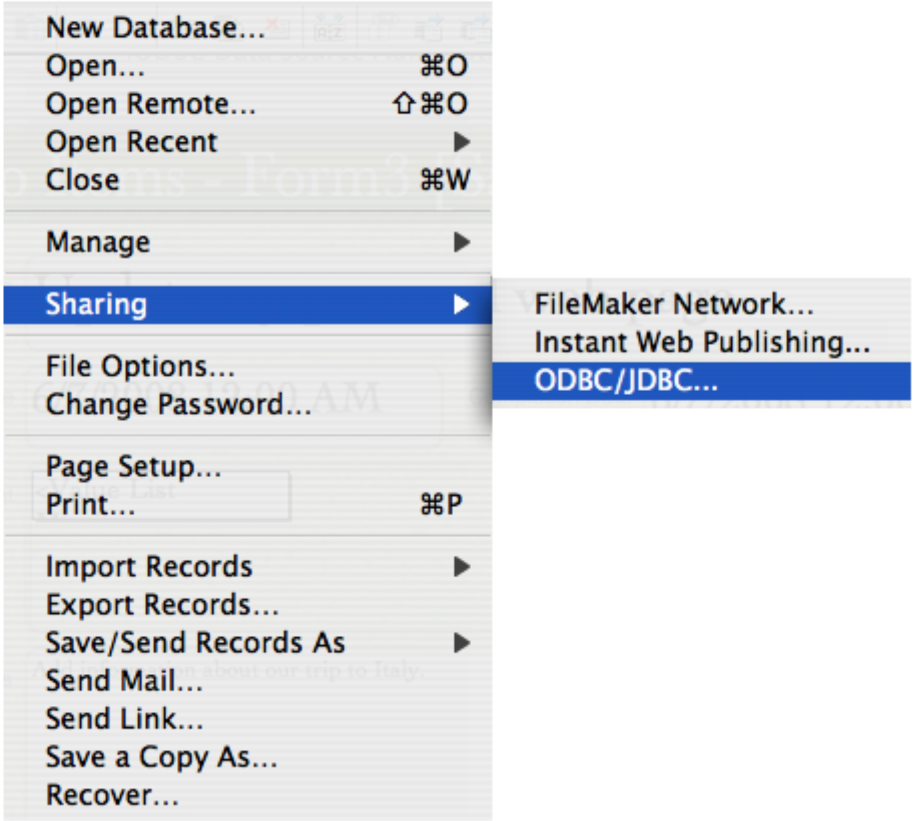


The ODBC data source has the following configuration:

```
FileMaker ODBC Driver 11.0.58

Data Source Name: example_fmp_dsn11
Description:
Host: localhost
Database: Default
Auto-detect Language Settings: No
Application Uses "Wide" API: No
Treat Text Types as Unicode: No
Multi-Byte Encoding: UTF-8
Log Long Running Queries: No
```

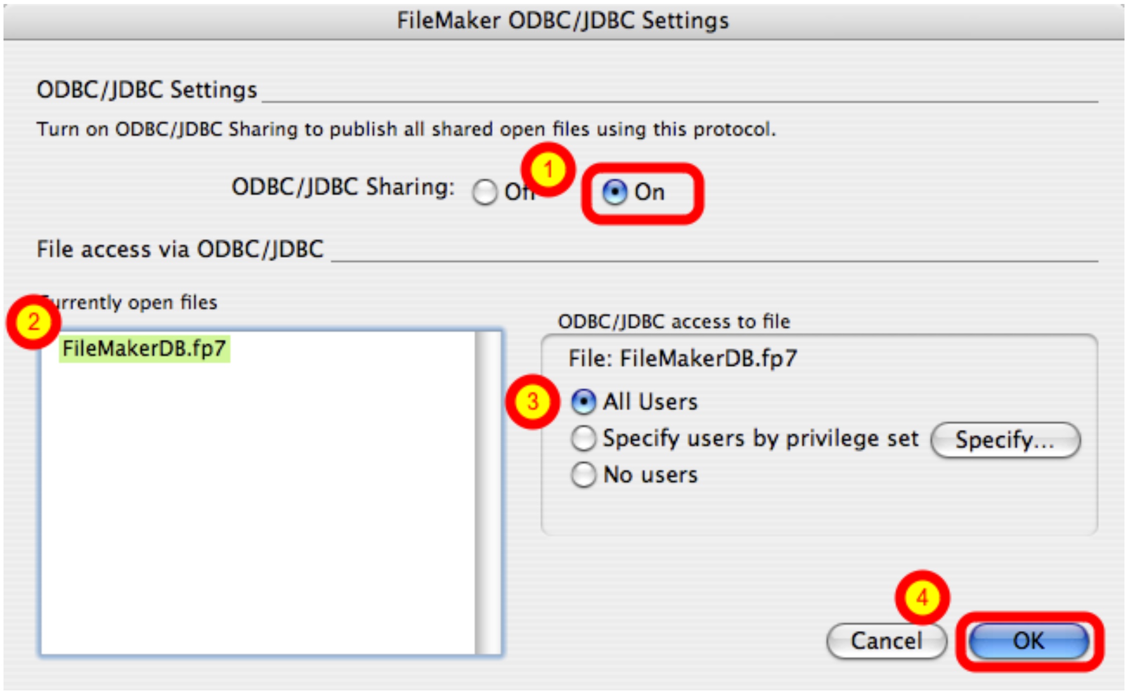
(1) Click the Done button to save the new ODBC DSN.

Enable ODBC/JDBC Sharing within FileMaker Pro



ODBC/JDBC Sharing needs to be enabled within any FileMaker Pro database file which is going to be used with the FileMaker ODBC driver.

Select Sharing Parameters



FileMaker ODBC/JDBC Settings

ODBC/JDBC Settings

Turn on ODBC/JDBC Sharing to publish all shared open files using this protocol.

ODBC/JDBC Sharing: ○ Off ● On

File access via ODBC/JDBC

Currently open files

FileMakerDB.fp7

ODBC/JDBC access to file

File: FileMakerDB.fp7

● All Users

○ Specify users by privilege set

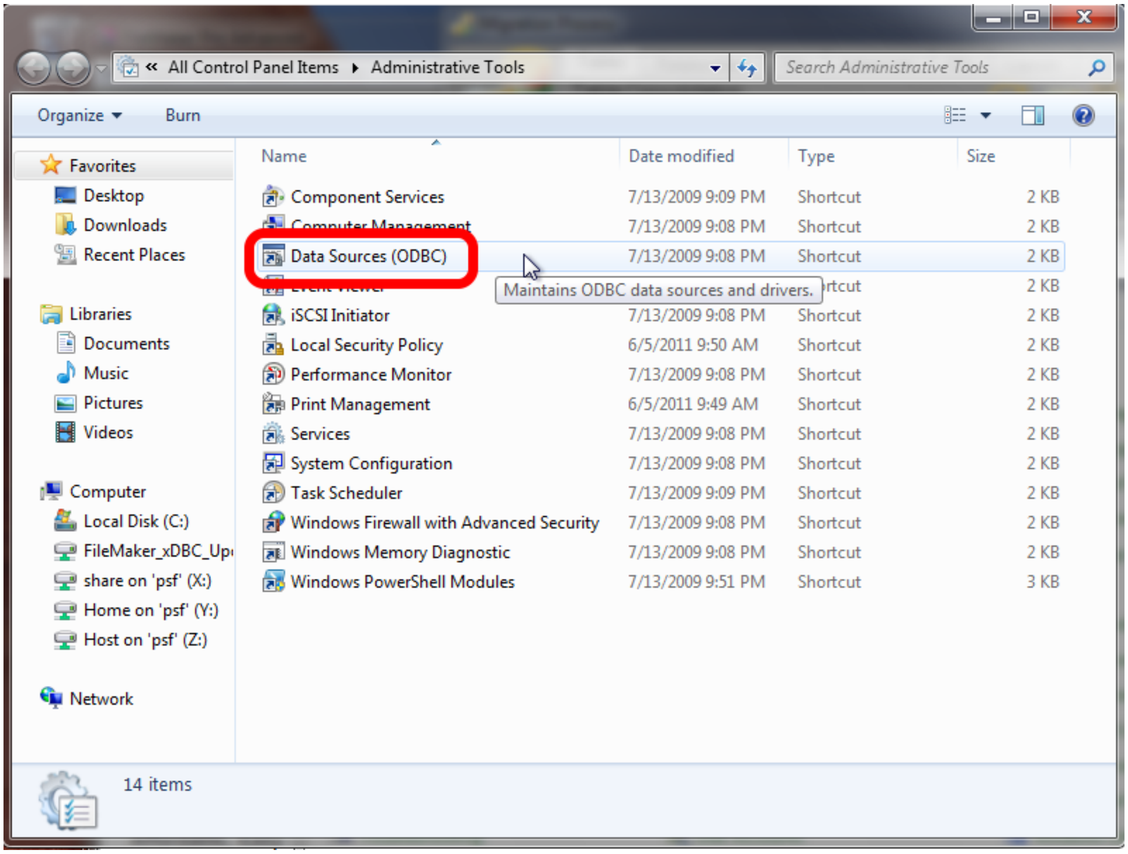
○ No users

(1) Turn on ODBC/JDBC Sharing, (2) for the currently open FileMaker Pro database file, (3) for All Users, then (4) click the Ok button.

Install FileMaker ODBC Driver - Windows - FileMaker 11+

Install the FileMaker ODBC driver from within the xDBC folder on the installation CD or CD image file.

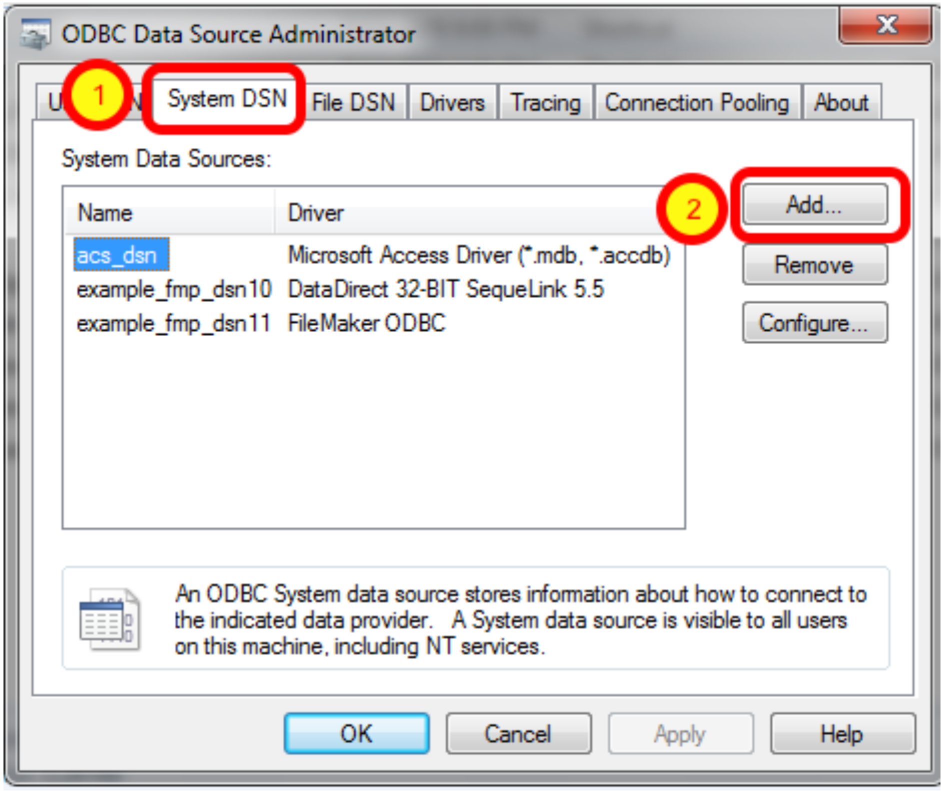
Create ODBC DSN - Windows



Name	Date modified	Type	Size
Component Services	7/13/2009 9:09 PM	Shortcut	2 KB
Computer Management	7/13/2009 9:08 PM	Shortcut	2 KB
Data Sources (ODBC)	7/13/2009 9:08 PM	Shortcut	2 KB
Event Viewer		rtcut	2 KB
iSCSI Initiator	7/13/2009 9:08 PM	Shortcut	2 KB
Local Security Policy	6/5/2011 9:50 AM	Shortcut	2 KB
Performance Monitor	7/13/2009 9:08 PM	Shortcut	2 KB
Print Management	6/5/2011 9:49 AM	Shortcut	2 KB
Services	7/13/2009 9:08 PM	Shortcut	2 KB
System Configuration	7/13/2009 9:08 PM	Shortcut	2 KB
Task Scheduler	7/13/2009 9:09 PM	Shortcut	2 KB
Windows Firewall with Advanced Security	7/13/2009 9:08 PM	Shortcut	2 KB
Windows Memory Diagnostic	7/13/2009 9:08 PM	Shortcut	2 KB
Windows PowerShell Modules	7/13/2009 9:51 PM	Shortcut	3 KB

Maintains ODBC data sources and drivers.

To create a new System ODBC DSN, open the Control Panel > Administrative Tools > Data Sources (ODBC) Control Panel.



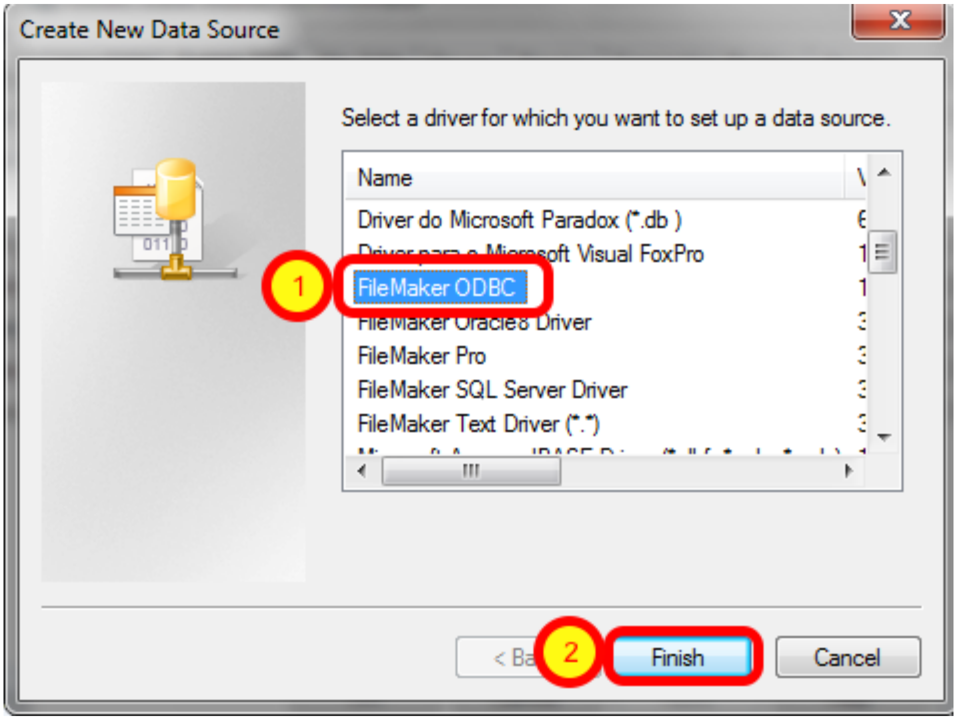
System Data Sources:

Name	Driver
acs_dsn	Microsoft Access Driver (*.mdb, *.accdb)
example_fmp_dsn10	DataDirect 32-BIT SequeLink 5.5
example_fmp_dsn11	FileMaker ODBC

An ODBC System data source stores information about how to connect to the indicated data provider. A System data source is visible to all users on this machine, including NT services.

(1) Click the System DSN tab, then (2) click the Add button.

Select ODBC Driver - FileMaker 11 - Windows

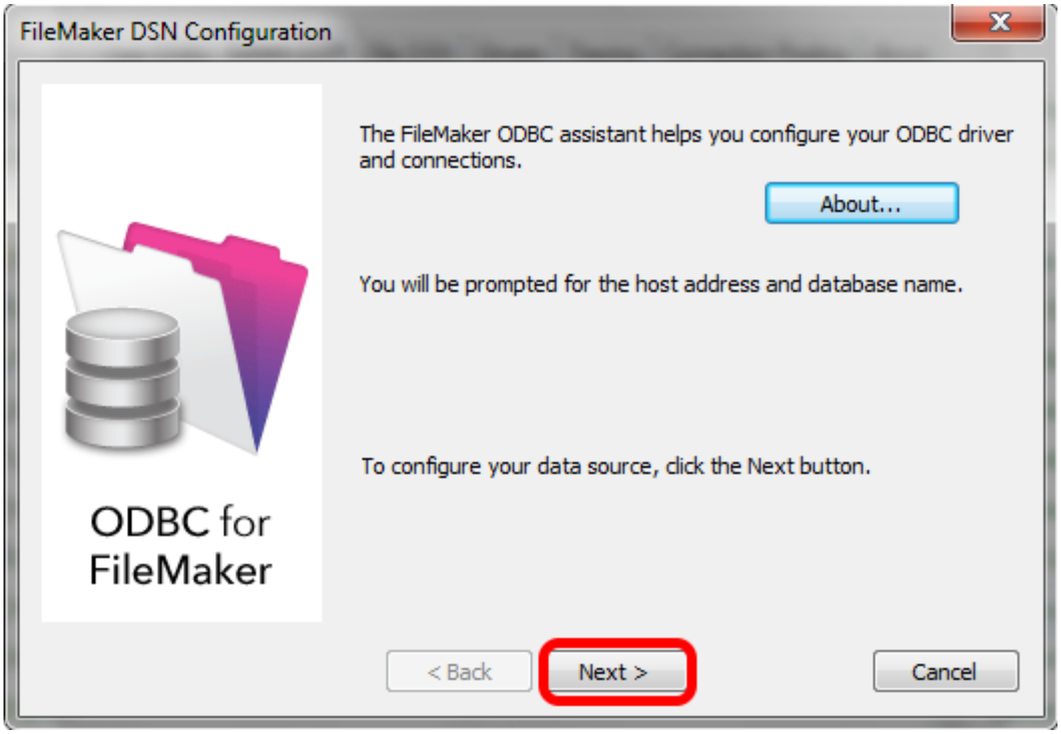


Select a driver for which you want to set up a data source.

Name
Driver do Microsoft Paradox (*.db)
Driver para o Microsoft Visual FoxPro
FileMaker ODBC
FileMaker Oracle8 Driver
FileMaker Pro
FileMaker SQL Server Driver
FileMaker Text Driver (*.*)

(1) Select the FileMaker ODBC driver, then (2) click the Finish button.

ODBC Driver Configuration - FileMaker 11+ - Windows (1)



FileMaker DSN Configuration

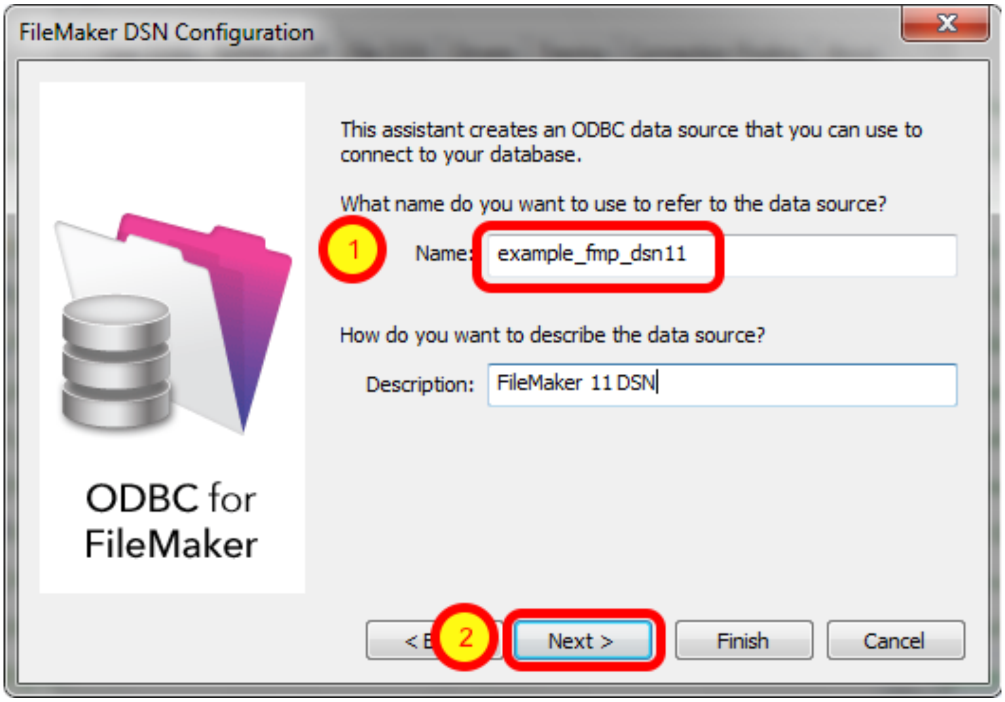
The FileMaker ODBC assistant helps you configure your ODBC driver and connections.

You will be prompted for the host address and database name.

To configure your data source, click the Next button.

Click Next button.
Click the About button if you need to find the version number for the driver.

ODBC Driver Configuration - FileMaker 11+ - Windows (2)



FileMaker DSN Configuration

This assistant creates an ODBC data source that you can use to connect to your database.

What name do you want to use to refer to the data source?

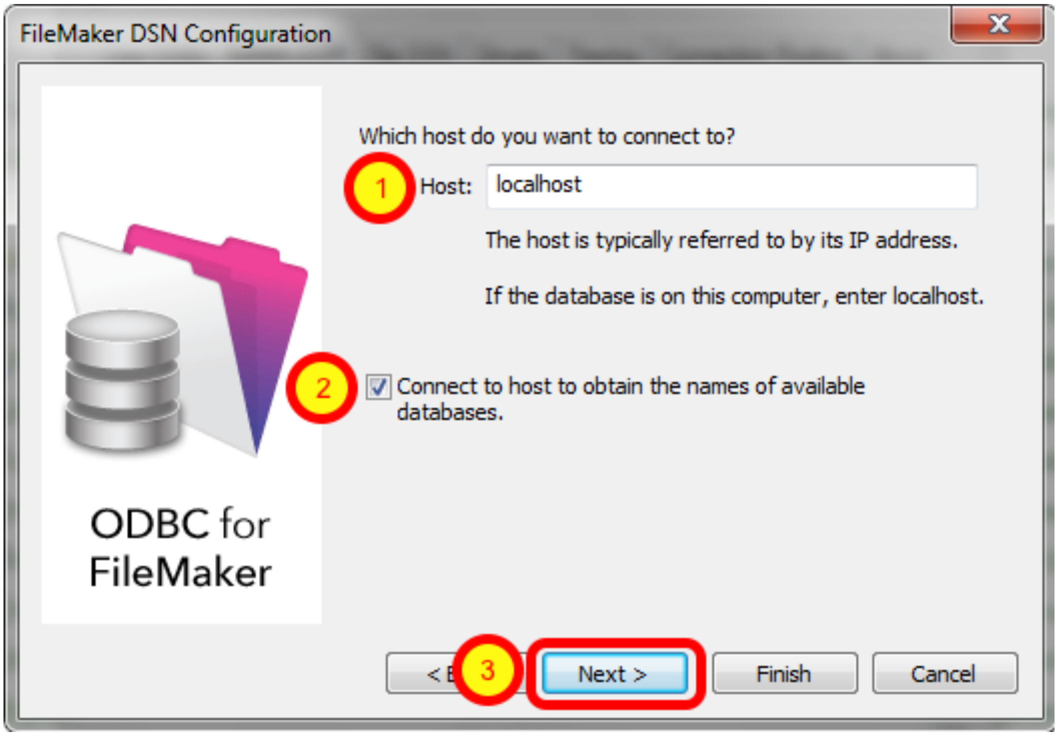
Name: example_fmp_dsn11

How do you want to describe the data source?

Description: FileMaker 11 DSN

1) Enter a name for the ODBC DSN, (2) click the Next button.

ODBC Driver Configuration - FileMaker 11+ - Windows (3)



FileMaker DSN Configuration

Which host do you want to connect to?

Host: localhost

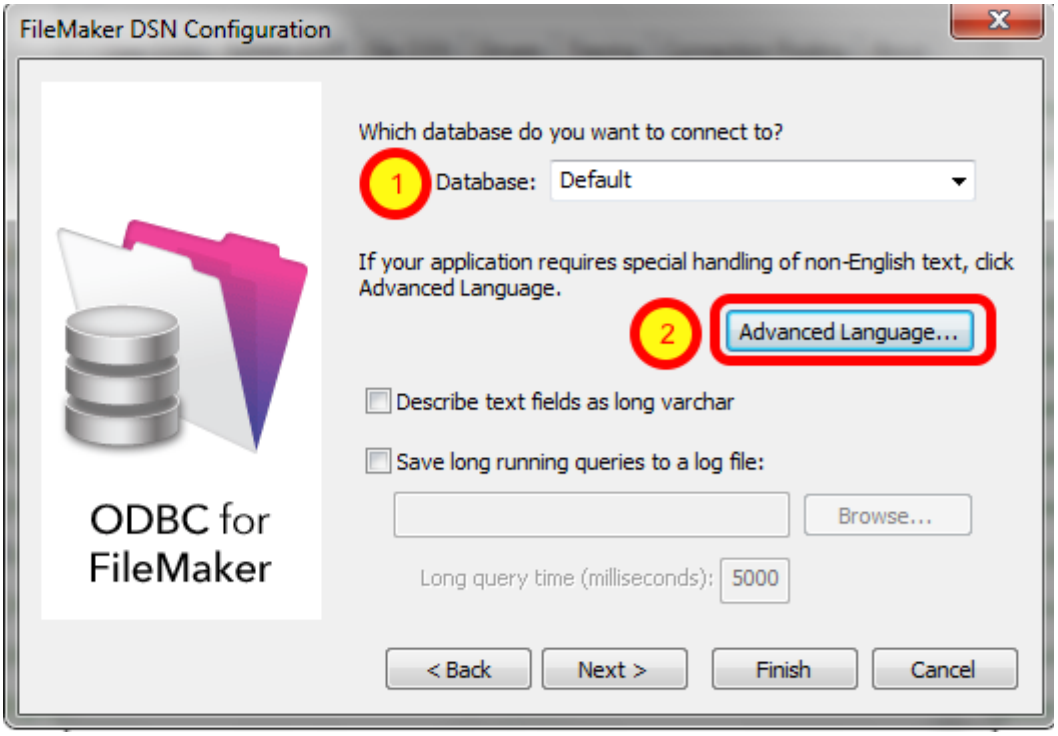
The host is typically referred to by its IP address.

If the database is on this computer, enter localhost.

☑ Connect to host to obtain the names of available databases.

1) If you are using a local copy of FileMaker Pro or FileMaker Pro Advanced, then use "localhost" as the host name. Otherwise use the IP address or hostname of the FileMaker Advanced Server.
2) You may optionally click the "Connect to host ..." button to get a list of database files.
3) Click the Next button.

ODBC Driver Configuration - FileMaker 11+ - Windows (4)



FileMaker DSN Configuration

Which database do you want to connect to?

Database: Default

If your application requires special handling of non-English text, click Advanced Language.

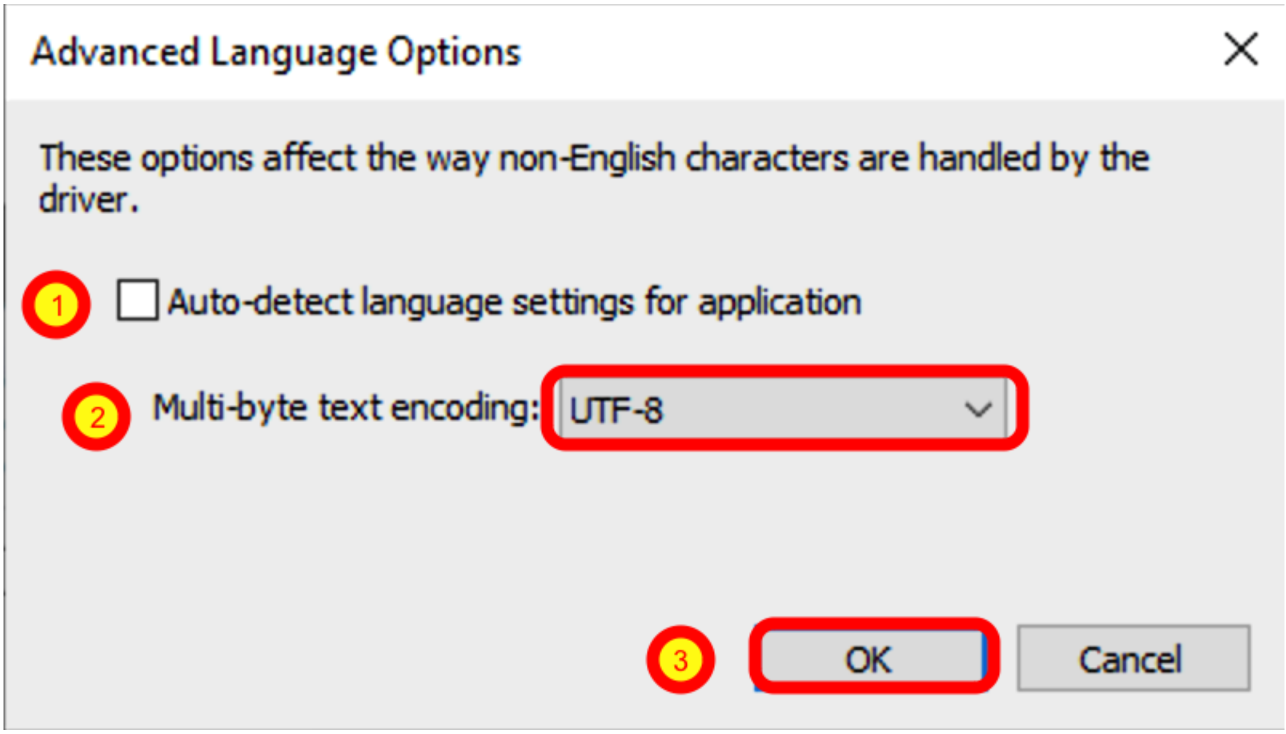
□ Describe text fields as long varchar

□ Save long running queries to a log file:

Long query time (milliseconds): 5000

1) Enter "Default" as the database name. Or if you have multiple database files open locally or on the server, then select the database name from the list or enter the name of the database. Using the Default keyword means that you can avoid changing the setup of the ODBC DSN when using different database files - but it means that only one file can be open at a time.
2) Click the Advanced Language button.

ODBC Driver Configuration - FileMaker 11 - Windows (5)



Advanced Language Options

These options affect the way non-English characters are handled by the driver.

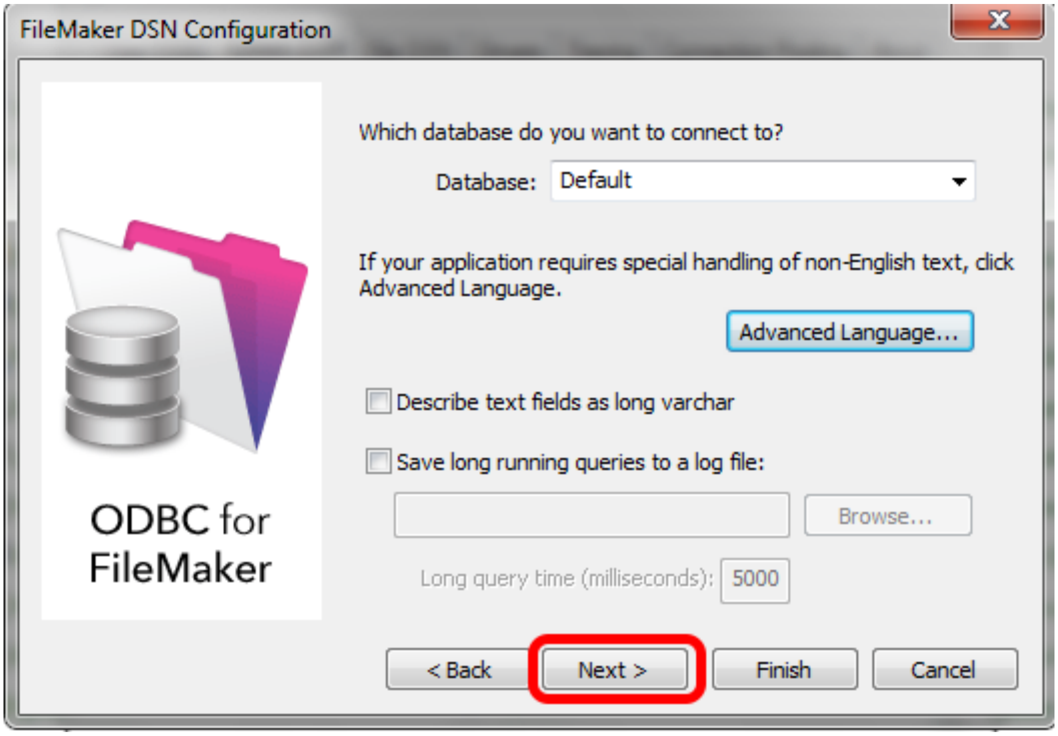
1 □ Auto-detect language settings for application

2 Multi-byte text encoding: UTF-8

1) Uncheck the "Auto-detect language settings" button, 2) select System (WinLatin1) for the text encoding, 3) click the Ok button.

It has been found that for proper character encoding, the following configuration works well:
1) Install the FileMaker database and FmPro Migrator on Windows. If using a 64-bit version of Windows, use the ODBC Administrator - 32 bit (not the 64 bit) to create your ODBC DSN when using FileMaker Pro or FileMaker Pro Advanced.
2) In the FileMaker ODBC driver, disable the Auto-Detect Language checkbox in the Advanced Language dialog.
3) Set the Multi-byte Text Encoding menu to UTF-8 in the Advanced Language dialog.
4) Enable the Perform Translation for Character Data checkbox on the last panel of the SQL Server ODBC DSN configuration.

ODBC Driver Configuration - FileMaker 11+ - Windows (6)



Which database do you want to connect to?

Database: Default

If your application requires special handling of non-English text, click Advanced Language.

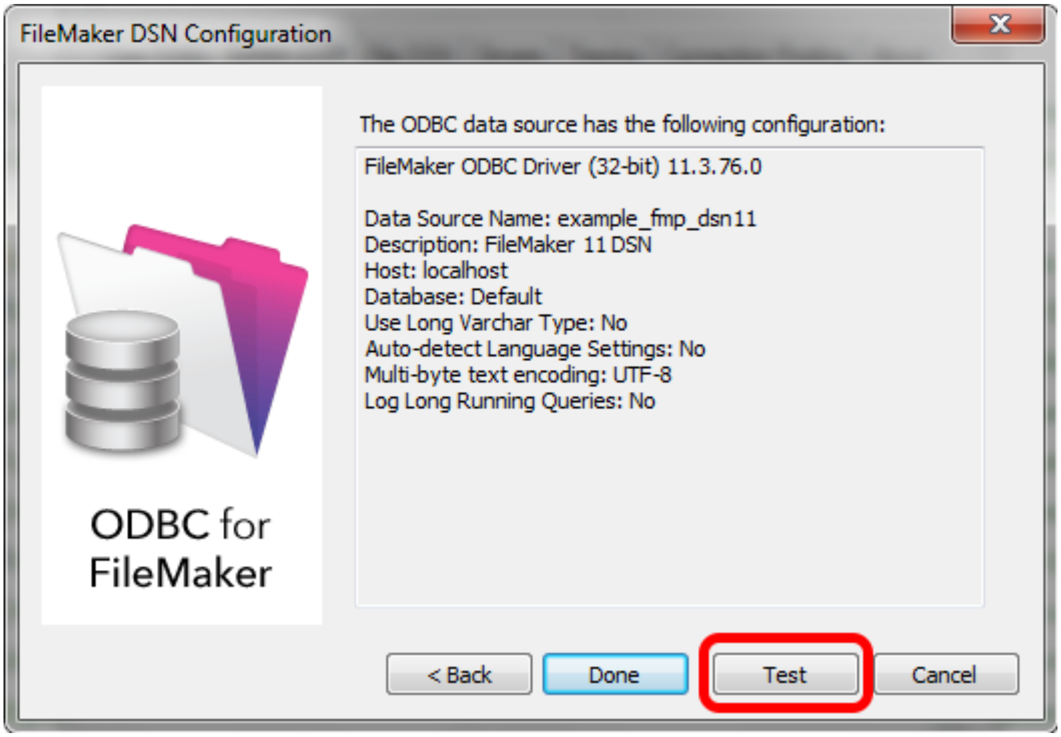
□ Describe text fields as long varchar

□ Save long running queries to a log file:

Long query time (milliseconds): 5000

Click the Next button.

ODBC Driver Configuration - FileMaker 11+ - Windows (7)

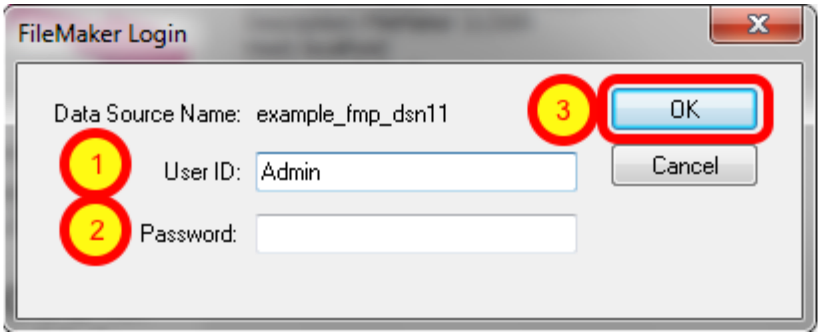


The ODBC data source has the following configuration:

```
FileMaker ODBC Driver (32-bit) 11.3.76.0

Data Source Name: example_fmp_dsn11
Description: FileMaker 11 DSN
Host: localhost
Database: Default
Use Long Varchar Type: No
Auto-detect Language Settings: No
Multi-byte text encoding: UTF-8
Log Long Running Queries: No
```

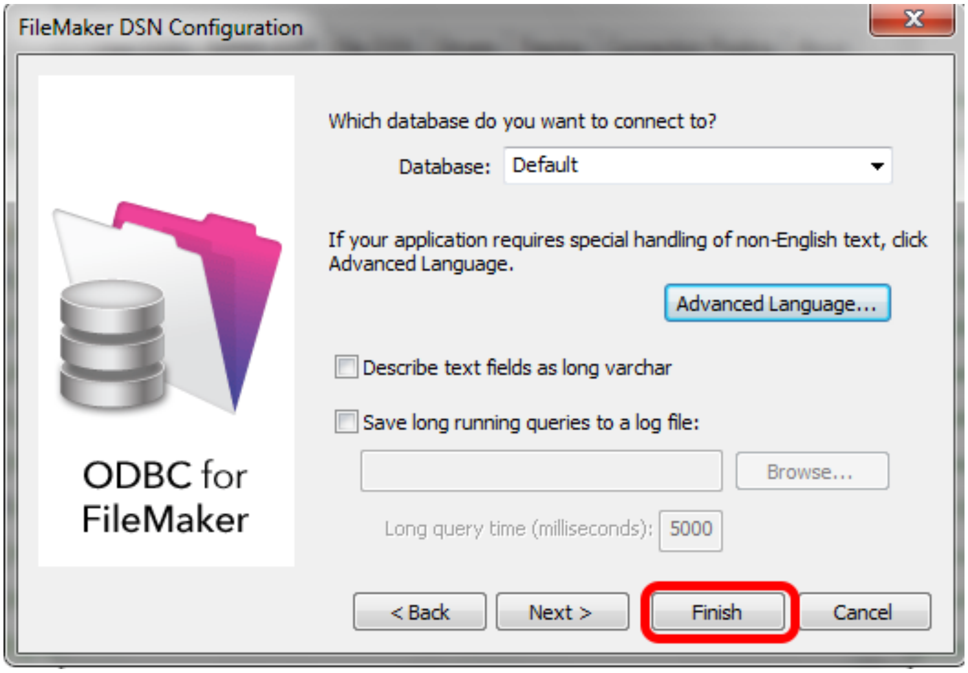
Click the Test button.

ODBC Driver Configuration - FileMaker 11+ - Windows (8)



1) Enter Username, 2) Password (if needed), 3) click the Ok button.
Review the test results, make corrections as necessary until the test passes.
<u>ODBC Error Troubleshooting Steps:</u>
1) Verify that the FileMaker database file is open.
2) Verify that ODBC sharing is enabled for the account entered in the connection test dialog prompt.
3) Make sure that there aren't two different copies of FileMaker running at the same time. Both older and current versions of FileMaker use TCP/IP Port# 2399 for ODBC connectivity. If this type of conflict occurs, close both versions of the FileMaker application, and only open the one copy you want to serve the database file being migrated.
4) Try rebooting the computer (after clicking Ok to save changes and close this ODBC dialog). Then test the DSN again after rebooting.

ODBC Driver Configuration - FileMaker 11+ - Windows (9)

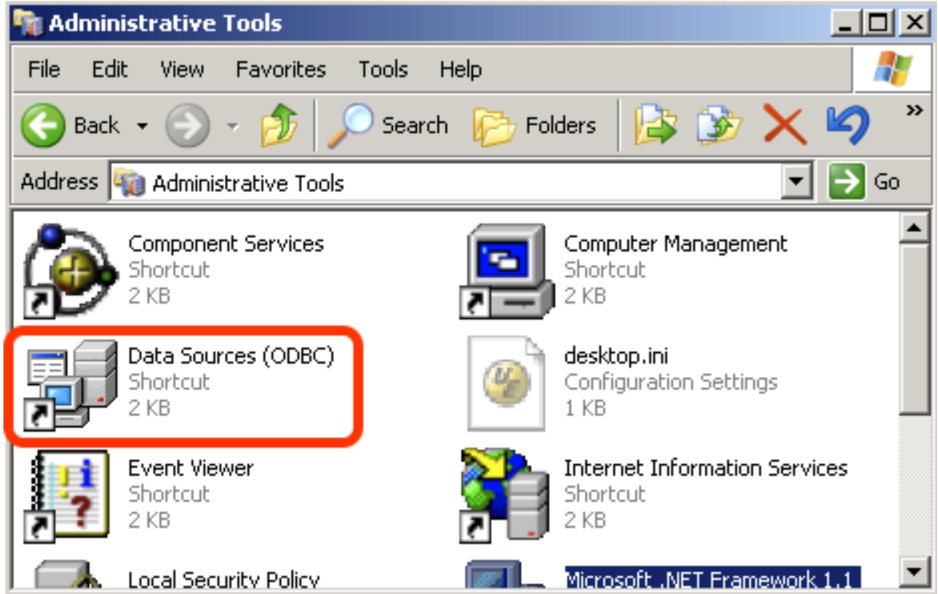

FileMaker DSN Configuration
Which database do you want to connect to?
Database: Default
If your application requires special handling of non-English text, click Advanced Language.
Advanced Language...
Describe text fields as long varchar
Save long running queries to a log file:
Browse...
Long query time (milliseconds): 5000
ODBC for FileMaker
< Back
Next >
Finish
Cancel

Click the Finish button.

Create Access ODBC DSN - Windows

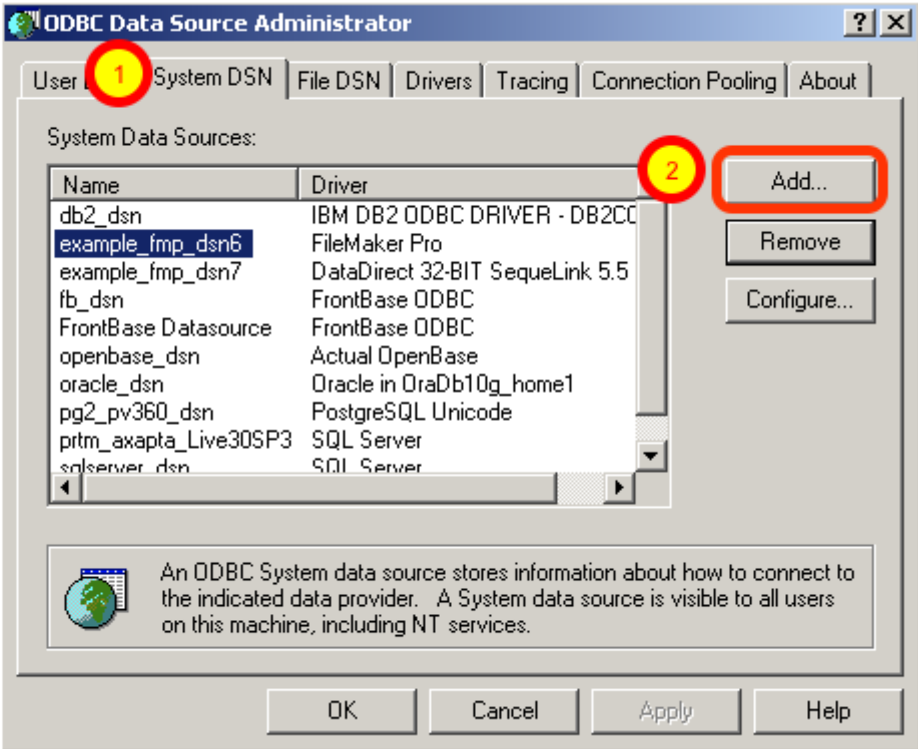
The Microsoft Access ODBC driver is installed automatically with every copy of the Windows operating system. This process shows how to create a System DSN to connect to an Access database (any version of Access).

Open ODBC Manager Control Panel


Administrative Tools
File Edit View Favorites Tools Help
Back Search Folders
Address Administrative Tools Go
Component Services Shortcut 2 KB
Computer Management Shortcut 2 KB
Data Sources (ODBC) Shortcut 2 KB
desktop.ini Configuration Settings 1 KB
Event Viewer Shortcut 2 KB
Internet Information Services Shortcut 2 KB
Local Security Policy
Microsoft .NET Framework 1.1

Open the ODBC Data Source Administrator Control Panel, located in Control Panel -> Administrative Tools.

Create New System ODBC DSN



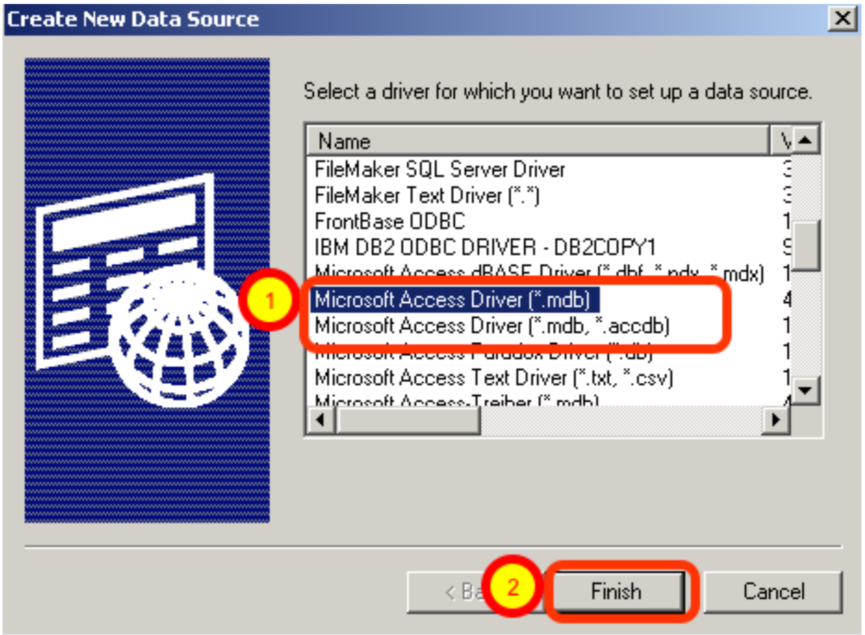
System Data Sources:

Name	Driver
db2_dsn	IBM DB2 ODBC DRIVER - DB2CC
example_fmp_dsn6	FileMaker Pro
example_fmp_dsn7	DataDirect 32-BIT SequeLink 5.5
fb_dsn	FrontBase ODBC
FrontBase Datasource	FrontBase ODBC
openbase_dsn	Actual OpenBase
oracle_dsn	Oracle in OraDb10g_home1
pg2_pv360_dsn	PostgreSQL Unicode
prtm_axapta_Live30SP3	SQL Server
sqlserver_dsn	SQL Server

An ODBC System data source stores information about how to connect to the indicated data provider. A System data source is visible to all users on this machine, including NT services.

(1) Click the System DSN tab, click the Add button to create a new ODBC DSN.

Select Microsoft Access Driver (*.mdb)

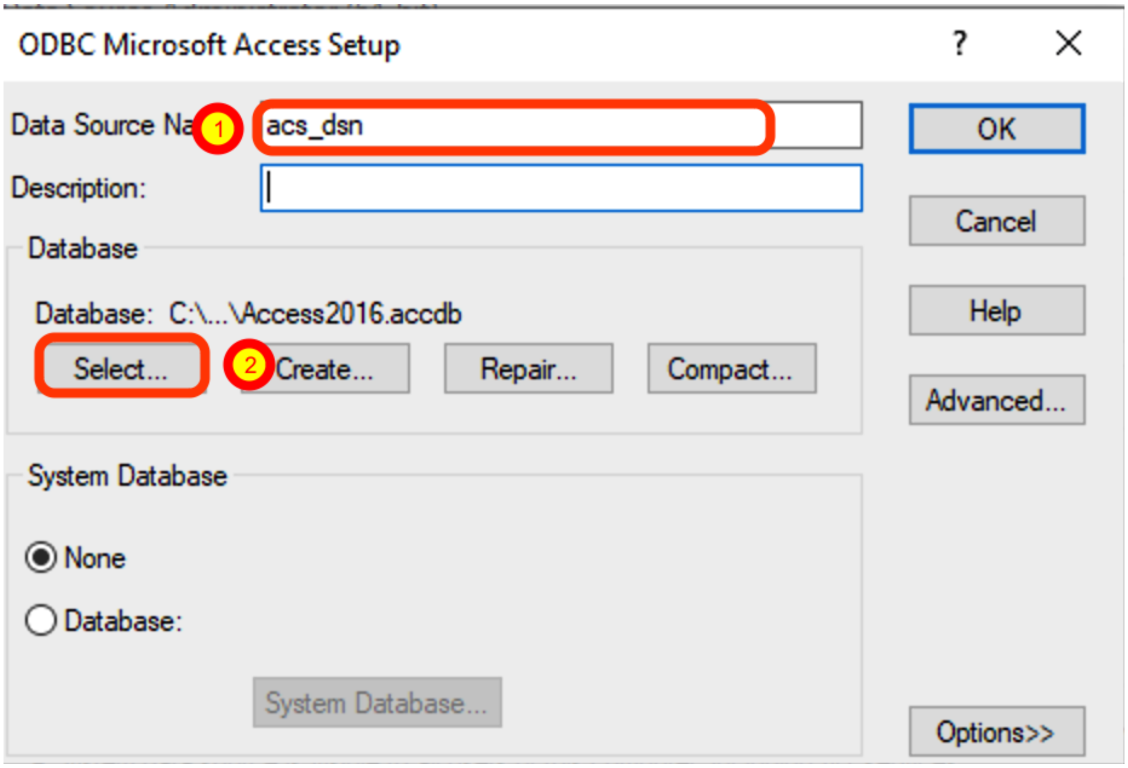


Select a driver for which you want to set up a data source.

Name
FileMaker SQL Server Driver
FileMaker Text Driver (*.*)
FrontBase ODBC
IBM DB2 ODBC DRIVER - DB2COPY1
Microsoft Access dBASE Driver (*.dbf, *.ndx, *.mdx)
Microsoft Access Driver (*.mdb)
Microsoft Access Driver (*.mdb, *.accdb)
Microsoft Access Paradox Driver (*.db)
Microsoft Access Text Driver (*.txt, *.csv)
Microsoft Access-Treiber (*.mdb)

(1) Select the Microsoft Access Driver (*.mdb), (2) click the Finish button.
Note: If you need to connect to an Access 2007+ .accdb database file, select the driver having *.accdb in its title, which is shown in this screenshot just below the *.mdb driver.

Enter ODBC DSN Name



Data Source Name: acs_dsn

Description:

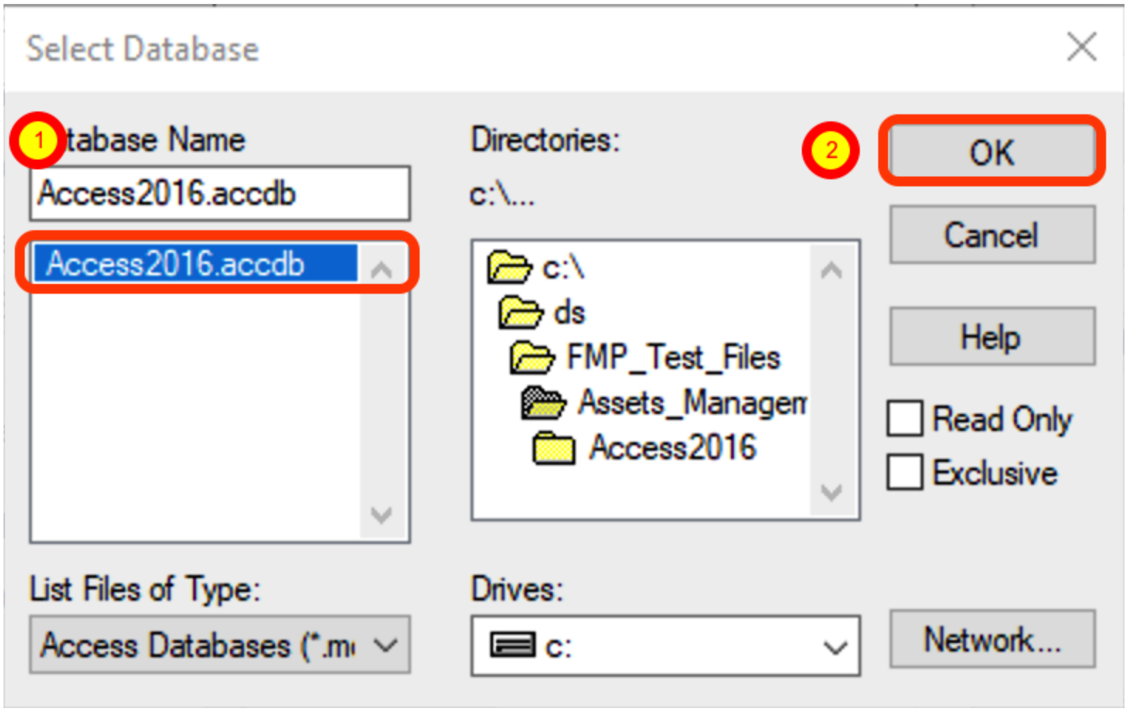
Database: C:\...\Access2016.accdb

● None

○ Database:

(1) Enter the name for this new ODBC DSN. This name either needs to match the name of the DSN which is already entered into FmPro Migrator or you can change the name of the DSN in FmPro Migrator to match the one you enter here.
(2) Click the Select... button to select the Access database filename.

Select the Access Database File - Click Ok Button



Select Database

Database Name: Access2016.accdb

Directories: c:\...

List Files of Type: Access Databases (*.m

Drives: c:

□ Read Only

□ Exclusive

(1) Select the Access file, (2) click the Ok button to complete the creation of the ODBC DSN.

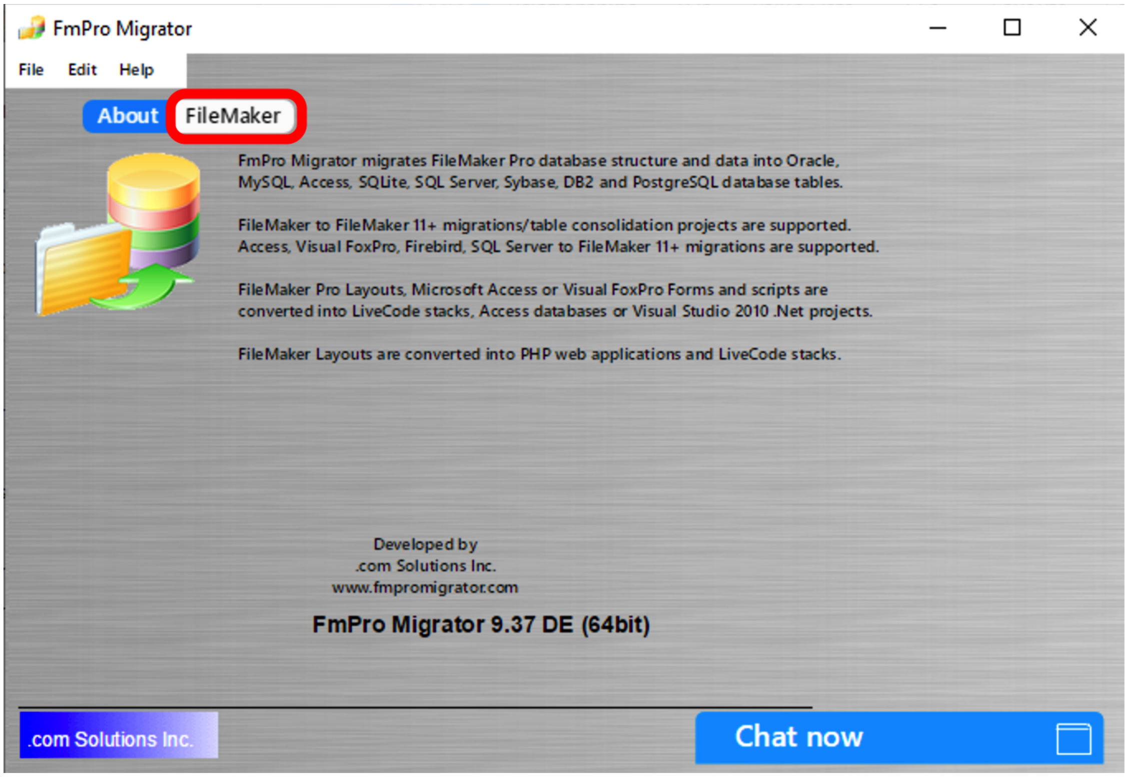
Step 1

Step 1 - Get Info - FileMaker to Access

Step 1 - Install FileMaker ODBC Driver

FmPro Migrator uses the FileMaker ODBC driver provided by FileMaker Inc. to transfer data from FileMaker databases.
ODBC driver and ODBC DSN creation info is documented in previous chapters of this manual.

Step 1 - Get Info - Select FileMaker tab



FmPro Migrator migrates FileMaker Pro database structure and data into Oracle, MySQL, Access, SQLite, SQL Server, Sybase, DB2 and PostgreSQL database tables.

FileMaker to FileMaker 11+ migrations/table consolidation projects are supported. Access, Visual FoxPro, Firebird, SQL Server to FileMaker 11+ migrations are supported.

FileMaker Pro Layouts, Microsoft Access or Visual FoxPro Forms and scripts are converted into LiveCode stacks, Access databases or Visual Studio 2010 .Net projects.

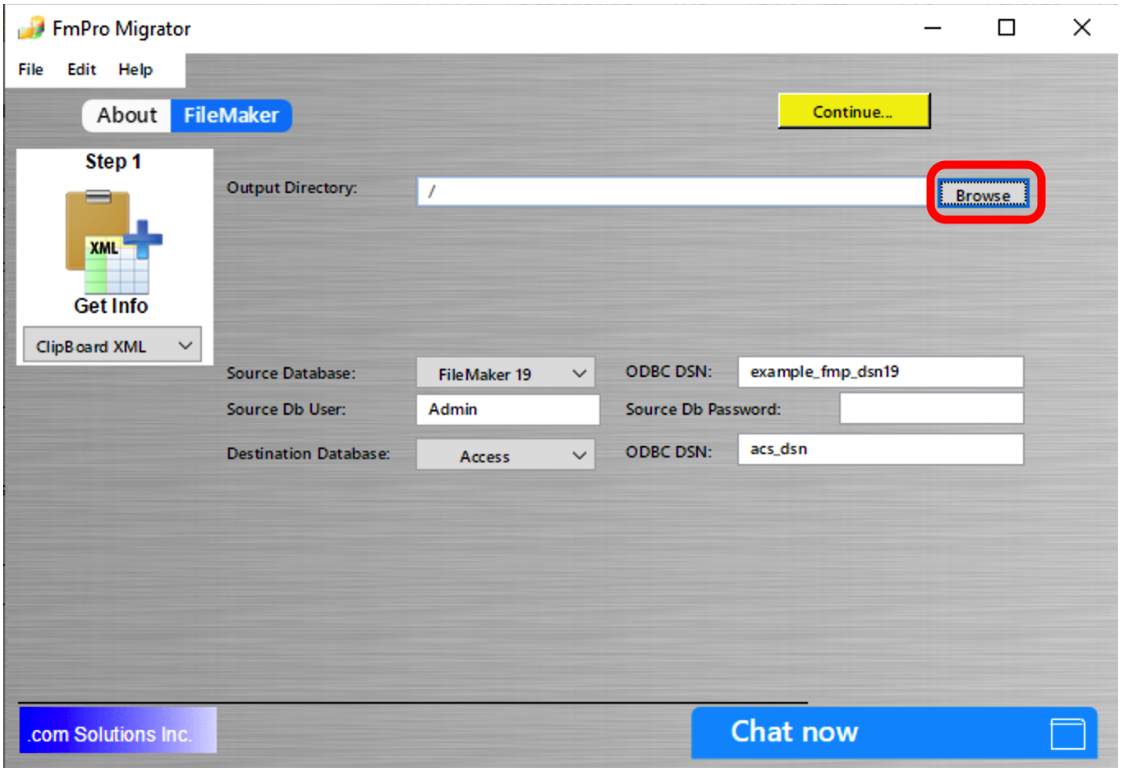
FileMaker Layouts are converted into PHP web applications and LiveCode stacks.

Developed by
.com Solutions Inc.
www.fmpromigrator.com

FmPro Migrator 9.37 DE (64bit)

Launch FmPro Migrator, then click on the FileMaker tab at the top of the window.

Step 1 - Get Info - Select Output Directory



ClipBoard XML

Output Directory: /

Source Database: FileMaker 19

ODBC DSN: example_fmp_dsn19

Source Db User: Admin

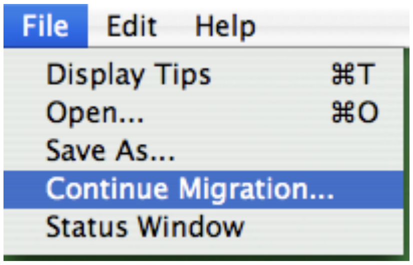
Source Db Password:

Destination Database: Access

ODBC DSN: acs_dsn

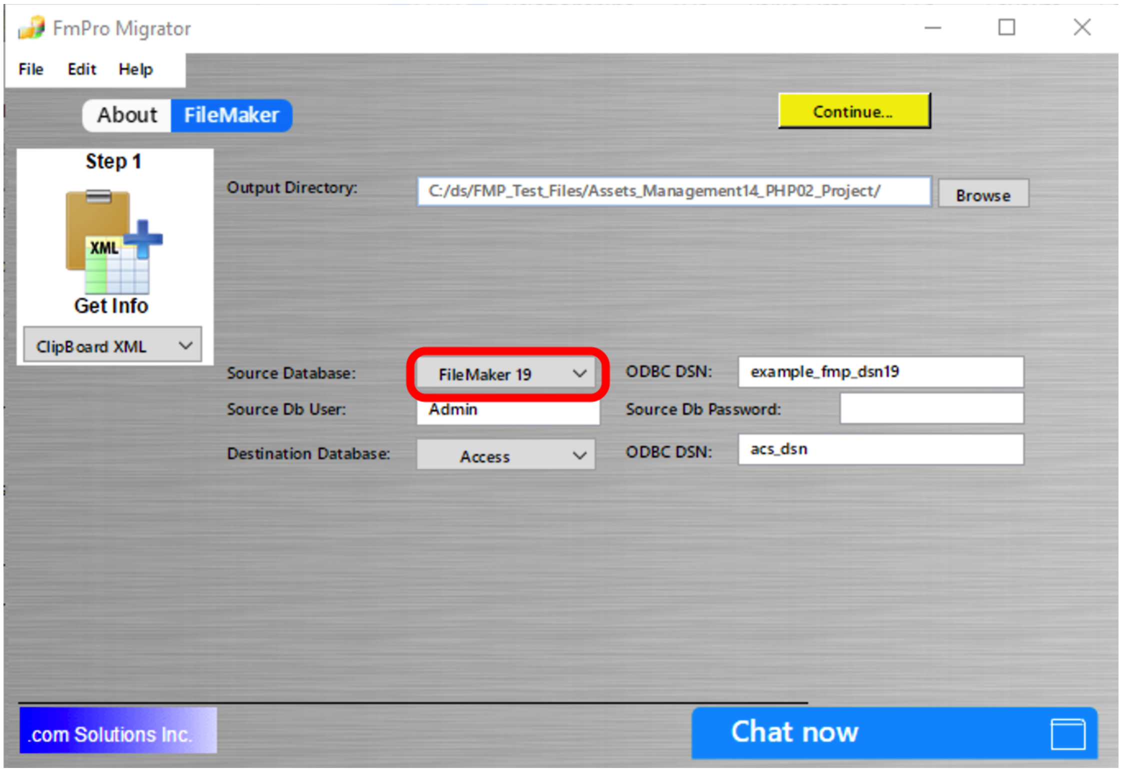
FmPro Migrator stores migration process information within a SQLite database file named MigrationProcess.db3. This file contains the metadata for the source database file(s), including tables, fields, relationships, table creation SQL code, value lists, layouts, scripts and status info. Therefore the first step in the migration process is to select the Browse button to select the output folder FmPro Migrator will use when creating the MigrationProcess.db3 file.

Continue Migration...



Tip: To restart an existing migration project, select the output directory, then select Continue Migration... from the File menu.
Click the yellow Continue button to open the Migration Process window.

Step 1 - Get Info - Enter Source Database Info



ClipBoard XML

Output Directory: C:/ds/FMP_Test_Files/Assets_Management14_PHP02_Project/

Source Database: FileMaker 19

ODBC DSN: example_fmp_dsn19

Source Db User: Admin

Source Db Password:

Destination Database: Access

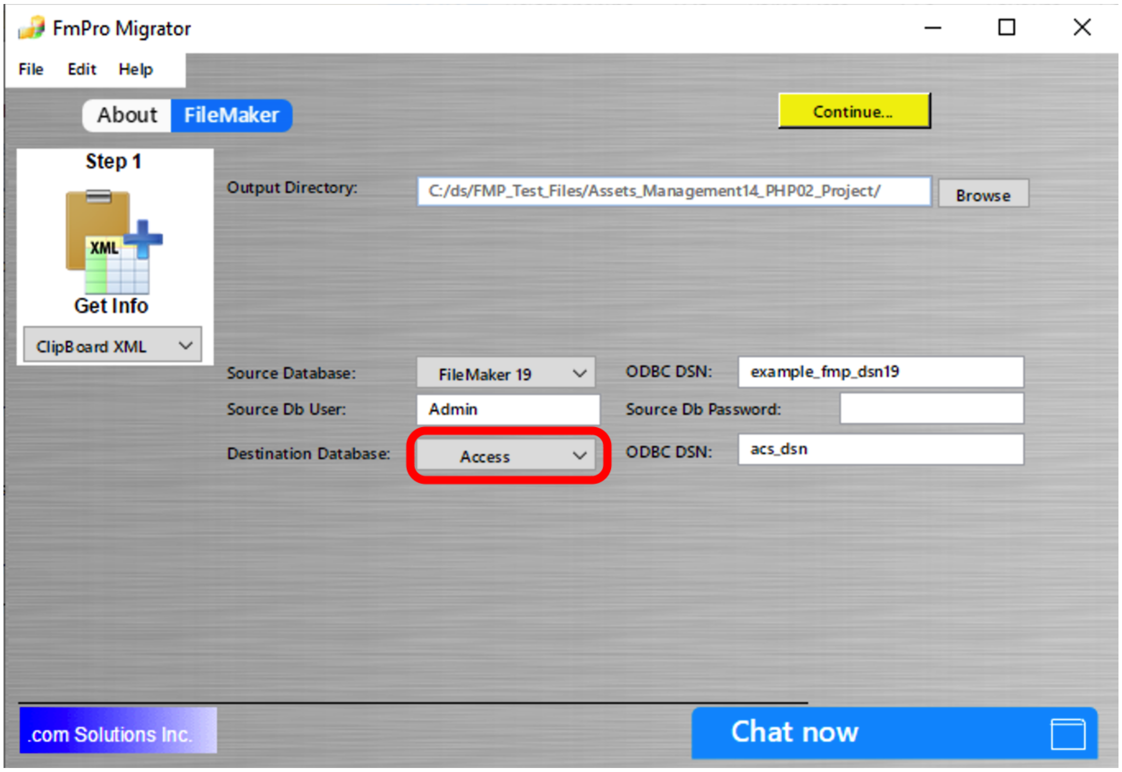
ODBC DSN: acs_dsn

Select the type of source database from the Source Database menu.
Note: Since there isn't a macOS compatible FileMaker ODBC driver for FileMaker 5/6, these older database files should be converted to FileMaker 12 or higher in order to perform the migration on macOS.

Either use the existing ODBC DSN name "example_fmp_dsn" or change this name to reflect the name of a System ODBC DSN you have created on your computer.

Enter the Username and Password required to access the source database file. If a password has not been configured within a FileMaker 7+ file, set the username to "Admin" with the password field empty. The user account entered here needs to have full access to the database file and needs to have ODBC access privileges within the file. This is the default access for the Admin account.

Step 1 - Get Info - Enter Destination Database Info



ClipBoard XML

Output Directory: C:/ds/FMP_Test_Files/Assets_Management14_PHP02_Project/

Source Database: FileMaker 19

ODBC DSN: example_fmp_dsn19

Source Db User: Admin

Source Db Password:

Destination Database: Access

ODBC DSN: acs_dsn

(1) Select Access as the destination database type.

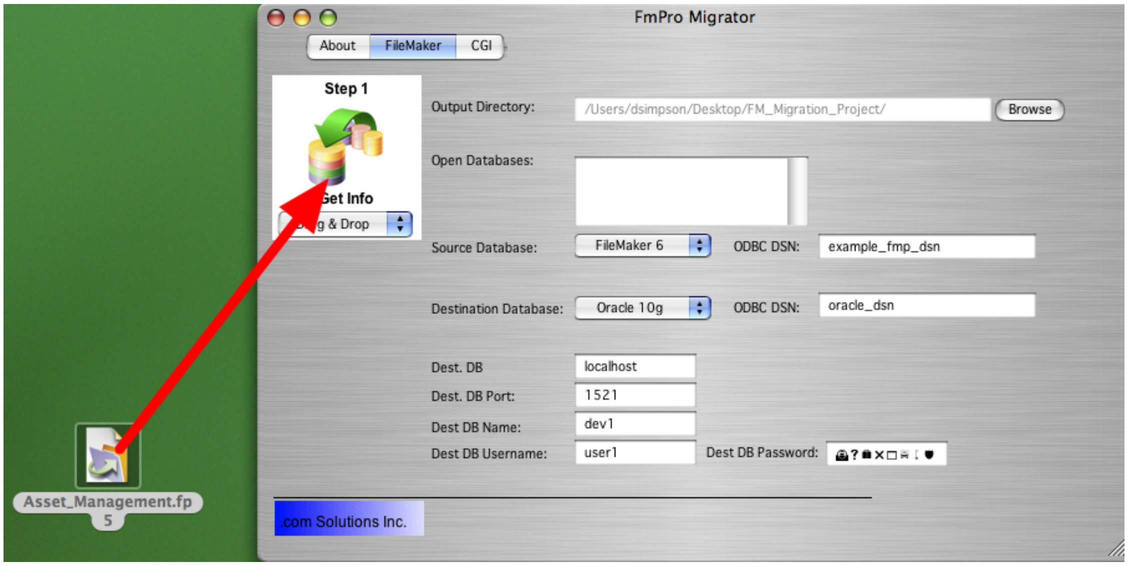
Step 1 - Get Info

FmPro Migrator needs to get info about tables, fields, field types and repeating fields status within the source FileMaker database file(s). This info is used to create the tables/fields within the destination database and is used to copy data between the databases.

There are multiple ways FmPro Migrator can use to get info about the source database file, as listed below. Use the method which works best for your computer OS and the version of the FileMaker database file being migrated.

Select the pop-up menu below the Step 1 Get Info icon to select the method you want to use to get info from the source database file(s). This pop-up menu is context-sensitive, based upon the selected source database type and the platform where FmPro Migrator is running (macOS or Windows).

Step 1 - Get Info - Using Drag & Drop - macOS & Windows - Older FileMaker Versions Only



Output Directory: /Users/dsimpson/Desktop/FM_Migration_Project/

Open Databases:

Source Database: FileMaker 6 ODBC DSN: example_fmp_dsn

Destination Database: Oracle 10g ODBC DSN: oracle_dsn

Dest. DB: localhost

Dest. DB Port: 1521

Dest DB Name: dev1

Dest DB Username: user1 Dest DB Password:

<u>The Drag & Drop Get Info method is only used for older FileMaker versions:</u>

FmPro Migrator reads the metadata directly from older FileMaker database files, including FileMaker 2, 3, 4, 5 and 6 on macOS and Windows. This method of getting info from FileMaker provides a very accurate representation of the source database file, including calculation formula definitions and repeating field counts for each field. This method is deprecated, because more accurate info is obtained by copying tables via the clipboard as shown in the next step. Sometimes, deleted fields can still be seen in the old file if it hasn't been saved as a new empty clone copy (no records).

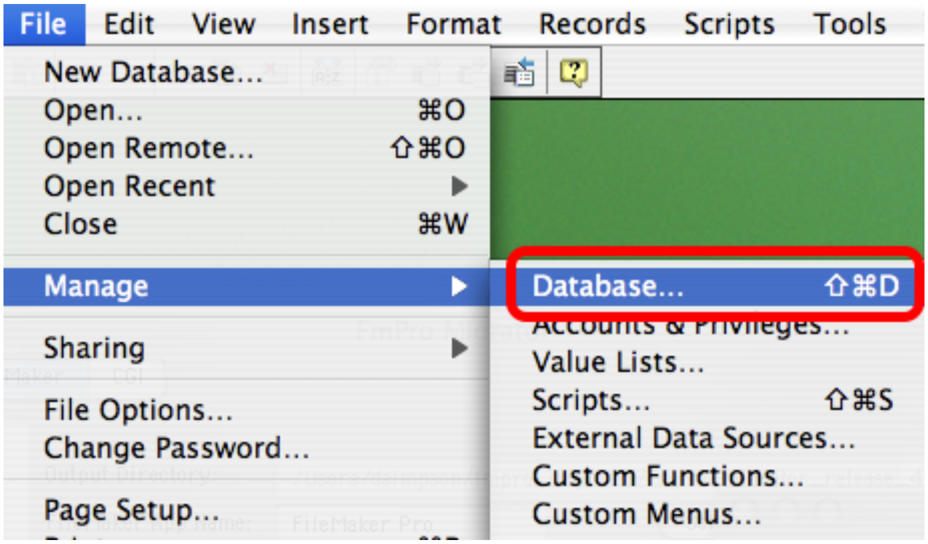
Drag & Drop the empty copy of the source database file onto the Step 1 Get Info icon.

Repeat these steps for each source database file which needs migrated.

Note: Since there is no FileMaker ODBC driver available for older versions of FileMaker on macOS, if you use this method to get info from your source database file(s), you will will then need to upgrade the file(s) to FileMaker 7+ prior to performing the migration. You will then need to change the source database type from FileMaker 5/6 to FileMaker 7+ within FmPro Migrator.

Also, this method of gathering info from the source database does not read relationships, value lists, layouts or scripts from the database file. To obtain these additional pieces of info from the file, the file will need to be upgraded to FileMaker 7+ and a DDR XML file will need to be created with FileMaker Pro.

Step 1 - Get Info - Using ClipBoard XML - macOS & Windows

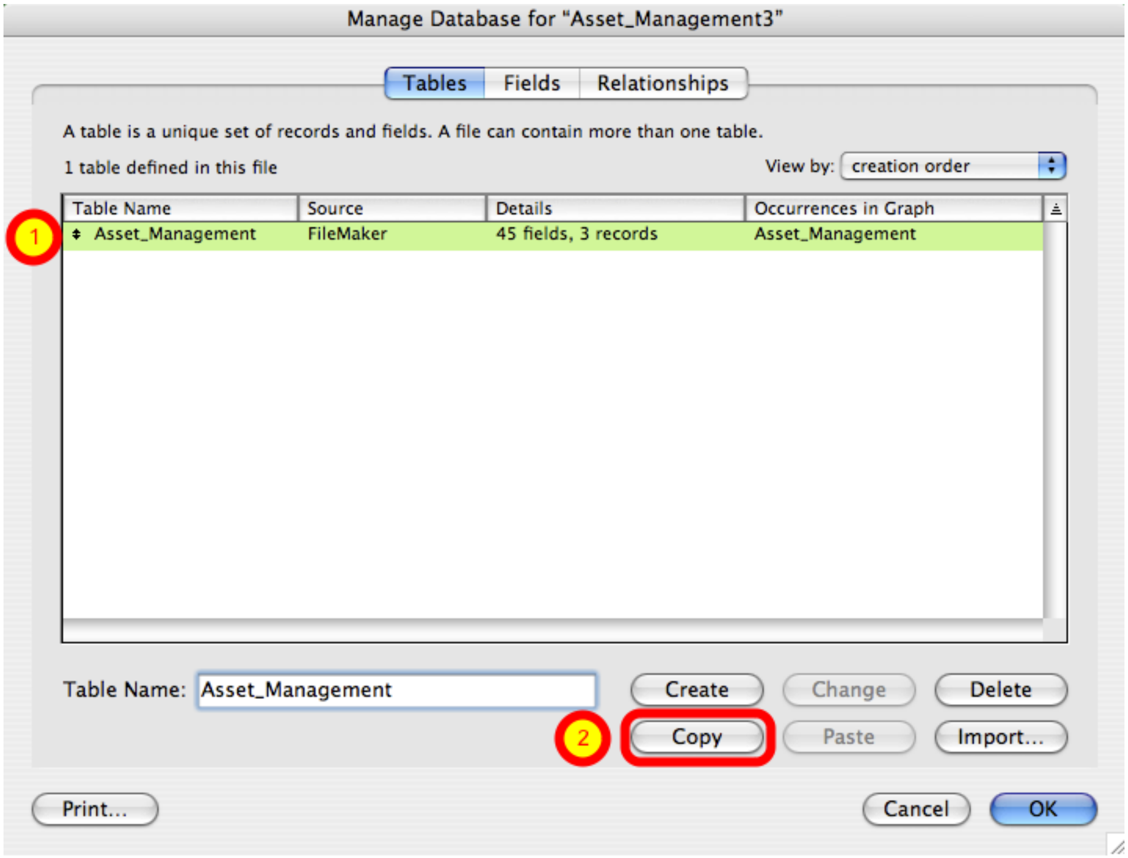


The most desirable method for obtaining info about FileMaker 7+ database files on macOS and Windows is thru copying the table structure using FileMaker Pro (version 17+). This method copies the table creation XML code from the clipboard and includes all of the info required to create the tables, fields, repeating field count and calculation formulas.

Open the FileMaker database file.

Select Define/Manage Database from the File menu within FileMaker Pro Advanced.

Step 1 - Get Info - Copy Table XML



Manage Database for "Asset_Management3"

A table is a unique set of records and fields. A file can contain more than one table.

1 table defined in this file

View by: creation order

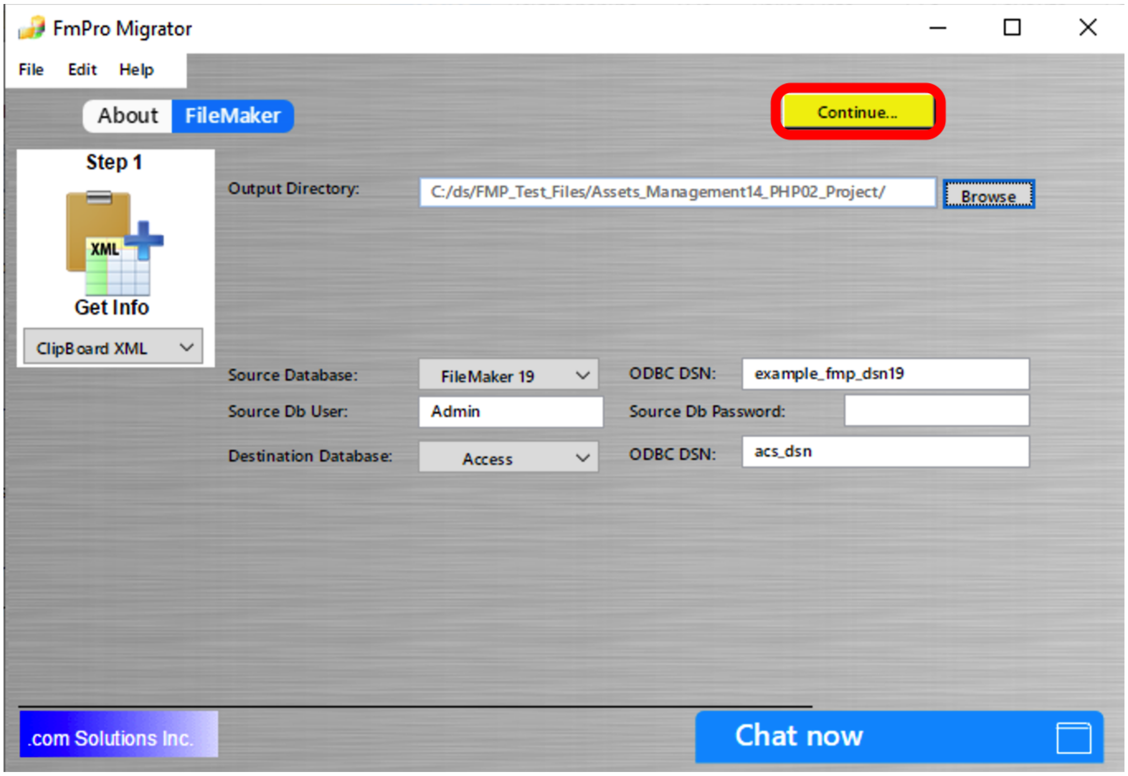
Table Name	Source	Details	Occurrences in Graph
Asset_Management	FileMaker	45 fields, 3 records	Asset_Management

Table Name: Asset_Management

(1) Select all of the tables on the Tables tab, (2) click the Copy button.

Once the Table XML info has been put onto the ClipBoard, the Paste button will become active. Click the Cancel or Ok buttons to close the Define/Manage Database dialog.

Step 1 - Get Info - Continue Button



Output Directory: C:/ds/FMP_Test_Files/Assets_Management14_PHP02_Project/

Source Database: FileMaker 19

ODBC DSN: example_fmp_dsn19

Source Db User: Admin

Source Db Password:

Destination Database: Access

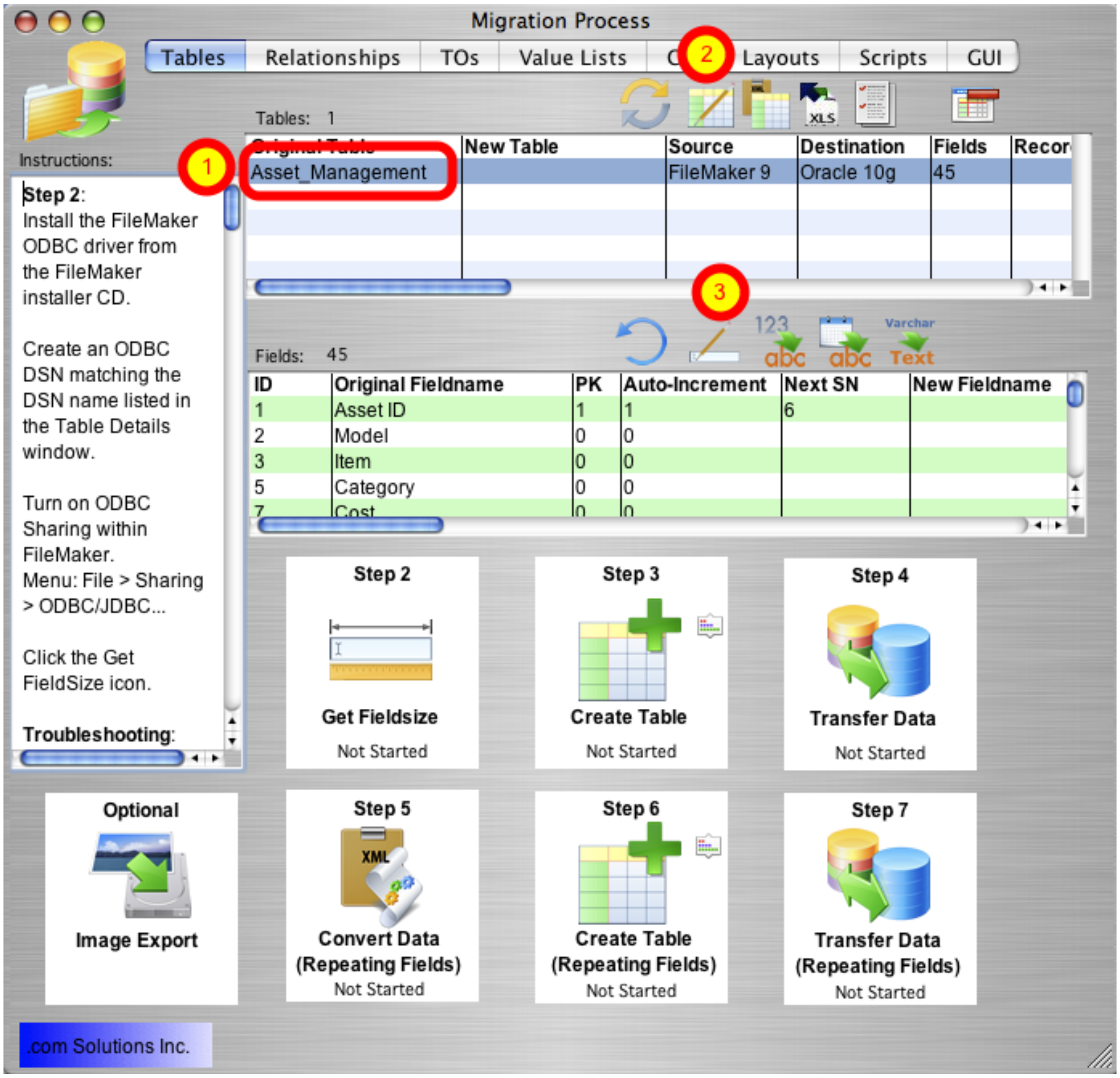
ODBC DSN: acs_dsn

Within FmPro Migrator, click the Step 1 Get Info button (with the ClipBoard XML menu option selected). The yellow Continue... button will become visible once FmPro Migrator has completed processing the Table XML from the ClipBoard.

Step 2

Step 2 - Get Fieldsize

Step 2 - Get Fieldsize - Window Features



Step 2:
Install the FileMaker ODBC driver from the FileMaker installer CD.

Create an ODBC DSN matching the DSN name listed in the Table Details window.

Turn on ODBC Sharing within FileMaker.
Menu: File > Sharing > ODBC/JDBC...

Click the Get FieldSize icon.

Troubleshooting:

Tables: 1

Original Table	New Table	Source	Destination	Fields	Recor
Asset_Management		FileMaker 9	Oracle 10g	45	

Fields: 45

ID	Original Fieldname	PK	Auto-Increment	Next SN	New Fieldname
1	Asset ID	1	1	6	
2	Model	0	0		
3	Item	0	0		
5	Category	0	0		
7	Cost	0	0		

Note: Prior to clicking the Step 2 Get Fieldsize step, an ODBC DSN should have already been created and tested to make sure that FmPro Migrator can communicate with the FileMaker database. It is recommended that this process be done on the local computer to read the FileMaker database instead of running on FileMaker Server. It might be necessary to delete Unstored Calc and Summary fields in the FileMaker database in order to perform the Get Fieldsize or Transfer Data steps - and you don't want to accidentally delete fields from the server while users are running the database.

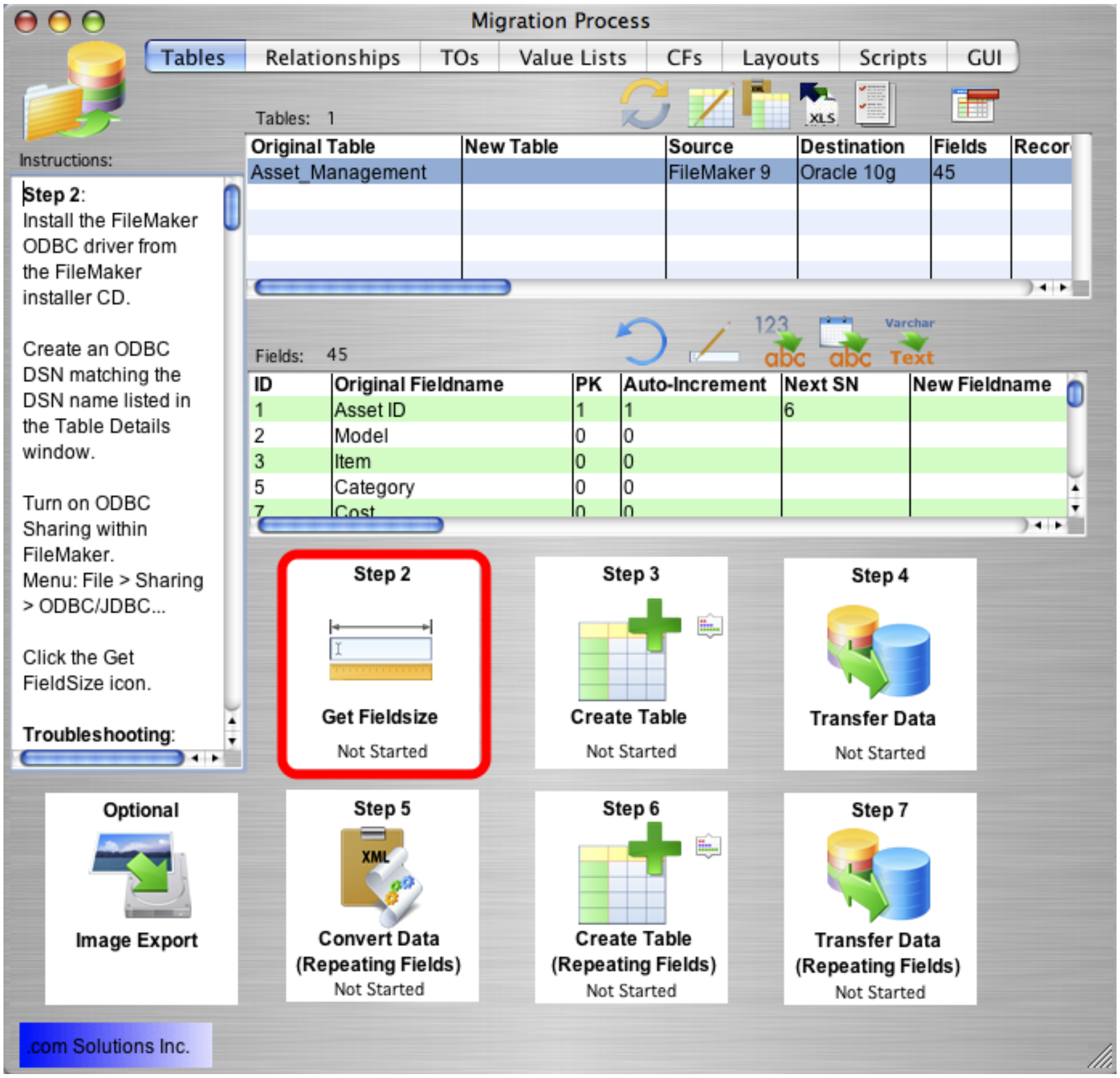
(1) Click on a table in the Tables list. Once a table is selected, a list of fields contained within the table will be displayed in the Fields list. These fields represent the info gathered from the source database by FmPro Migrator.

(2) Clicking the Table Details button opens the Table Details window, which enables you to view

and make changes to the individual parameters for the table.

(3) The Field Details button provides you with the ability to view and make changes at the field level.

Step 2 - Get Fieldsize



Instructions:

Step 2:
Install the FileMaker ODBC driver from the FileMaker installer CD.

Create an ODBC DSN matching the DSN name listed in the Table Details window.

Turn on ODBC Sharing within FileMaker.
Menu: File > Sharing > ODBC/JDBC...

Click the Get FieldSize icon.

Troubleshooting:

Tables: 1

Original Table	New Table	Source	Destination	Fields	Recor
Asset_Management		FileMaker 9	Oracle 10g	45	

Fields: 45

ID	Original Fieldname	PK	Auto-Increment	Next SN	New Fieldname
1	Asset ID	1	1	6	
2	Model	0	0		
3	Item	0	0		
5	Category	0	0		
7	Cost	0	0		

Click the Step 2 Get Fieldsize button. FmPro Migrator will make an ODBC connection to the source FileMaker Pro database, and put up a progress dialog as it is reading thru the records. After the records have been successfully read from the table, the status menu at the bottom of the Step 2 button will change from "Not Started" to "Completed". The status will be set to "Failed" if the Get Fieldsize step fails for some reason.

If the Get Fieldsize step fails:
1) Check to make sure that the source database is open within FileMaker Pro on the local computer.
2) Make sure that ODBC sharing is enabled.

3) For FileMaker 7+ databases, make sure that there is a TO on the RelationshipGraph which exactly matches the name of the base table being queried by FmPro Migrator.
4) Open the Define/Manage Database dialog and verify the existence of the base table within the FileMaker Pro database.

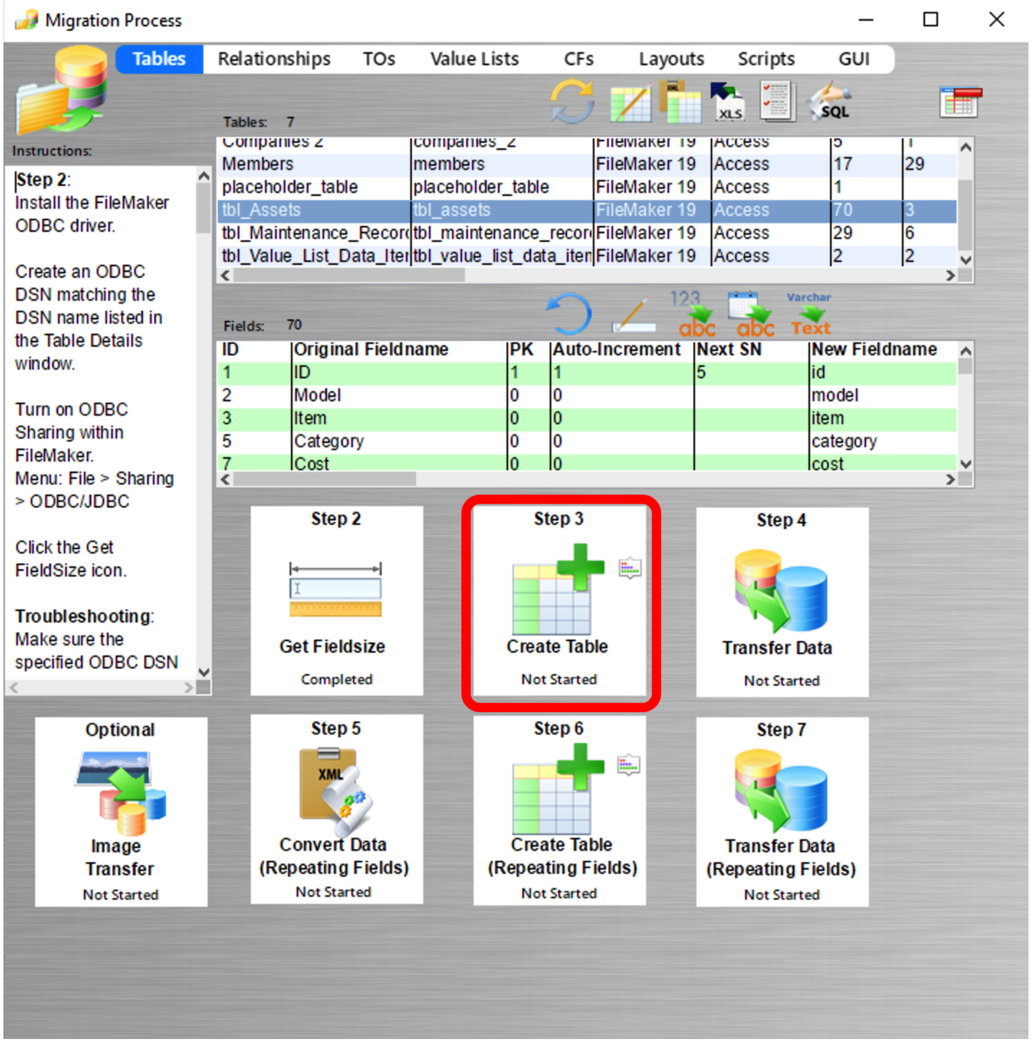
Step 3

Step 3 - Create Table - FileMaker to Access

Once the Get Fieldsize step has been completed, FmPro Migrator has enough information to create the SQL code defining the table in the destination database. The fieldsize info is automatically used to create table columns of the correct size when creating the table in the Access database.
Text fields containing more than 255 characters are converted to Access LongText/Memo columns and FileMaker Pro container fields are converted into Access OLE Object columns.

Note: Before clicking the Step 3 - Create Table or Step 4 - Transfer Data steps, make sure that the ODBC DSN for the Access database has been created. Otherwise the Create Table step will fail and may cause FmPro Migrator to crash.

Step 3 - Create Table



Tables: 7

Companies 2	companies_2	FileMaker 19	Access	5	1
Members	members	FileMaker 19	Access	17	29
placeholder_table	placeholder_table	FileMaker 19	Access	1	
tbl_Assets	tbl_assets	FileMaker 19	Access	70	3
tbl_Maintenance_Recor	tbl_maintenance_recor	FileMaker 19	Access	29	6
tbl_Value_List_Data_Iter	tbl_value_list_data_iten	FileMaker 19	Access	2	2

Fields: 70

ID	Original Fieldname	PK	Auto-Increment	Next SN	New Fieldname
1	ID	1	1	5	id
2	Model	0	0		model
3	Item	0	0		item
5	Category	0	0		category
7	Cost	0	0		cost

Step 2:
Install the FileMaker ODBC driver.

Create an ODBC DSN matching the DSN name listed in the Table Details window.

Turn on ODBC Sharing within FileMaker.
Menu: File > Sharing > ODBC/JDBC

Click the Get FieldSize icon.

Troubleshooting:
Make sure the specified ODBC DSN

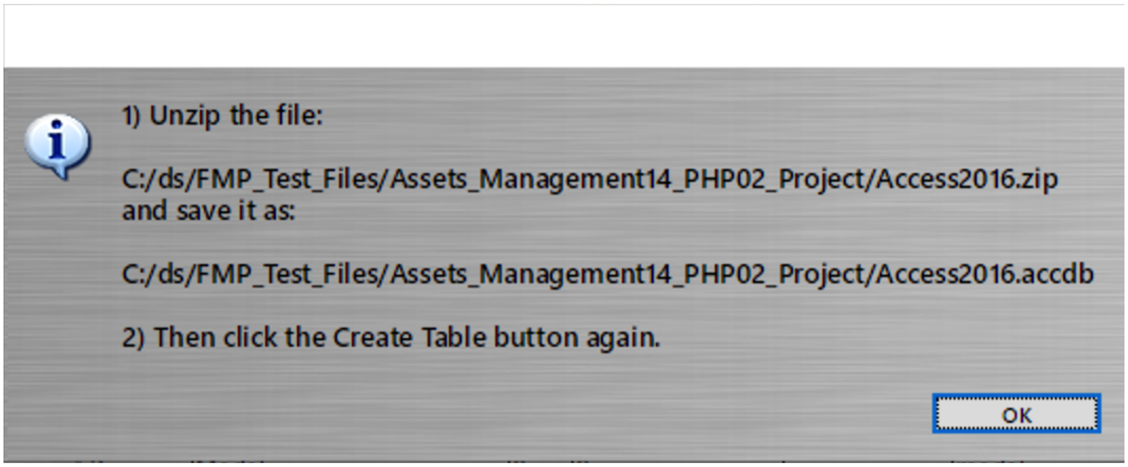
Click the Step 3 Create Table button to create the table in the destination database.

FmPro Migrator will generate the table creation SQL code, connect to the destination database and create the table. Once the table has been successfully created, the status menu under the Step 3 Create Table button will be changed to "Completed".

Holding down the shift key while clicking the Step 3 Create Table button drops and re-creates the table in the destination database.

Warning: Dropping the table in the destination database causes the loss of all data within the table.

Create Table Dialog - Access2016.zip File



1) Unzip the file:

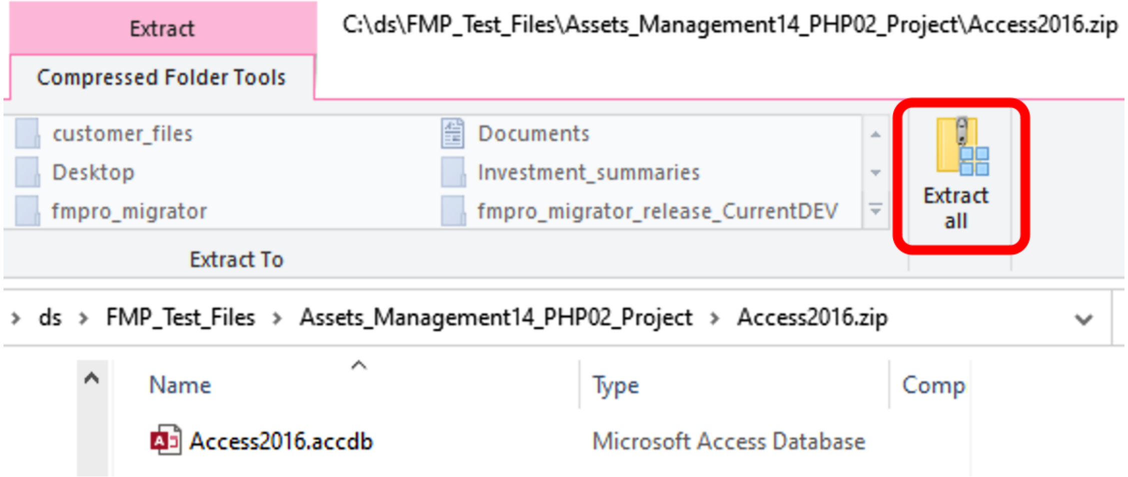
C:/ds/FMP_Test_Files/Assets_Management14_PHP02_Project/Access2016.zip
and save it as:

C:/ds/FMP_Test_Files/Assets_Management14_PHP02_Project/Access2016.accdb

2) Then click the Create Table button again.

FmPro Migrator will automatically save and open the Access2016.zip file within the output directory. This will be done when creating the first database table within the Access database.

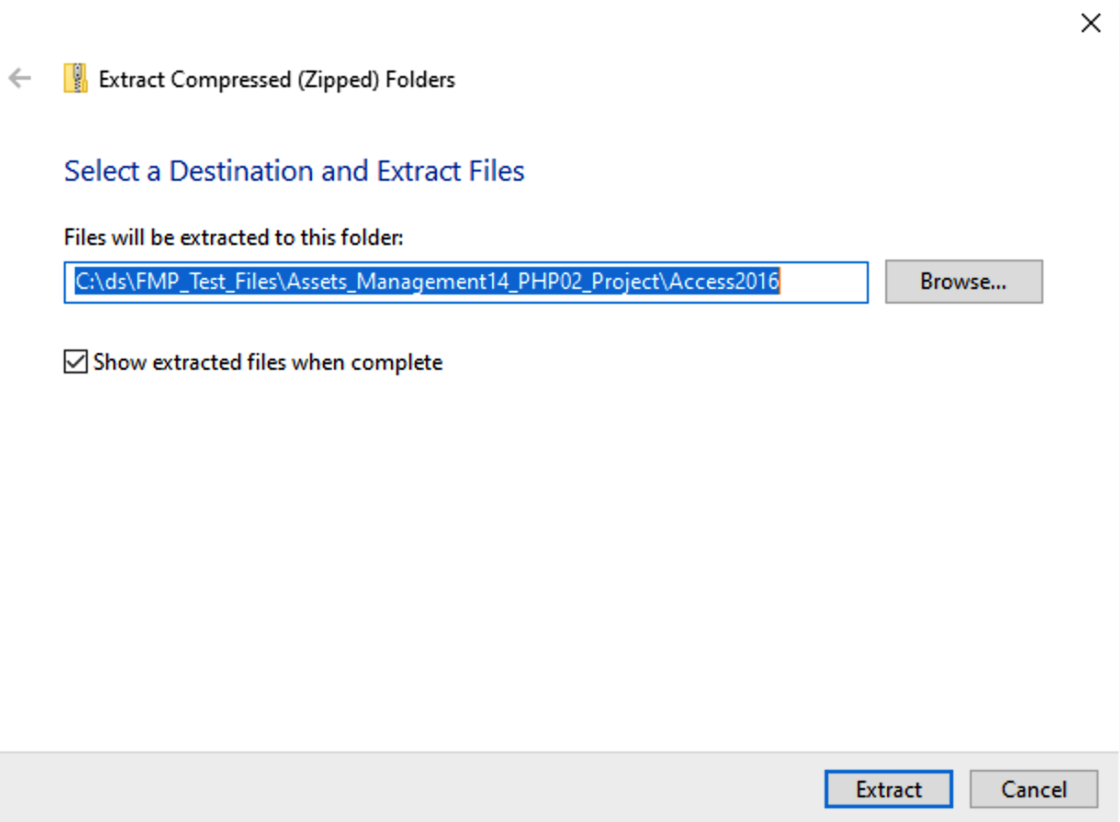
Unzip the Access2016.accdb File



Extract
C:\ds\FMP_Test_Files\Assets_Management14_PHP02_Project\Access2016.zip
Compressed Folder Tools
customer_files
Desktop
fmpro_migrator
Documents
Investment_summaries
fmpro_migrator_release_CurrentDEV
Extract all
Extract To
ds › FMP_Test_Files › Assets_Management14_PHP02_Project › Access2016.zip
Name
Type
Comp
Access2016.accdb
Microsoft Access Database

Click the Extract All button and extract the Access2016.accdb file into the output directory so that FmPro Migrator can locate the file automatically. If you extract the file to another location, the file DSN created by FmPro Migrator won't find the database when creating the table.

Click Trust Publisher or Open Button within Security Dialog



Select a Destination and Extract Files

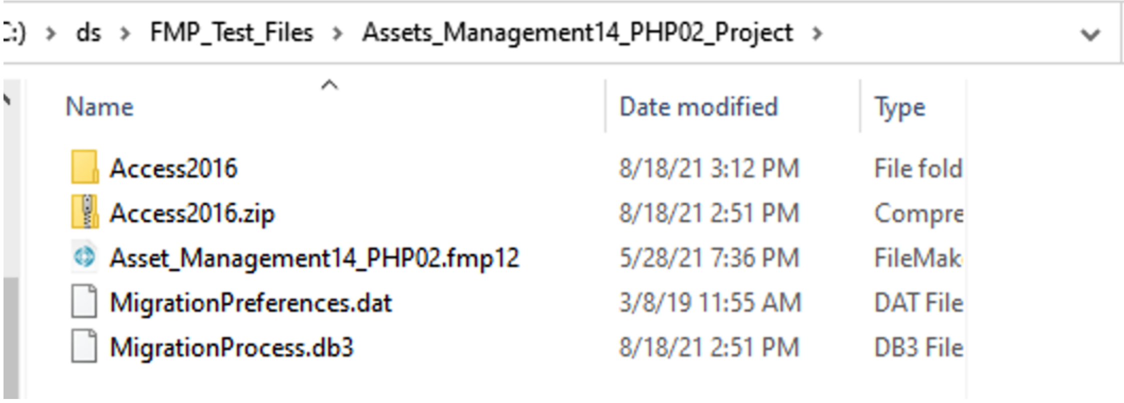
Files will be extracted to this folder:

C:\ds\FMP_Test_Files\Assets_Management14_PHP02_Project\Access2016

☑ Show extracted files when complete

The Windows Extract Compressed (Zipped) Folders window will usually select the correct output folder (the same folder where the zip file was written by FmPro Migrator).

Move Access2016.accdb File

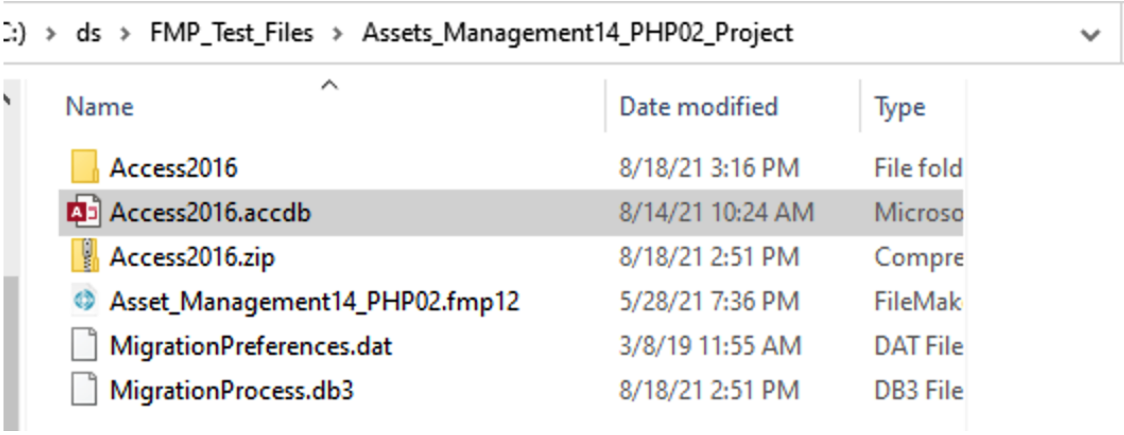


:) › ds › FMP_Test_Files › Assets_Management14_PHP02_Project ›

Name	Date modified	Type
Access2016	8/18/21 3:12 PM	File fold
Access2016.zip	8/18/21 2:51 PM	Compre
Asset_Management14_PHP02.fmp12	5/28/21 7:36 PM	FileMak
MigrationPreferences.dat	3/8/19 11:55 AM	DAT File
MigrationProcess.db3	8/18/21 2:51 PM	DB3 File

Windows will place the extracted Access2016.accdb file into a folder - which **must be moved manually** into the folder containing the MigrationProcess.db3 file.

Access2016.accdb File in Output Directory

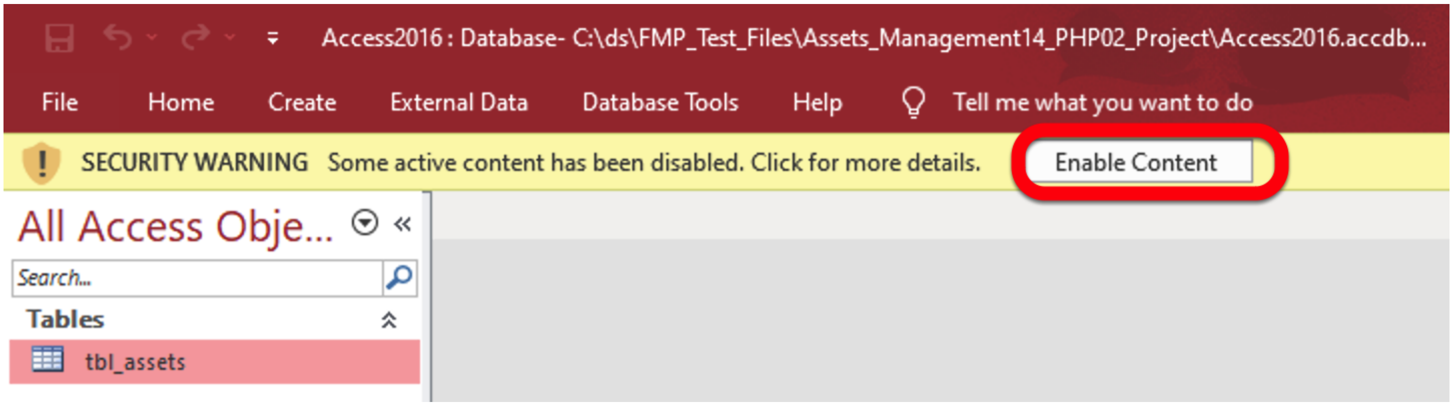


:) › ds › FMP_Test_Files › Assets_Management14_PHP02_Project

Name	Date modified	Type
Access2016	8/18/21 3:16 PM	File fold
Access2016.accdb	8/14/21 10:24 AM	Microso
Access2016.zip	8/18/21 2:51 PM	Compre
Asset_Management14_PHP02.fmp12	5/28/21 7:36 PM	FileMak
MigrationPreferences.dat	3/8/19 11:55 AM	DAT File
MigrationProcess.db3	8/18/21 2:51 PM	DB3 File

It is easy to use the Windows contextual menus to simply Cut then Paste the file into the correct location - as shown here.

Opening the Access2016.accdb File

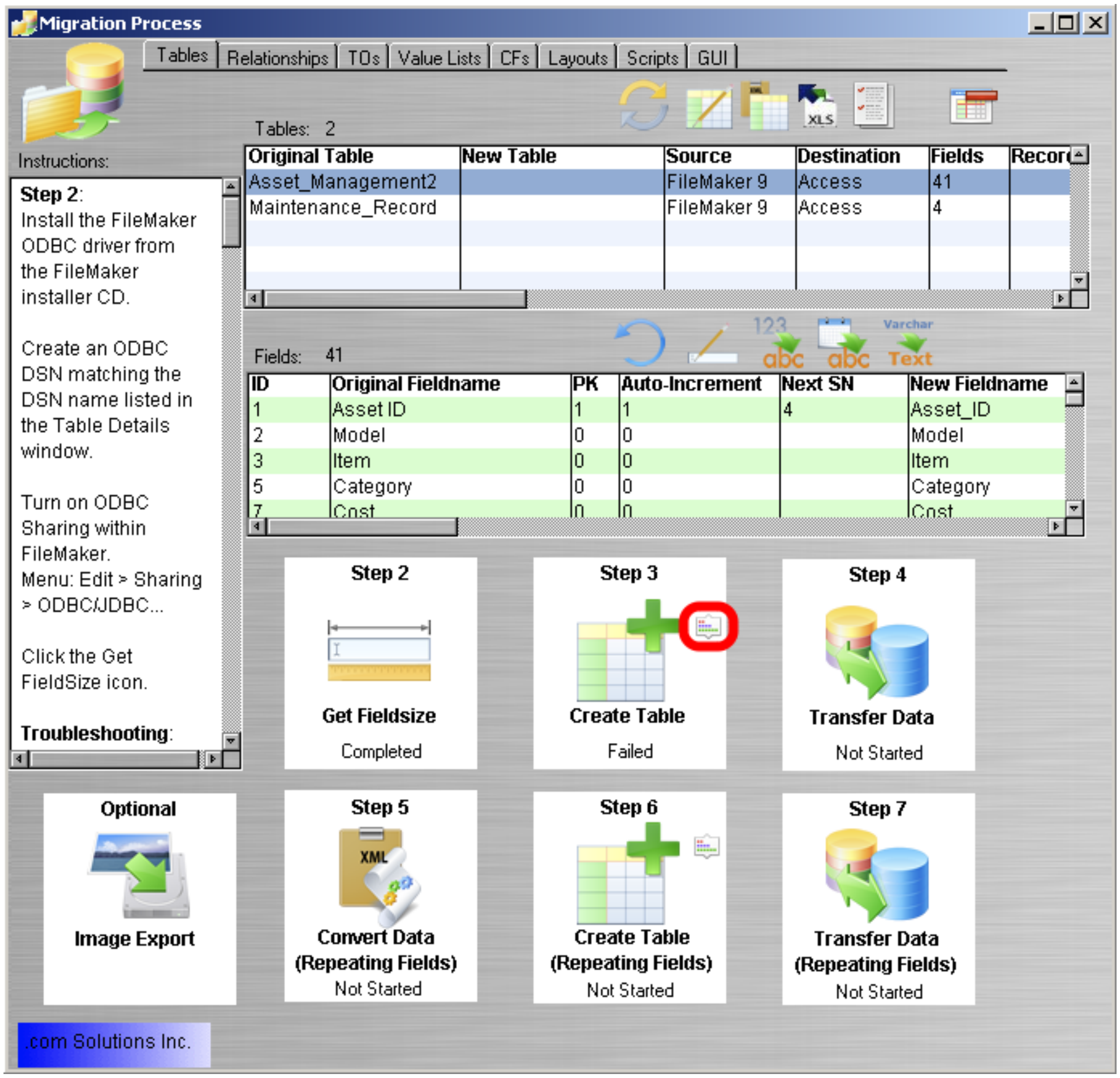


Access2016 : Database- C:\ds\FMP_Test_Files\Assets_Management14_PHP02_Project\Access2016.accdb...

SECURITY WARNING Some active content has been disabled. Click for more details.

Click the Enable Content button when first opening the Access2016.accdb file.

Step 3 - Table Creation SQL Details



Instructions:

Step 2:
Install the FileMaker ODBC driver from the FileMaker installer CD.

Create an ODBC DSN matching the DSN name listed in the Table Details window.

Turn on ODBC Sharing within FileMaker.
Menu: Edit > Sharing > ODBC/JDBC...

Click the Get FieldSize icon.

Troubleshooting:

Tables: 2

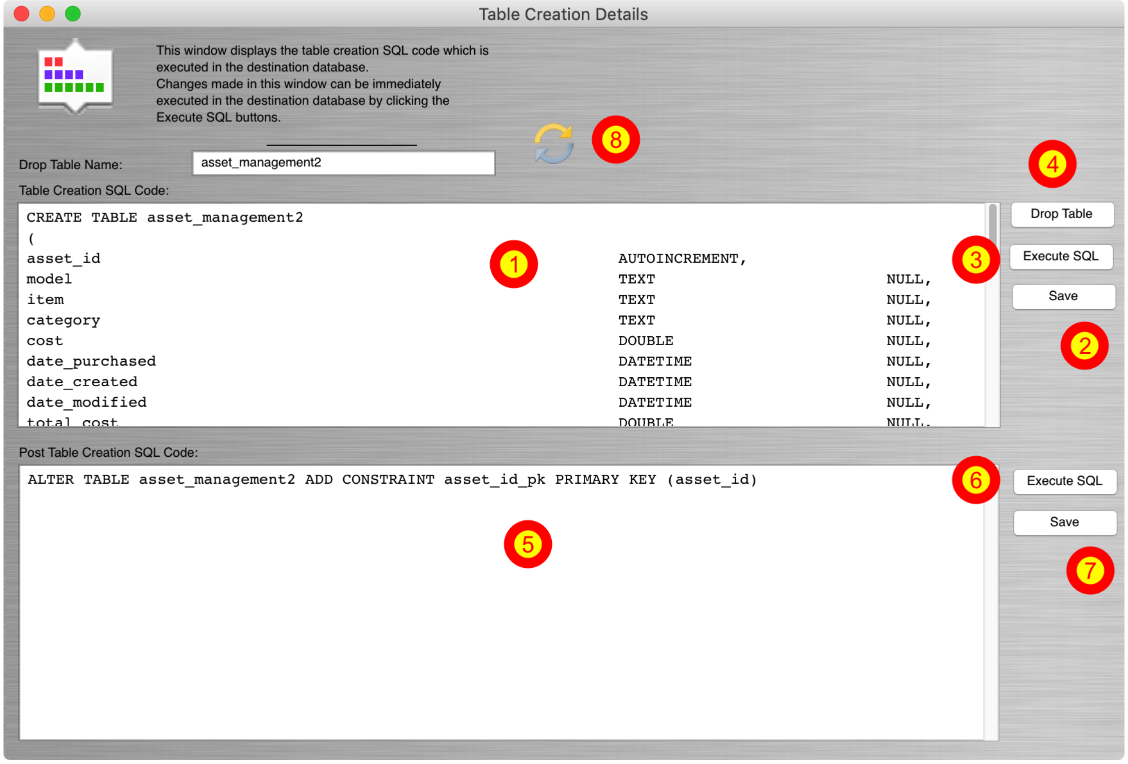
Original Table	New Table	Source	Destination	Fields	Recor
Asset_Management2		FileMaker 9	Access	41	
Maintenance_Record		FileMaker 9	Access	4	

Fields: 41

ID	Original Fieldname	PK	Auto-Increment	Next SN	New Fieldname
1	Asset ID	1	1	4	Asset_ID
2	Model	0	0		Model
3	Item	0	0		Item
5	Category	0	0		Category
7	Cost	0	0		Cost

if problems occur while creating the table in the destination database, you can review the table creation SQL code generated by FmPro Migrator by clicking the Table Creation Details button, next to the Step 3 Create Table icon.

Step 3 - Create Table - Table Creation Details Window



This window displays the table creation SQL code which is executed in the destination database.
Changes made in this window can be immediately executed in the destination database by clicking the Execute SQL buttons.

Drop Table Name: asset_management2

Table Creation SQL Code:

```
CREATE TABLE asset_management2
(
asset_id                                                AUTOINCREMENT,
model                                                   TEXT                 NULL,
item                                                    TEXT                 NULL,
category                                                TEXT                 NULL,
cost                                                    DOUBLE               NULL,
date_purchased                                          DATETIME             NULL,
date_created                                            DATETIME             NULL,
date_modified                                           DATETIME             NULL,
total_cost                                              DOUBLE               NULL,
```

Post Table Creation SQL Code:

```
ALTER TABLE asset_management2 ADD CONSTRAINT asset_id_pk PRIMARY KEY (asset_id)
```

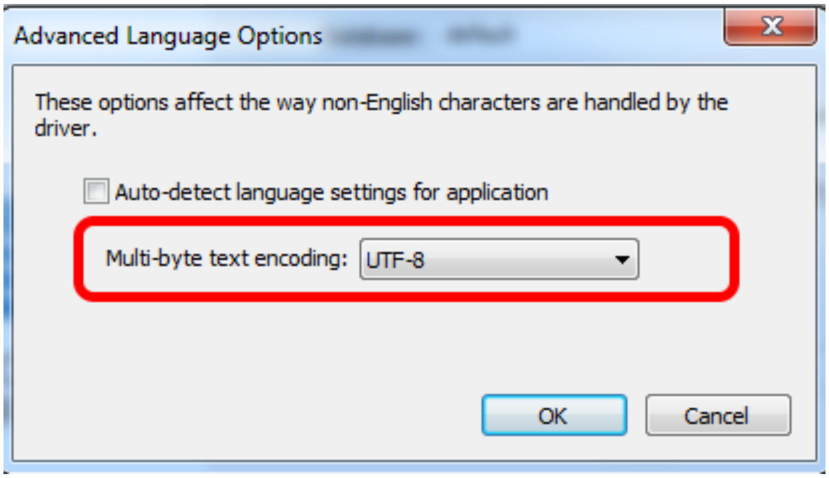
From within the Table Creation Details window, the Table Creation SQL code can be (1) manually edited, (2) saved or (3) re-executed in the Access database. The table can also be (4) dropped in the Access database.

The Post Table Creation SQL code can also be (5) manually edited, (6) executed, and (7) saved.

Clicking the (8) refresh button re-loads the previously stored version of the SQL code from the SQLite database.

Step 4

Step 4 - Transfer Data - Unicode Characters



Advanced Language Options

These options affect the way non-English characters are handled by the driver.

□ Auto-detect language settings for application

Multi-byte text encoding: UTF-8

As of FmPro Migrator 7.58, NVARCHAR/NVARCHAR(MAX) columns are created within SQL Server databases.

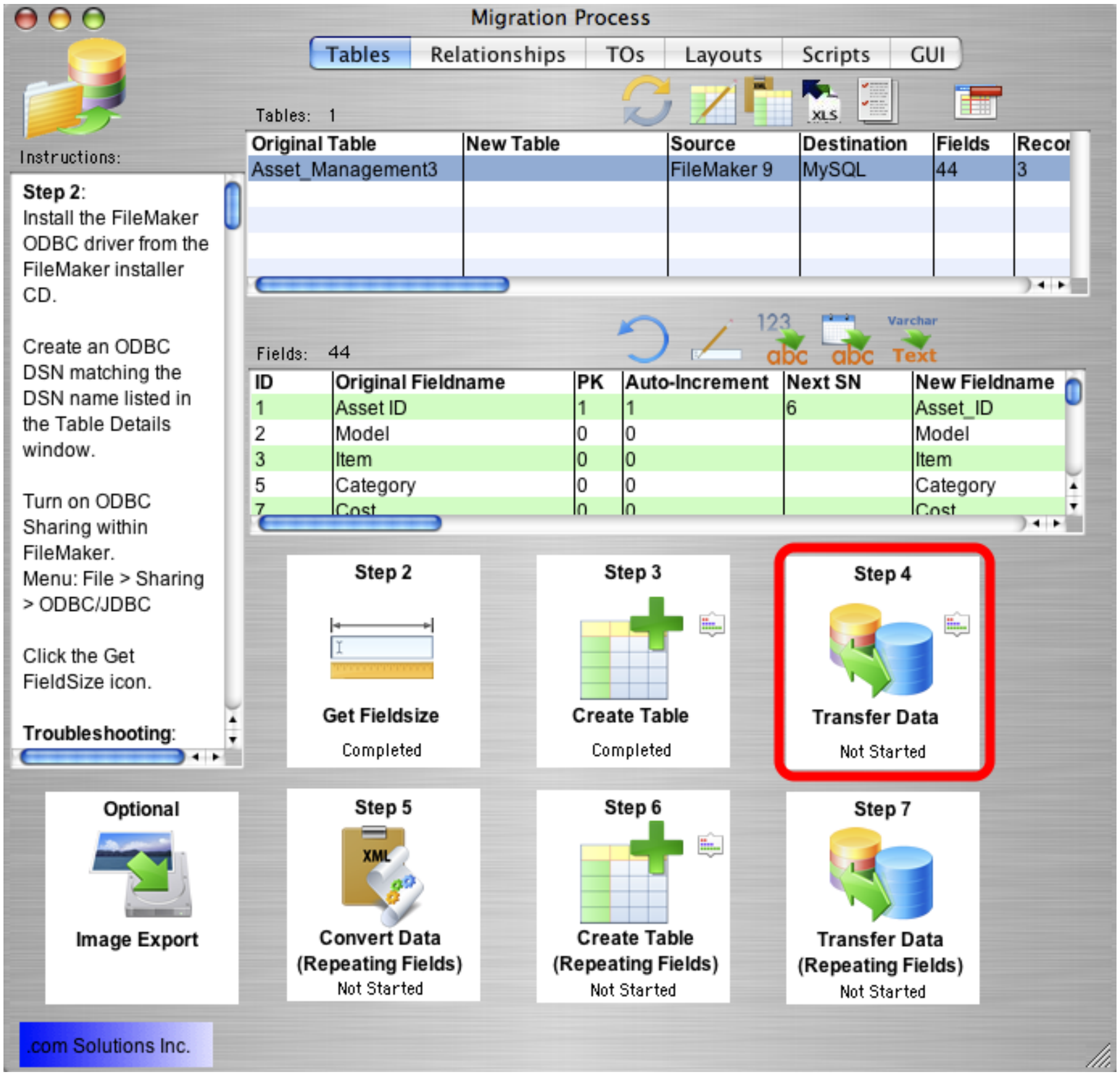
As of FmPro Migrator 9.37, FmPro Migrator always uses the SQL_WVARCHAR binding within the Perl script to insure proper UTF-8 encoding of the data for Access databases.

In general you should always select UTF-8 in the Advanced Language Options section of the FileMaker ODBC Driver 11+ versions.

Note: For most databases, it is no longer necessary to hold down the Shift key when clicking the Transfer Data step for full Unicode compatibility. For Microsoft Access and SQL Server databases, FmPro Migrator always performs this task for you automatically, as indicated by a small command prompt window showing the data transfer in progress. Nothing is actually displayed in this window presently. If you wanted to see the data transfer in progress, you could use the command prompt on Windows to run the xferData.EXE application which then runs the xferData.pl script.

DSN Configuration Note: Since the xferData.EXE app is used for transferring data to Access databases, the ODBC DSN for the Access database will need to be configured to point to the Access2016.accdb file. The automatically generated file DSN is only used for creating the table in the Access database. So if you can create the table, but not transfer the data - the ODBC DSN configured in FmPro Migrator probably needs updated to use the Access2016.accdb database file.

Step 4 - Transfer Data



Instructions:

Step 2:
Install the FileMaker ODBC driver from the FileMaker installer CD.

Create an ODBC DSN matching the DSN name listed in the Table Details window.

Turn on ODBC Sharing within FileMaker.
Menu: File > Sharing > ODBC/JDBC

Click the Get FieldSize icon.

Troubleshooting:

Tables: 1

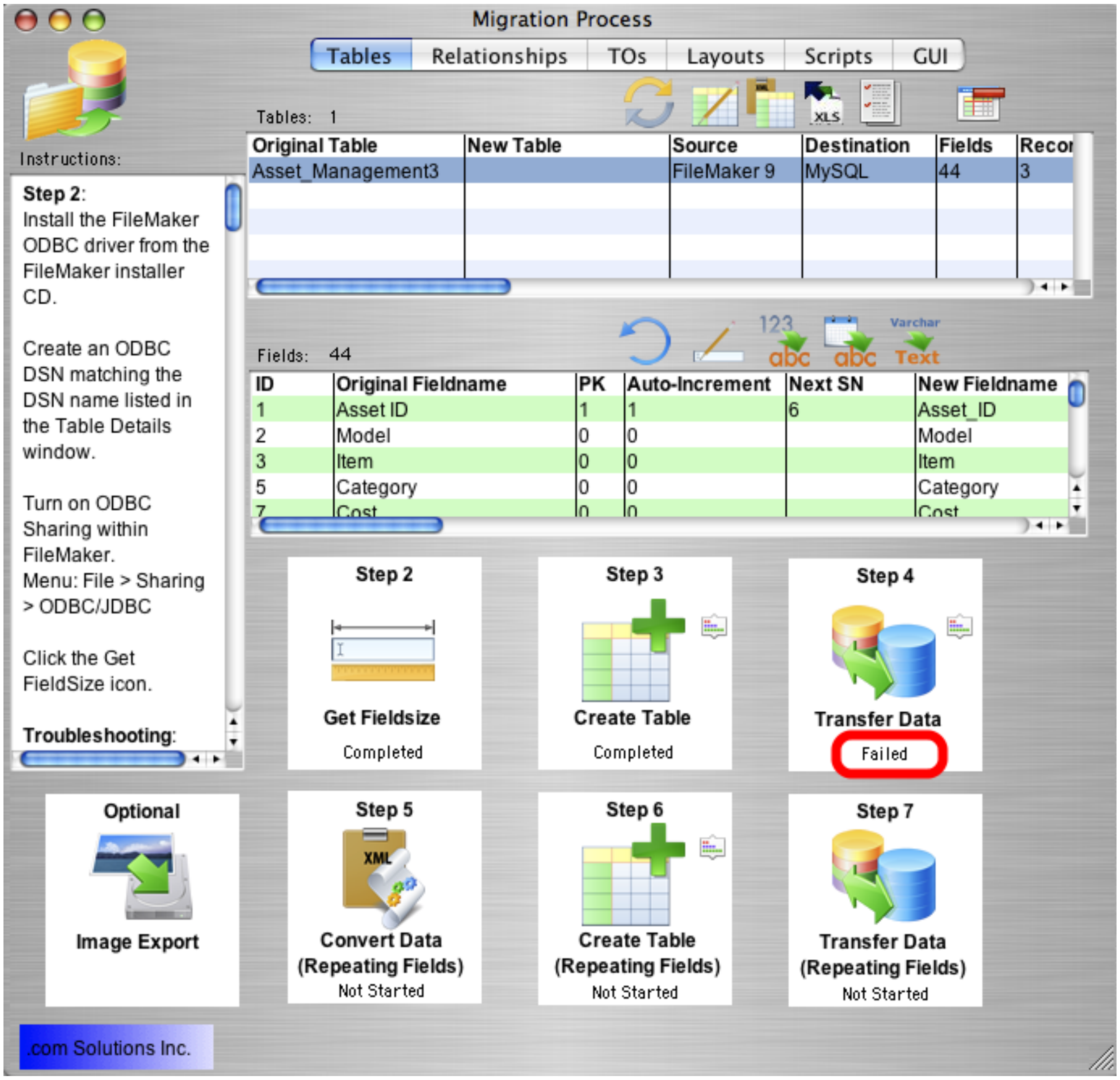
Original Table	New Table	Source	Destination	Fields	Recor
Asset_Management3		FileMaker 9	MySQL	44	3

Fields: 44

ID	Original Fieldname	PK	Auto-Increment	Next SN	New Fieldname
1	Asset ID	1	1	6	Asset_ID
2	Model	0	0		Model
3	Item	0	0		Item
5	Category	0	0		Category
7	Cost	0	0		Cost

Click the Step 4 Transfer Data button to transfer data from the source table in the FileMaker database to the newly created table in the destination database. Once the data has been transferred successfully, the status menu below the Step 4 Transfer Data button will change from "Not Started" to "Completed".

Step 4 - Transfer Data - Troubleshooting



Instructions:

Step 2:
Install the FileMaker ODBC driver from the FileMaker installer CD.

Create an ODBC DSN matching the DSN name listed in the Table Details window.

Turn on ODBC Sharing within FileMaker.
Menu: File > Sharing > ODBC/JDBC

Click the Get FieldSize icon.

Troubleshooting:

Tables: 1

Original Table	New Table	Source	Destination	Fields	Recor
Asset_Management3		FileMaker 9	MySQL	44	3

Fields: 44

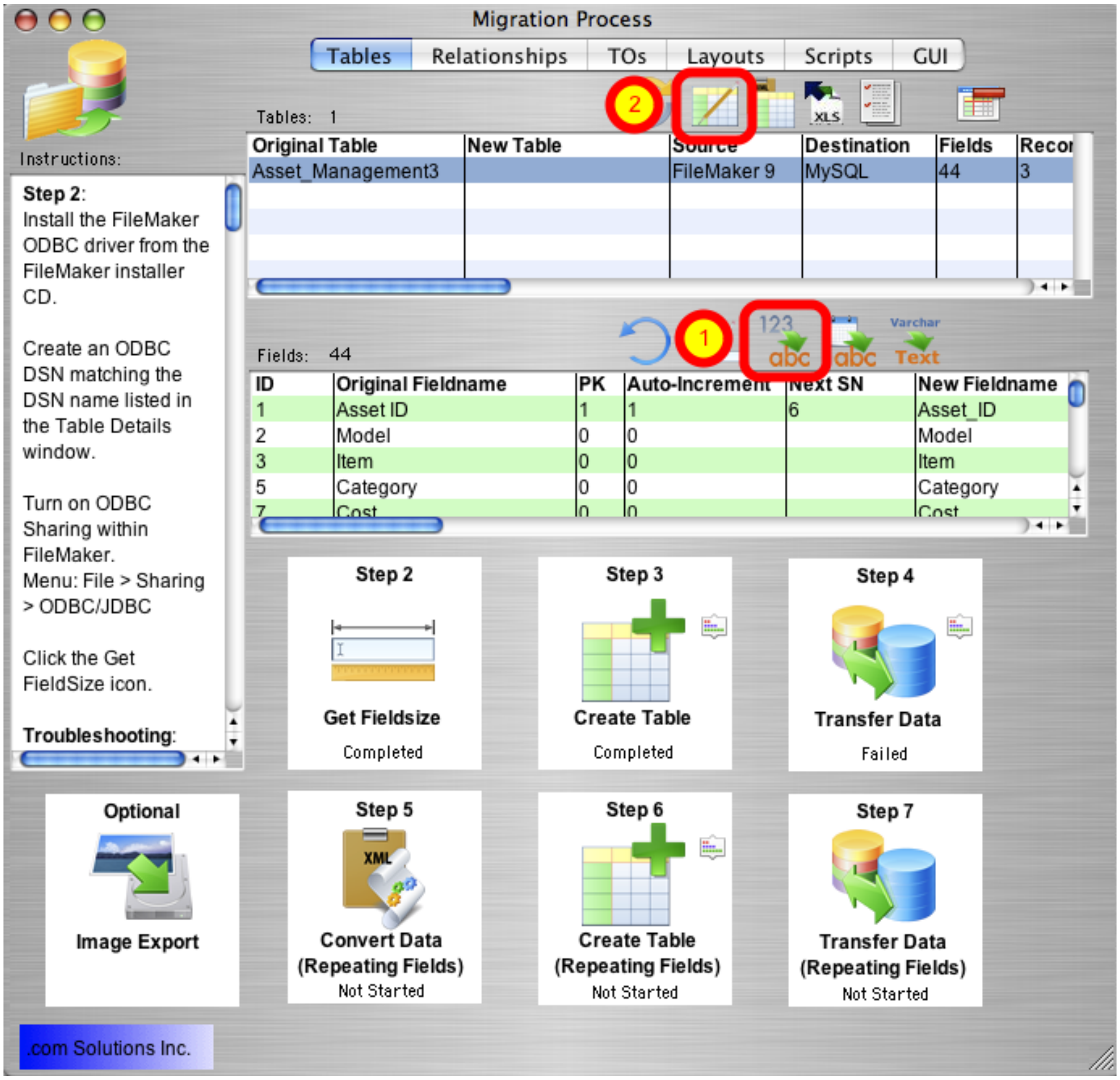
ID	Original Fieldname	PK	Auto-Increment	Next SN	New Fieldname
1	Asset ID	1	1	6	Asset_ID
2	Model	0	0		Model
3	Item	0	0		Item
5	Category	0	0		Category
7	Cost	0	0		Cost

Step 2 – Get Fieldsize – Completed
Step 3 – Create Table – Completed
Step 4 – Transfer Data – Failed
Optional – Image Export
Step 5 – Convert Data (Repeating Fields) – Not Started
Step 6 – Create Table (Repeating Fields) – Not Started
Step 7 – Transfer Data (Repeating Fields) – Not Started

If the data transfer process fails, an error message will be displayed containing the text of the error message returned by the destination database.

Some of the most common data transfer errors include:
FileMaker numeric fields containing non-numeric data.
FileMaker date fields containing non-date/time data.
FileMaker primary key fields containing duplicate or empty values.

Step 4 - Transfer Data - Troubleshooting - Non-Numeric Data Type Error



Migration Process

Tables: 1

Original Table	New Table	Source	Destination	Fields	Recor
Asset_Management3		FileMaker 9	MySQL	44	3

Instructions:

Step 2:
Install the FileMaker ODBC driver from the FileMaker installer CD.

Create an ODBC DSN matching the DSN name listed in the Table Details window.

Turn on ODBC Sharing within FileMaker.
Menu: File > Sharing > ODBC/JDBC

Click the Get FieldSize icon.

Troubleshooting:

Fields: 44

ID	Original Fieldname	PK	Auto-Increment	Next SN	New Fieldname
1	Asset ID	1	1	6	Asset_ID
2	Model	0	0		Model
3	Item	0	0		Item
5	Category	0	0		Category
7	Cost	0	0		Cost

Step 2 – Get Fieldsize – Completed

Step 3 – Create Table – Completed

Step 4 – Transfer Data – Failed

Optional – Image Export

Step 5 – Convert Data (Repeating Fields) – Not Started

Step 6 – Create Table (Repeating Fields) – Not Started

Step 7 – Transfer Data (Repeating Fields) – Not Started

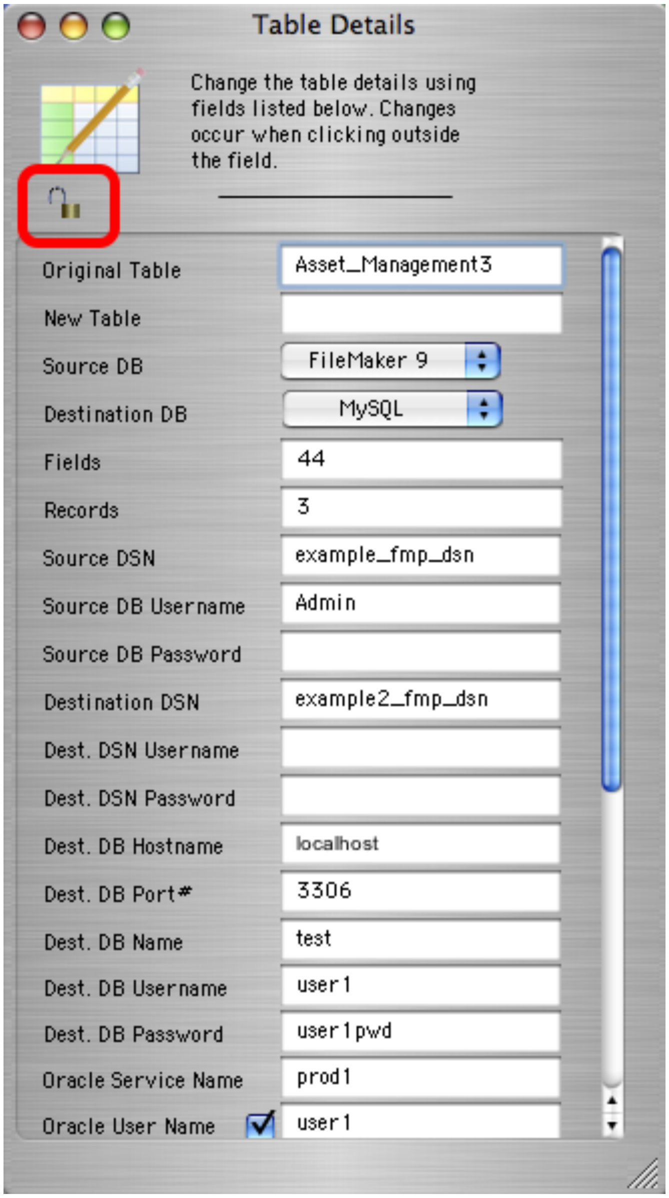
.com Solutions Inc.

The FileMaker application is very forgiving of the types of data which can be entered into each field. Therefore it is possible for non-numeric data to be entered into numeric fields if field validation has not been enabled for these fields. However SQL databases generally won't accept non-numeric data within numeric fields, which will cause an error during the data transfer process.

If this type of error occurs:
Option 1: Correct the data within the FileMaker database in order to resolve the problem. Then drop and re-create the table in the destination database and click the Step 4 Transfer Data button again.

Option 2: Correct the data after transferring the data to the destination database. To get the data transferred to the destination database, click the Set Numeric Fields to Text button, above the Fields list. This button instructs FmPro Migrator to change all of the numeric field types to varchar field types in the table creation SQL code for the destination table.

Step 4 - Transfer Data - Lock Table Details



Change the table details using fields listed below. Changes occur when clicking outside the field.

Original Table	Asset_Management3
New Table	
Source DB	FileMaker 9
Destination DB	MySQL
Fields	44
Records	3
Source DSN	example_fmp_dsn
Source DB Username	Admin
Source DB Password	
Destination DSN	example2_fmp_dsn
Dest. DSN Username	
Dest. DSN Password	
Dest. DB Hostname	localhost
Dest. DB Port#	3306
Dest. DB Name	test
Dest. DB Username	user1
Dest. DB Password	user1pwd
Oracle Service Name	prod1
Oracle User Name ☑	user1

After clicking the Numeric to Varchar, Date to Varchar or Varchar to TEXT buttons, open the Table Details window and click the lock icon. Setting the lock icon prevents FmPro Migrator from re-setting the field types back to their original values when clicking the Step 3 Create Table button.

Hold down the Shift key and click the Step 3 Create Table button to drop and re-create the table in the destination database, then click the Step 4 Transfer Data button again.

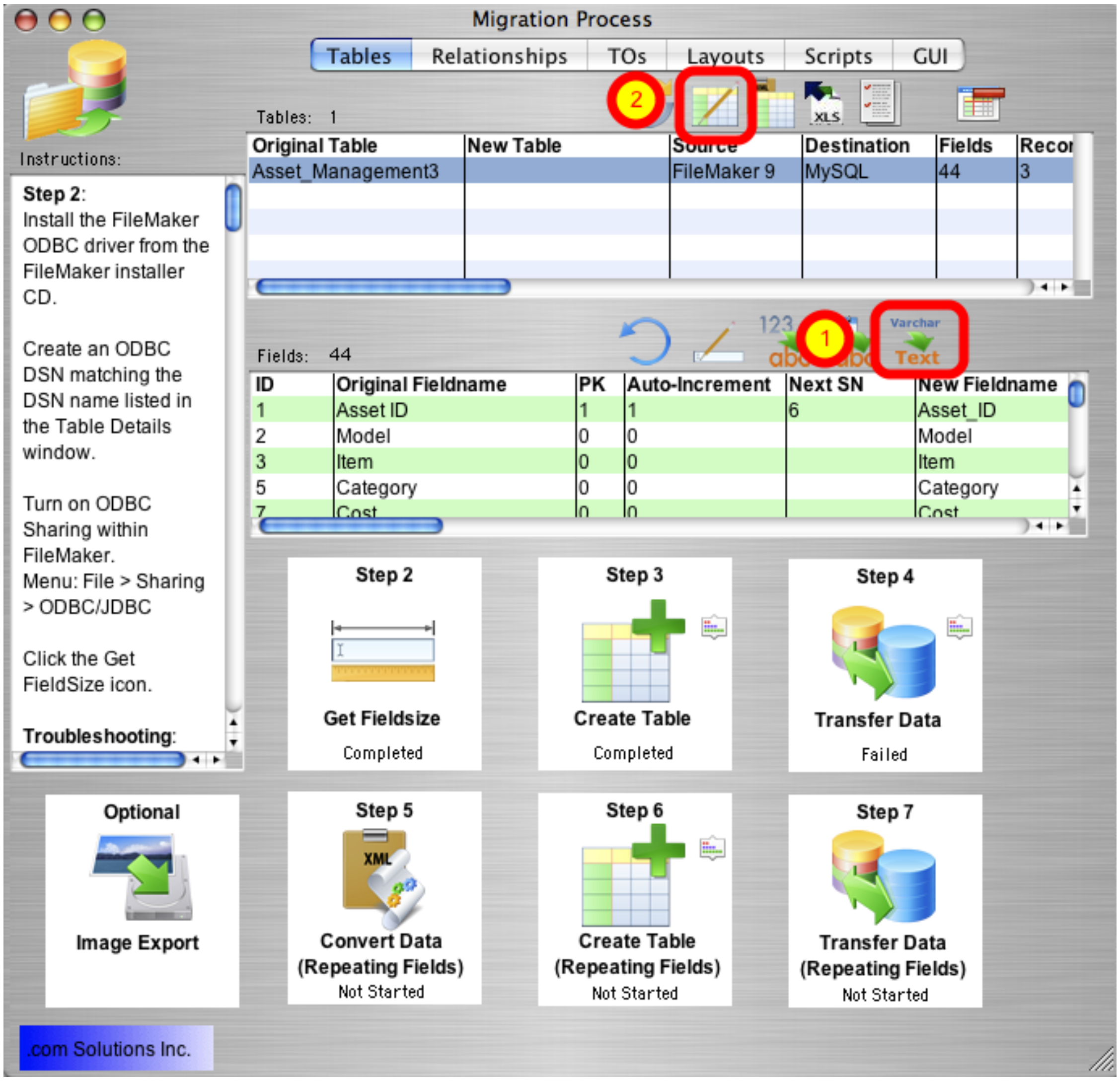
Note: You must generate table creation SQL code by clicking the Step 3 Create Table button at least once, before clicking the lock icon on the Table Details window.

Step 4 - Transfer Data - Troubleshooting - Non-Date/Time Data Type Error

If data transfer errors occur as a result of having non-date/time data within Date/Time fields, clicking the Set Date/Time Fields to Varchar button will change these column types to varchar columns in the destination database.

Open the Table Details window, click the Lock icon, then hold down the Shift key while clicking the Step 3 Create Table button.

Step 4 - Transfer Data - Varchar to TEXT Conversion



Tables: 1

Original Table	New Table	Source	Destination	Fields	Recor
Asset_Management3		FileMaker 9	MySQL	44	3

Fields: 44

ID	Original Fieldname	PK	Auto-Increment	Next SN	New Fieldname
1	Asset ID	1	1	6	Asset_ID
2	Model	0	0		Model
3	Item	0	0		Item
5	Category	0	0		Category
7	Cost	0	0		Cost

Instructions:

Step 2:
Install the FileMaker ODBC driver from the FileMaker installer CD.

Create an ODBC DSN matching the DSN name listed in the Table Details window.

Turn on ODBC Sharing within FileMaker.
Menu: File > Sharing > ODBC/JDBC

Click the Get FieldSize icon.

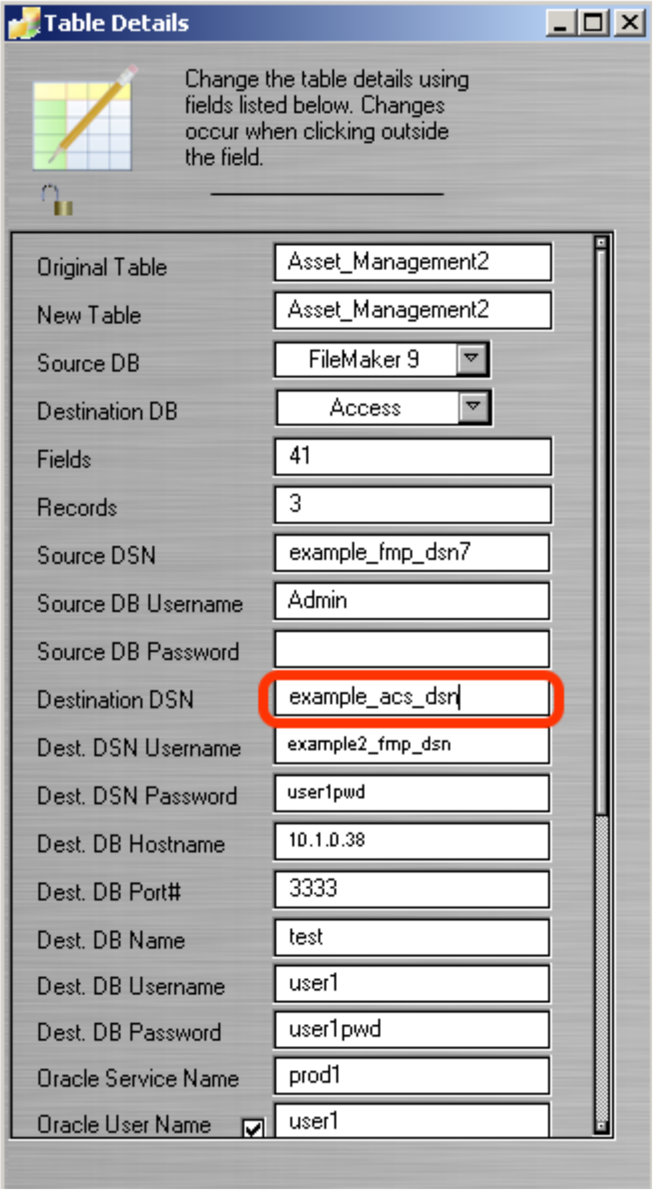
Troubleshooting:

Step 2 – Get Fieldsize – Completed
Step 3 – Create Table – Completed
Step 4 – Transfer Data – Failed
Optional – Image Export
Step 5 – Convert Data (Repeating Fields) – Not Started
Step 6 – Create Table (Repeating Fields) – Not Started
Step 7 – Transfer Data (Repeating Fields) – Not Started

It is not usually necessary to change Varchar columns to TEXT column types in the destination database. But occasionally you might need to perform this task. If you wanted to perform a quick migration, without running the Get Fieldsize step, you could just convert all of the Varchar columns to TEXT column types without worrying about the exact size of the data being transferred. FmPro Migrator will utilize the correct column type in the destination database for storing the largest amount of text.

After clicking the Varchar to TEXT button, open the Table Details window, click the Lock icon, then hold down the Shift key while clicking the Step 3 Create Table button. This process will drop and re-create the table in the destination database, creating the new table with the requested column type changes.

Step 4 - Transfer Data - Troubleshooting



Change the table details using fields listed below. Changes occur when clicking outside the field.

Original Table	Asset_Management2
New Table	Asset_Management2
Source DB	FileMaker 9
Destination DB	Access
Fields	41
Records	3
Source DSN	example_fmp_dsn7
Source DB Username	Admin
Source DB Password	
Destination DSN	example_acs_dsn
Dest. DSN Username	example2_fmp_dsn
Dest. DSN Password	user1pwd
Dest. DB Hostname	10.1.0.38
Dest. DB Port#	3333
Dest. DB Name	test
Dest. DB Username	user1
Dest. DB Password	user1pwd
Oracle Service Name	prod1
Oracle User Name ☑	user1

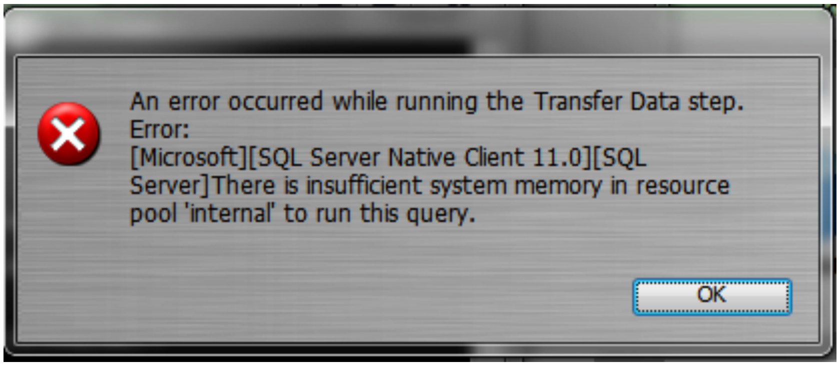
Error Message: Count Error (1).

If an unexpected error occurs during the data transfer process, there is an alternate data transfer process built into FmPro Migrator which can be used. Hold down the shift key when clicking on the Transfer Data button. The alternate data transfer method will open a separate console window and will run a compiled Perl program to transfer the data between databases.

If transferring data to an Access database, create an ODBC DSN to the destination database, as shown in the Table Details window. Open the Table Details window by double-clicking on the table in the list of tables.

Note: For some databases like SQL Server and Microsoft Access, this alternate data transfer method is always used, so you don't need to hold down the shift key.

Step 4 - Transfer Data Troubleshooting - SQL Server 2008 Memory Error



An error occurred while running the Transfer Data step.
Error:
[Microsoft][SQL Server Native Client 11.0][SQL Server]There is insufficient system memory in resource pool 'internal' to run this query.

SQL Server 2008 may fail to insert data which will display this error message in FmPro Migrator.

Troubleshooting Tip #1:
Install the latest SQL Server Service Pack (at least SP3).

Troubleshooting Tip #2:
Try using the following commands to free up memory:

```
DBCC FREESYSTEMCACHE ('ALL');
DBCC DROPCLEANBUFFERS;
DBCC FREEPROCCACHE;
DBCC FREESESSIONCACHE;
```

Troubleshooting Tip #3:
Upgrade to SQL Server 2012 or higher. Database tables which failed to transfer to SQL Server 2008 have successfully worked with SQL Server 2012+ versions.

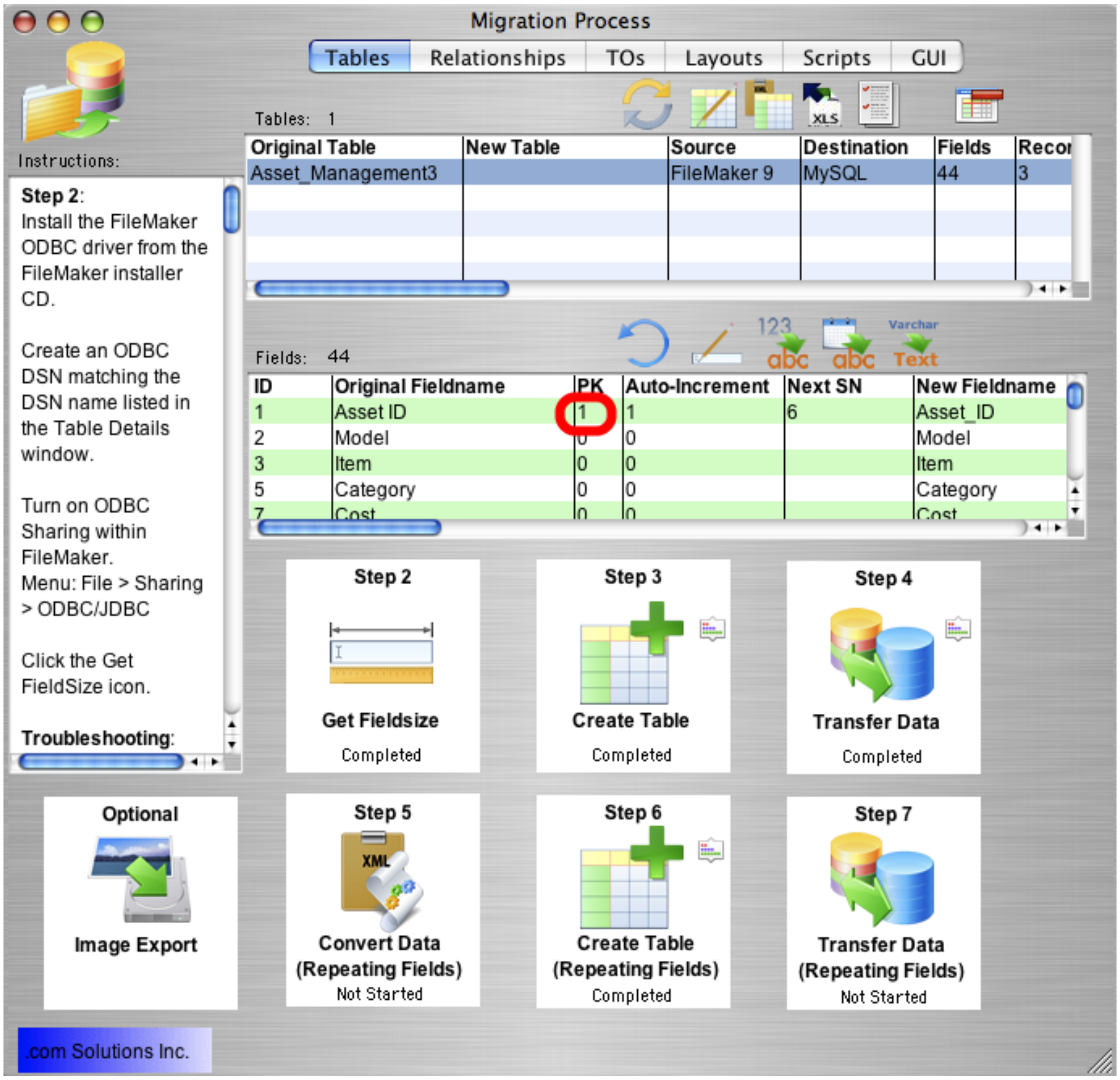
Step 5

Step 5 - Convert Data (Repeating Fields)

The buttons for Steps 5, 6 and 7 will only be visible if the source FileMaker Pro database table contains repeating fields.

FmPro Migrator converts non-relational repeating fields data into child records referencing the parent table record using the primary key of the parent table. The child table is created in the destination database using the naming convention *parent_table_repeating.* Each individual repeating fields data value is written into a separate record within this table.

Repeating Fields Processing - Preparation Steps



Instructions:

Step 2:
Install the FileMaker ODBC driver from the FileMaker installer CD.

Create an ODBC DSN matching the DSN name listed in the Table Details window.

Turn on ODBC Sharing within FileMaker.
Menu: File > Sharing > ODBC/JDBC

Click the Get FieldSize icon.

Troubleshooting:

Tables: 1

Original Table	New Table	Source	Destination	Fields	Recor
Asset_Management3		FileMaker 9	MySQL	44	3

Fields: 44

ID	Original Fieldname	PK	Auto-Increment	Next SN	New Fieldname
1	Asset ID	1	1	6	Asset_ID
2	Model	0	0		Model
3	Item	0	0		Item
5	Category	0	0		Category
7	Cost	0	0		Cost

In order to accurately transfer repeating fields data, FmPro Migrator needs to determine the Primary Key field within the source FileMaker database table. FmPro Migrator identifies the Primary Key field as being a field in which Unique and Not Empty data validation is configured. The Primary Key field contains a "1" in the PK column as shown in the field list.

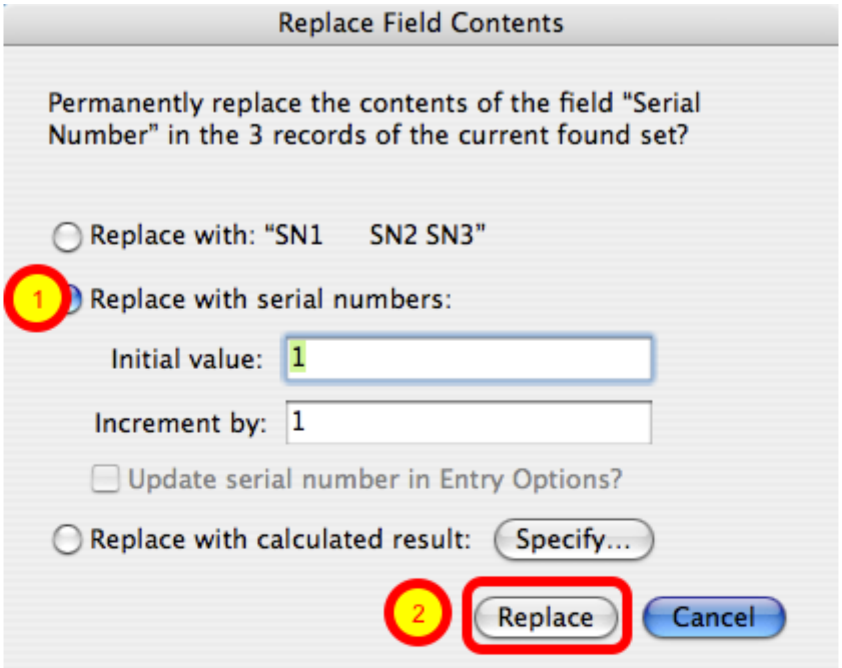
If the Primary Key field contains empty or non-unique values, then the data needs to be corrected, or a new Primary Key field needs to be defined. Only one field should be defined as the Primary Key, and this field should be defined as a numeric field, as SQL databases don't generally handle Auto-Incrementing of Text and Numeric values within the same field (as FileMaker easily does).

To create a new Primary Key field within the FileMaker database, create a numeric field having Unique and Not Empty field validation. Disable the Unique and Not Empty validation for the original Primary Key field.

Select the Records > Show All Records menu item, to make sure that all records will be affected by the Replace Field Contents feature.

After creating the new field, put the cursor into this new field, then select the Records > Replace Field Contents menu item.

Repeating Fields Processing - Creating New Primary Key



Replace Field Contents

Permanently replace the contents of the field "Serial Number" in the 3 records of the current found set?

○ Replace with: "SN1   SN2 SN3"

● Replace with serial numbers:

Initial value: 1

Increment by: 1

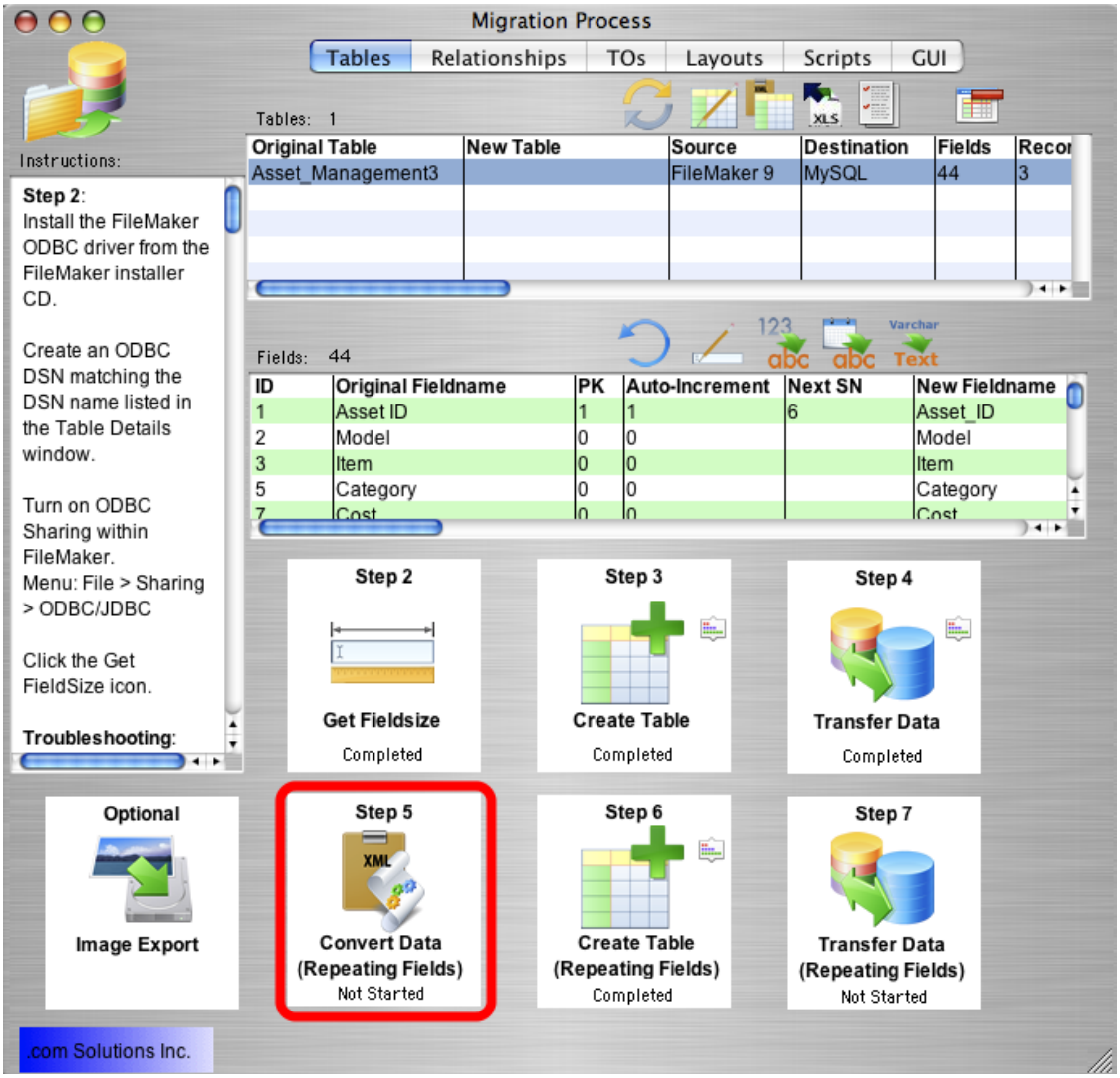
□ Update serial number in Entry Options?

○ Replace with calculated result: Specify...

Using Replace Field Contents, replaces the empty values within the new Primary Key field with an incrementing series of numbers, within all records of the source table.

After adding this new Primary Key field and replacing field contents, you will need to start the migration process over starting with Step 1, since the structure of the source table has changed and FmPro Migrator needs to be aware of any changes which have been made to the source table.

Step 5 - Convert Data (Repeating Fields) - FileMaker 7+ Only



Tables: 1

Original Table	New Table	Source	Destination	Fields	Recor
Asset_Management3		FileMaker 9	MySQL	44	3

Step 2:
Install the FileMaker ODBC driver from the FileMaker installer CD.

Create an ODBC DSN matching the DSN name listed in the Table Details window.

Turn on ODBC Sharing within FileMaker.
Menu: File > Sharing > ODBC/JDBC

Click the Get FieldSize icon.

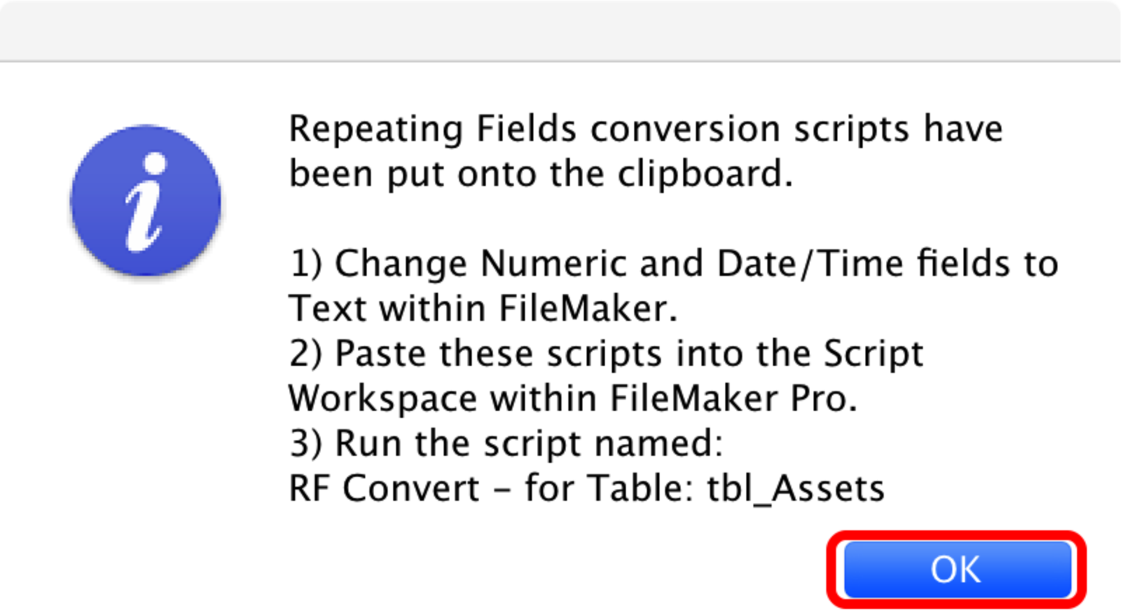
Troubleshooting:

Fields: 44

ID	Original Fieldname	PK	Auto-Increment	Next SN	New Fieldname
1	Asset ID	1	1	6	Asset_ID
2	Model	0	0		Model
3	Item	0	0		Item
5	Category	0	0		Category
7	Cost	0	0		Cost

For FileMaker 7+ source database tables, Step 5 processing is designed to change the contents of repeating fields data so that it can be transferred thru the FileMaker 7+ ODBC driver. FileMaker 7+ ODBC drivers are no longer capable of transferring data from repeating fields within the FileMaker database. To work-around this ODBC Driver limitation, FmPro Migrator moves all of the repeating fields data into the first repeating data value, and separates each repeat value with a TAB data separator.

When the Step 5 Convert Data (Repeating Fields) button is clicked, FmPro Migrator generates FileMaker Script Workspace scripts to convert the data within each repeating field data and puts this script onto the ClipBoard.



Repeating Fields conversion scripts have been put onto the clipboard.

1) Change Numeric and Date/Time fields to Text within FileMaker.
2) Paste these scripts into the Script Workspace within FileMaker Pro.
3) Run the script named:
RF Convert - for Table: tbl_Assets

Click the Ok button to the informational dialog.

Open the FileMaker Script Workspace, and paste the script from the ClipBoard into FileMaker.

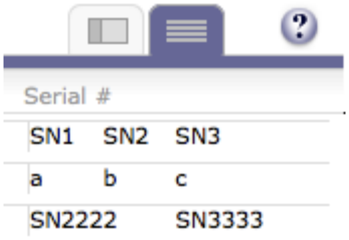
Manually change each of the repeating fields from Numeric or Date/Time format to Text within FileMaker. It is not necessary to re-gather info from FileMaker after making this change.

Run the script shown in the dialog (the one you just pasted into the Script Workspace).

Warning: Running the RF Convert script for a particular table is a task which should be performed **one-time only**. It should not be run more than once. To verify whether the RF Convert script has previously been run on a table, check to see if all of the repeating field data values have been moved to the first repeating field occurrence within the database.

If FileMaker gets interrupted while running the script, it's generally not possible to start the process over again with the remaining records. For instance, the script could stop if a data validation error occurs with one of the other fields of the database table. If this type of problem occurs, then it will be necessary to get a copy of the database file, copy and run this script again after solving the problem.

Step 5 - Convert Data (Repeating Fields) - Verifying Data Conversion



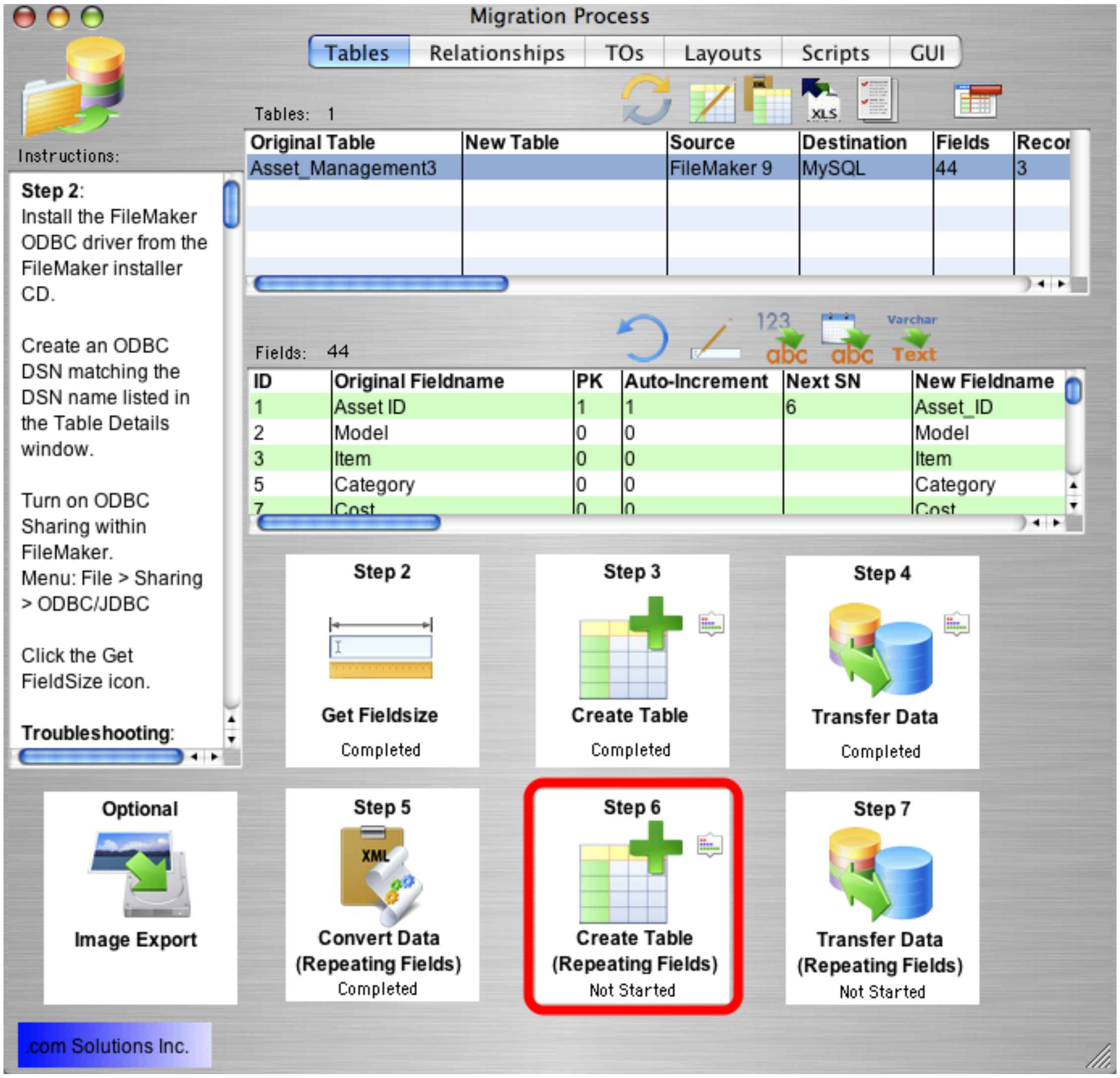
```
Serial #
SN1   SN2   SN3
a     b     c
SN2222      SN3333
```

This image shows the contents of a repeating field shown on the FileMaker layout as "Serial #", which has been successfully converted with the RF Convert ScriptMaker script. If the RF convert script had not been run, then there would be vertical lines separating each repeating value. But since all of the repeating field data values have been placed into the first repeat occurrence, these vertical lines are not visible.

FmPro Migrator sets the status of the Step 5 Convert Data (Repeating Fields) step to "In Progress" when you click the button to generate and put the script onto the ClipBoard. FmPro Migrator doesn't know when you have actually run the RF Convert script within the Script Workspace, so you should click the pop-up menu and set the status of this step to "Completed" once you have completed this task. Keeping the status selections updated can be helpful if you get interrupted while performing the migration, so that you will know which steps you have completed for each table to be migrated.

Step 6

Step 6 - Create Table (Repeating Fields)



Step 2:
Install the FileMaker ODBC driver from the FileMaker installer CD.

Create an ODBC DSN matching the DSN name listed in the Table Details window.

Turn on ODBC Sharing within FileMaker.
Menu: File > Sharing > ODBC/JDBC

Click the Get FieldSize icon.

Troubleshooting:

Tables: 1

Original Table	New Table	Source	Destination	Fields	Recor
Asset_Management3		FileMaker 9	MySQL	44	3

Fields: 44

ID	Original Fieldname	PK	Auto-Increment	Next SN	New Fieldname
1	Asset ID	1	1	6	Asset_ID
2	Model	0	0		Model
3	Item	0	0		Item
5	Category	0	0		Category
7	Cost	0	0		Cost

Click the Step 6 Create Table (Repeating Fields) button to create the repeating fields table in the destination database.

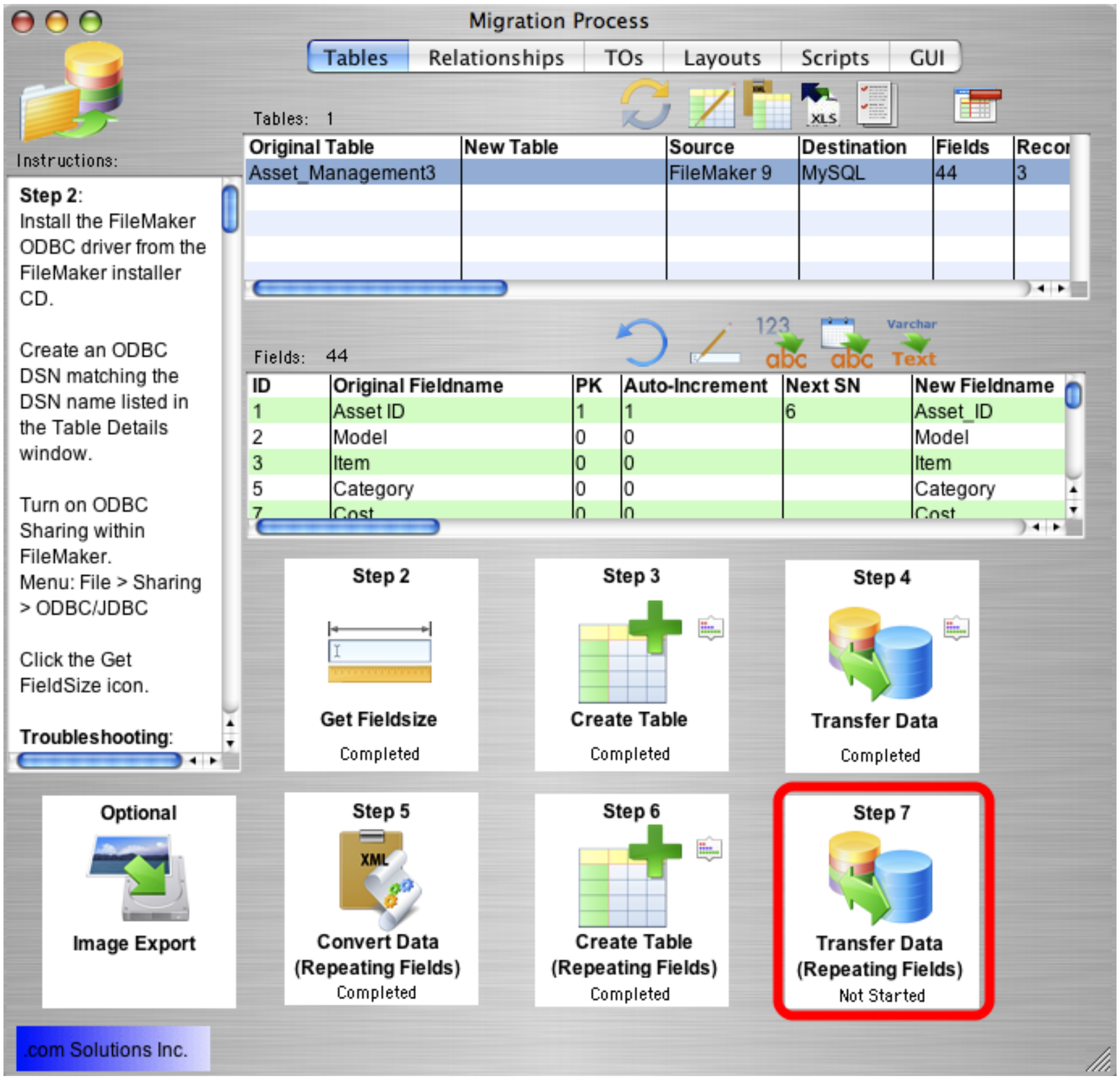
FmPro Migrator will generate the table creation SQL code, connect to the destination database and create the table. Once the repeating fields table has been successfully created, the status menu under the Step 6 Create Table (Repeating Fields) button will be changed to "Completed".

Holding down the shift key while clicking the Step 6 Create Table (Repeating Fields) button drops and re-creates the table in the destination database.

Warning: Dropping the table in the destination database causes the loss of all data within the table.

Step 7

Step 7 - Transfer Data (Repeating Fields)



Tables: 1

Original Table	New Table	Source	Destination	Fields	Recor
Asset_Management3		FileMaker 9	MySQL	44	3

Fields: 44

ID	Original Fieldname	PK	Auto-Increment	Next SN	New Fieldname
1	Asset ID	1	1	6	Asset_ID
2	Model	0	0		Model
3	Item	0	0		Item
5	Category	0	0		Category
7	Cost	0	0		Cost

Instructions:

Step 2:
Install the FileMaker ODBC driver from the FileMaker installer CD.

Create an ODBC DSN matching the DSN name listed in the Table Details window.

Turn on ODBC Sharing within FileMaker.
Menu: File > Sharing > ODBC/JDBC

Click the Get FieldSize icon.

Troubleshooting:

Click the Step 7 Transfer Data (Repeating Fields) button to transfer data from the source table in the FileMaker database to the newly created repeating fields table in the destination database. Once the data has been transferred successfully, the status menu below the Step 7 Transfer Data (Repeating Fields) button will change from "Not Started" to "Completed".

Step 7 - Transfer Data (Repeating Fields) - Troubleshooting

All of the same troubleshooting steps apply to transferring repeating fields data as were mentioned within the Step 4 instructions. But one additional consideration involves reviewing the data transferred into the repeating fields.

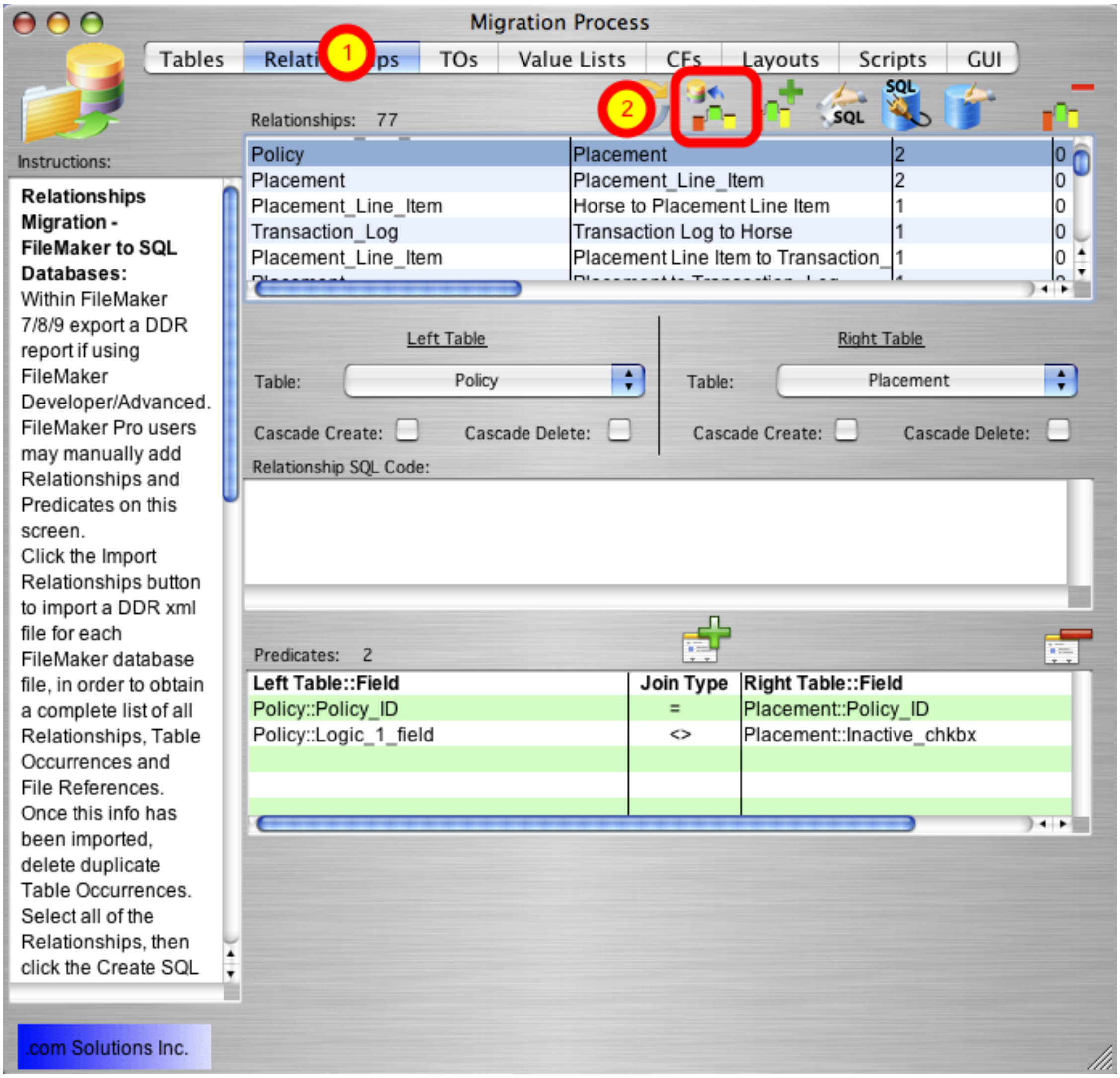
Reviewing the data validates whether the each of the steps were processed correctly. For instance, if only 1 repeating value was transferred to the destination table, then this could mean that the RF Convert script wasn't run or the contents of the source table fields weren't converted into Text prior to processing. If either of these problems occur, it is a simple matter to go back and re-do those processing steps, drop & re-create the table in the destination database then press the Step 7 button to transfer the data again.

Migrating Relationships

Migrating Relationships

FmPro Migrator reads FileMaker Pro relationship info from DDR XML files exported by FileMaker Pro. This relationship info is then used to create SQL code to re-create the relationships within the destination database.

Migrating Relationships - Import DDR XML File



Relationships: 77

Policy	Placement	2	0
Placement	Placement_Line_Item	2	0
Placement_Line_Item	Horse to Placement Line Item	1	0
Transaction_Log	Transaction Log to Horse	1	0
Placement_Line_Item	Placement Line Item to Transaction_	1	0

Instructions:

Relationships Migration - FileMaker to SQL Databases: Within FileMaker 7/8/9 export a DDR report if using FileMaker Developer/Advanced. FileMaker Pro users may manually add Relationships and Predicates on this screen. Click the Import Relationships button to import a DDR xml file for each FileMaker database file, in order to obtain a complete list of all Relationships, Table Occurrences and File References. Once this info has been imported, delete duplicate Table Occurrences. Select all of the Relationships, then click the Create SQL

Left Table

Table: Policy

Cascade Create: □ Cascade Delete: □

Right Table

Table: Placement

Cascade Create: □ Cascade Delete: □

Relationship SQL Code:

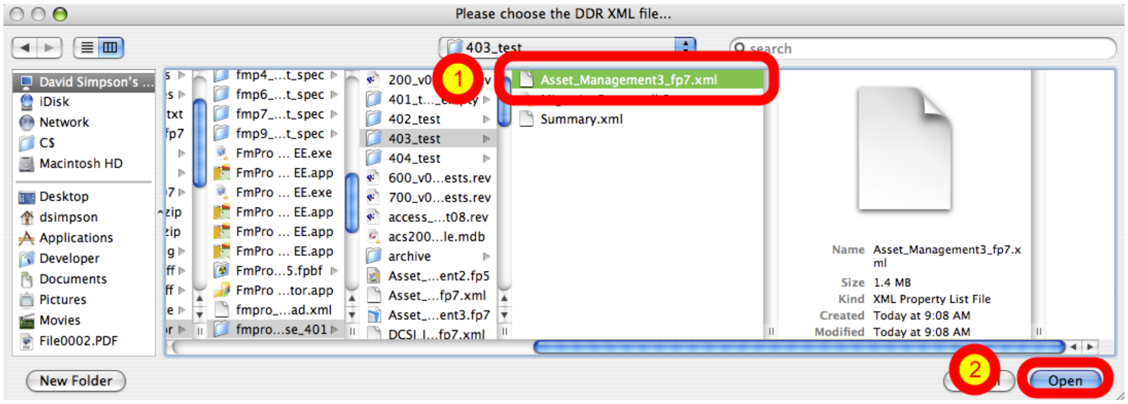
Predicates: 2

Left Table::Field	Join Type	Right Table::Field
Policy::Policy_ID	=	Placement::Policy_ID
Policy::Logic_1_field	<>	Placement::Inactive_chkbx

Within FmPro Migrator, (1) click on the Relationships tab, then (2) click on the Import Relationships button.

Note: Relationships will only be imported correctly if all of the base tables have already been created within the Tables tab.

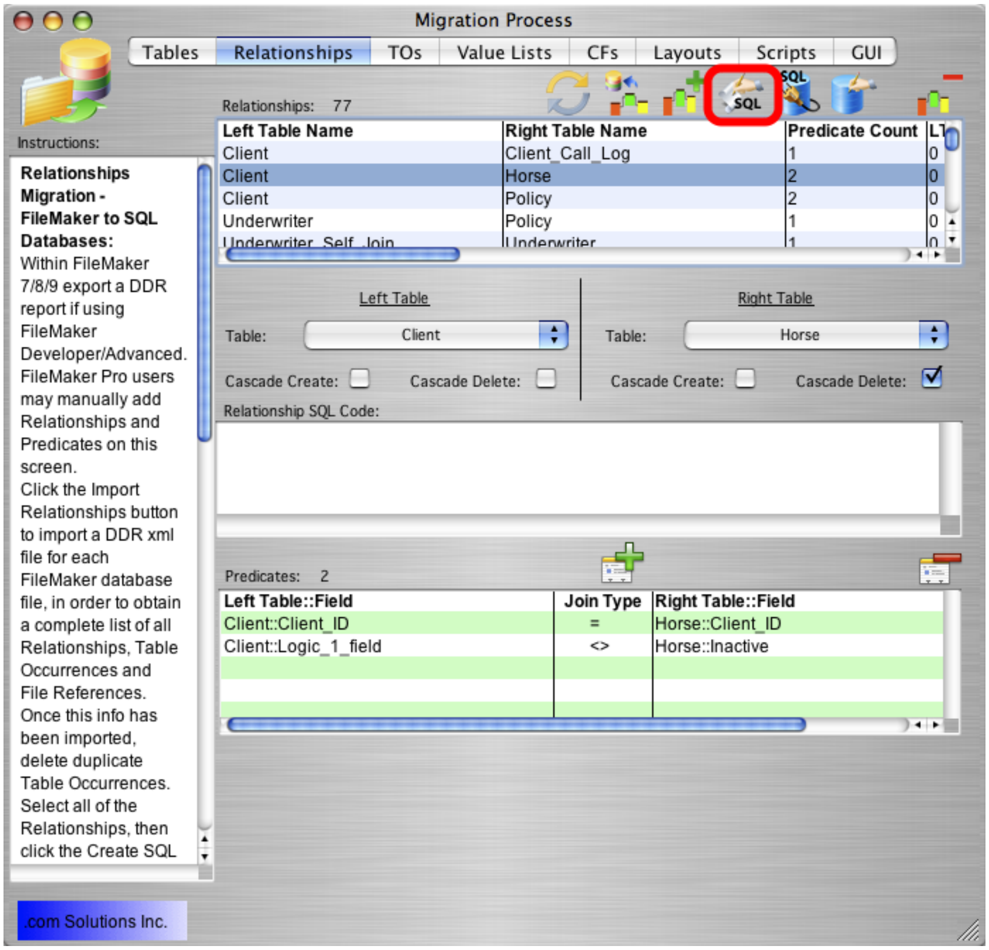
Migrating Relationships - Import DDR XML File - Select File


Please choose the DDR XML file...

Name Asset_Management3_fp7.xml
Size 1.4 MB
Kind XML Property List File
Created Today at 9:08 AM
Modified Today at 9:08 AM

(1) Select the exported DDR XML file, then (2) click the Open button.
Don't select the Summary.xml file.

Migrating Relationships - Generate Relationship SQL Code


Migration Process

Relationships: 77

Left Table Name	Right Table Name	Predicate Count	L1
Client	Client_Call_Log	1	0
Client	Horse	2	0
Client	Policy	2	0
Underwriter	Policy	1	0
Underwriter_Self_Join	Underwriter	1	0

Instructions:

Relationships Migration - FileMaker to SQL Databases: Within FileMaker 7/8/9 export a DDR report if using FileMaker Developer/Advanced. FileMaker Pro users may manually add Relationships and Predicates on this screen. Click the Import Relationships button to import a DDR xml file for each FileMaker database file, in order to obtain a complete list of all Relationships, Table Occurrences and File References. Once this info has been imported, delete duplicate Table Occurrences. Select all of the Relationships, then click the Create SQL

Left Table — Table: Client — Cascade Create: □ Cascade Delete: □

Right Table — Table: Horse — Cascade Create: □ Cascade Delete: ☑

Relationship SQL Code:

Predicates: 2

Left Table::Field	Join Type	Right Table::Field
Client::Client_ID	=	Horse::Client_ID
Client::Logic_1_field	<>	Horse::Inactive

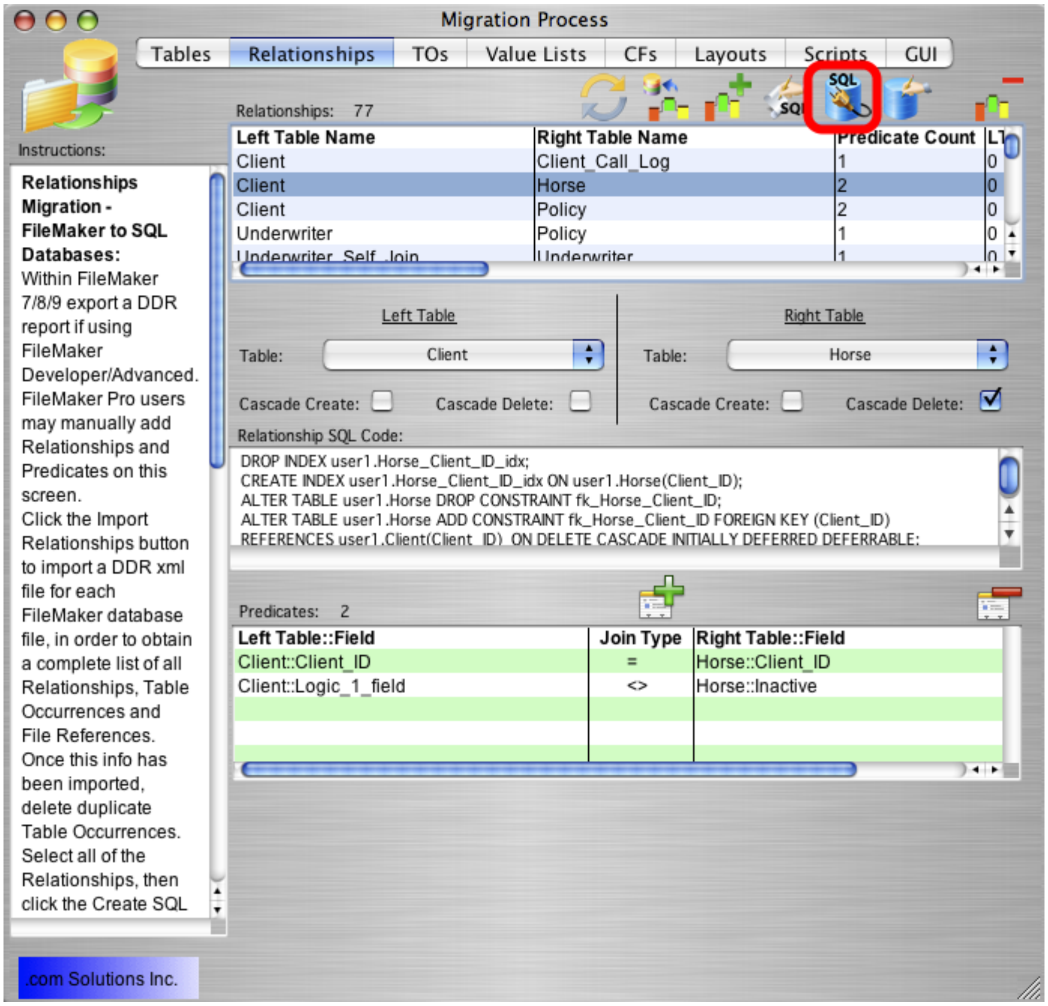
Select one ore more relationships from the list, click on the Generate Relationship SQL button.

If the relationship cannot be generated from the FileMaker Pro relationship, error text will be displayed within the Relationship SQL Code field. Otherwise, the generated SQL code will be displayed in the Relationship SQL Code field.

The most common reasons relationship code won't be generated are:
1) Having a missing primary key in the table.
2) The TO name specified in the relationship does not represent a base table. Relationship SQL code can only be generated for base tables, since these are the only tables which will get created in the destination database.

Any base tables which don't contain a primary key, will be listed in a error dialog when the Generate Relationship SQL button is clicked. This is just an informative dialog to let you know that relationships won't be generated for the tables which don't have a primary key. Relationship SQL code will still be generated for all of the remaining selected tables.

Migrating Relationships - Create Relationships in Destination Database



Relationships: 77

Left Table Name	Right Table Name	Predicate Count	L1
Client	Client_Call_Log	1	0
Client	Horse	2	0
Client	Policy	2	0
Underwriter	Policy	1	0
Underwriter_Self_Join	Underwriter	1	0

Left Table

Table: Client

Cascade Create: □ Cascade Delete: □

Right Table

Table: Horse

Cascade Create: □ Cascade Delete: ☑

Relationship SQL Code:

```
DROP INDEX user1.Horse_Client_ID_idx;
CREATE INDEX user1.Horse_Client_ID_idx ON user1.Horse(Client_ID);
ALTER TABLE user1.Horse DROP CONSTRAINT fk_Horse_Client_ID;
ALTER TABLE user1.Horse ADD CONSTRAINT fk_Horse_Client_ID FOREIGN KEY (Client_ID)
REFERENCES user1.Client(Client_ID) ON DELETE CASCADE INITIALLY DEFERRED DEFERRABLE;
```

Predicates: 2

Left Table::Field	Join Type	Right Table::Field
Client::Client_ID	=	Horse::Client_ID
Client::Logic_1_field	<>	Horse::Inactive

Relationships Migration - FileMaker to SQL Databases: Within FileMaker 7/8/9 export a DDR report if using FileMaker Developer/Advanced. FileMaker Pro users may manually add Relationships and Predicates on this screen. Click the Import Relationships button to import a DDR xml file for each FileMaker database file, in order to obtain a complete list of all Relationships, Table Occurrences and File References. Once this info has been imported, delete duplicate Table Occurrences. Select all of the Relationships, then click the Create SQL

Click the Transfer Relationships button to create the selected relationships in the destination database.

<u>Note</u>: Most SQL databases don't allow the creation of relationships with the same flexibility as FileMaker Pro. Some databases don't allow joining different types of data together, and FileMaker also allows entire lists of values to be matched more easily. If the relationships cannot be created using this automated method within FmPro Migrator, then the relationships may need to be created manually in the destination database. Also, you don't always require creating a relationship in all situations if the relationship is managed via software application code.

In some situations, it might be better to re-design the relationships within the FileMaker database before performing the final conversion. FmPro Migrator makes it easy to iterate thru multiple options to see which one works best for your database conversion project.

Image Export

Image Transfer (Optional Step)

Images (and other types of data within FileMaker 7+) can be transferred directly into supported SQL databases [but not Microsoft Access]. In order to perform this type of export, you need to have at least one container field within the FileMaker Pro database table and a primary key within the table.

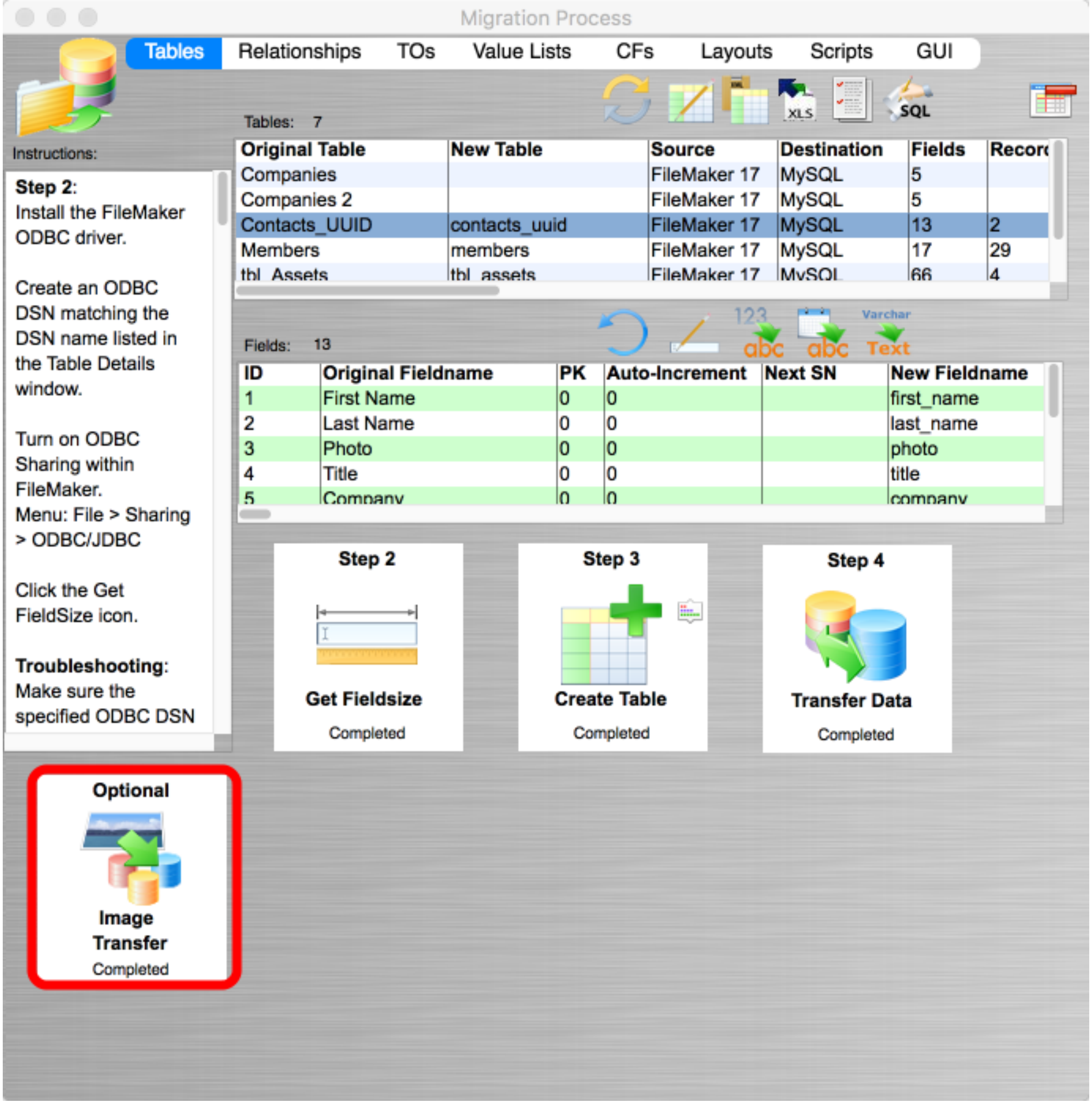
Image Transfer - Prerequisites

Binary files of all types are supported when the FileMaker Insert File... feature has been used to insert the file. This feature would commonly be used for storing spreadsheets and other files.

The database table must have a primary key (including numeric and non-numeric primary keys consisting of UUIDs).

FmPro Migrator is able to transfer FileMaker container field data from embedded container fields.
The transfer of the following container field data is not supported:
External Container Fields
Files stored as a Reference
Very old unsupported embedded data types including: including: XMLO - FileMaker layout, JP2 - JPEG 2000, Quicktime, PICT, MacPaint, FPX - FlashPix, .SGI, TGA - Targa

Image Transfer



Tables: 7

Original Table	New Table	Source	Destination	Fields	Recor
Companies		FileMaker 17	MySQL	5	
Companies 2		FileMaker 17	MySQL	5	
Contacts_UUID	contacts_uuid	FileMaker 17	MySQL	13	2
Members	members	FileMaker 17	MySQL	17	29
tbl_Assets	tbl_assets	FileMaker 17	MySQL	66	4

Fields: 13

ID	Original Fieldname	PK	Auto-Increment	Next SN	New Fieldname
1	First Name	0	0		first_name
2	Last Name	0	0		last_name
3	Photo	0	0		photo
4	Title	0	0		title
5	Company	0	0		company

Instructions:

Step 2:
Install the FileMaker ODBC driver.

Create an ODBC DSN matching the DSN name listed in the Table Details window.

Turn on ODBC Sharing within FileMaker.
Menu: File > Sharing > ODBC/JDBC

Click the Get FieldSize icon.

Troubleshooting:
Make sure the specified ODBC DSN

Step 2
Get Fieldsize
Completed

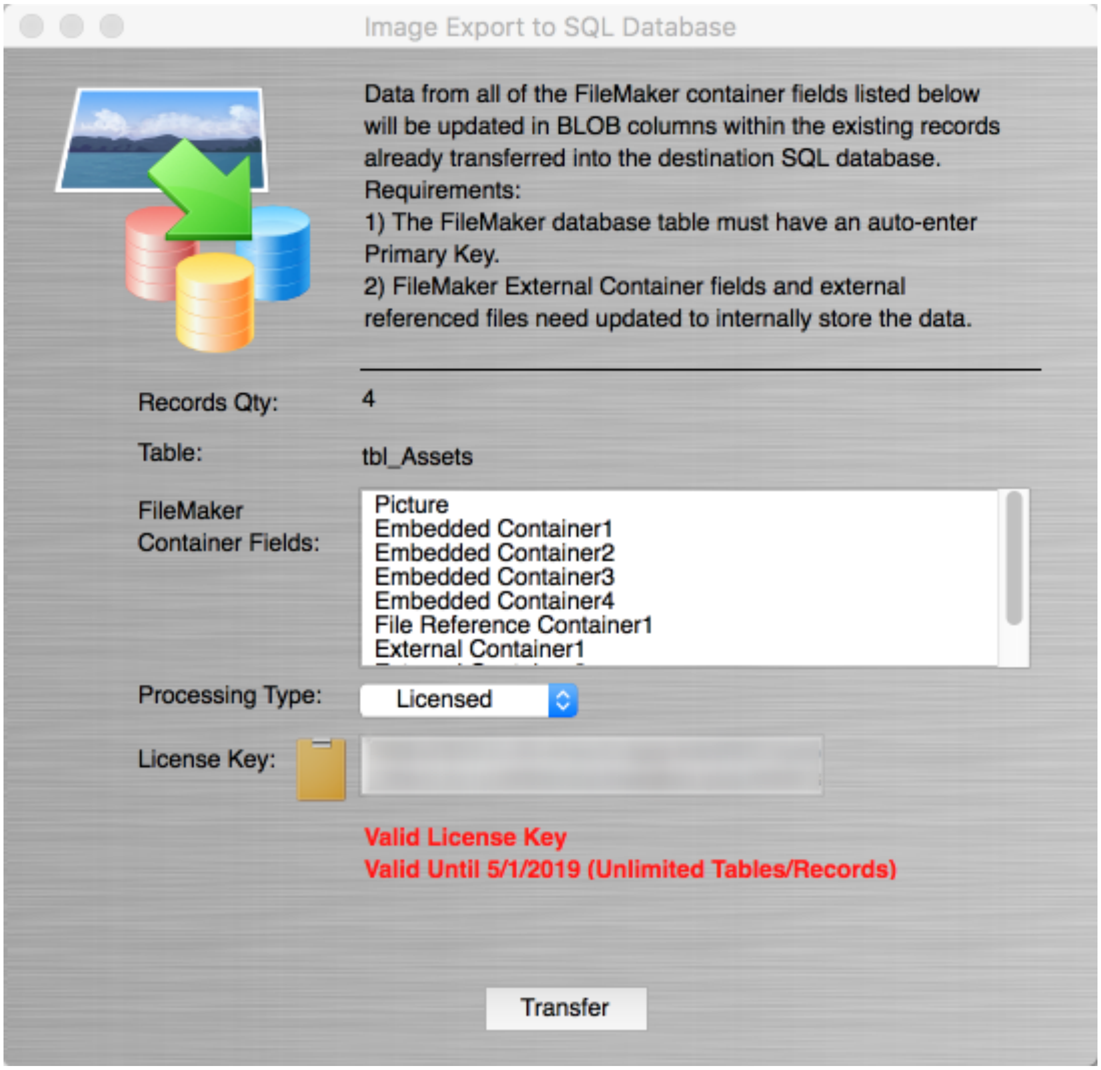
Step 3
Create Table
Completed

Step 4
Transfer Data
Completed

Optional
Image Transfer
Completed

Click on the Image Transfer button to transfer images from the FileMaker Pro database table. This button will not be displayed if there are no container fields within the table.

Image Transfer Window



Data from all of the FileMaker container fields listed below will be updated in BLOB columns within the existing records already transferred into the destination SQL database.
Requirements:
1) The FileMaker database table must have an auto-enter Primary Key.
2) FileMaker External Container fields and external referenced files need updated to internally store the data.

Records Qty: 4

Table: tbl_Assets

FileMaker Container Fields:
Picture
Embedded Container1
Embedded Container2
Embedded Container3
Embedded Container4
File Reference Container1
External Container1

Processing Type: Licensed

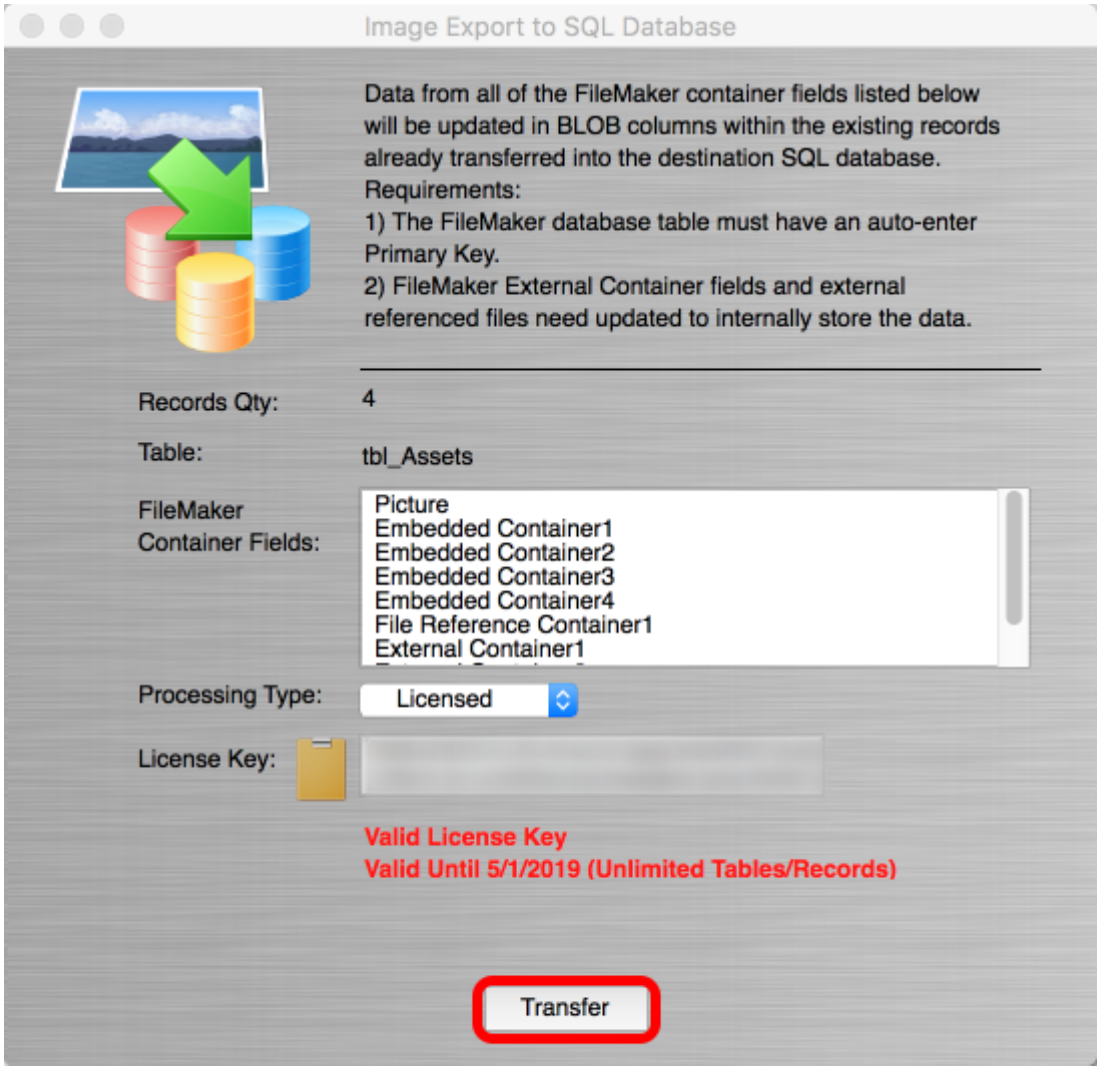
License Key:

Valid License Key
Valid Until 5/1/2019 (Unlimited Tables/Records)

If the Demo feature is used, 5 records containing container field data will be transferred.

Using the license key supplied with FmPro Migrator, all records containing container fields will be transferred to the SQL database. To use the supplied license key, select Licensed from the Processing Type menu, copy the FileMaker Container fields license key from your purchase email, then click on the clipboard icon.

Image - Transfer



Data from all of the FileMaker container fields listed below will be updated in BLOB columns within the existing records already transferred into the destination SQL database.
Requirements:
1) The FileMaker database table must have an auto-enter Primary Key.
2) FileMaker External Container fields and external referenced files need updated to internally store the data.

Records Qty: 4

Table: tbl_Assets

FileMaker Container Fields:
Picture
Embedded Container1
Embedded Container2
Embedded Container3
Embedded Container4
File Reference Container1
External Container1

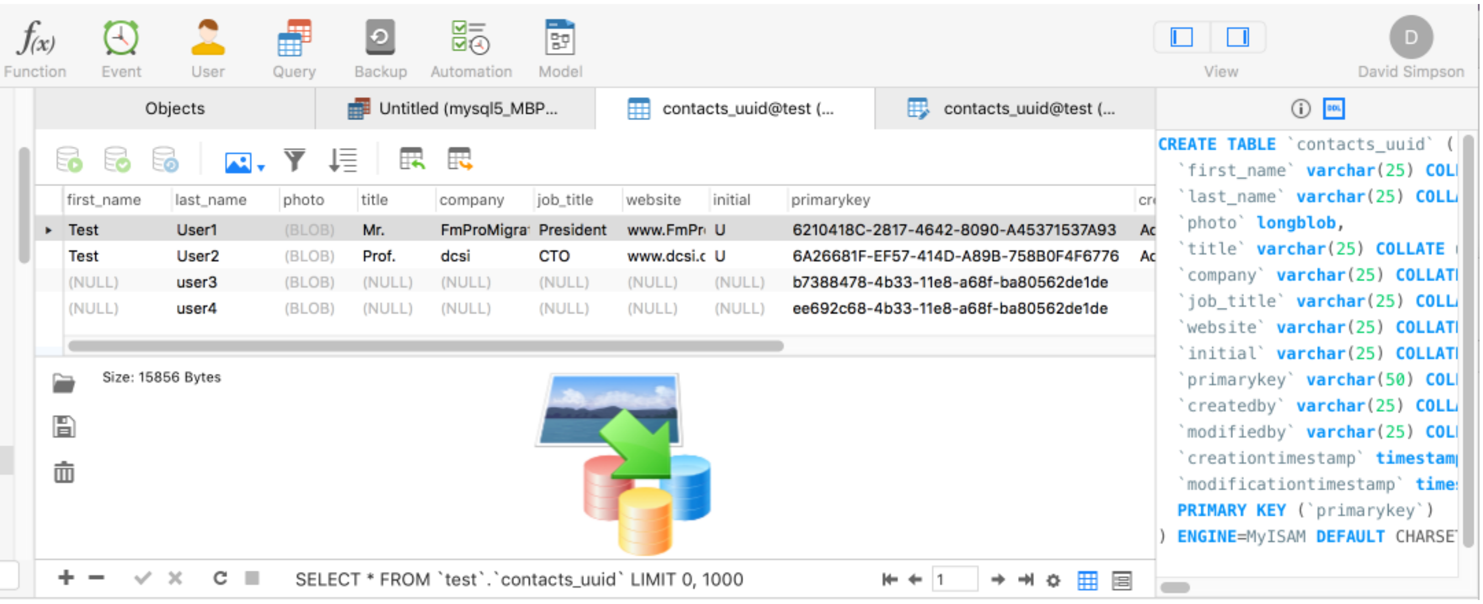
Processing Type: Licensed

License Key:

Valid License Key
Valid Until 5/1/2019 (Unlimited Tables/Records)

Click the Transfer button. FmPro Migrator will read thru all of the records and transfer container field data for all of the records into the SQL database.

Imager Transfer - and UUID PK Displayed - MySQL



first_name	last_name	photo	title	company	job_title	website	initial	primarykey	cr
Test	User1	(BLOB)	Mr.	FmProMigra	President	www.FmPr	U	6210418C-2817-4642-8090-A45371537A93	Ac
Test	User2	(BLOB)	Prof.	dcsi	CTO	www.dcsi.c	U	6A26681F-EF57-414D-A89B-758B0F4F6776	Ac
(NULL)	user3	(BLOB)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	b7388478-4b33-11e8-a68f-ba80562de1de	
(NULL)	user4	(BLOB)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	ee692c68-4b33-11e8-a68f-ba80562de1de	

```
CREATE TABLE `contacts_uuid` (
  `first_name` varchar(25) COL
  `last_name` varchar(25) COLL
  `photo` longblob,
  `title` varchar(25) COLLATE
  `company` varchar(25) COLLAT
  `job_title` varchar(25) COLL
  `website` varchar(25) COLLAT
  `initial` varchar(25) COLLAT
  `primarykey` varchar(50) COL
  `createdby` varchar(25) COLL
  `modifiedby` varchar(25) COL
  `creationtimestamp` timestam
  `modificationtimestamp` time
  PRIMARY KEY (`primarykey`)
) ENGINE=MyISAM DEFAULT CHARSE
```

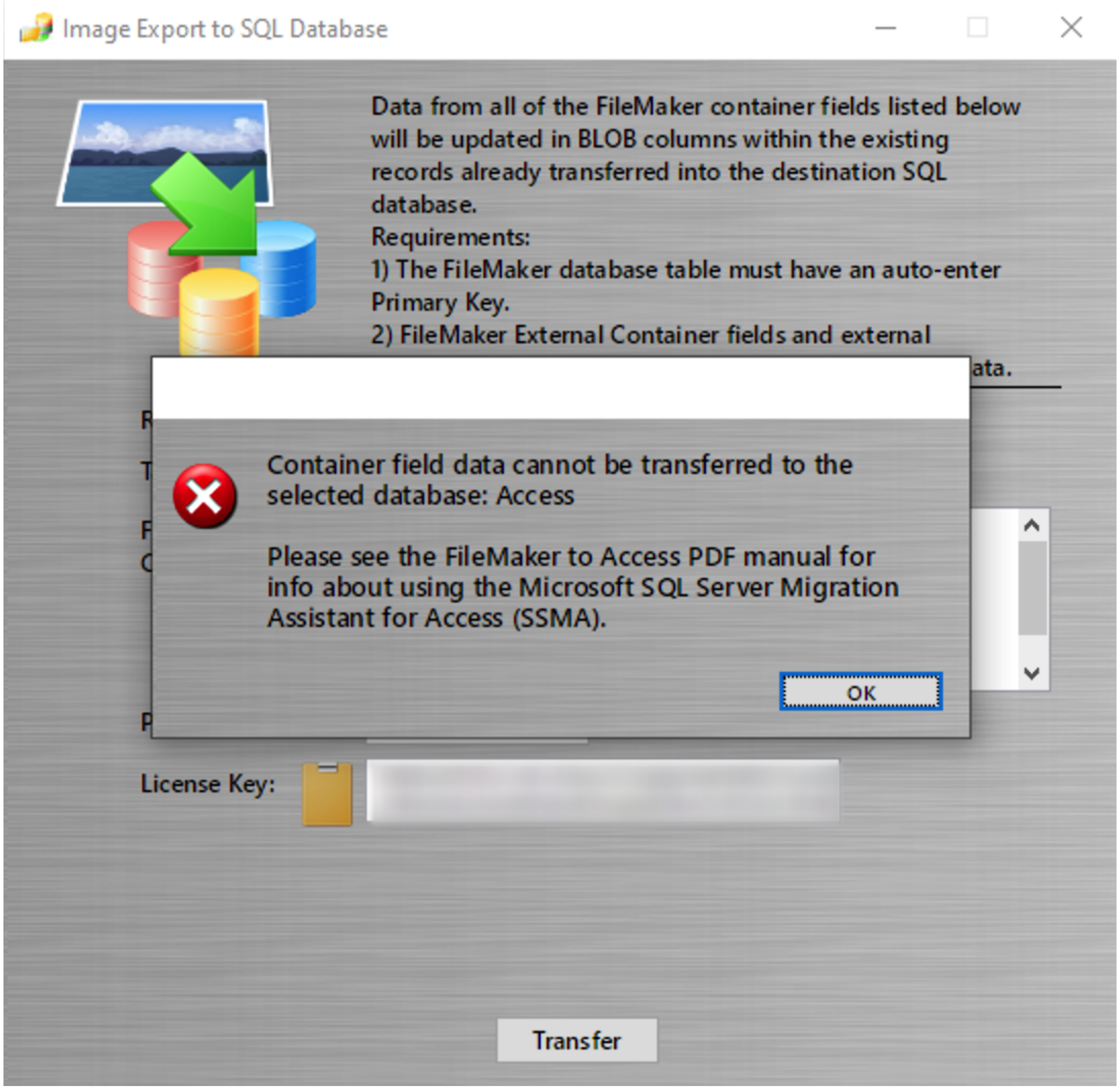
```
SELECT * FROM `test`.`contacts_uuid` LIMIT 0, 1000
```

For MySQL database servers, FileMaker fields having an Auto-Enter Get (UUID) calculation will be created with a MySQL database trigger having the same functionality. The first 2 records were created in FileMaker and transferred into MySQL. The last 2 records were created directly in MySQL. Notice that the UUID values look a little different - in MySQL only the first section of the UUID changes value.

Image Transfer to Microsoft Access

Direct transfer of images into Microsoft Access from FileMaker databases is not supported. But you can still accomplish your goal a different way.
1) You could transfer data for all fields except the container fields into Access.
2) Then use the Microsoft SQL Server Migration Assistant for Access utility (SSMA) to transfer the tables and data into SQL Server. This utility will make linked tables in Access to the tables in SQL Server.
3) Use FmPro Migrator to transfer the images into SQL Server.
4) Write some Access VBA code to work with the linked BLOB data.

Access Image Transfer Error Dialog



Data from all of the FileMaker container fields listed below will be updated in BLOB columns within the existing records already transferred into the destination SQL database.
Requirements:
1) The FileMaker database table must have an auto-enter Primary Key.
2) FileMaker External Container fields and external

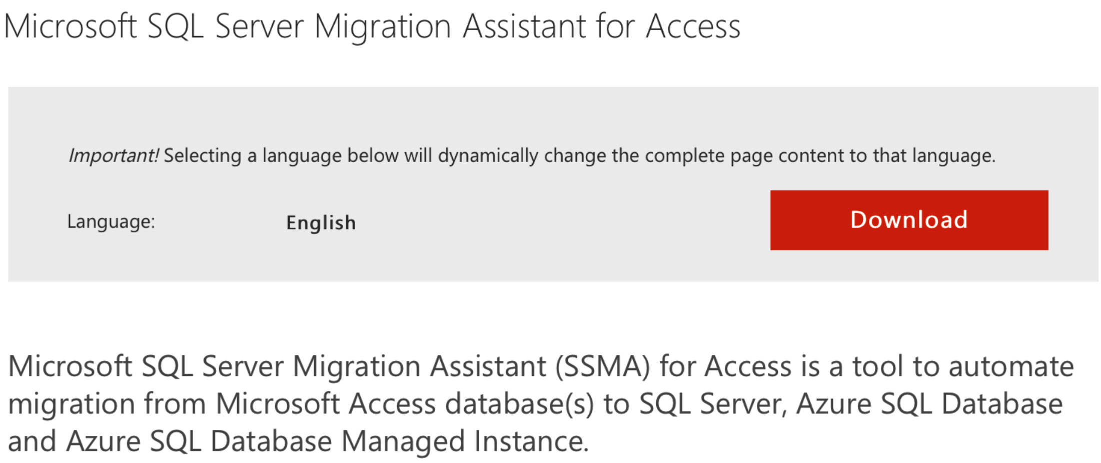
Container field data cannot be transferred to the selected database: Access

Please see the FileMaker to Access PDF manual for info about using the Microsoft SQL Server Migration Assistant for Access (SSMA).

FmPro Migrator warns that the direct transfer of images isn't supported when the Transfer button is clicked. If this dialog wasn't displayed, the data transfer would be attempted, but every record with fail with a Data type mismatch error message similar to the following text:

Record: ID 1
[Microsoft][ODBC Microsoft Access Driver] Data type mismatch in criteria expression.
UPDATE tbl_assets SET picture = :1 WHERE id = '1';

Microsoft SQL Server Migration Assistant for Access



Microsoft SQL Server Migration Assistant for Access

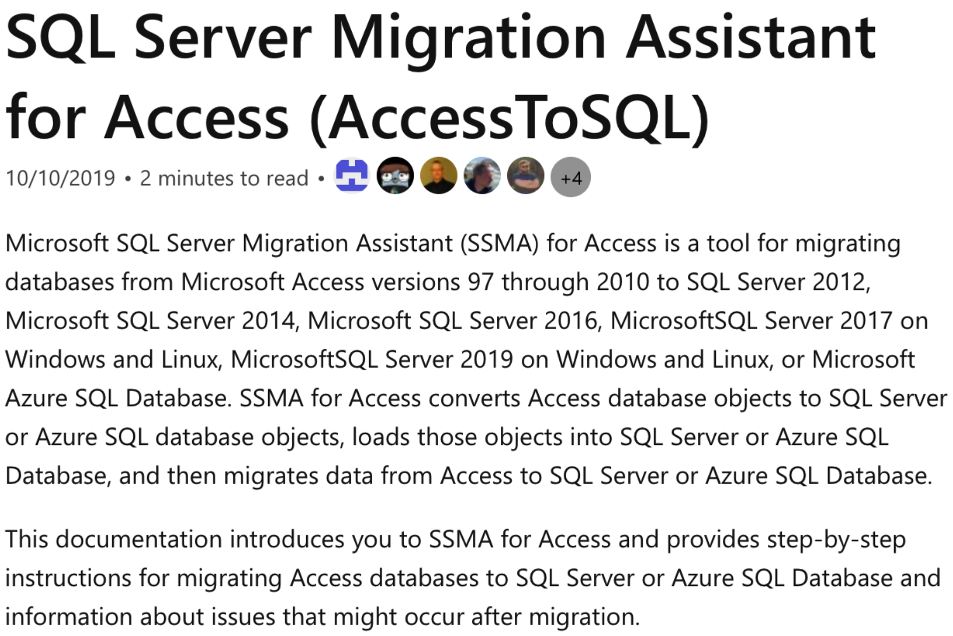
Important! Selecting a language below will dynamically change the complete page content to that language.

Language: English

Microsoft SQL Server Migration Assistant (SSMA) for Access is a tool to automate migration from Microsoft Access database(s) to SQL Server, Azure SQL Database and Azure SQL Database Managed Instance.

Microsoft recommends using SQL Server as the data storage database for multi-user databases anyhow, so this utility will also help with that process. This tool also supports Azure SQL databases in addition to SQL Server instances.

SSMA Usage Details



SQL Server Migration Assistant for Access (AccessToSQL)

10/10/2019 • 2 minutes to read •

Microsoft SQL Server Migration Assistant (SSMA) for Access is a tool for migrating databases from Microsoft Access versions 97 through 2010 to SQL Server 2012, Microsoft SQL Server 2014, Microsoft SQL Server 2016, MicrosoftSQL Server 2017 on Windows and Linux, MicrosoftSQL Server 2019 on Windows and Linux, or Microsoft Azure SQL Database. SSMA for Access converts Access database objects to SQL Server or Azure SQL database objects, loads those objects into SQL Server or Azure SQL Database, and then migrates data from Access to SQL Server or Azure SQL Database.

This documentation introduces you to SSMA for Access and provides step-by-step instructions for migrating Access databases to SQL Server or Azure SQL Database and information about issues that might occur after migration.

This support article provides more details about using the SSMA utility. The link for this article is:
https://docs.microsoft.com/en-us/sql/ssma/access/sql-server-migration-assistant-for-access-accesstos

Transfer Images into SQL Server with FmPro Migrator

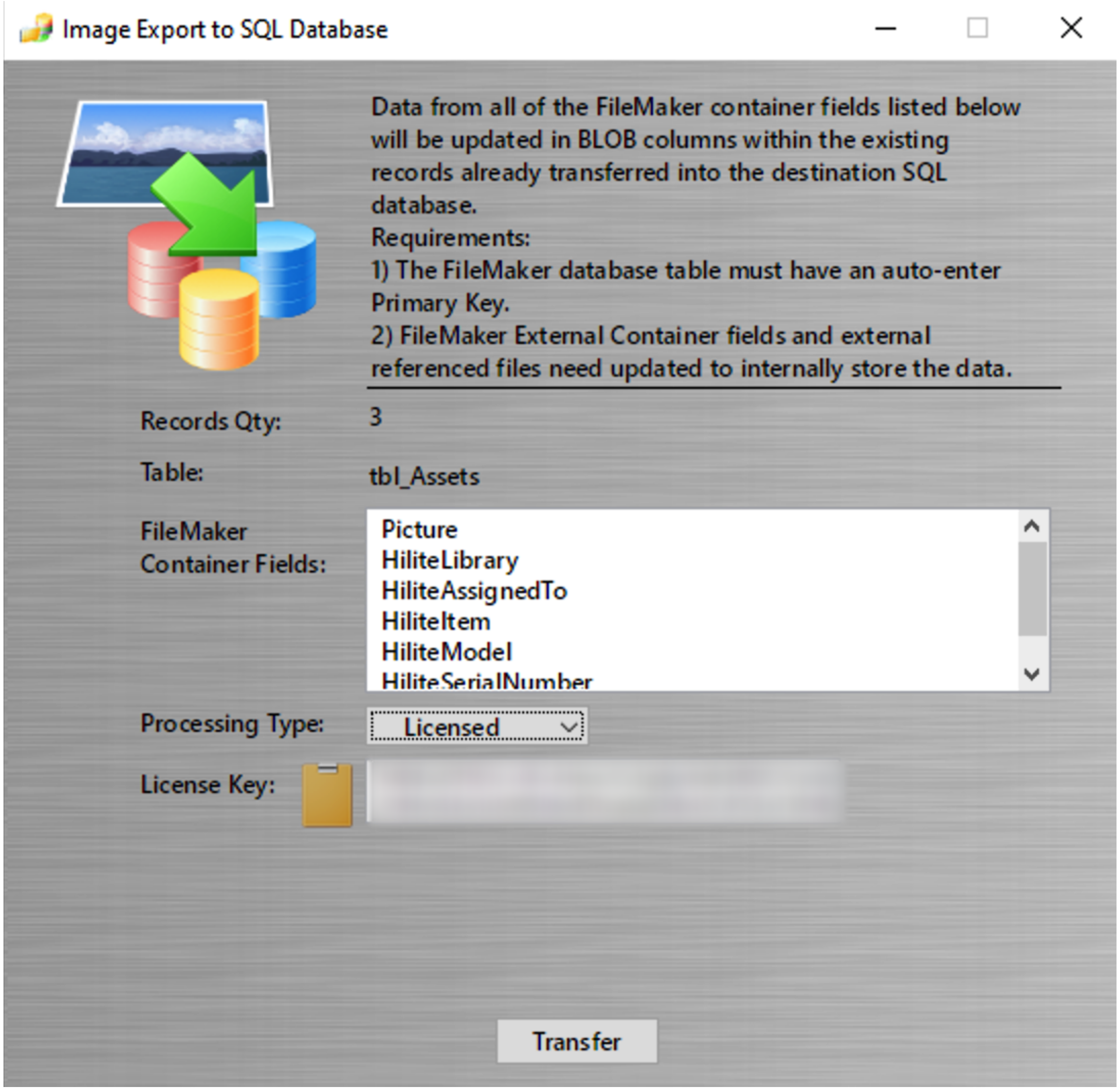


Image Export to SQL Database

Data from all of the FileMaker container fields listed below will be updated in BLOB columns within the existing records already transferred into the destination SQL database.
Requirements:
1) The FileMaker database table must have an auto-enter Primary Key.
2) FileMaker External Container fields and external referenced files need updated to internally store the data.

Records Qty: 3

Table: tbl_Assets

FileMaker Container Fields:
Picture
HiliteLibrary
HiliteAssignedTo
HiliteItem
HiliteModel
HiliteSerialNumber

Processing Type: Licensed

License Key:

The process of transferring images to SQL Server is straightforward, using the Image Export to SQL Database feature.

VBA Sample Code for SQL Server Linked BLOB Columns - Posting Link

Stream from sql server table to ms Access form

Archived Forums > Access for Developers

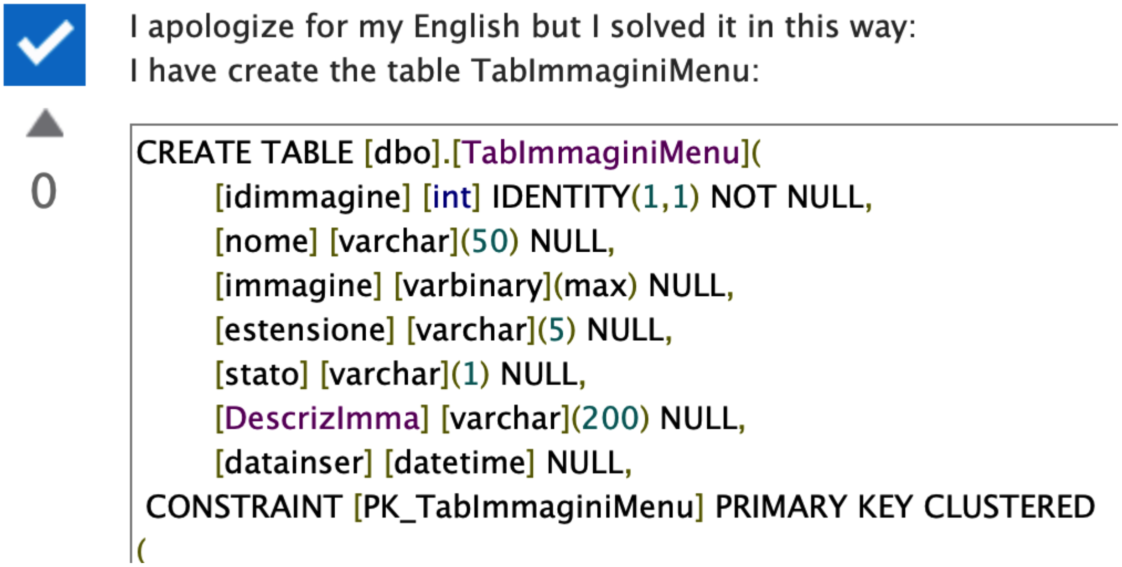
Question

Alert me

This archived forum posting provides some sample code for working with linked SQL Server BLOB columns within Microsoft Access databases. The link for this posting is: https://social.msdn.microsoft.com/Forums/en-US/abd9edcd-2cd8-48c8-b249-a4a4e23df2c0/stream-fi

The following steps include the sample code for the article, in case it cannot be found in the future.

VBA Sample Code for SQL Server Linked BLOB Columns - Table SQL



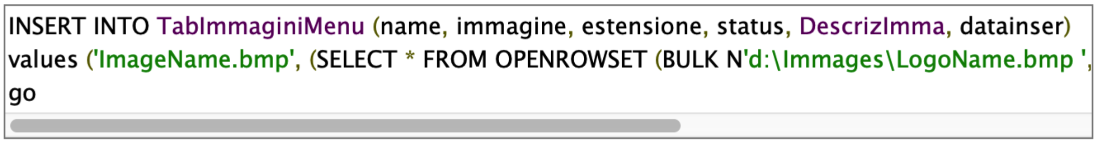
I apologize for my English but I solved it in this way:
I have create the table TabImmaginiMenu:

```
CREATE TABLE [dbo].[TabImmaginiMenu](
	[idimmagine] [int] IDENTITY(1,1) NOT NULL,
	[nome] [varchar](50) NULL,
	[immagine] [varbinary](max) NULL,
	[estensione] [varchar](5) NULL,
	[stato] [varchar](1) NULL,
	[DescrizImma] [varchar](200) NULL,
	[datainser] [datetime] NULL,
 CONSTRAINT [PK_TabImmaginiMenu] PRIMARY KEY CLUSTERED
(
```

```
CREATE TABLE [dbo].[TabImmaginiMenu](
    [idimmagine] [int] IDENTITY(1,1) NOT NULL,
    [nome] [varchar](50) NULL,
    [immagine] [varbinary](max) NULL,
    [estensione] [varchar](5) NULL,
    [stato] [varchar](1) NULL,
    [DescrizImma] [varchar](200) NULL,
    [datainser] [datetime] NULL,
 CONSTRAINT [PK_TabImmaginiMenu] PRIMARY KEY CLUSTERED
(
    [idimmagine] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO

ALTER TABLE [dbo].[TabImmaginiMenu] ADD  CONSTRAINT [DF_TabImmaginiMenu_stato]
DEFAULT ((0)) FOR [stato]
GO
```

VBA Sample Code for SQL Server Linked BLOB Columns - Upload Images SQL Code

to upload images to SQL Server

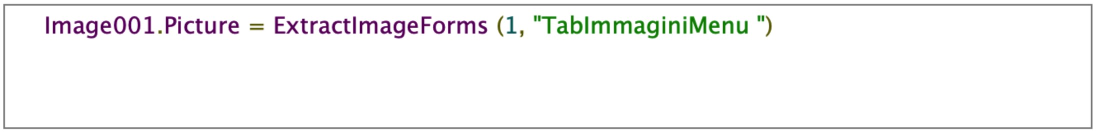


```
INSERT INTO TabImmaginiMenu (name, immagine, estensione, status, DescrizImma, datainser)
values ('ImageName.bmp', (SELECT * FROM OPENROWSET (BULK N'd:\Immages\LogoName.bmp ',
go
```

```
INSERT INTO TabImmaginiMenu (name, immagine, estensione, status, DescrizImma, datainser)
values ('ImageName.bmp', (SELECT * FROM OPENROWSET (BULK
N'd:\Immages\LogoName.bmp ', SINGLE_BLOB) as T1),' bmp ',' 1 ',' LogoName ', getdate ())
go
```

VBA Sample Code for SQL Server Linked BLOB Columns - Load Image When Opening Form

to load the image into the form on opening I enter this step

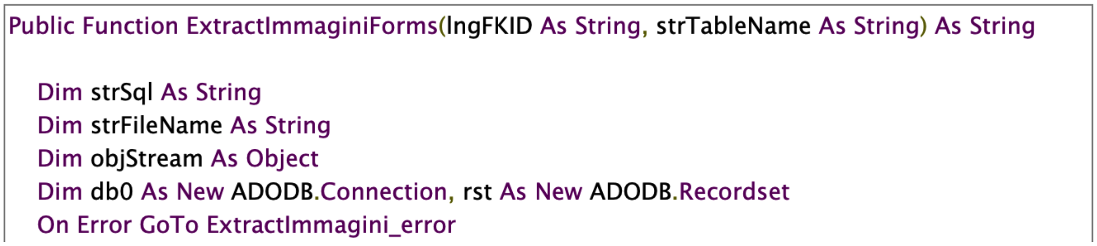


```
  Image001.Picture = ExtractImageForms (1, "TabImmaginiMenu ")
```

```
 Image001.Picture = ExtractImageForms (1, "TabImmaginiMenu ")
```

VBA Sample Code for SQL Server Linked BLOB Columns - VBA ExtractImmaginiForms Function

and this is the function



```
Public Function ExtractImmaginiForms(lngFKID As String, strTableName As String) As String

   Dim strSql As String
   Dim strFileName As String
   Dim objStream As Object
   Dim db0 As New ADODB.Connection, rst As New ADODB.Recordset
   On Error GoTo ExtractImmagini_error
```

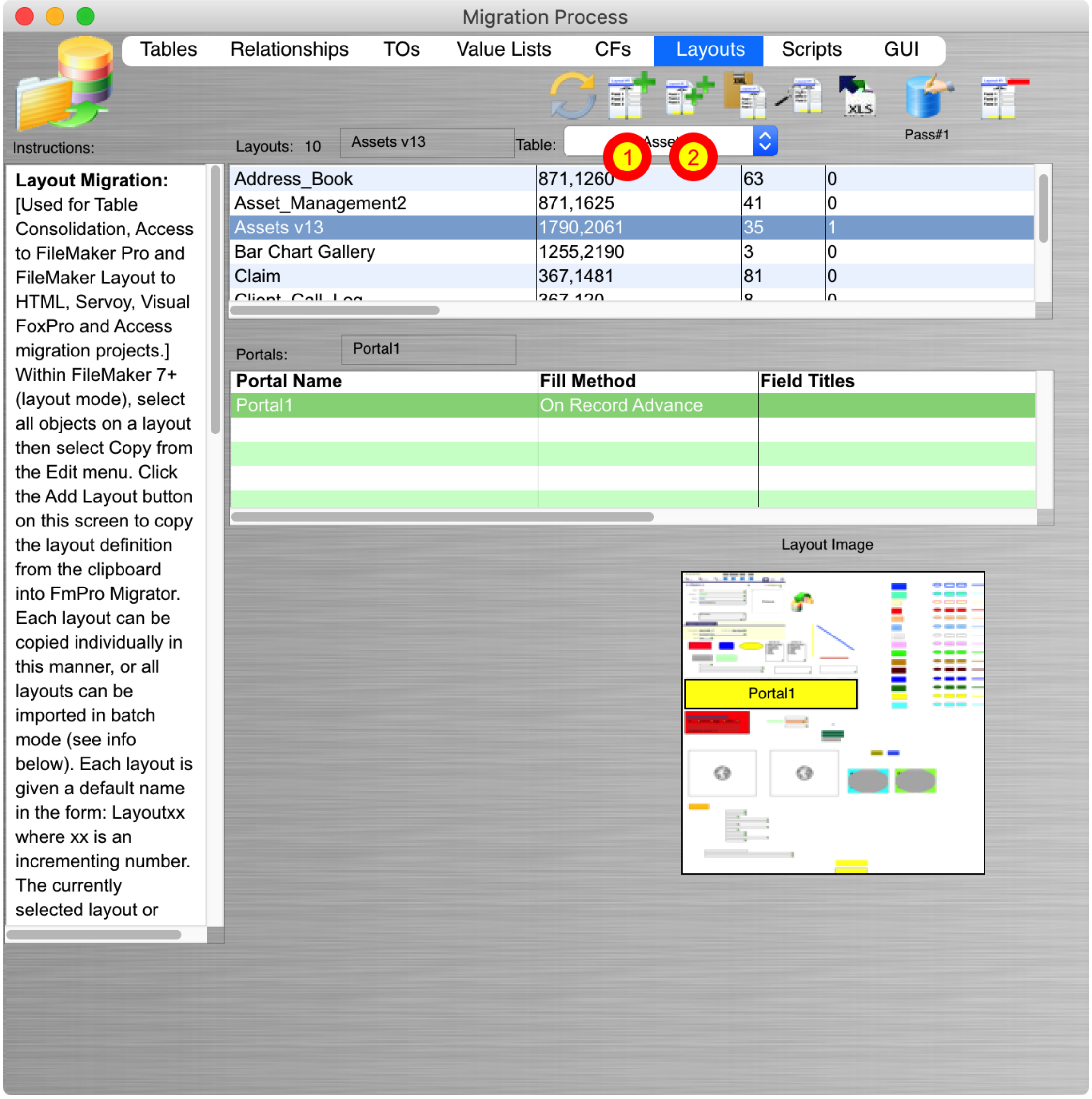
```
Public Function ExtractImmaginiForms(lngFKID As String, strTableName As String) As String

   Dim strSql As String
   Dim strFileName As String
   Dim objStream As Object
   Dim db0 As New ADODB.Connection, rst As New ADODB.Recordset
   On Error GoTo ExtractImmagini_error
   strSql = "SELECT idimmagine, Immagine, nome FROM " & strTableName & " WHERE
idimmagine = " & lngFKID
   db0.ConnectionString = oledbstrconnect
   db0.Open
   rst.Open strSql, db0, adOpenKeyset, adLockOptimistic
   If rst.RecordCount = 0 Then
      GoTo CloseUp
   End If
   On Error GoTo ExtractImmagini_error
   Set objStream = CreateObject("ADODB.Stream")
   With objStream
      .Type = adTypeBinary
      .Open
      .Write rst.Fields("Immagine").Value
      strFileName = CurrentProject.Path & "\temp\" & rst!nome
      .SaveToFile strFileName, adSaveCreateOverWrite
   End With

   ExtractImmaginiForms = strFileName
CloseUp:
   On Error Resume Next
   If rst.state = adStateOpen Then rst.Close: Set rst = Nothing
   Set rst = Nothing
   Set objStream = Nothing
Exit Function
ExtractImmagini_error:
MsgBox "Error. " & lngFKID, vbOKOnly
GoTo CloseUp
End Function
```

Migrating Layouts & Scripts to Access

Importing FileMaker Layouts

Prior to importing layouts, export the DDR XML file for the FileMaker database. On the Relationships tab, click the import button to import the relationships, TOs, Value Lists, Custom Functions into FmPro Migrator.

Layouts Tab



Instructions:

Layout Migration: [Used for Table Consolidation, Access to FileMaker Pro and FileMaker Layout to HTML, Servoy, Visual FoxPro and Access migration projects.] Within FileMaker 7+ (layout mode), select all objects on a layout then select Copy from the Edit menu. Click the Add Layout button on this screen to copy the layout definition from the clipboard into FmPro Migrator. Each layout can be copied individually in this manner, or all layouts can be imported in batch mode (see info below). Each layout is given a default name in the form: Layoutxx where xx is an incrementing number. The currently selected layout or

Layouts: 10 Assets v13 Table: Asse

Address_Book	871,1260	63	0
Asset_Management2	871,1625	41	0
Assets v13	1790,2061	35	1
Bar Chart Gallery	1255,2190	3	0
Claim	367,1481	81	0
Client_Call_Log	367,120	8	0

Portals: Portal1

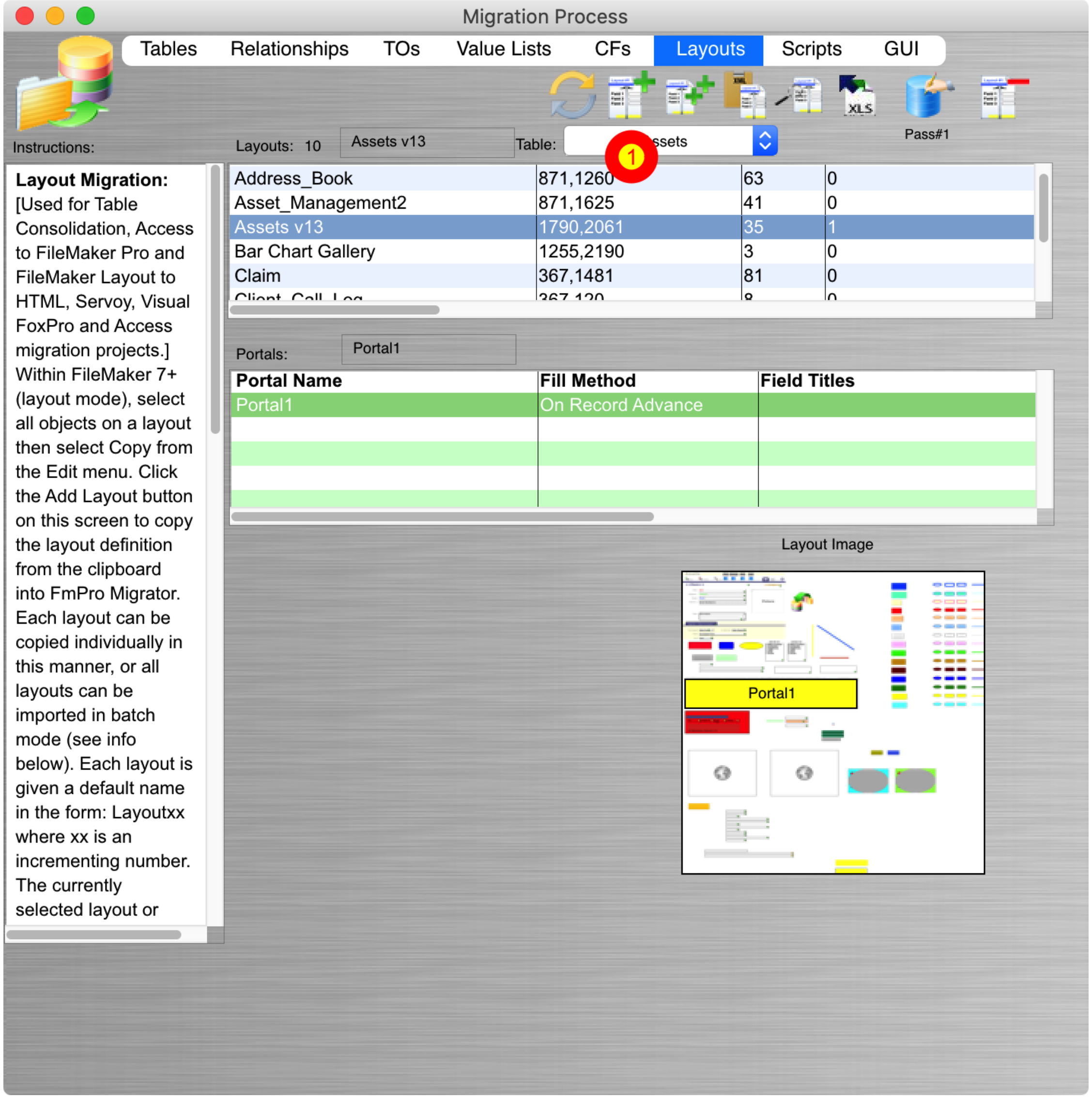
Portal Name	Fill Method	Field Titles
Portal1	On Record Advance	

The Layouts tab provides an interface for managing layouts during a migration project. Layouts can be imported from FileMaker either one at a time [1] or in batch mode [2].

Clicking on a layout in the list displays details for the layout, including a thumbnail version of the layout image.

Clicking on a portal displays the portal rectangle on the thumbnail image.

Import Individual Layouts - One at a Time



Migration Process

Instructions:

Layout Migration: [Used for Table Consolidation, Access to FileMaker Pro and FileMaker Layout to HTML, Servoy, Visual FoxPro and Access migration projects.] Within FileMaker 7+ (layout mode), select all objects on a layout then select Copy from the Edit menu. Click the Add Layout button on this screen to copy the layout definition from the clipboard into FmPro Migrator. Each layout can be copied individually in this manner, or all layouts can be imported in batch mode (see info below). Each layout is given a default name in the form: Layoutxx where xx is an incrementing number. The currently selected layout or

Layouts: 10 Assets v13 Table: ssets

Address_Book	871,1260	63	0
Asset_Management2	871,1625	41	0
Assets v13	1790,2061	35	1
Bar Chart Gallery	1255,2190	3	0
Claim	367,1481	81	0
Client_Call_Log	367,120	8	0

Portals: Portal1

Portal Name	Fill Method	Field Titles
Portal1	On Record Advance	

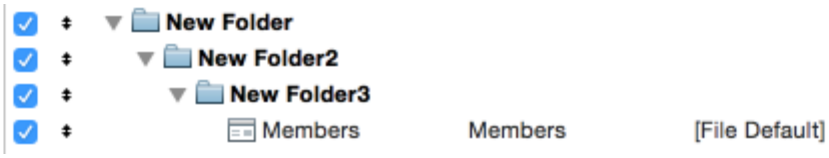
For small databases having under 20 layouts, it is easy to import layouts one at a time. To copy a layout via the clipboard:

Switch FileMaker from browse to layout mode, select the contents of the layout with Command-A (macOS) or Control-A (Windows). Select Copy or Command-C (macOS), Control-C (Windows). This will copy the contents of the layout onto the clipboard, ready for pasting into FmPro Migrator.

(1) Click the Add layout from clipboard button to import a single layout which has already been copied to the clipboard.

Then select the name of the Table associated with the layout and enter a name for the layout.

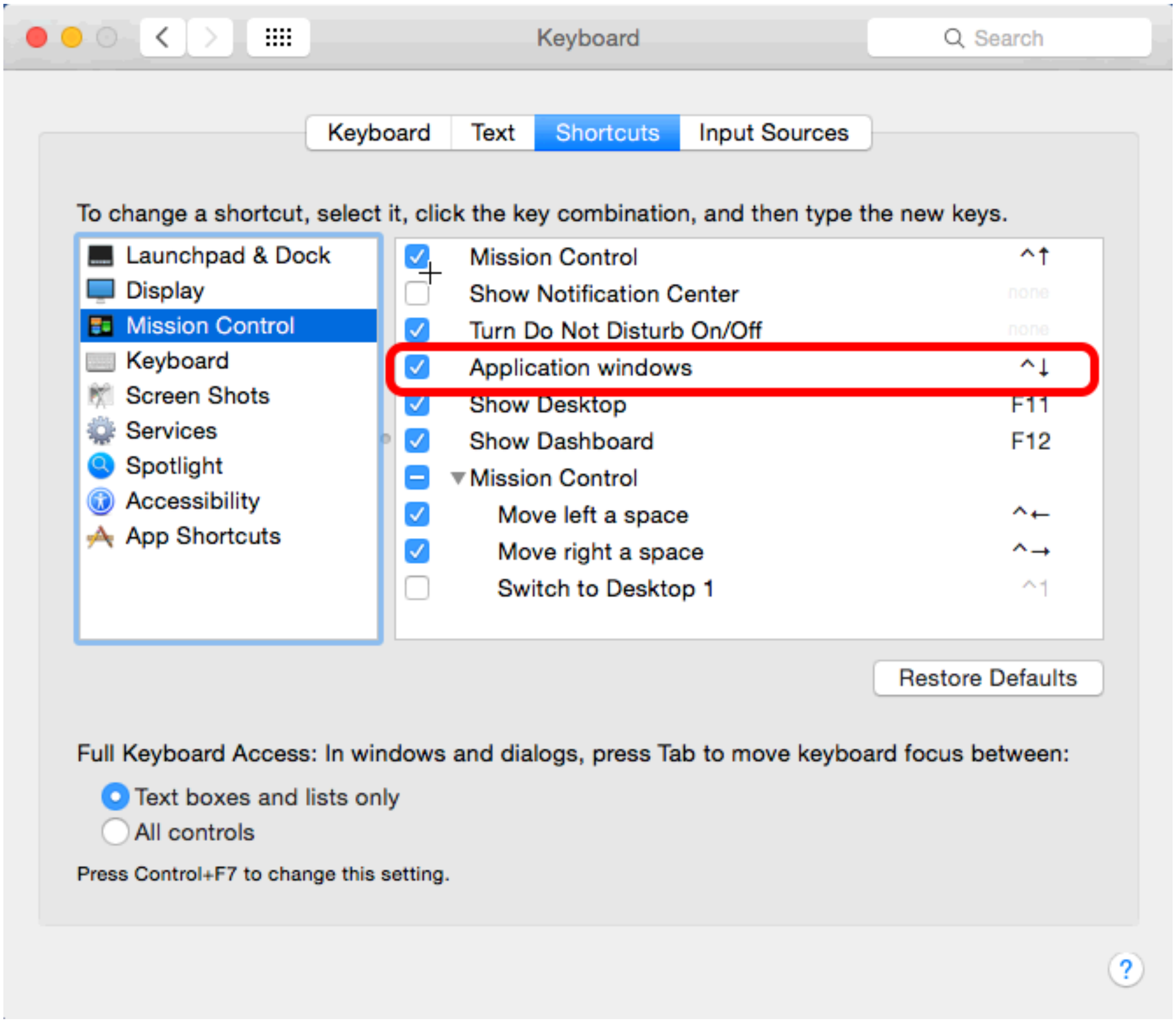
Layouts Within Nested Folders



☑ ▼ New Folder
☑ ▼ New Folder2
☑ ▼ New Folder3
☑ Members Members [File Default]

In almost all cases, FmPro Migrator should be able to import layouts which are nested within folders. If for some reason, this doesn't work when performing the automated batch importing - move the layouts outside the nested folders, and re-export the DDR XML file again, prior to performing the batch import feature.

Mission Control - Shortcut Key Conflict with Command Up Arrow in FileMaker (macOS)



Keyboard

Search

Keyboard | Text | Shortcuts | Input Sources

To change a shortcut, select it, click the key combination, and then type the new keys.

- Launchpad & Dock
- Display
- Mission Control
- Keyboard
- Screen Shots
- Services
- Spotlight
- Accessibility
- App Shortcuts

☑	Mission Control	^↑
□	Show Notification Center	none
☑	Turn Do Not Disturb On/Off	none
☑	Application windows	^↓
☑	Show Desktop	F11
☑	Show Dashboard	F12
⊟	▼Mission Control	
☑	Move left a space	^←
☑	Move right a space	^→
□	Switch to Desktop 1	^1

Restore Defaults

Full Keyboard Access: In windows and dialogs, press Tab to move keyboard focus between:

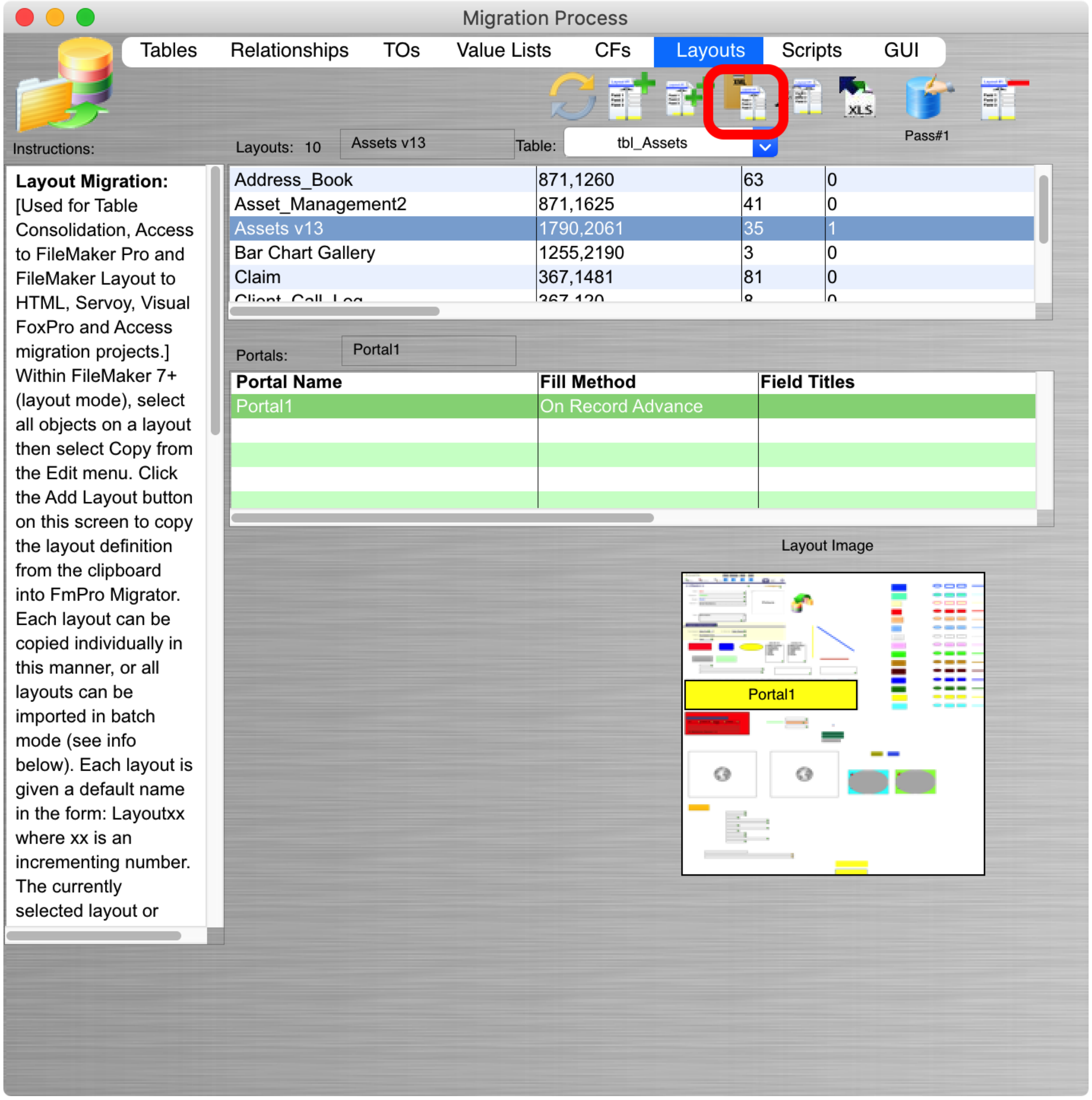
● Text boxes and lists only
○ All controls

Press Control+F7 to change this setting.

?

On macOS, one of the default shortcut keys for Mission Control conflicts with the FileMaker Shift Down Arrow shortcut used for advancing to the next layout. FmPro Migrator will import the same layout many times if this conflict exits. Before importing layouts, disable this conflicting shortcut key in the System Preferences -> Keyboard panel for Mission Control. It is a good idea to disable 3rd party automation tools which might also have conflicting shortcut keys.

Click Import All Layouts Button - Batch Importing



Instructions:

Layout Migration: [Used for Table Consolidation, Access to FileMaker Pro and FileMaker Layout to HTML, Servoy, Visual FoxPro and Access migration projects.] Within FileMaker 7+ (layout mode), select all objects on a layout then select Copy from the Edit menu. Click the Add Layout button on this screen to copy the layout definition from the clipboard into FmPro Migrator. Each layout can be copied individually in this manner, or all layouts can be imported in batch mode (see info below). Each layout is given a default name in the form: Layoutxx where xx is an incrementing number. The currently selected layout or

Layouts: 10 Assets v13 Table: tbl_Assets

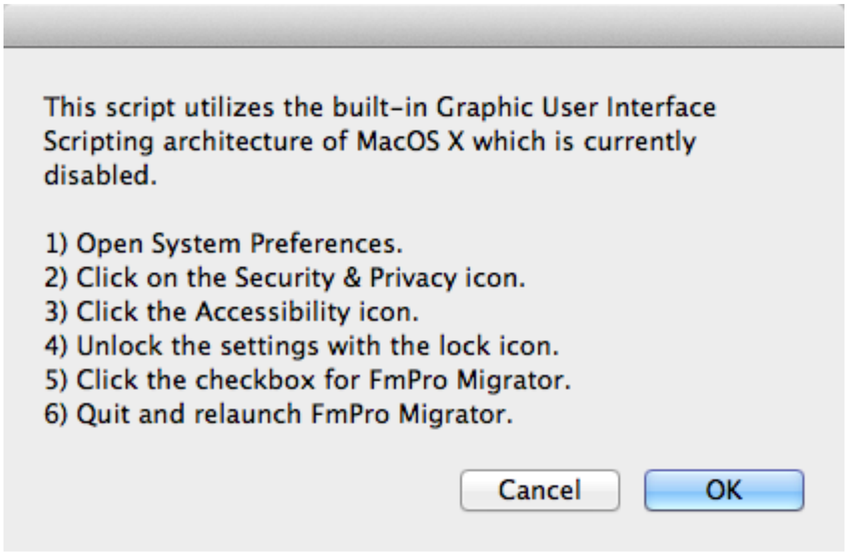
Address_Book	871,1260	63	0
Asset_Management2	871,1625	41	0
Assets v13	1790,2061	35	1
Bar Chart Gallery	1255,2190	3	0
Claim	367,1481	81	0
Client_Call_Log	367,120	8	0

Portals: Portal1

Portal Name	Fill Method	Field Titles
Portal1	On Record Advance	

The Import All Layouts button starts the process of importing all of the layouts.

macOS 10.9+ Security Dialog

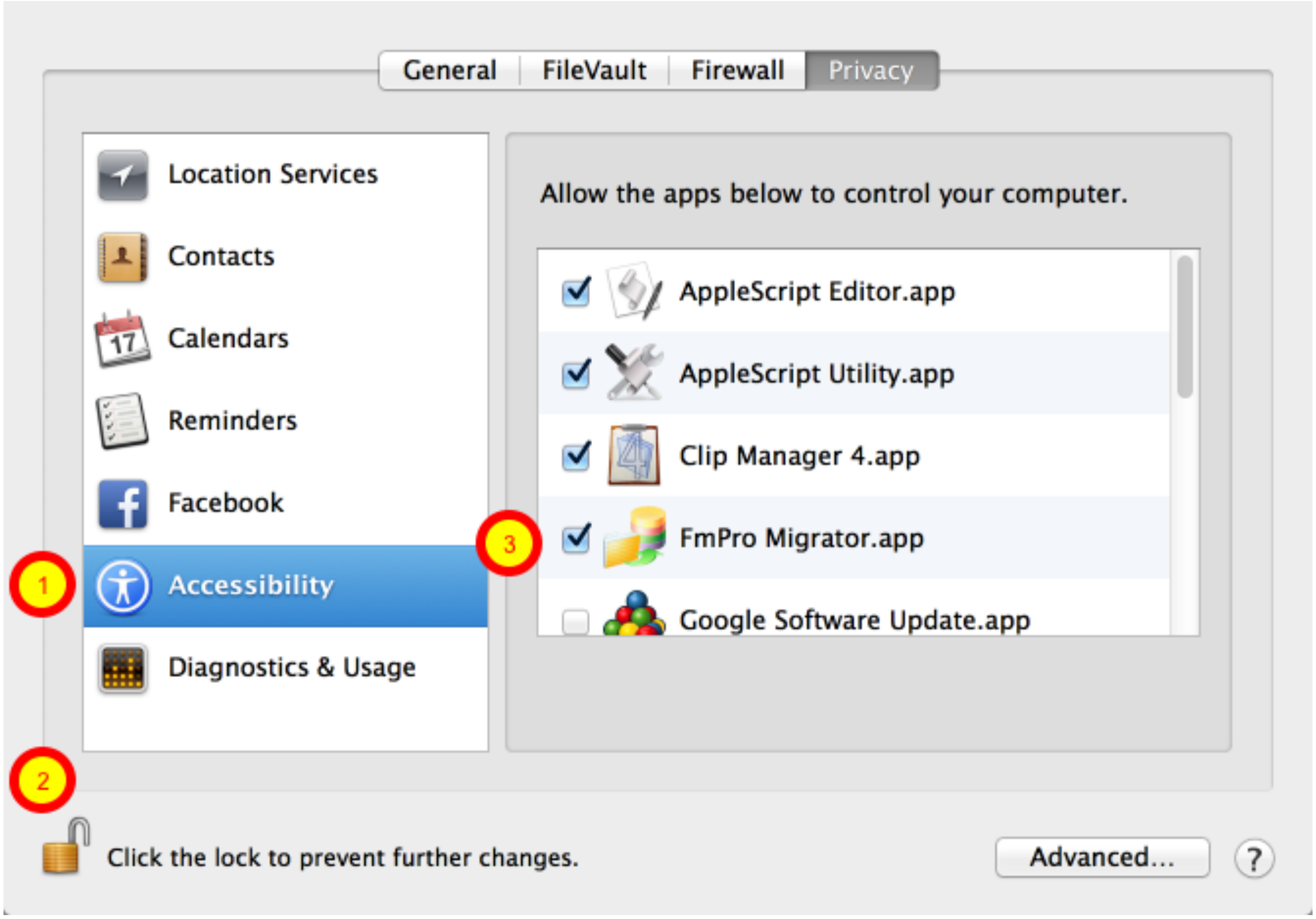


This script utilizes the built-in Graphic User Interface Scripting architecture of MacOS X which is currently disabled.

1) Open System Preferences.
2) Click on the Security & Privacy icon.
3) Click the Accessibility icon.
4) Unlock the settings with the lock icon.
5) Click the checkbox for FmPro Migrator.
6) Quit and relaunch FmPro Migrator.

On macOS, FmPro Migrator uses AppleScript to perform batch processing tasks with FileMaker database files. Starting with macOS 10.9 (Mavericks), each application which uses the Accessibility API to control the computer needs to be authorized. FmPro Migrator performs a test to verify that it has been authorized to control the computer via AppleScript GUI scripting, and presents this dialog if it has not yet been enabled.

macOS Security Preferences



Allow the apps below to control your computer.

- ☑ AppleScript Editor.app
- ☑ AppleScript Utility.app
- ☑ Clip Manager 4.app
- ☑ FmPro Migrator.app
- □ Google Software Update.app

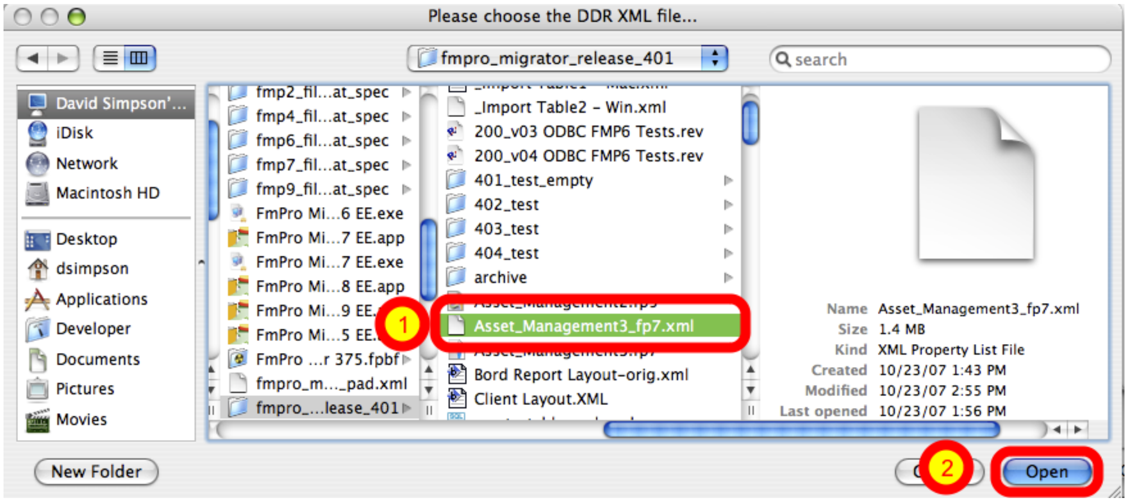
Click the lock to prevent further changes.

In the macOS System Preferences, click the Security & Privacy icon, (1) click the Accessibility item in the list, (2) click the lock button to allow changes, (3) click the checkmark next to the FmPro Migrator application.

Note: You will need to perform this unlocking process once, for each new version of FmPro Migrator is installed on your computer.

After allowing Accessibility access for FmPro Migrator, it will often be possible to immediately click on the same GUI batch processing button in FmPro Migrator to proceed with the migration. Sometimes you need to re-launch FmPro Migrator after making this change, and it can be helpful to close the System Preferences window to insure that macOS has recognized the change.

Select the DDR XML File


Please choose the DDR XML file...

Name Asset_Management3_fp7.xml
Size 1.4 MB
Kind XML Property List File
Created 10/23/07 1:43 PM
Modified 10/23/07 2:55 PM
Last opened 10/23/07 1:56 PM

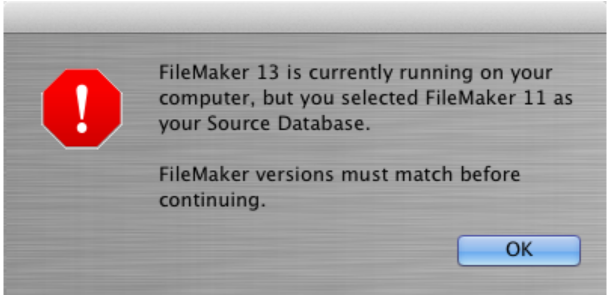
(1) Select a DDR XML file, (2) click the Open button.

Note: FmPro Migrator requires a Database Design Report as a DDR XML file created with FileMaker Developer/Advanced 7+/FileMaker Pro 17+. This DDR XML file is used by FmPro Migrator to count the number of layouts which will be copied via the clipboard, as well as the order in which the layouts are found within the FileMaker database file. The DDR XML file should be re-created if any layouts have been deleted or moved within the file by using the Manage Layouts dialog.

This DDR XML file needs to be created with the same version of FileMaker Pro which is being used for the conversion project. The version of FileMaker Pro should remain the same during the entire project.

FmPro Migrator will loop thru all of the layouts and capture them via the clipboard by using AppleScript (macOS) or WinAutomation (Windows) to drive the FileMaker graphical interface.

Troubleshooting - Mis-Matched FileMaker Versions Error [macOS]

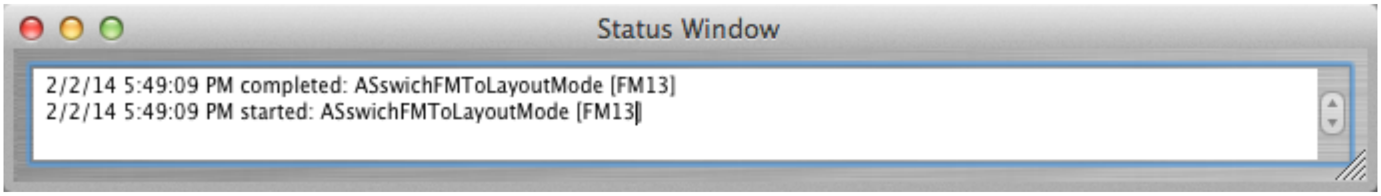

FileMaker 13 is currently running on your computer, but you selected FileMaker 11 as your Source Database.

FileMaker versions must match before continuing.

FmPro Migrator needs to have an accurate idea of which version of FileMaker Pro is currently running on your computer in order to use the correct AppleScript code to perform each task. This means that it is important to select the correct Source Database and Destination Database on the main FmPro Migrator window.

If you are capturing layouts from FileMaker Pro, FmPro Migrator will verify that the source database version you selected is the actual version which is running on your computer [see dialog above].

Troubleshooting - Locks Up with No Processing Completed [macOS]



```
2/2/14 5:49:09 PM completed: ASswichFMToLayoutMode [FM13]
2/2/14 5:49:09 PM started: ASswichFMToLayoutMode [FM13]
```

If a batch processing task starts, but appears to lock up with no activity, then force quit and re-launch FmPro Migrator from the Finder and open the Status Window by selecting the File -> Status Window menu item. Click the batch processing button again, and review the results in the Status Window.

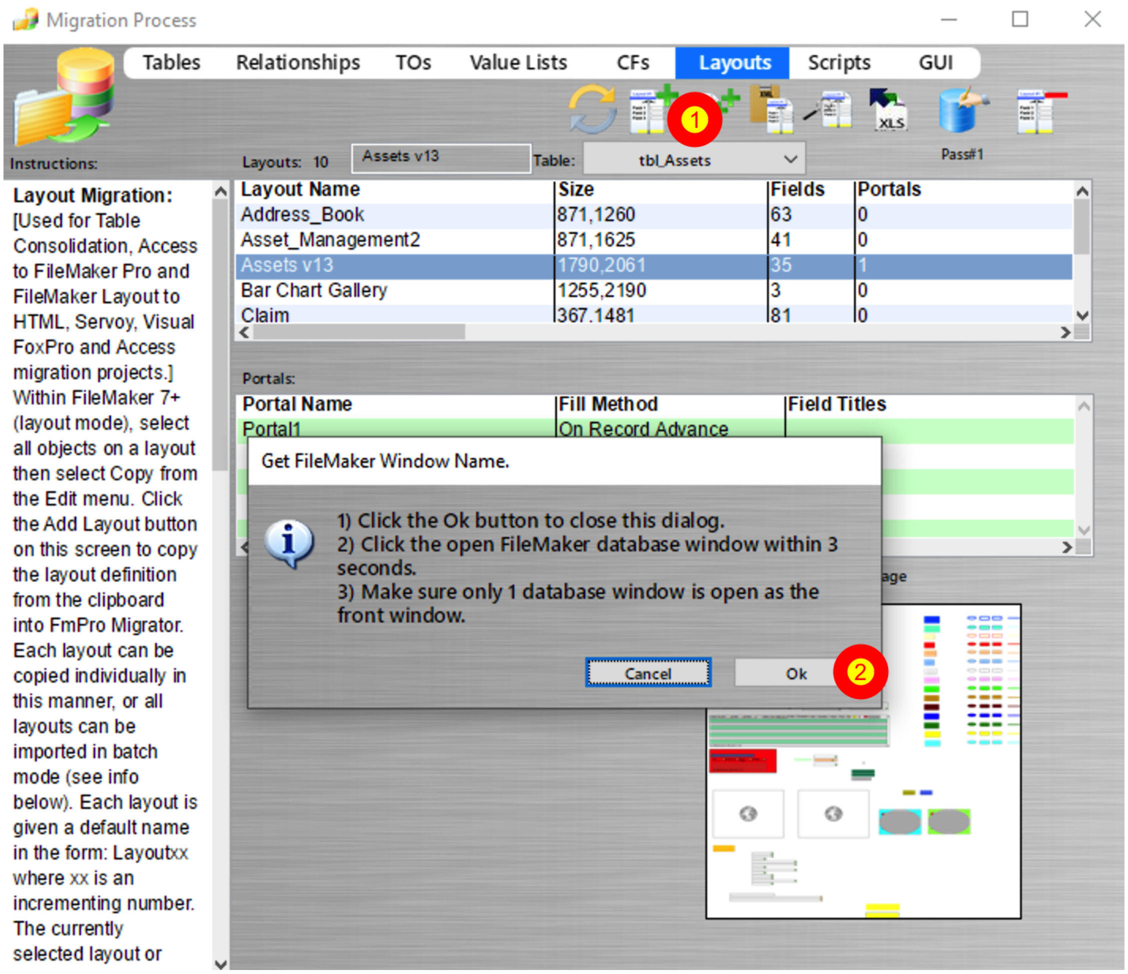
FmPro Migrator will continue to log the results of its AppleScript processing handlers in the Status Window as long as the window is open.

As shown above, the ASswitchFMToLayoutMode handler has run, but nothing else has completed. This type of problem can be due to having 2 copies of FileMaker Pro Advanced open at the same time. Some of the commands might get sent to one copy of FileMaker and other commands could get sent to the 2nd copy of FileMaker.

If a single copy of FileMaker Pro is running, but the FileMaker database file is not open, a similar problem might also occur.

For more details about the tasks performed by the AppleScript handlers, please see the Troubleshooting AppleScript GUI Automation part of this manual.

Click Import All Layouts Button - Batch Importing #1 (Windows)



Instructions:

Layouts: 10 Assets v13 Table: tbl_Assets

Layout Migration: [Used for Table Consolidation, Access to FileMaker Pro and FileMaker Layout to HTML, Servoy, Visual FoxPro and Access migration projects.] Within FileMaker 7+ (layout mode), select all objects on a layout then select Copy from the Edit menu. Click the Add Layout button on this screen to copy the layout definition from the clipboard into FmPro Migrator. Each layout can be copied individually in this manner, or all layouts can be imported in batch mode (see info below). Each layout is given a default name in the form: Layoutxx where xx is an incrementing number. The currently selected layout or

Layout Name	Size	Fields	Portals
Address_Book	871,1260	63	0
Asset_Management2	871,1625	41	0
Assets v13	1790,2061	35	1
Bar Chart Gallery	1255,2190	3	0
Claim	367,1481	81	0

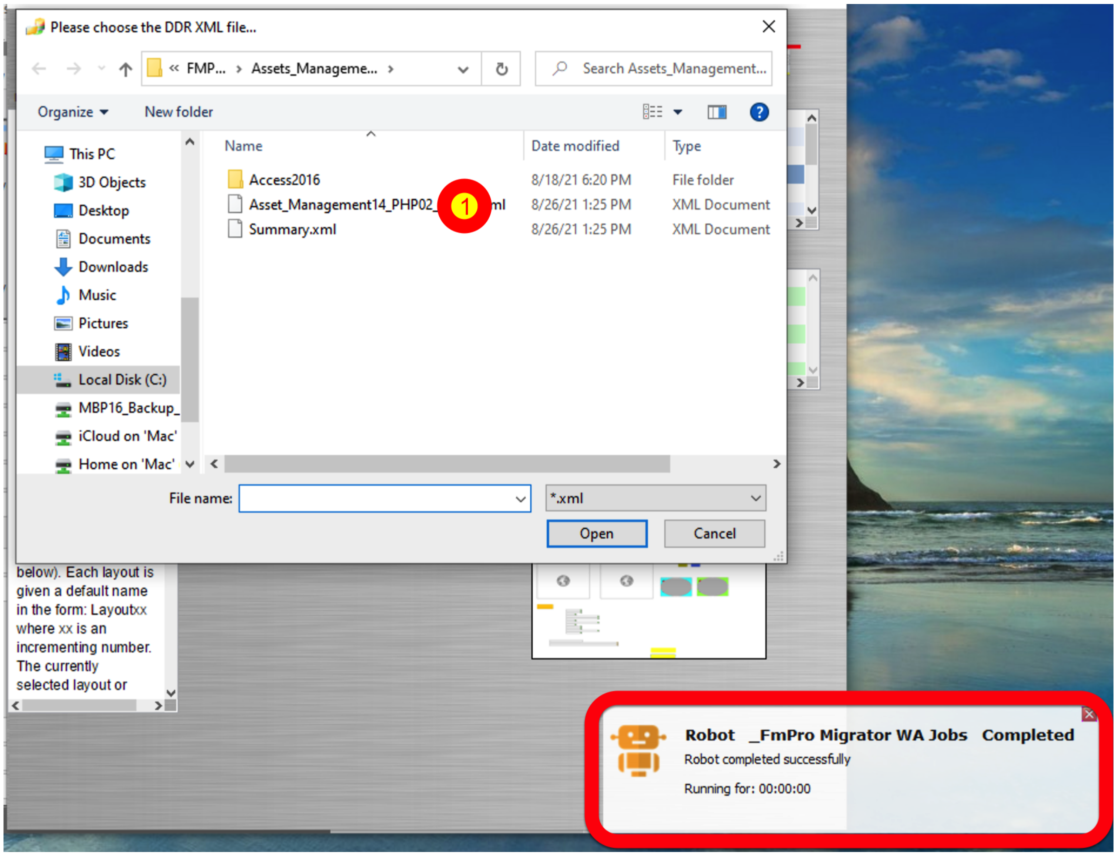
Portals:

Portal Name	Fill Method	Field Titles
Portal1	On Record Advance	

Get FileMaker Window Name.

1) Click the Ok button to close this dialog.
2) Click the open FileMaker database window within 3 seconds.
3) Make sure only 1 database window is open as the front window.

Click Import All Layouts Button - Batch Importing #2 (Windows)



Name	Date modified	Type
Access2016	8/18/21 6:20 PM	File folder
Asset_Management14_PHP02_ ml	8/26/21 1:25 PM	XML Document
Summary.xml	8/26/21 1:25 PM	XML Document

below). Each layout is given a default name in the form: Layoutxx where xx is an incrementing number. The currently selected layout or

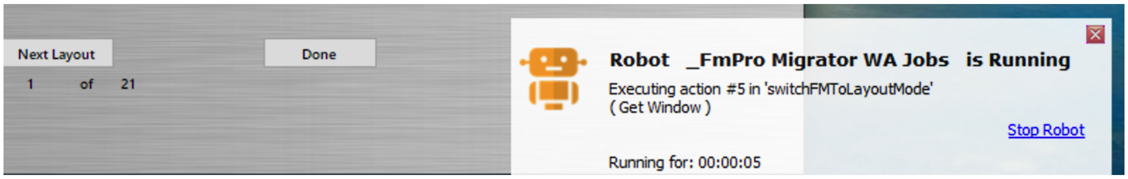
Robot _FmPro Migrator WA Jobs Completed

Robot completed successfully

Running for: 00:00:00

On Windows, a compiled WinAutomation Job is used to automate the importing of layouts.

Click Import All Layouts Button - Batch Importing #3 (Windows)



Robot _FmPro Migrator WA Jobs is Running

Executing action #5 in 'switchFMToLayoutMode'
(Get Window)

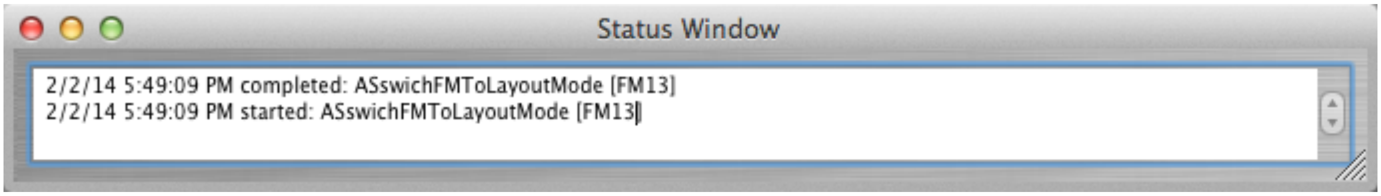
Running for: 00:00:05

FmPro Migrator will work thru the list of layouts which were read from the DDR XML file and will show the progress at the bottom of the Layouts tab. During processing, multiple WinAutomation job actions will be displayed during the processing.

Troubleshooting AppleScript GUI Automation [macOS]

FmPro Migrator uses embedded AppleScript code to automate importing Layouts, and building Custom Functions, Value Lists, Table Occurrences, Relationships, Scripts and Layouts in FileMaker database files. Detailed info about this AppleScript code is included in this section of the manual.

Troubleshooting - Locks Up with No Processing Completed [macOS]



```
2/2/14 5:49:09 PM completed: ASswichFMToLayoutMode [FM13]
2/2/14 5:49:09 PM started: ASswichFMToLayoutMode [FM13]
```

If a batch processing task starts, but appears to lock up with no activity, then force quit and re-launch FmPro Migrator from the Finder and open the Status Window by selecting the File -> Status Window menu item. Click the batch processing button again, and review the results in the Status Window.

FmPro Migrator will continue to log the results of its AppleScript processing handlers in the Status Window as long as the window is open.

As shown above, the ASswitchFMToLayoutMode handler has run, but nothing else has completed. This type of problem can be due to having 2 copies of FileMaker Pro Advanced open at the same time. Some of the commands might get sent to one copy of FileMaker and other commands could get sent to the 2nd copy of FileMaker.

If a single copy of FileMaker Pro Advanced is running, but the FileMakerDB.fp7/FileMakerDB.fmp12 file is not open, a similar problem might also occur. Generally, FmPro Migrator should detect these types of situations and warn you automatically at the start of automated process.

Troubleshooting - List of AppleScript Handlers

FmPro Migrator uses the following handlers containing dynamically compiled AppleScript code:

ASCheckFMVersion - check the currently running FileMaker version to make sure it matches the selected version

AScheckGUIScriptingMode - Checks to determine if GUI Scripting is enabled for use by AppleScript. If false, it will issue the command to enable macOS Accessibility features on macOS 10.8 or below. For macOS 10.9, the user will be prompted to open System Preferences to enable Accessibility for FmPro Migrator.

AScloseFMCustomFunctionsWindow - Closes the “Manage Custom Functions...” window by clicking the Ok button.

AScloseFMManageScriptsWindow - Closes the “Manage Scripts...” window by clicking on the close button, selects first script in list in preparation for creating the next script.

AScloseFMRelationshipGraphWindow - Sends a return keystroke to close the FileMaker Relationship Graph window.

AScloseFMValueListsWindow - Closes the “Manage Value Lists...” window by clicking the Ok button.

AScopyFMLayout - Copies the current FileMaker layout into FmPro Migrator.

AScopyLayout2FM - Creates a new empty layout in FileMaker (Pass #1), sets layout name, sets TO Name for new layout, deletes layout header, footer parts, pastes the layout XML already on the clipboard, clicks Yes to extend objects dialog #1, #2.

AScreateFMCustomFunction - Reads Custom Function creation parameters from the Clipboard. Creates a new FileMaker Custom Function, having the proper name, parameters and code, then saves it.

AScreateFMCustomValueList - Creates a FileMaker custom value list and saves it. Field based value lists are also created, but the properties need to be set manually.

AScreateFMRelationship - Reads relationship creation parameters from the Clipboard, storing info in 10 variables, creates a new relationship or a predicate for an existing relationship which is already being worked on.

AScreateFMTO - Gets the TOName and Base Table Name from the Clipboard then creates a new FileMaker TO. Note: Setting the TO color is not supported.

ASdeleteFMLayoutParts (function) - Deletes the Header and Footer layout parts from a new empty layout, leaving the Body part.

ASgetTOListPosition (function) - Returns the position number of a TO name within a list of TOs. The returned value is used for selecting a specific line (TO name) within a popup menu. This needs to be done with a numeric index to the TO name instead of passing in the actual TO name via keystrokes.

ASglobalsAndHandlers (function) - returns the AppleScript code having the global variables and commonly used handlers

ASgotoNextFMLayout (function) - Advances to the next FileMaker layout while in layout mode, by sending the control down arrow keystroke.

ASopenFMCustomFunctionsWindow - Opens the “Manage Custom Functions” window using the File menu.

ASopenFMEditRelationhipWindow - Opens Edit Relationship window using command shift r keystroke, waits for window to open.

ASopenFMEditRelationhipWindow - Opens Edit Relationship window, waits for window to open.

ASopenFMManageScriptsWindow - Opens the “Manage Scripts...” window using the Command Shift S keystroke.

ASopenFMManageTablesWindow - Opens “Manage Database” window, clicks on the Tables tab.

ASopenFMRelationshipGraphWindow - Opens “Manage Database” window, clicks on the Relationship Graph tab. NOTE: Doesn't work with 8,9,10 because window doesn't refresh when Relationship Graph tab is clicked.

ASopenFMValueListsWindow - Opens the “Manage Value Lists...” window using the File menu."

ASpasteFMScript - Clicks the first row of the scripts displayed in the “Manage Scripts...” window. Pastes the current script XML on the clipboard using Command V keys.

ASreplaceLayout2FMPPass2 - Selects the contents of the current layout in Layout mode, deletes the contents, pastes the layout contents already on the clipboard. Used during Pass #2 processing during Table Consolidation projects.

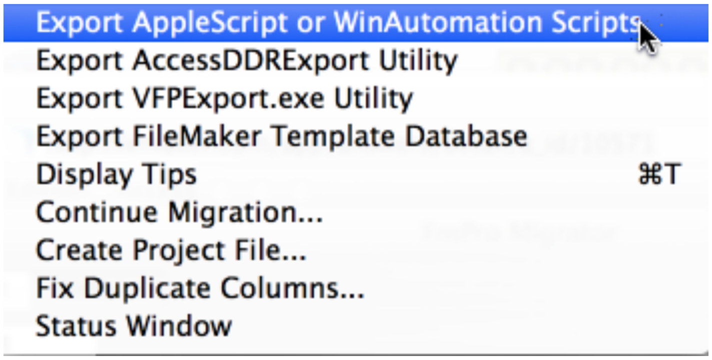
ASsaveFMRelationship - Sends keystroke return key to click the Ok button of the Edit Relationship window to save the current relationship.

ASsaveFMRelationship (function) - Sends return key keystroke to click the Ok button of the Edit Relationship window to save the current relationship. Sends another return key keystroke if the Rename TO dialog is displayed.

ASswitchFMToBrowseMode - Switches to Browse Mode using Command b keys.

ASswitchFMToLayoutMode - Switches to Layout Mode, using Command L keys, advances to layout #1.

Export AppleScript Code

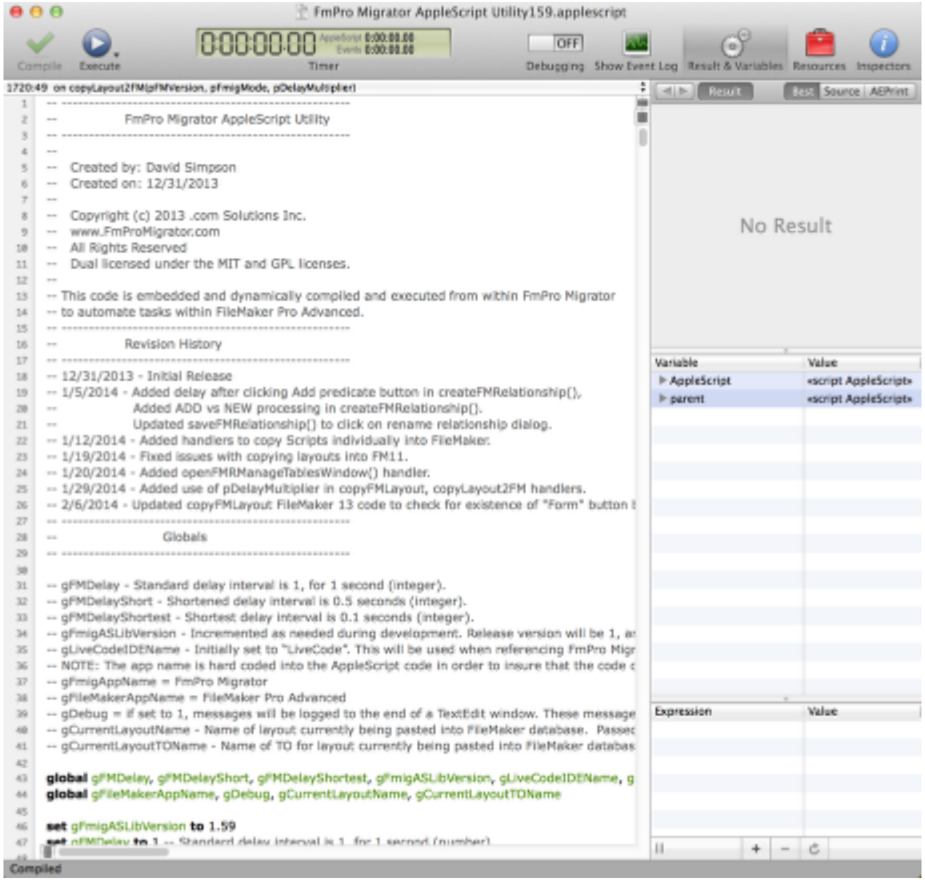

Export AppleScript or WinAutomation Scripts
Export AccessDDRExport Utility
Export VFPExport.exe Utility
Export FileMaker Template Database
Display Tips ⌘T
Continue Migration...
Create Project File...
Fix Duplicate Columns...
Status Window

On macOS, the AppleScript code is embedded within FmPro Migrator therefore no installation is required.

It may be helpful to review the AppleScript source code for troubleshooting purposes, therefore it can be exported to the output directory by selecting the File -> Export AppleScript or WinAutomation Scripts menu.

Selecting this menu will save a text file of AppleScript code, a LiveCode AppleScript test stack and a LiveCode performance testing stack to the output directory.
These files have been released as open source for use by the FileMaker community.

FmPro Migrator AppleScript Utility???.applescript File

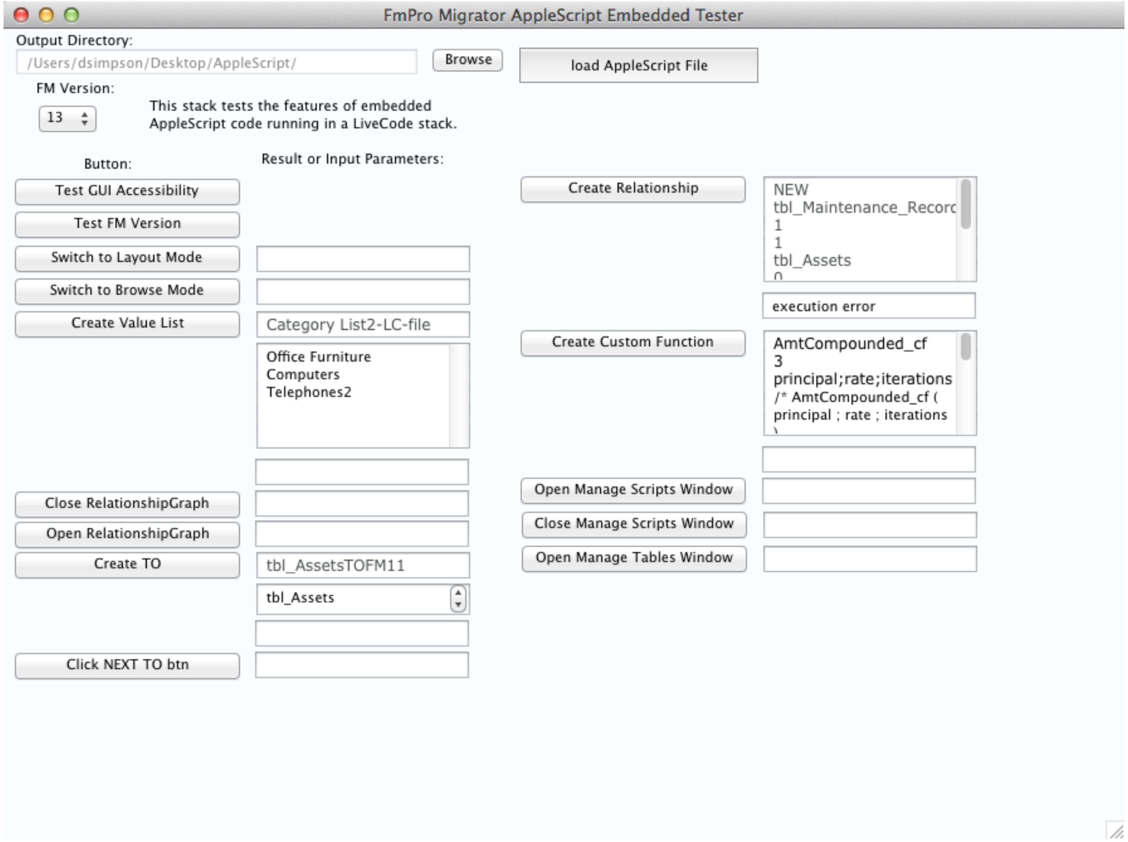


```
-- ----------------------------------------------------
--              FmPro Migrator AppleScript Utility
-- ----------------------------------------------------
--
--   Created by: David Simpson
--   Created on: 12/31/2013
--
--   Copyright (c) 2013 .com Solutions Inc.
--   www.FmProMigrator.com
--   All Rights Reserved
--   Dual licensed under the MIT and GPL licenses.
--
-- This code is embedded and dynamically compiled and executed from within FmPro Migrator
-- to automate tasks within FileMaker Pro Advanced.
-- ----------------------------------------------------
--              Revision History
-- ----------------------------------------------------
-- 12/31/2013 - Initial Release
-- 1/5/2014 - Added delay after clicking Add predicate button in createFMRelationship().
--                Added ADD vs NEW processing in createFMRelationship().
--                Updated saveFMRelationship() to click on rename relationship dialog.
-- 1/12/2014 - Added handlers to copy Scripts individually into FileMaker.
-- 1/19/2014 - Fixed issues with copying layouts into FM11.
-- 1/20/2014 - Added openFMRManageTablesWindow() handler.
-- 1/29/2014 - Added use of pDelayMultiplier in copyFMLayout, copyLayout2FM handlers.
-- 2/6/2014 - Updated copyFMLayout FileMaker 13 code to check for existence of "Form" button t
-- ----------------------------------------------------
--                  Globals
-- ----------------------------------------------------


-- gFMDelay - Standard delay interval is 1, for 1 second (integer).
-- gFMDelayShort - Shortened delay interval is 0.5 seconds (integer).
-- gFMDelayShortest - Shortest delay interval is 0.1 seconds (integer).
-- gFmigASLibVersion - Incremented as needed during development. Release version will be 1, a
-- gLiveCodeIDEName - Initially set to "LiveCode". This will be used when referencing FmPro Migr
-- NOTE: The app name is hard coded into the AppleScript code in order to insure that the code c
-- gFmigAppName = FmPro Migrator
-- gFileMakerAppName = FileMaker Pro Advanced
-- gDebug = if set to 1, messages will be logged to the end of a TextEdit window. These message
-- gCurrentLayoutName - Name of layout currently being pasted into FileMaker database.  Passec
-- gCurrentLayoutTOName - Name of TO for layout currently being pasted into FileMaker databas

global gFMDelay, gFMDelayShort, gFMDelayShortest, gFmigASLibVersion, gLiveCodeIDEName, g
global gFileMakerAppName, gDebug, gCurrentLayoutName, gCurrentLayoutTOName

set gFmigASLibVersion to 1.59
set gFMDelay to 1 -- Standard delay interval is 1, for 1 second (number)
```

Variable	Value
▶ AppleScript	«script AppleScript»
▶ parent	«script AppleScript»

The AppleScript code embedded within FmPro Migrator follows the naming strategy used within the FmPro Migrator AppleScript Utility???.applescript text file.

FmPro Migrator AppleScript Embedded Tester.livecode Stack



Output Directory: /Users/dsimpson/Desktop/AppleScript/

FM Version: 13

This stack tests the features of embedded AppleScript code running in a LiveCode stack.

Category List2-LC-file

Office Furniture
Computers
Telephones2

tbl_AssetsTOFM11

tbl_Assets

NEW
tbl_Maintenance_Recorc
1
1
tbl_Assets
0

execution error

AmtCompounded_cf
3
principal;rate;iterations
/* AmtCompounded_cf (
principal ; rate ; iterations

Many of the AppleScript handlers embedded within FmPro Migrator have been converted into LiveCode handlers within the FmPro Migrator AppleScript Embedded Tester stack file.
This stack file serves as a test harness for testing the embedded AppleScript code, prior to implementing it within FmPro Migrator. This stack is also released as open source software.

Download a copy of LiveCode from the www.livecode.com website to open this stack file.

Right click the mouse on any of the buttons to see the code which is being called.

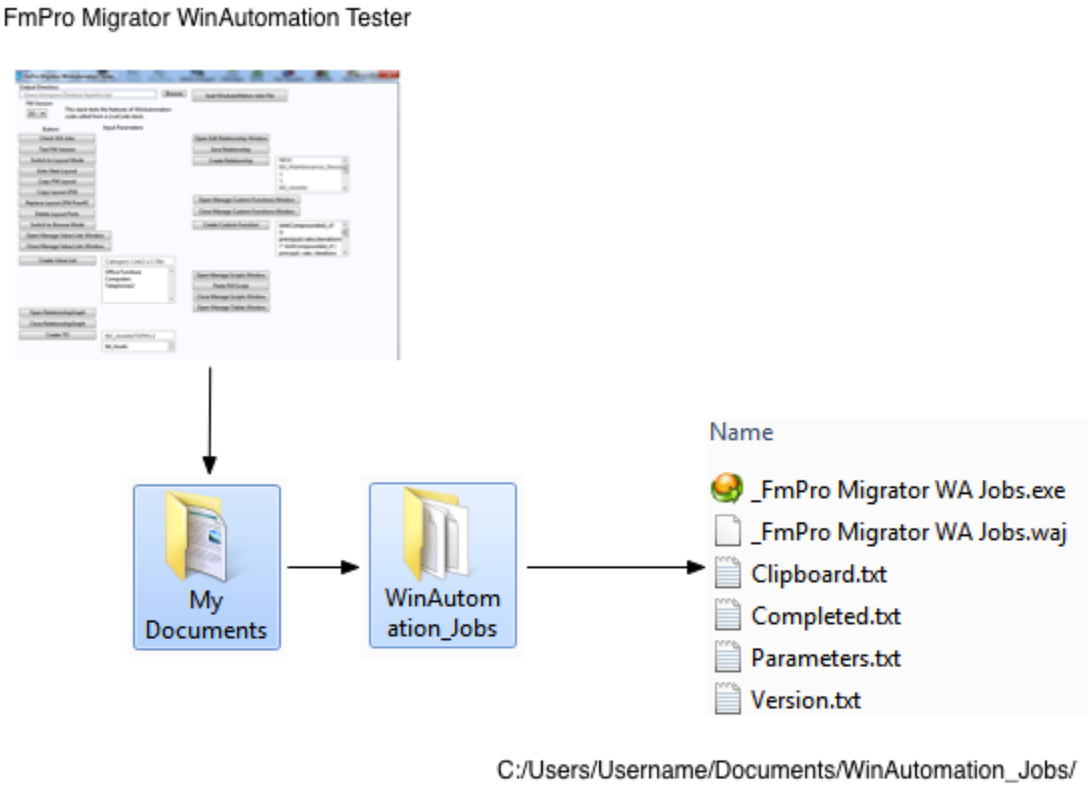
Select the LiveCode Object -> Stack Script menu item to see the stack level handlers.

About the FmPro Migrator WinAutomation Tester [Windows]

The FmPro Migrator WinAutomation Tester is a LiveCode 9 stack file used to test WinAutomation jobs [now called Robots] used to automate tasks within a FileMaker Pro Advanced 11 - 19 database.
A copy of LiveCode can be downloaded from the LiveCode.com website.

v 19.1
10/13/2020
Added support for FileMaker Pro 19.1, WinAutomation Jobs version is now v8. Since the EXE files aren't embedded in this stack anymore, this message is displayed when clicking the CheckWAJobs button.
The version numbering of this PDF matches the version of latest FileMaker release.

Architecture


FmPro Migrator WinAutomation Tester
My Documents
WinAutomation_Jobs
Name
_FmPro Migrator WA Jobs.exe
_FmPro Migrator WA Jobs.waj
Clipboard.txt
Completed.txt
Parameters.txt
Version.txt
C:/Users/Username/Documents/WinAutomation_Jobs/

Process Flow:

1) The FmPro Migrator WinAutomation Tester stack writes the Parameters.txt file and an optional Clipboard.txt file into the WinAutomation_Jobs directory and the Completed.txt file is deleted.

2) The FmPro Migrator WinAutomation Tester stack executes the compiled WinAutomation jobs application file (_FmPro Migrator WA Jobs.exe), and then enters a loop, waiting for completion of

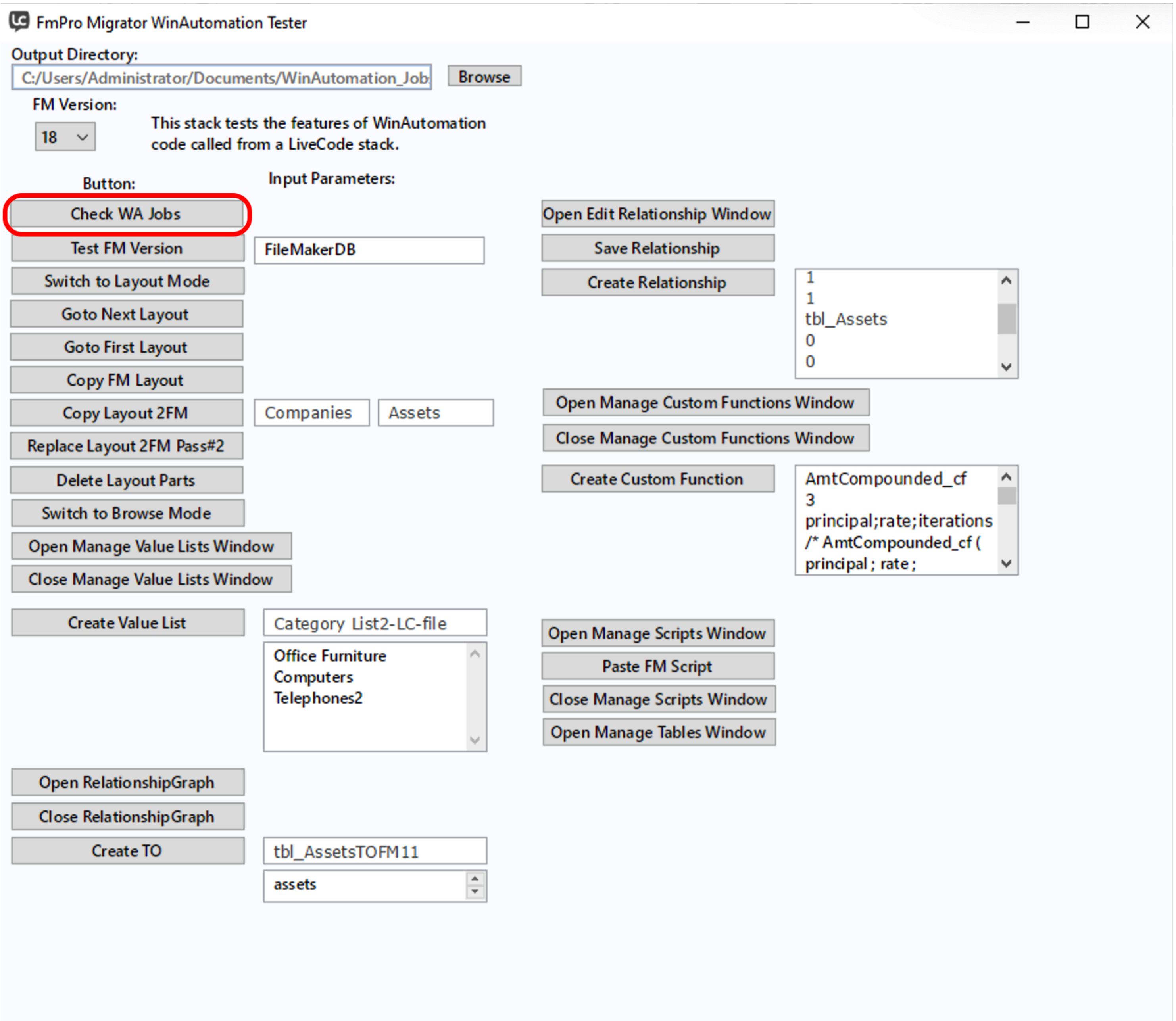
the job.

3) As soon as the WinAutomation jobs application runs, it opens the Parameters.txt file in order to determine which function to run.

4) Once the requested function has been completed, the Completed.txt file is written by the WinAutomation jobs application, and the application exits.

5) The The FmPro Migrator WinAutomation Tester stack exits the infinite loop based upon finding the existence of the Completed.txt file, and returns control back to the user.

Setup



FmPro Migrator WinAutomation Tester

Output Directory: C:/Users/Administrator/Documents/WinAutomation_Job

FM Version: 18

This stack tests the features of WinAutomation code called from a LiveCode stack.

Button: Input Parameters:

FileMakerDB

Companies Assets

Category List2-LC-file

Office Furniture
Computers
Telephones2

tbl_AssetsTOFM11

assets

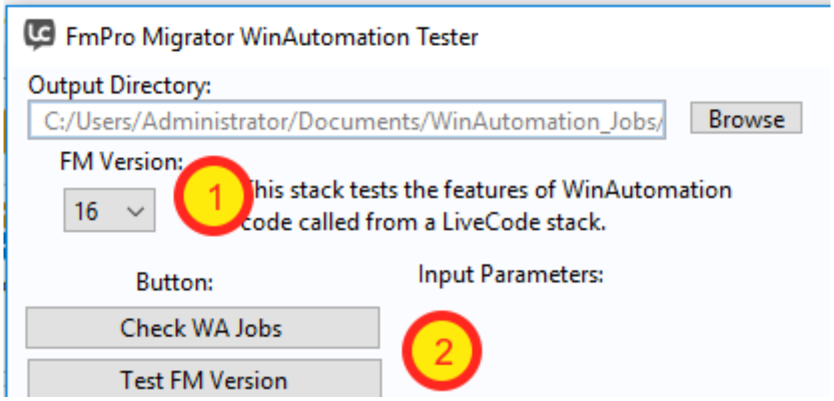
1
1
tbl_Assets
0
0

AmtCompounded_cf
3
principal;rate;iterations
/* AmtCompounded_cf (
principal ; rate ;

Clicking the Check WA Jobs button creates the WinAutomation_Jobs folder within the Documents directory. The _FmPro Migrator WA Jobs.exe and _FmPro Migrator WA Jobs.waj binary files are no longer written into the the WinAutomation_Jobs folder, since these files would have been written by FmPro Migrator. The Version.txt file is also written.

This task only needs to be done once.

Verifying FileMaker Pro Advanced Version


FmPro Migrator WinAutomation Tester

Output Directory: C:/Users/Administrator/Documents/WinAutomation_Jobs/

FM Version: 16

This stack tests the features of WinAutomation code called from a LiveCode stack.

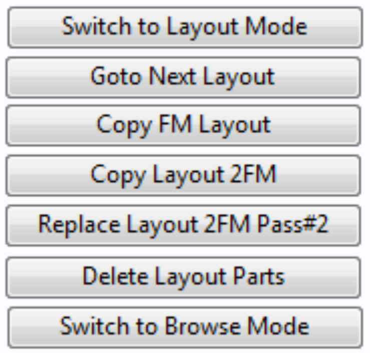
Input Parameters:

Launch the FileMaker Pro Advanced application. The FileMaker Pro Advanced application needs to already be running in order for any of these tasks to complete.

However it is important to verify that the correct version of FileMaker is running before running the automation jobs.

1) Select the version of the currently running FileMaker Pro application from the FM Version menu.
2) Click the Test FM Version button. If the versions match, then no error message will be displayed. Otherwise an error dialog is displayed.

Running Automation Tasks



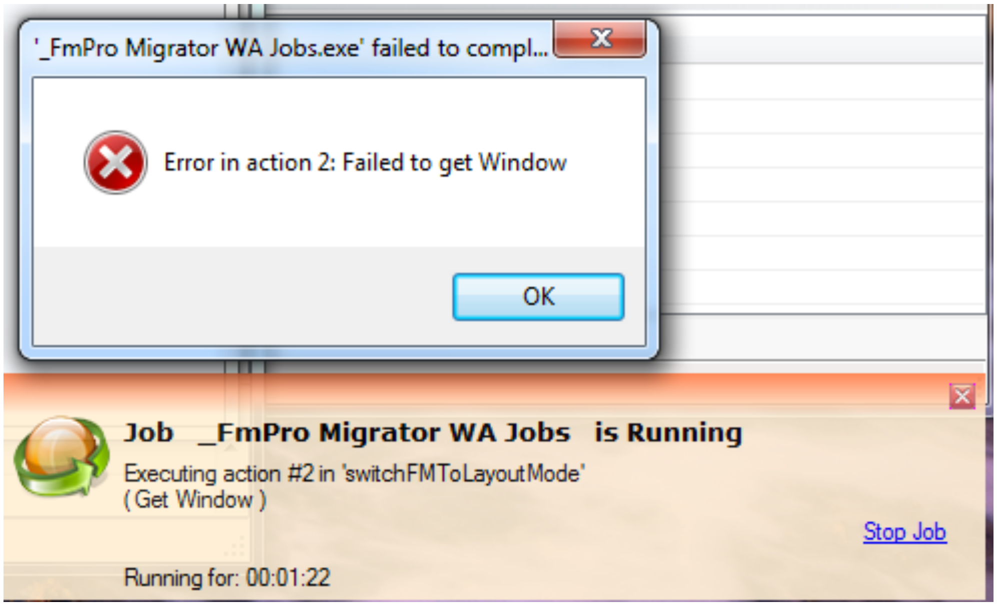
One of the simplest tasks to test is switching between Browse and Layout modes. These tasks require no setup prior to running.

Tasks like Copy Layout 2FM, Replace Layout 2FM Pass#2 - require that a FileMaker layout already be copied onto the clipboard when the button is clicked.

The Goto Next Layout task, requires that FileMaker be in Layout mode.

Tooltips are provided for task buttons which require a layout or script to be on the clipboard before running.

Troubleshooting the Compiled WinAutomation Application



'_FmPro Migrator WA Jobs.exe' failed to compl...

Error in action 2: Failed to get Window

Job _FmPro Migrator WA Jobs is Running

Executing action #2 in 'switchFMToLayoutMode'
(Get Window)

Running for: 00:01:22

The compiled WinAutomation Jobs file is compiled with the option to display Notifications and Error Message Dialogs, for troubleshooting purposes.
This error in the switchFMToLayoutMode function is due to FileMaker Pro Advanced not being found running at the time the job is running.

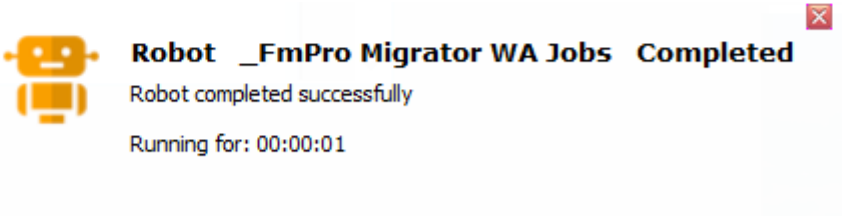
Troubleshooting WinAutomation GUI Automation [Windows]

FmPro Migrator includes a compiled WinAutomation job to provide GUI automation, at no cost. FmPro Migrator automatically writes this executable job file into a new WinAutomation_Jobs folder within the user's Documents directory, and then executes this file as needed.
Note: Starting with WinAutomation v6, these Jobs are now called Robots, and the icons in the windows look a little different compared to previous versions - but the functionality remains the same.

FileMaker Versions Supported - FM11, FM15+

On Windows, the WinAutomation jobs/robots are currently supported and tested with FileMaker Pro Advanced 11 and FileMaker Pro Advanced 15+.

Windows OS - Must be set to use English



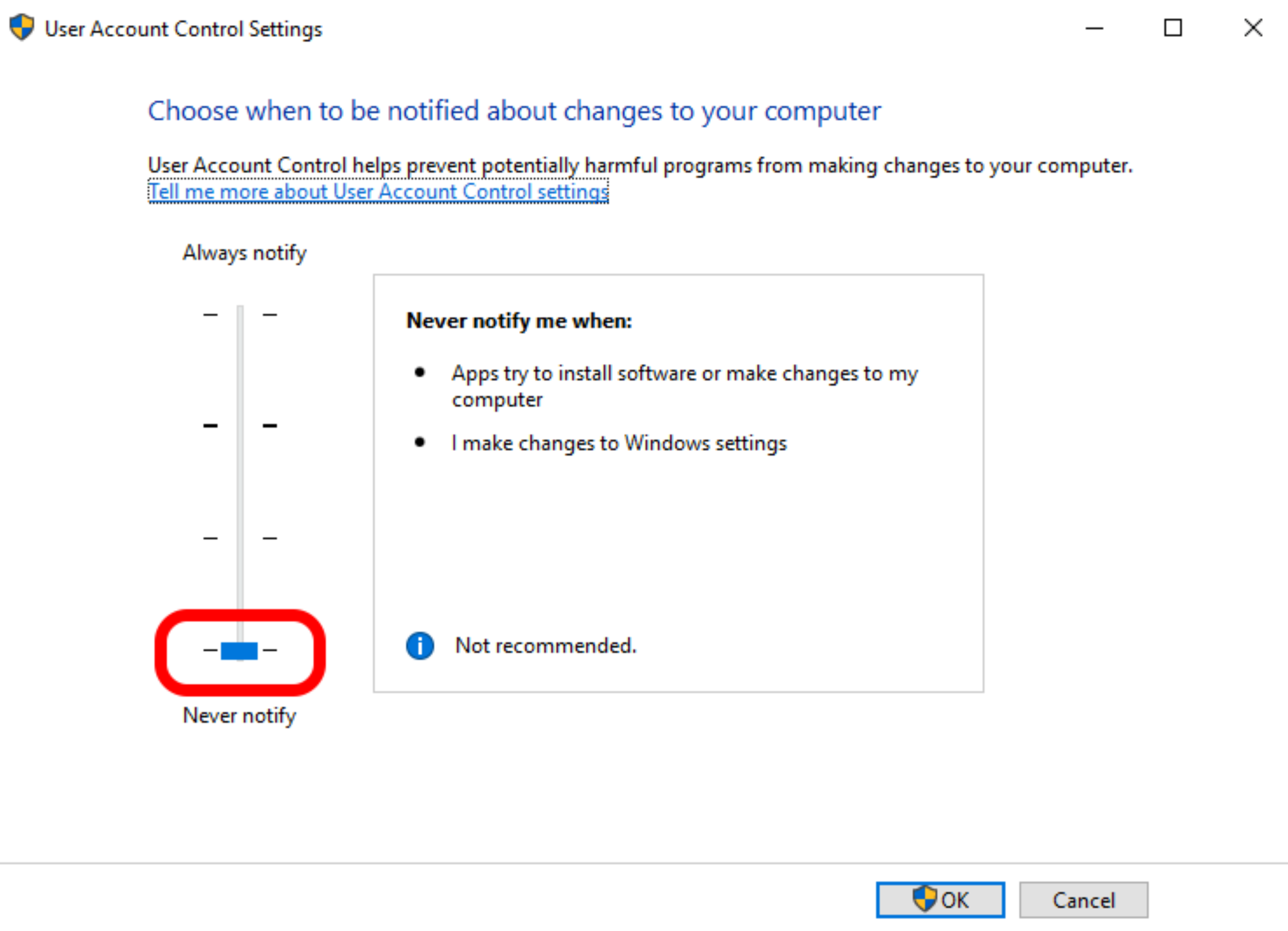
Robot _FmPro Migrator WA Jobs Completed

Robot completed successfully

Running for: 00:00:01

The WinAutomation Job/Robot actions always look for the English text of the controls & windows being used within FileMaker Pro Advanced. For this reason, the English language needs to be used for the Windows operating system.

Windows 10 User Account Control Settings



User Account Control Settings

Choose when to be notified about changes to your computer

User Account Control helps prevent potentially harmful programs from making changes to your computer.
Tell me more about User Account Control settings

Never notify me when:

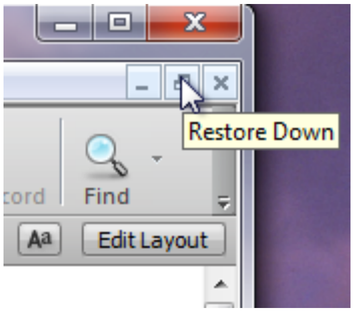
- Apps try to install software or make changes to my computer
- I make changes to Windows settings

Not recommended.

Windows 10 sometimes seems to be overly enthusiastic about notifying the user whenever an app is launched. It may sometimes be necessary to temporarily disable this feature during GUI automation tasks. FmPro Migrator runs the compiled WinAutomation Job/Robot file many times automatically during processing - and you don't want the EXE to be blocked waiting for the user to click to allow it to run every time.
Most of the time, this setting doesn't cause any problems, but if it does cause a problem it can be changed in the Control Panel:

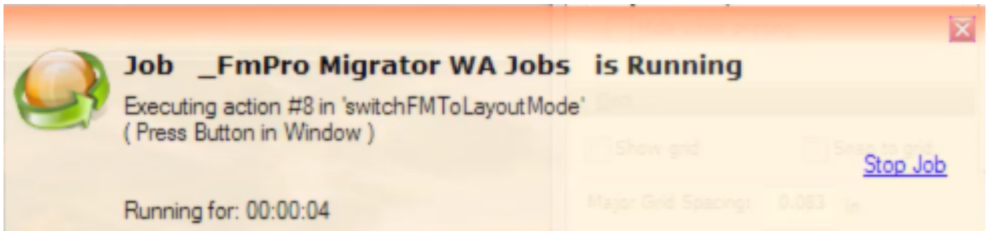
This setting is found in: Control Panel -> Security & Maintenance -> User Account Control Settings.

Maximized Window Problem



In order for the compiled WinAutomation job to find menu items within the FileMaker user interface, the database window should not be maximized.
If the window has been maximized, click the restore down button to solve this problem.

WinAutomation Notification Window



Job _FmPro Migrator WA Jobs is Running

Executing action #8 in 'switchFMToLayoutMode'
(Press Button in Window)

Running for: 00:00:04

The compiled WinAutomation job (or robot) file has been developed with notifications enabled to allow troubleshooting during the processing.
In this screenshot:

_FmPro Migrator WA Jobs - Is the name of the compiled job file.

switchFMToLayoutMode - Is the name of the function within the job file which is currently running.

action #8 - Is the number of the action being executed within this particular function.

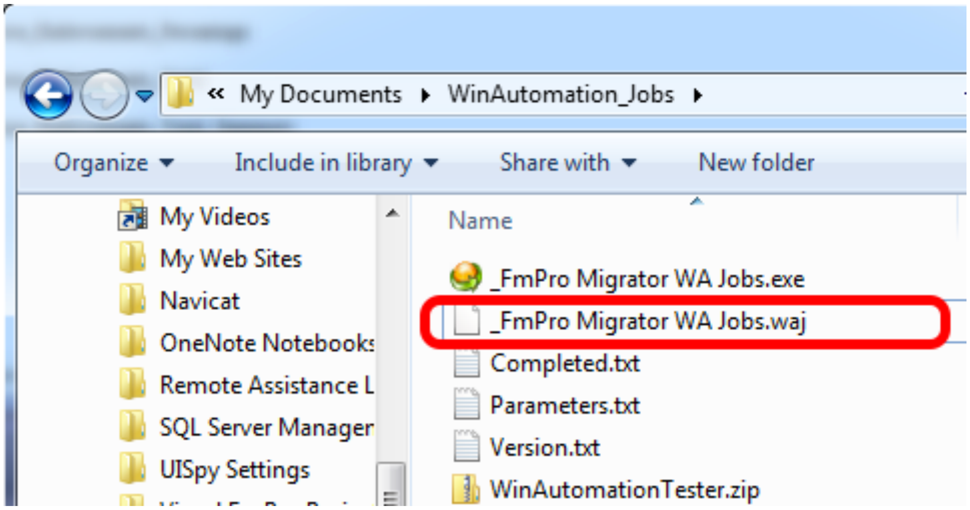
(Press Button in Window) - This is the name of the action being executed. In this example, pressing the OK button on the Specify Number window to goto layout #1.

WinAutomation - Wrong Job Executing

If you notice that the wrong jobs are executing, then it may be necessary to perform the following steps. This might occur if a job action fails during processing. There could also be multiple copies of the _WinAutomation_Jobs.exe application running.
1) Exit the FmPro Migrator application.

2) Delete the Parameters.txt and Clipboard.txt files from within the C:\My Documents\WinAutomation_Jobs folder.
3) Restart Windows.

Localization



The Actions in the compiled WinAutomation job file have been developed using the English version of FileMaker Pro. The WinAutomation Actions could be changed to support a different language. This change would need to be made for each button, window name reference and field specified within the functions of the job file.

The source code for the job file is written into the C\<username>\My Documents\WinAutomation_Jobs folder as a file named _FmPro Migrator WA Jobs.waj. This file is available under open source licensing, and modifications can be made by the user if necessary, for instance for updating the job file to work with localized versions of FileMaker.

After making changes in the WinAutomation Job Designer, save the changes, recompile and save the compiled job into the WinAutomation_Jobs folder using WinAutomation Professional. WinAutomation Professional is required in order to compile WinAutomation jobs into executable files.

WinAutomation Function Names

closeFMCustomFunctionsWindow - Closes the “Manage Custom Functions...” window by clicking the Ok button.

closeFMManageScriptsWindow - Closes the “Manage Scripts...” window by clicking on the close button, selects first script in list in preparation for creating the next script.

closeFMRelationshipGraphWindow - Closes the FileMaker Relationship Graph window by clicking the Ok button.

closeFMValueListsWindow - Closes the “Manage Value Lists...” window by clicking the Ok button.

copyFMLayout - Copies the current FileMaker layout into FmPro Migrator.

copyLayout2FM - Creates a new empty layout in FileMaker (Pass #1), sets layout name, sets TO Name for new layout, deletes layout header, footer parts, pastes the layout XML already on the clipboard, clicks Yes to extend objects dialog #1, #2.

createFMCustomFunction - Reads Custom Function creation parameters from the Clipboard. Creates a new FileMaker Custom Function, having the proper name, parameters and code, then saves it.

createFMCustomValueList - Creates a FileMaker custom value list and saves it. Field based value lists are also created, but the properties need to be set manually.

createFMRelationship - Reads relationship creation parameters from the Clipboard, storing info in 10 variables, creates a new relationship or a predicate for an existing relationship which is already being worked on.

createFMTO - Gets the TOName and Base Table Name from the Clipboard then creates a new FileMaker TO. Note: Setting the TO color is not supported.

deleteFMLayoutParts - Deletes the Header and Footer layout parts from a new empty layout, leaving the Body part.

getTOListPosition - Returns the position number of a TO name within a list of TOs. The returned value is used for selecting a specific line (TO name) within a popup menu. This is done with a numeric index to the TO name instead of passing in the actual TO name via keystrokes.

gotoNextFMLayout - Advances to the next FileMaker layout while in layout mode, by sending the control down arrow keystroke.

openFMCustomFunctionsWindow - Opens the “Manage Custom Functions” window using the File menu.

openFMEditRelationhipWindow - Opens Edit Relationship window by clicking the Add Relationship button, waits for window to open.

openFMManageScriptsWindow - Opens the “Manage Scripts...” window using the Control Shift S keystroke.

openFMManageTablesWindow - Opens “Manage Database” window, clicks on the Tables tab.

openFMRelationshipGraphWindow - Opens “Manage Database” window, clicks on the Relationship Graph tab.

openFMValueListsWindow - Opens the “Manage Value Lists...” window using the File menu."

pasteFMScript - Clicks the first row of the scripts displayed in the “Manage Scripts...” window. Pastes the current script XML on the clipboard using Control V keys.

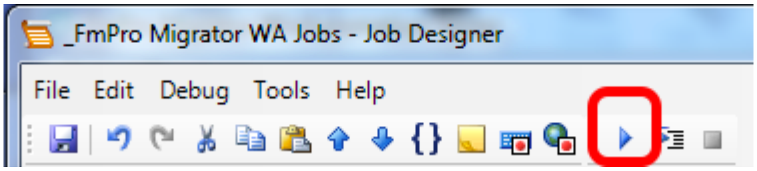
replaceLayout2FMPPass2 - Selects the contents of the current layout in Layout mode, deletes the contents, pastes the layout contents already on the clipboard. Used during Pass #2 processing during Table Consolidation projects.

saveFMRelationship - Clicks the Ok button of the Edit Relationship window to save the current relationship. If the rename TO dialog is displayed, clicks the Ok button to that dialog also.

switchFMToBrowseMode - Switches to Browse Mode using Control B keys.

switchFMToLayoutMode - Switches to Layout Mode, using Control L keys, advances to layout #1.

Troubleshooting WinAutomation Jobs within WinAutomation

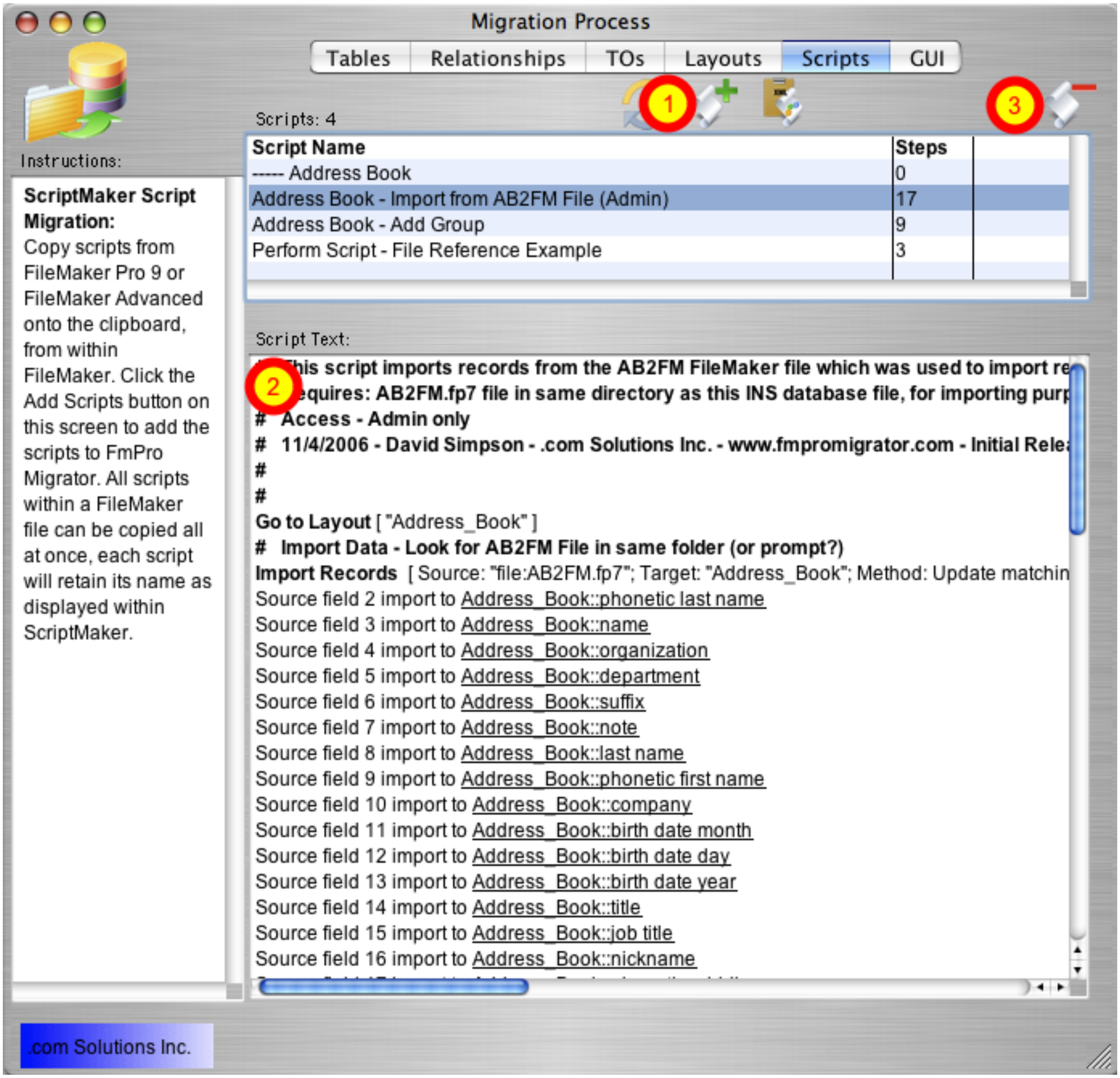


The _FmPro Migrator WA Jobs.waj file contains the source code for the automation job. Import this file into the WinAutomation Console to test, update or debug the jobs.

To run the jobs within the WinAutomation Console, first run the job using the FmPro Migrator WinAutomation Tester stack (documented in a separate chapter of this PDF), in order to setup the Parameters.txt file and the Clipboard.txt file.
Then click the Run button in the Job Designer. Breakpoints can be set as needed and variables can be inspected during debugging.

Copy Script Workspace Scripts

Scripts Tab



Scripts: 4

Instructions:

ScriptMaker Script Migration: Copy scripts from FileMaker Pro 9 or FileMaker Advanced onto the clipboard, from within FileMaker. Click the Add Scripts button on this screen to add the scripts to FmPro Migrator. All scripts within a FileMaker file can be copied all at once, each script will retain its name as displayed within ScriptMaker.

Script Name	Steps
----- Address Book	0
Address Book - Import from AB2FM File (Admin)	17
Address Book - Add Group	9
Perform Script - File Reference Example	3

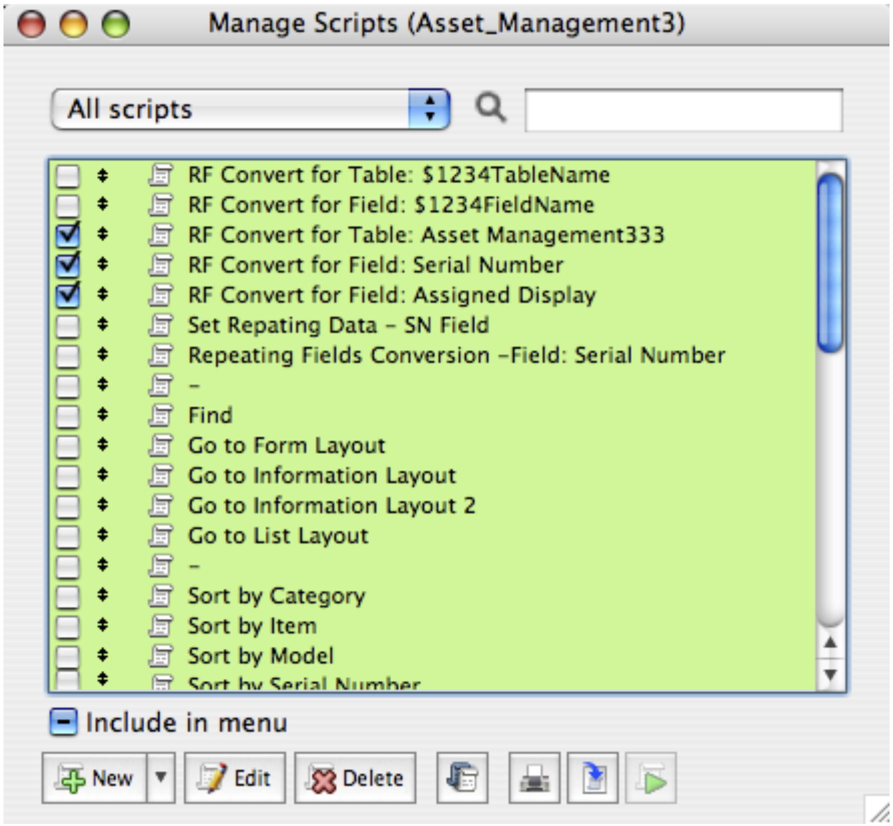
Script Text:

```
# This script imports records from the AB2FM FileMaker file which was used to import re
# Requires: AB2FM.fp7 file in same directory as this INS database file, for importing purp
# Access - Admin only
# 11/4/2006 - David Simpson - .com Solutions Inc. - www.fmpromigrator.com - Initial Relea
#
#
Go to Layout [ "Address_Book" ]
# Import Data - Look for AB2FM File in same folder (or prompt?)
Import Records [ Source: "file:AB2FM.fp7"; Target: "Address_Book"; Method: Update matchin
Source field 2 import to Address_Book::phonetic last name
Source field 3 import to Address_Book::name
Source field 4 import to Address_Book::organization
Source field 5 import to Address_Book::department
Source field 6 import to Address_Book::suffix
Source field 7 import to Address_Book::note
Source field 8 import to Address_Book::last name
Source field 9 import to Address_Book::phonetic first name
Source field 10 import to Address_Book::company
Source field 11 import to Address_Book::birth date month
Source field 12 import to Address_Book::birth date day
Source field 13 import to Address_Book::birth date year
Source field 14 import to Address_Book::title
Source field 15 import to Address_Book::job title
Source field 16 import to Address_Book::nickname
```

The Scripts tab provides an interface for managing FileMaker Script Workspace scripts during a migration project. Scripts can be captured [1] from FileMaker by copying them to the ClipBoard within the Script Workspace window within FileMaker Pro.

Scripts can also be viewed within FmPro Migrator by clicking each script and viewing the text version of the script within the Script Text field [2]. Duplicate or unneeded scripts can also be deleted by selecting the scripts and clicking the Delete button [3].

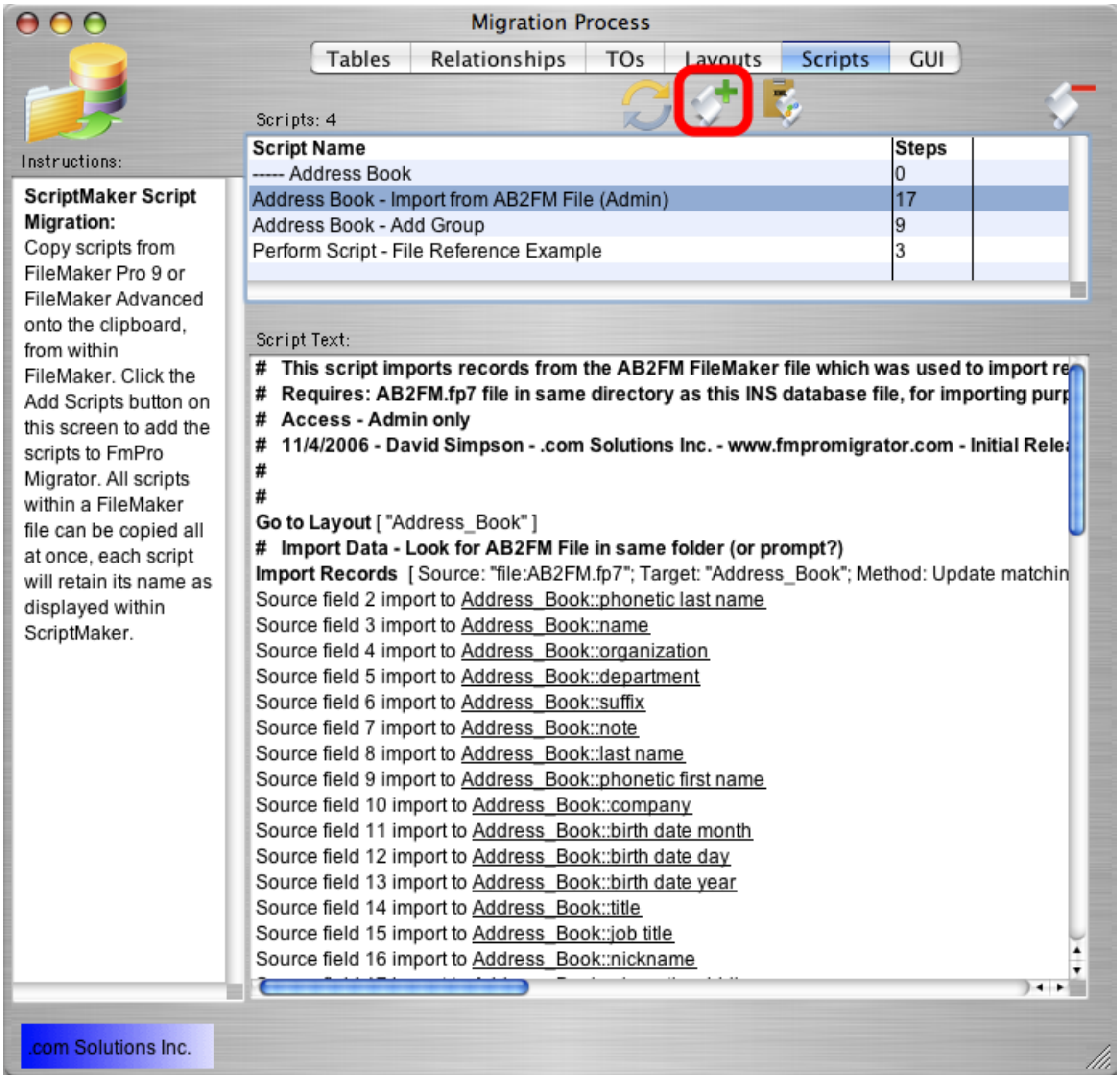
Copy Script Workspace Scripts From FileMaker to ClipBoard



□ RF Convert for Table: $1234TableName
□ RF Convert for Field: $1234FieldName
☑ RF Convert for Table: Asset Management333
☑ RF Convert for Field: Serial Number
☑ RF Convert for Field: Assigned Display
□ Set Repating Data - SN Field
□ Repeating Fields Conversion -Field: Serial Number
□ -
□ Find
□ Go to Form Layout
□ Go to Information Layout
□ Go to Information Layout 2
□ Go to List Layout
□ -
□ Sort by Category
□ Sort by Item
□ Sort by Model
□ Sort by Serial Number

Within FileMaker Advanced, open the Script Workspace window, select all of the scripts and select Copy from the Edit menu.

Paste Script Workspace Scripts From ClipBoard Into FmPro Migrator



ScriptMaker Script Migration: Copy scripts from FileMaker Pro 9 or FileMaker Advanced onto the clipboard, from within FileMaker. Click the Add Scripts button on this screen to add the scripts to FmPro Migrator. All scripts within a FileMaker file can be copied all at once, each script will retain its name as displayed within ScriptMaker.

Script Name	Steps
----- Address Book	0
Address Book - Import from AB2FM File (Admin)	17
Address Book - Add Group	9
Perform Script - File Reference Example	3

```
#  This script imports records from the AB2FM FileMaker file which was used to import re
#  Requires: AB2FM.fp7 file in same directory as this INS database file, for importing purp
#  Access - Admin only
#  11/4/2006 - David Simpson - .com Solutions Inc. - www.fmpromigrator.com - Initial Relea
#
#
Go to Layout [ "Address_Book" ]
#  Import Data - Look for AB2FM File in same folder (or prompt?)
Import Records  [ Source: "file:AB2FM.fp7"; Target: "Address_Book"; Method: Update matchin
Source field 2 import to Address_Book::phonetic last name
Source field 3 import to Address_Book::name
Source field 4 import to Address_Book::organization
Source field 5 import to Address_Book::department
Source field 6 import to Address_Book::suffix
Source field 7 import to Address_Book::note
Source field 8 import to Address_Book::last name
Source field 9 import to Address_Book::phonetic first name
Source field 10 import to Address_Book::company
Source field 11 import to Address_Book::birth date month
Source field 12 import to Address_Book::birth date day
Source field 13 import to Address_Book::birth date year
Source field 14 import to Address_Book::title
Source field 15 import to Address_Book::job title
Source field 16 import to Address_Book::nickname
```

Click the Add Scripts from ClipBoard button to paste the scripts into FmPro Migrator.

You will perform this task for every source database file which contains scripts to be migrated.

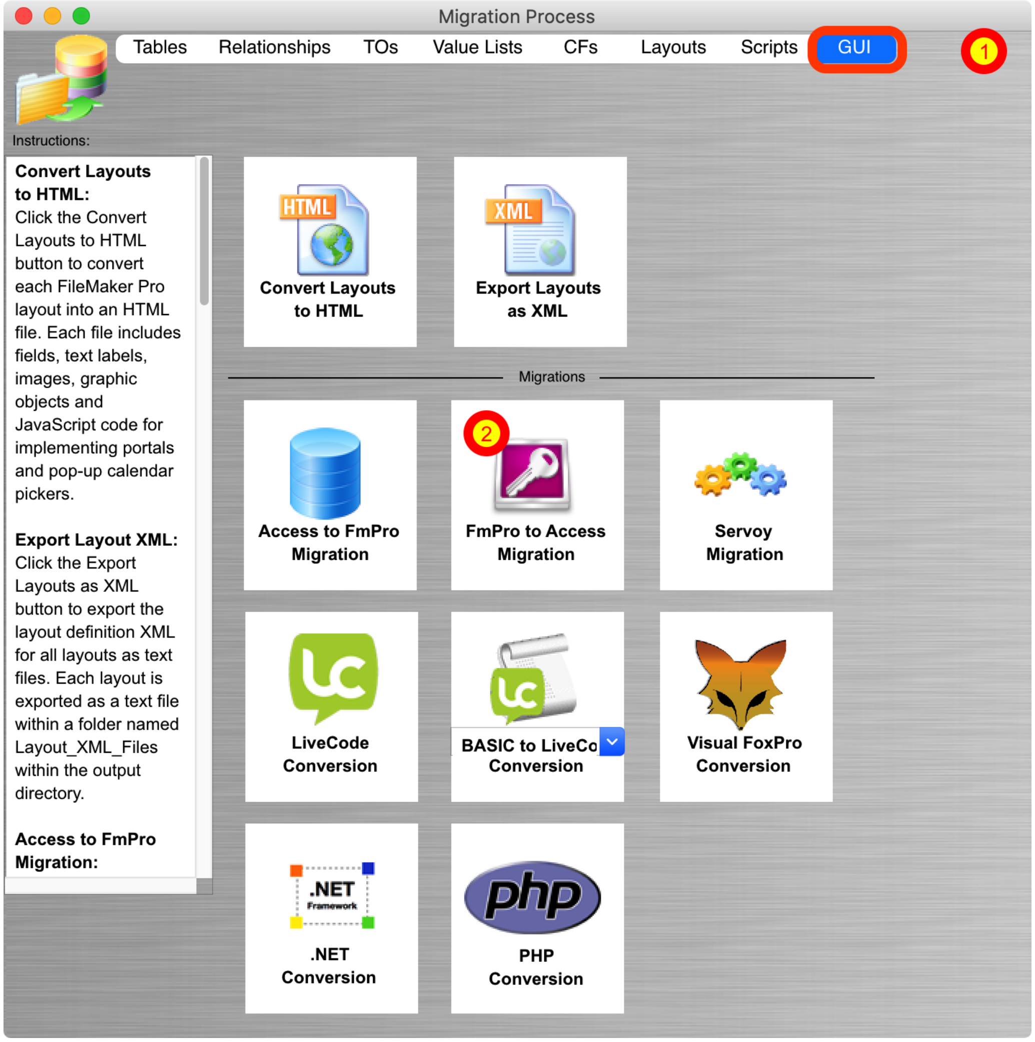
Open FmPro to Access Dialog

The previous steps have shown how to import the FileMaker database into FmPro Migrator on macOS and Windows. You may use either platform for importing the FileMaker database metadata into FmPro Migrator.

Once it is time to create the VBA scripts for the Access database, this task could be done on macOS or Windows and the scripts could be copied over to a Windows VM for running within the Access database.

It is also possible to perform all of the importing on macOS, and then copy the MigrationProcess.db3 FmPro Migrator project file over to a Windows VM where the Windows version of FmPro Migrator could be run.

Click FmPro to Access Migration Service Button



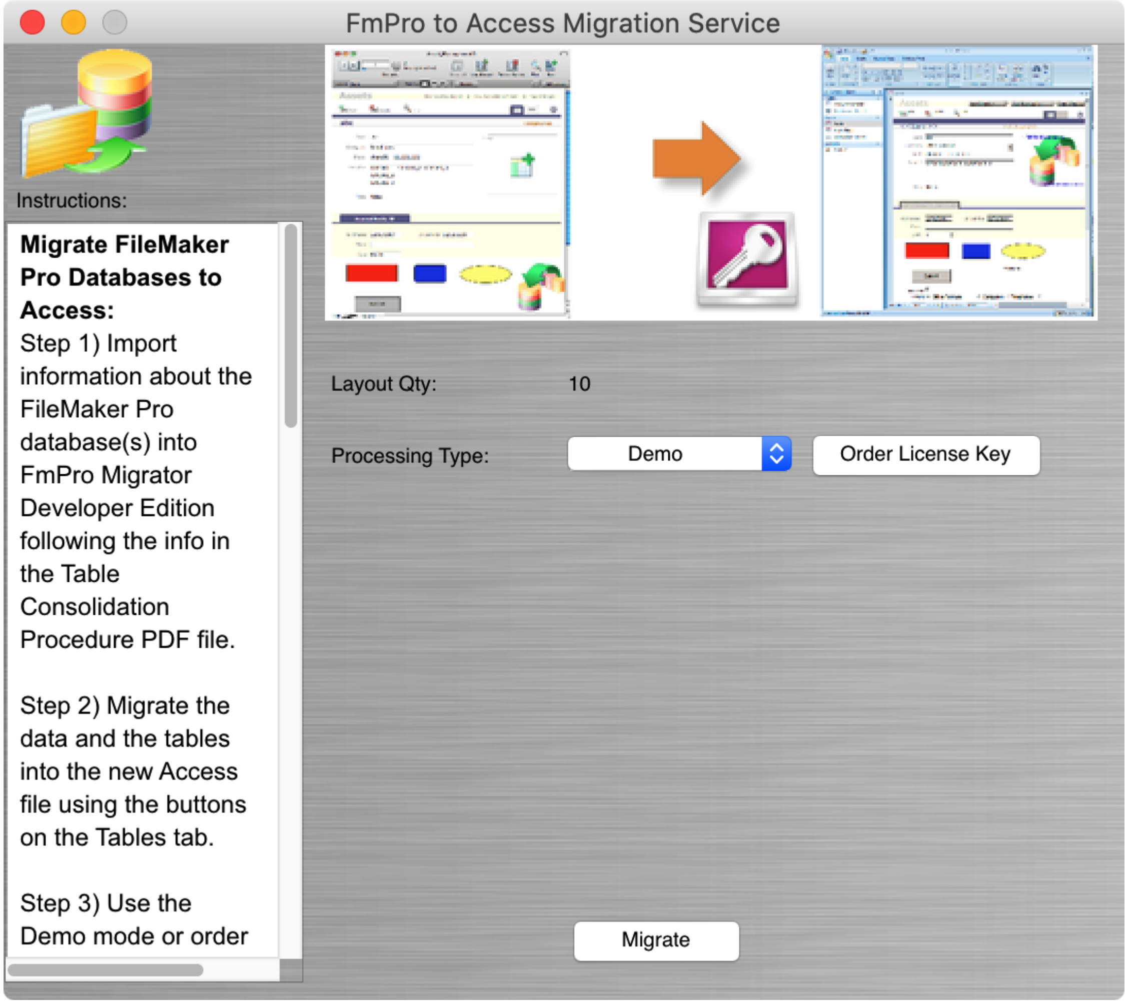
Convert Layouts to HTML: Click the Convert Layouts to HTML button to convert each FileMaker Pro layout into an HTML file. Each file includes fields, text labels, images, graphic objects and JavaScript code for implementing portals and pop-up calendar pickers.

Export Layout XML: Click the Export Layouts as XML button to export the layout definition XML for all layouts as text files. Each layout is exported as a text file within a folder named Layout_XML_Files within the output directory.

Access to FmPro Migration:

1) Click on the GUI folder tab within the FmPro Migrator Migration Process window.
2) Click on the FmPro to Access Migration button.

FmPro to Access - Demo Mode



Instructions:

Migrate FileMaker Pro Databases to Access:
Step 1) Import information about the FileMaker Pro database(s) into FmPro Migrator Developer Edition following the info in the Table Consolidation Procedure PDF file.

Step 2) Migrate the data and the tables into the new Access file using the buttons on the Tables tab.

Step 3) Use the Demo mode or order

Layout Qty: 10

Processing Type: Demo

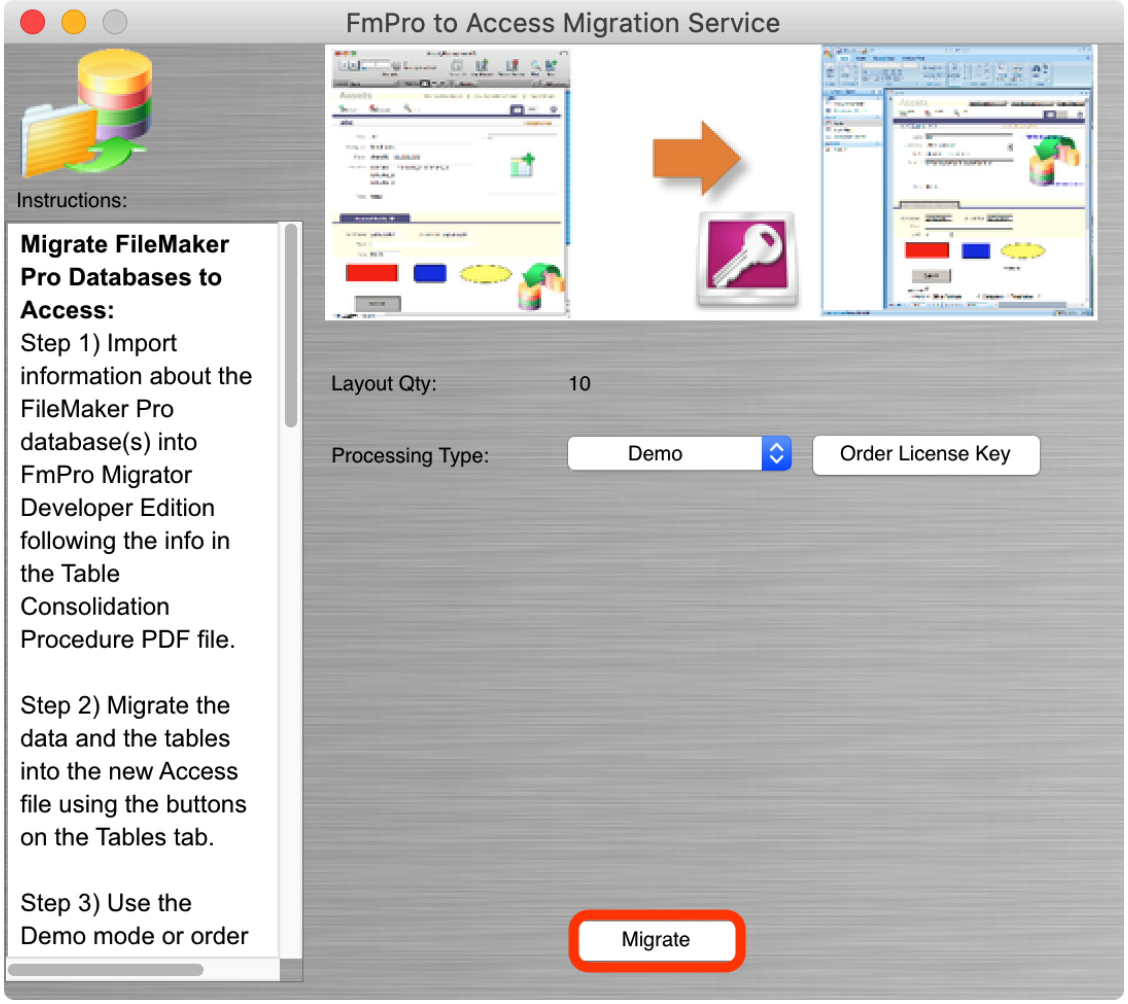
By default, FmPro Migrator sets the Processing Type for the FmPro to Access Migration Service to Demo mode. When running in Demo mode, 5 layouts and 5 scripts are processed - and all other objects including relationships, globals and value lists are migrated.

FmPro Migrator will convert each FileMaker Pro layout into a Visual Basic .bas file which will re-create the layout as a Form, SubForm or Report within the Access database file. Each FileMaker Pro script will be converted into a separate Visual Basic procedure within the _FmProConvertedScriptsVBA.bas file.

Using Demo Mode - FmPro to Access

FmPro Migrator includes a Demo mode for the FmPro to Access Migration Service. Demo mode enables Access developers to fully test the migration capabilities of the migration service with a limited number of layouts and scripts.
Demo mode also enables Access developers to quickly create "Proof of Concept" conversion projects for prospective clients who need to migrate their existing FileMaker Pro database solutions.

Click Migrate Button



FmPro to Access Migration Service

Instructions:

Migrate FileMaker Pro Databases to Access:
Step 1) Import information about the FileMaker Pro database(s) into FmPro Migrator Developer Edition following the info in the Table Consolidation Procedure PDF file.

Step 2) Migrate the data and the tables into the new Access file using the buttons on the Tables tab.

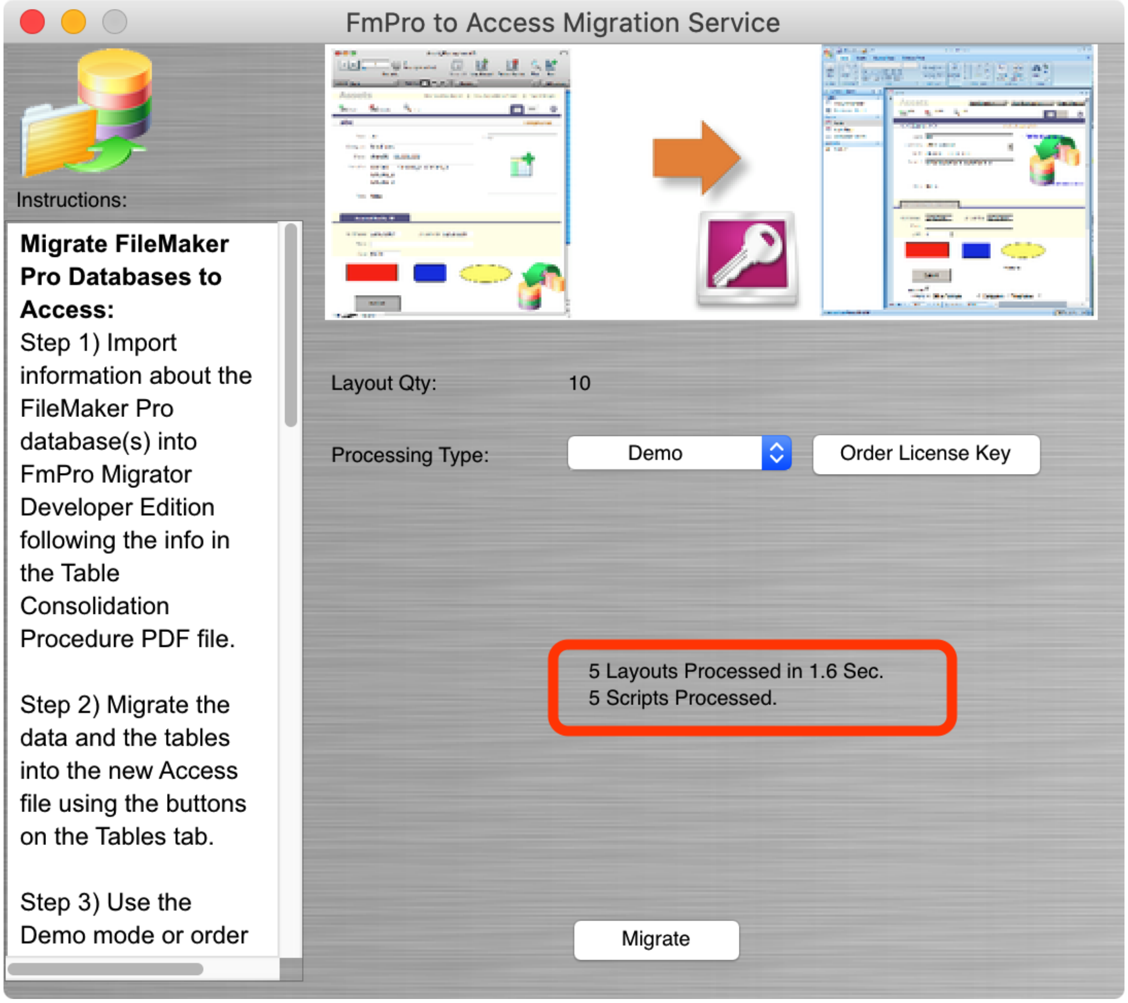
Step 3) Use the Demo mode or order

Layout Qty: 10

Processing Type: Demo

Click the Migrate button to create the Visual Basic files to create each Form, SubForm and Report in the Access database. A progress dialog will be displayed to show the progress as layouts and scripts are processed.

Migration Statistics



Instructions:

Migrate FileMaker Pro Databases to Access:
Step 1) Import information about the FileMaker Pro database(s) into FmPro Migrator Developer Edition following the info in the Table Consolidation Procedure PDF file.

Step 2) Migrate the data and the tables into the new Access file using the buttons on the Tables tab.

Step 3) Use the Demo mode or order

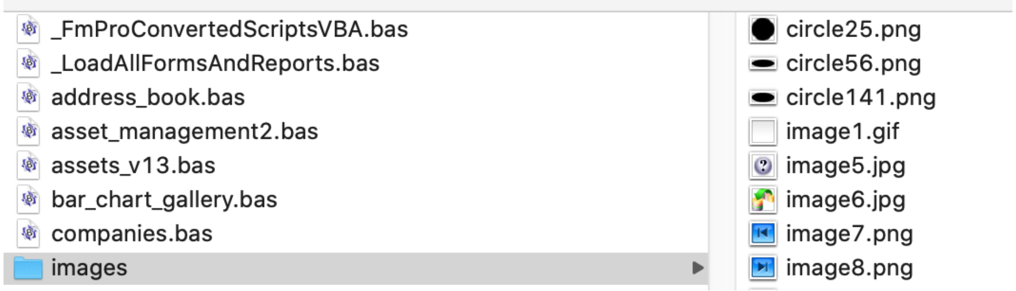
Layout Qty: 10

Processing Type: Demo

5 Layouts Processed in 1.6 Sec.
5 Scripts Processed.

After the migration has been completed, the processing statistics will be displayed above the Migrate button. FmPro Migrator will typically process layouts in less than one second per layout. As seen in this screenshot the layouts and and scripts and value lists were processed in less than 2 seconds.

Review Generated Files



_FmProConvertedScriptsVBA.bas
_LoadAllFormsAndReports.bas
address_book.bas
asset_management2.bas
assets_v13.bas
bar_chart_gallery.bas
companies.bas
images

circle25.png
circle56.png
circle141.png
image1.gif
image5.jpg
image6.jpg
image7.png
image8.png

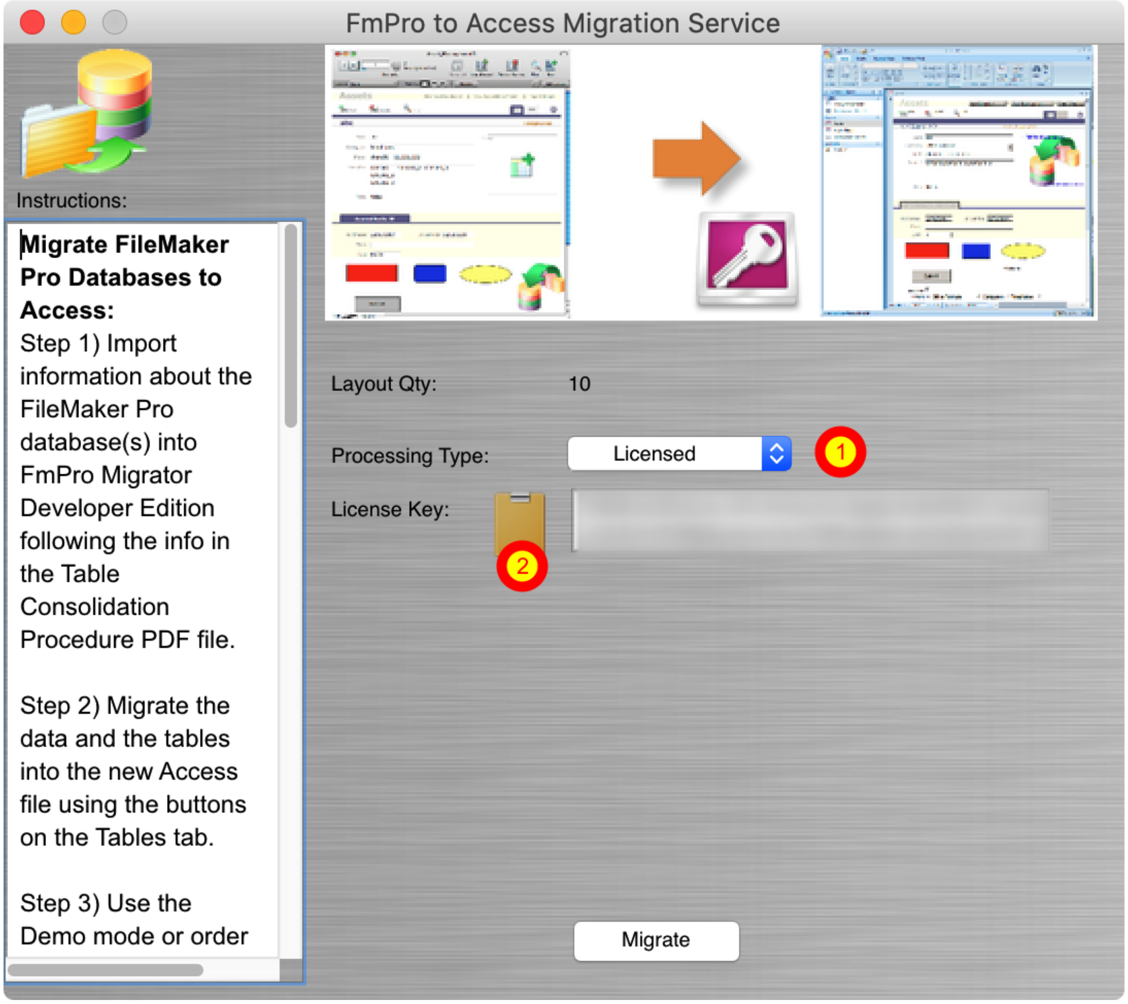
The generated Visual Basic and image files are written into a directory named AccessDBFiles within the FmPro Migrator output directory.

Using Licensed Mode - FmPro to Access

Using the FmPro to Access Migration Service in Demo mode, limits processing tasks to 5 layouts and 5 scripts. Ordering a license key lifts these operating limits for this service. Using the License Key allows for the processing of an unlimited number of database files and scripts for the purchased layout quantity during the duration of the license key.

Clicking on the Order License Key button in Demo mode will open the FmPro Migrator website order page.

Entering a License Key



Migrate FileMaker Pro Databases to Access:
Step 1) Import information about the FileMaker Pro database(s) into FmPro Migrator Developer Edition following the info in the Table Consolidation Procedure PDF file.

Step 2) Migrate the data and the tables into the new Access file using the buttons on the Tables tab.

Step 3) Use the Demo mode or order

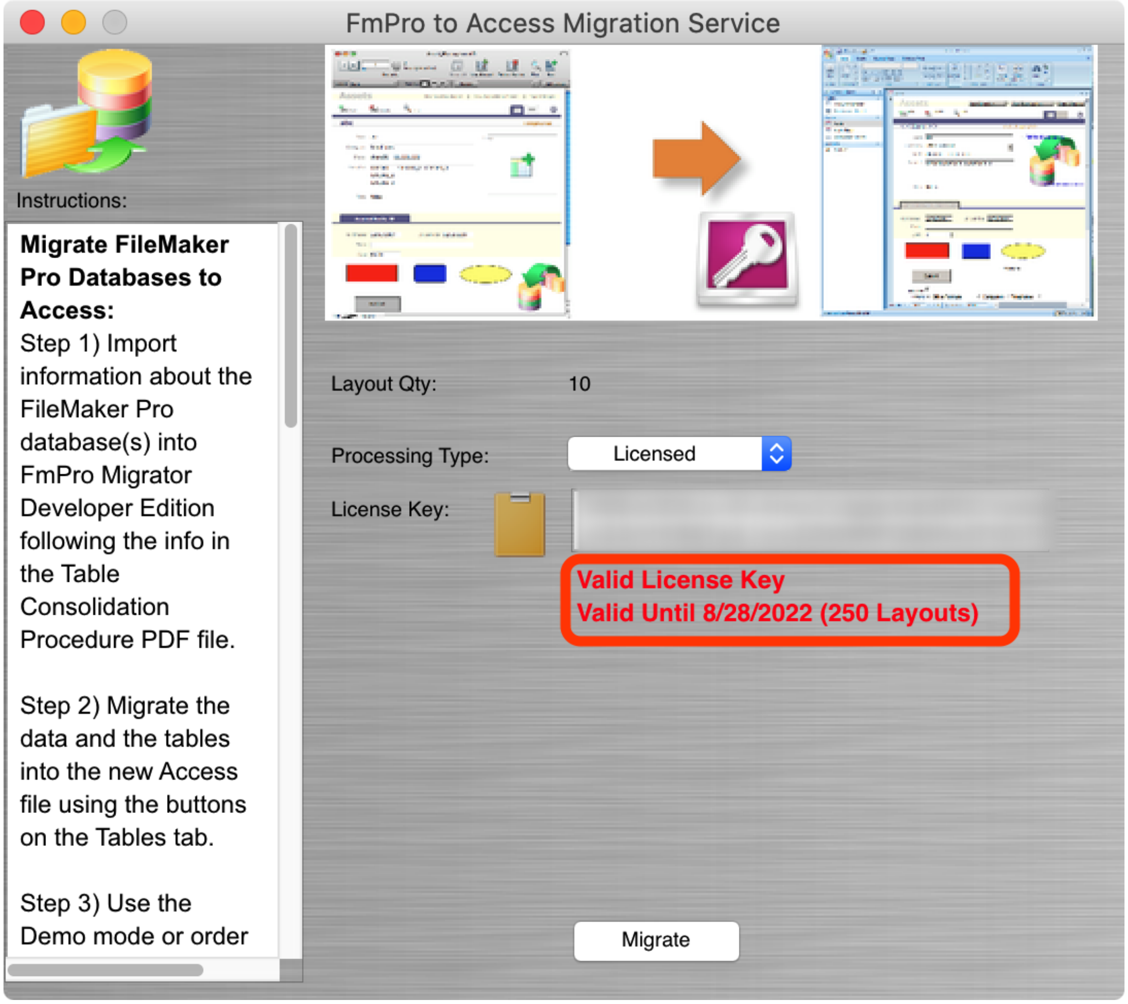
Layout Qty: 10

Processing Type: Licensed

License Key:

(1) Select Licensed from the Processing Type menu.
(2) Copy the License Key from the order confirmation email. Click the Clipboard icon in the FmPro to Access window.

Validating the License Key



FmPro to Access Migration Service

Instructions:

Migrate FileMaker Pro Databases to Access:
Step 1) Import information about the FileMaker Pro database(s) into FmPro Migrator Developer Edition following the info in the Table Consolidation Procedure PDF file.

Step 2) Migrate the data and the tables into the new Access file using the buttons on the Tables tab.

Step 3) Use the Demo mode or order

Layout Qty: 10

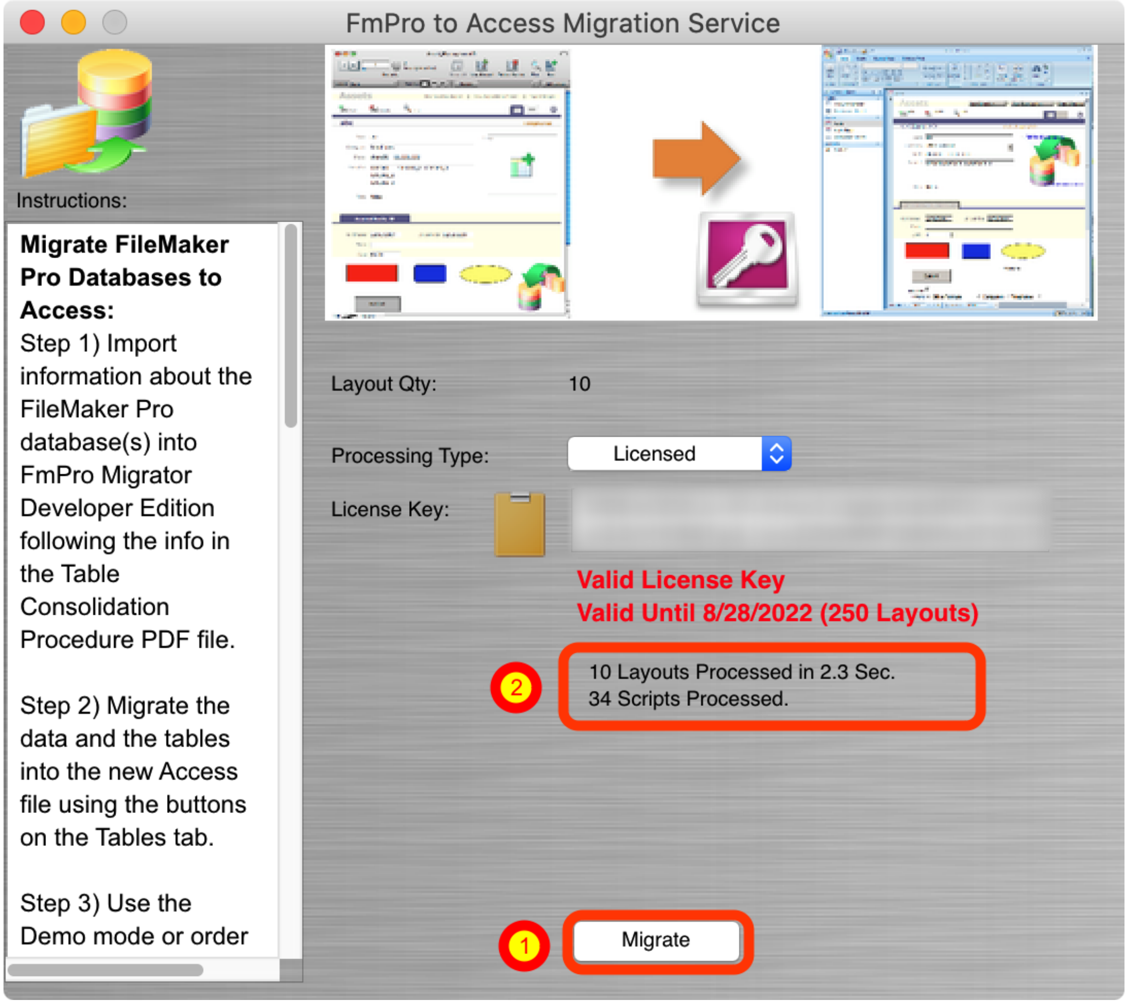
Processing Type: Licensed

License Key:

Valid License Key
Valid Until 8/28/2022 (250 Layouts)

FmPro Migrator copies the license key from the clipboard, validates the key with the license key server via the internet and then displays the layout quantity and expiration date for the key.

Migration Statistics - Licensed Mode



Instructions:

Migrate FileMaker Pro Databases to Access:
Step 1) Import information about the FileMaker Pro database(s) into FmPro Migrator Developer Edition following the info in the Table Consolidation Procedure PDF file.

Step 2) Migrate the data and the tables into the new Access file using the buttons on the Tables tab.

Step 3) Use the Demo mode or order

Layout Qty: 10

Processing Type: Licensed

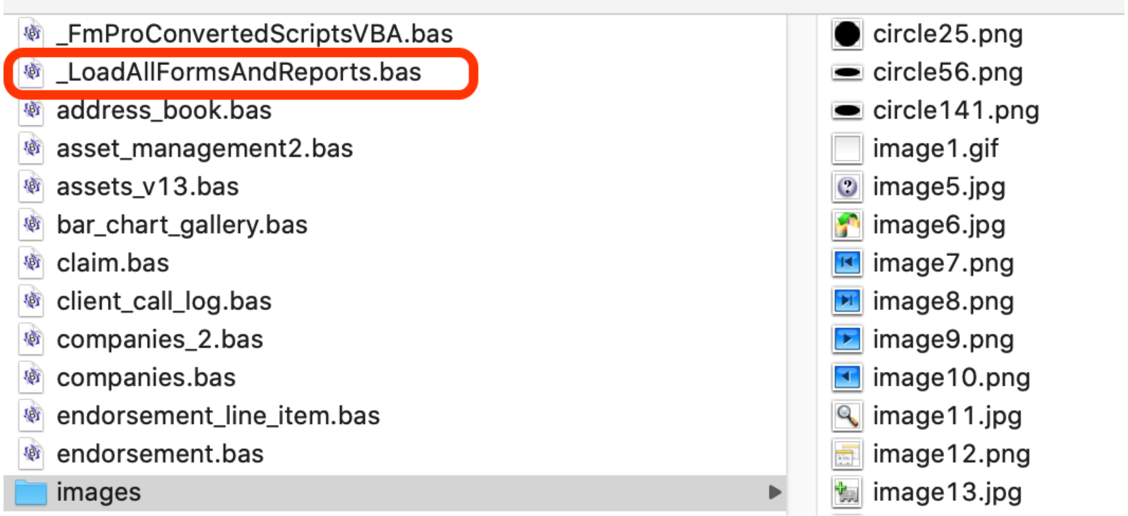
License Key:

Valid License Key
Valid Until 8/28/2022 (250 Layouts)

10 Layouts Processed in 2.3 Sec.
34 Scripts Processed.

(1) Clicking the Migrate button with the license key enables the processing of up to the number of layouts as specified in the license key and an unlimited number of scripts, with (2) the resulting processing statistics displayed below the license key field.

Generated Files



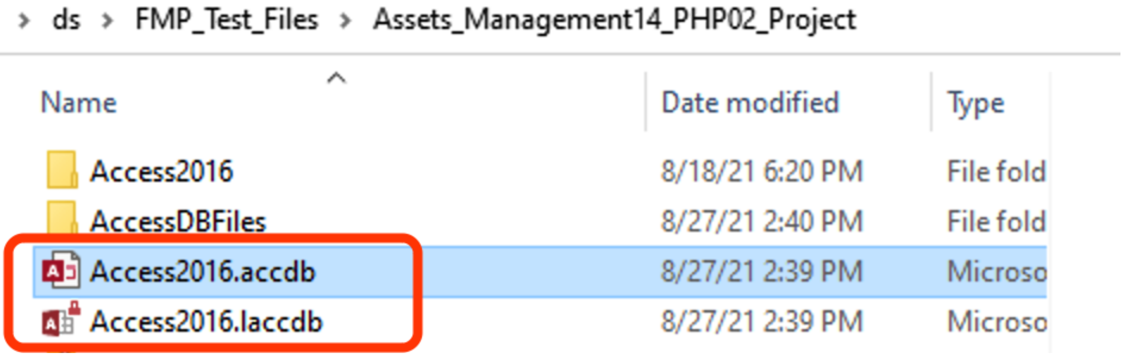
_FmProConvertedScriptsVBA.bas
_LoadAllFormsAndReports.bas
address_book.bas
asset_management2.bas
assets_v13.bas
bar_chart_gallery.bas
claim.bas
client_call_log.bas
companies_2.bas
companies.bas
endorsement_line_item.bas
endorsement.bas
images

circle25.png
circle56.png
circle141.png
image1.gif
image5.jpg
image6.jpg
image7.png
image8.png
image9.png
image10.png
image11.jpg
image12.png
image13.jpg

The generated Visual Basic files will be found within the AccessDBFiles folder within the output directory.

The _LoadAllFormsAndReports.bas file will load and run each of the 10 Visual Basic files in order to re-create each of the layouts as new forms, subforms and reports within the Access database.

Close FmPro Migrator

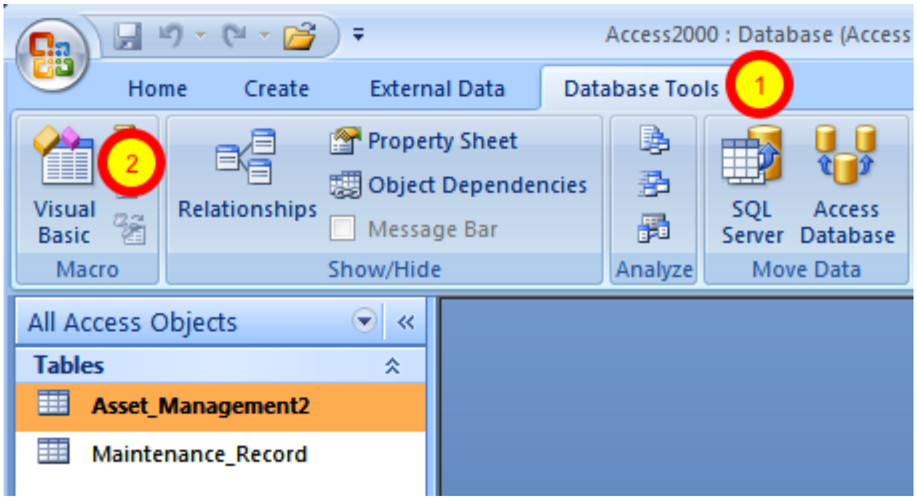


› ds › FMP_Test_Files › Assets_Management14_PHP02_Project

Name	Date modified	Type
Access2016	8/18/21 6:20 PM	File fold
AccessDBFiles	8/27/21 2:40 PM	File fold
Access2016.accdb	8/27/21 2:39 PM	Microso
Access2016.laccdb	8/27/21 2:39 PM	Microso

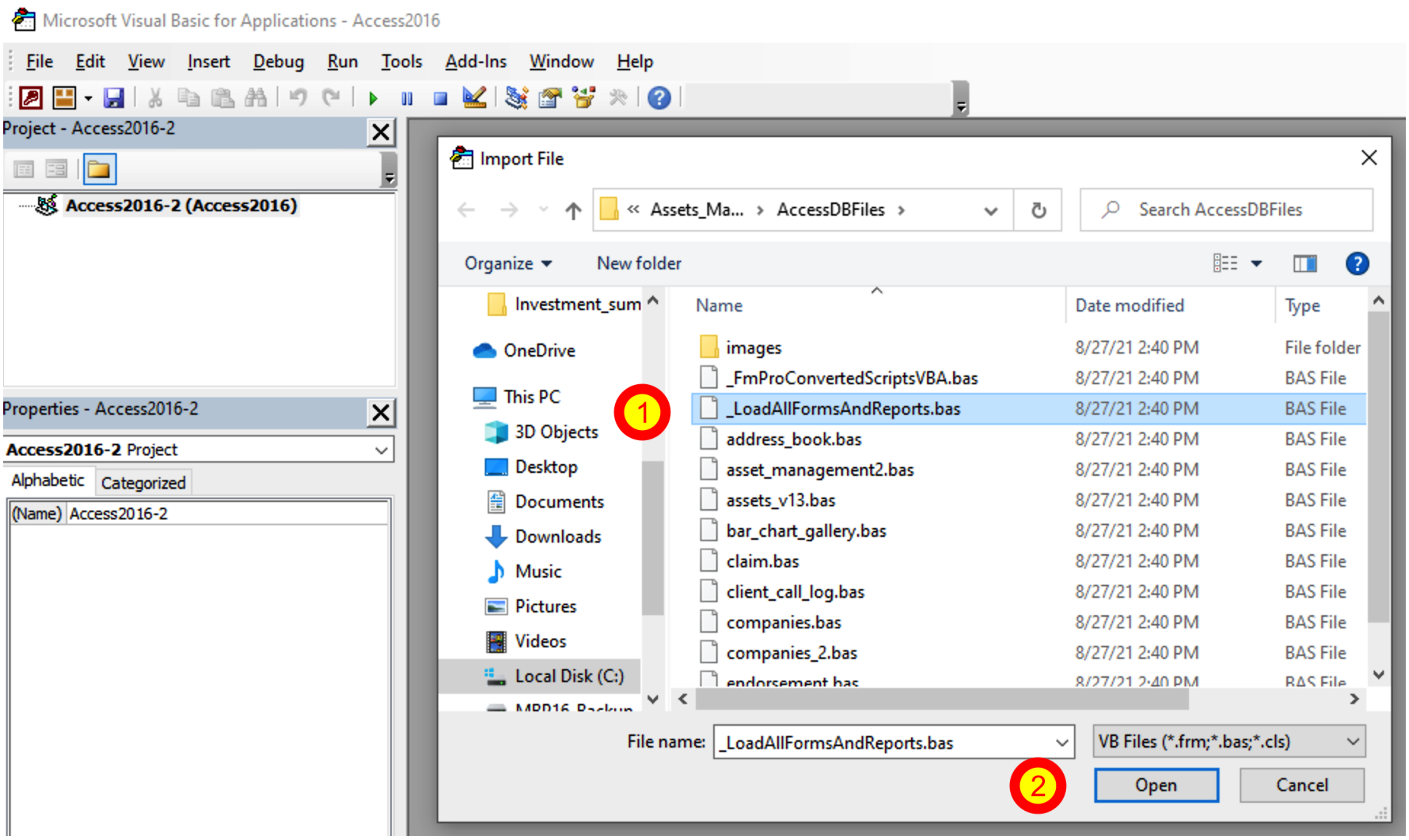
After the Visual Basic files have been created, close FmPro Migrator, in order to release the lock on the Access2016.mdb file held by the ODBC driver. The lock file is shown as the Access2016.laccdb file in the screenshot shown here. This lock is held on the file when the tables are created and data is transferred to the file. Closing FmPro Migrator releases this lock on the file so it can be opened within Microsoft Access.

Open the Access2016.accdb file



Open the Access2016.accdb file within Microsoft Access. This file is an Access 2007 formatted file, but it can be opened with any later version of Access.
(1) Click the Database Tools tab, (2) click on the Visual Basic icon to open the Visual Basic editor.

Import File... _LoadAllFormsAndReports.bas



(Name) Access2016-2

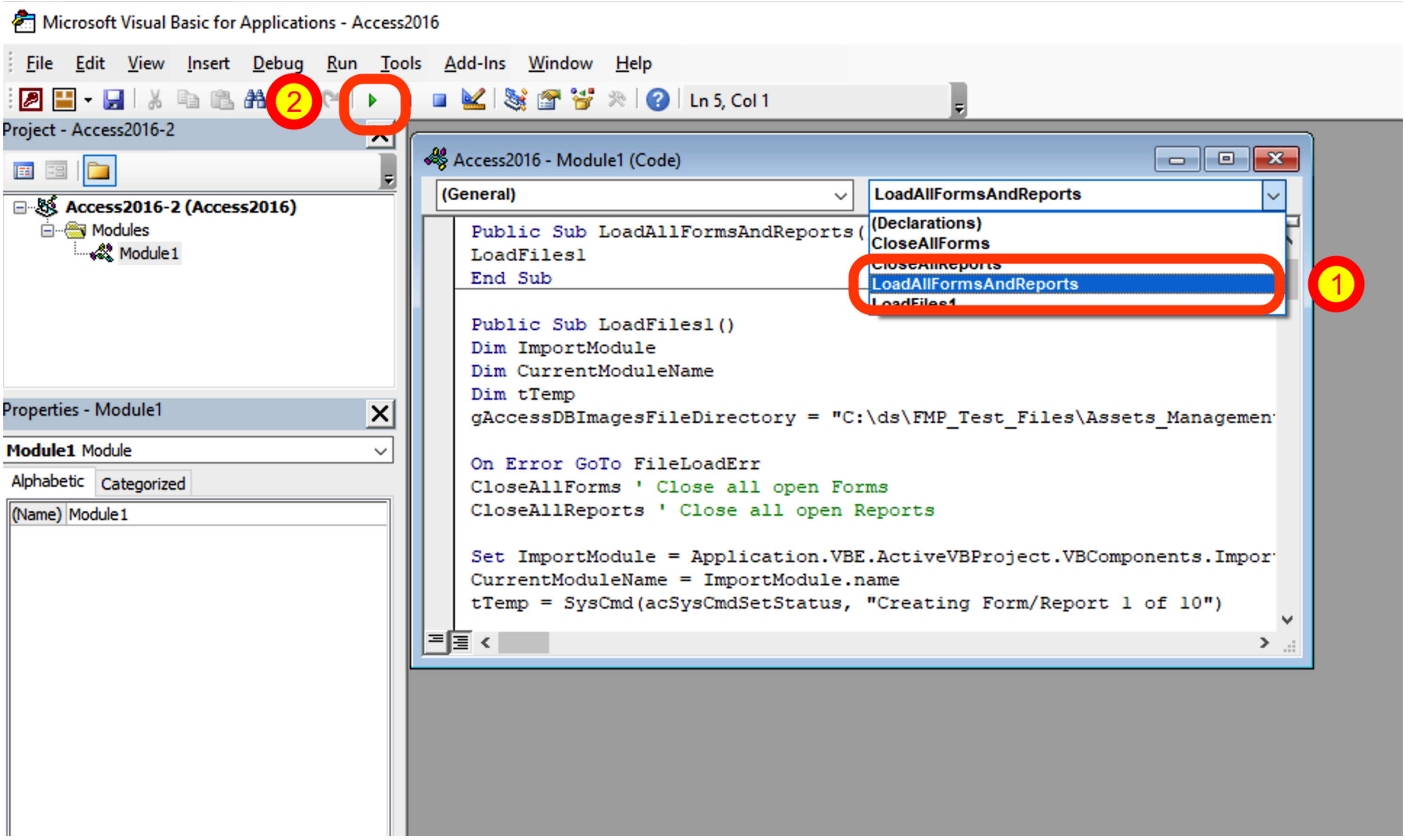
Name	Date modified	Type
images	8/27/21 2:40 PM	File folder
_FmProConvertedScriptsVBA.bas	8/27/21 2:40 PM	BAS File
_LoadAllFormsAndReports.bas	8/27/21 2:40 PM	BAS File
address_book.bas	8/27/21 2:40 PM	BAS File
asset_management2.bas	8/27/21 2:40 PM	BAS File
assets_v13.bas	8/27/21 2:40 PM	BAS File
bar_chart_gallery.bas	8/27/21 2:40 PM	BAS File
claim.bas	8/27/21 2:40 PM	BAS File
client_call_log.bas	8/27/21 2:40 PM	BAS File
companies.bas	8/27/21 2:40 PM	BAS File
companies_2.bas	8/27/21 2:40 PM	BAS File
endorsement.bas	8/27/21 2:40 PM	BAS File

File name: _LoadAllFormsAndReports.bas

VB Files (*.frm;*.bas;*.cls)

From within the Visual Basic editor window, select Import File..., and (1) select the _LoadAllFormsAndReports.bas file, (2) click the Open button.

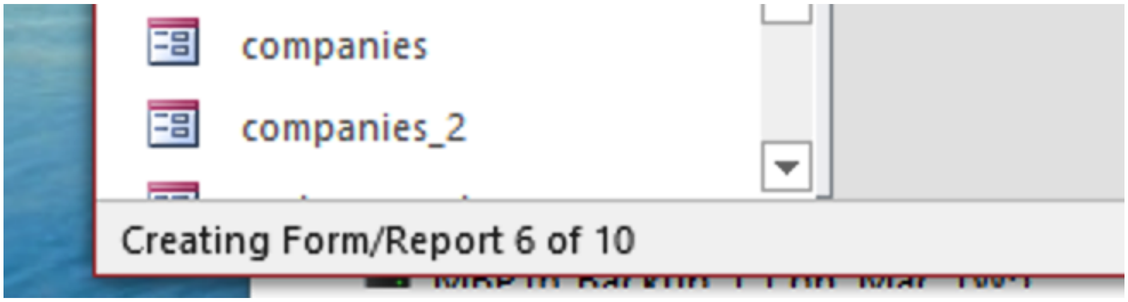
Run LoadAllFormsAndReports Procedure



```
Public Sub LoadAllFormsAndReports(
LoadFiles1
End Sub

Public Sub LoadFiles1()
Dim ImportModule
Dim CurrentModuleName
Dim tTemp
gAccessDBImagesFileDirectory = "C:\ds\FMP_Test_Files\Assets_Managemen

On Error GoTo FileLoadErr
CloseAllForms ' Close all open Forms
CloseAllReports ' Close all open Reports

Set ImportModule = Application.VBE.ActiveVBProject.VBComponents.Impor
CurrentModuleName = ImportModule.name
tTemp = SysCmd(acSysCmdSetStatus, "Creating Form/Report 1 of 10")
```

(1) Select the procedure named LoadAllFormsAndReports from the procedure menu, then click on the Run button.

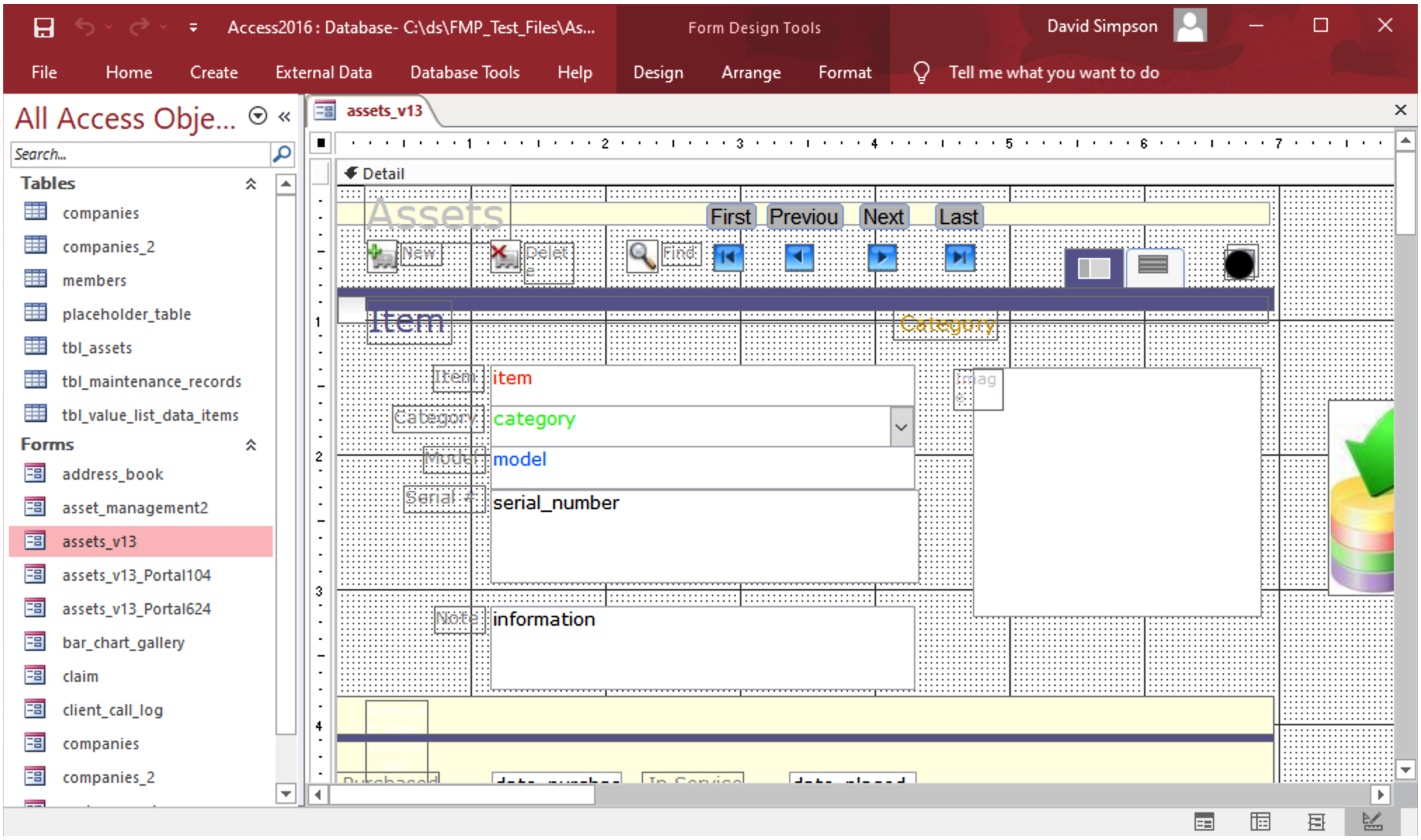
Progress Display



Creating Form/Report 6 of 10

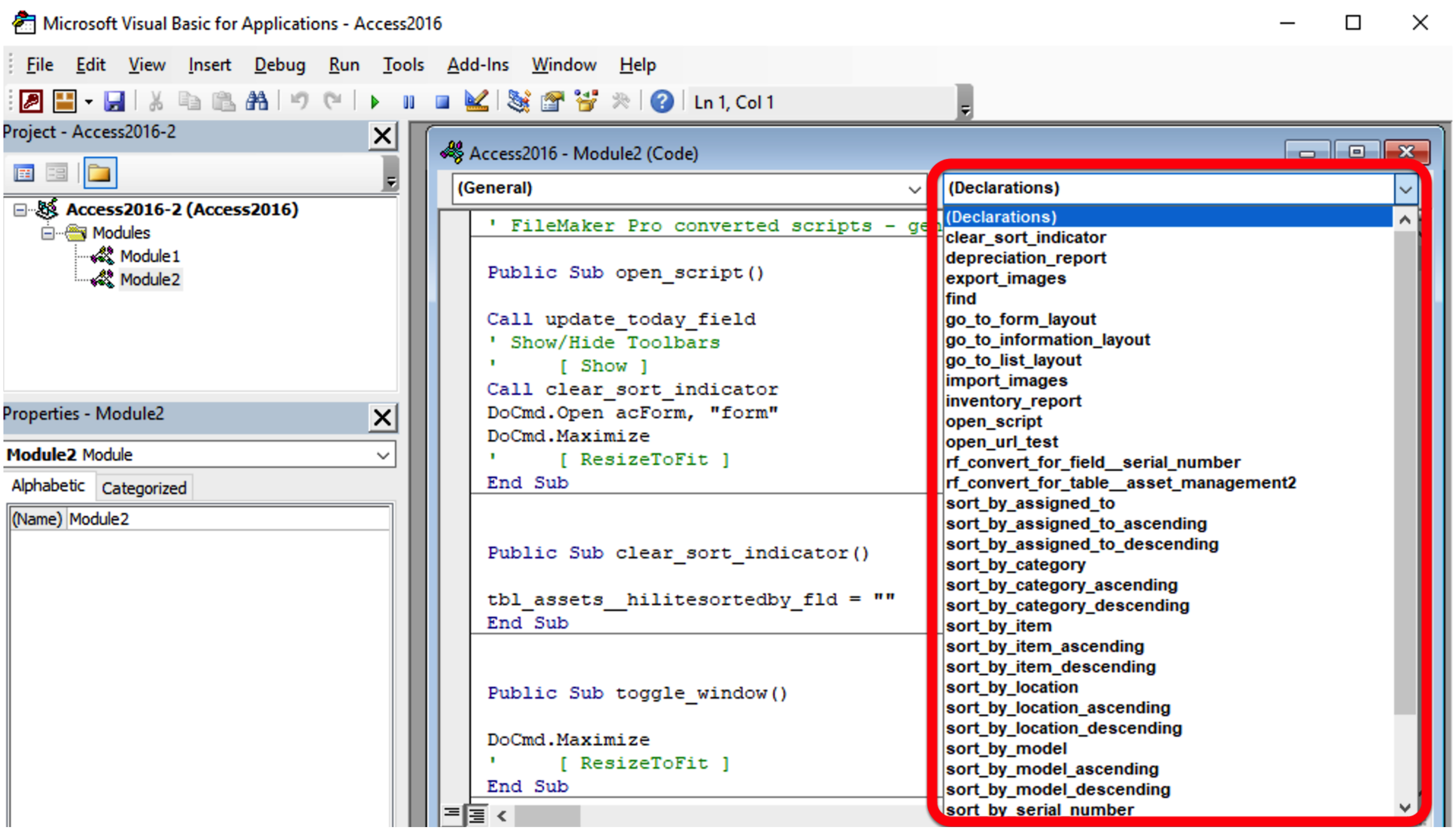
As the Forms, SubForms and Reports are being created, a progress message will be displayed in the lower left corner of the main Access database window.

Viewing Created Forms and Reports



Click on any of the Forms, SubForms or Reports shown in the selection window. The form will be displayed, having the same fields, text labels, images and graphics as the original layout in the FileMaker Pro database.

Open _FmProConvertedScriptsVBA.bas File



```
' FileMaker Pro converted scripts - ge

Public Sub open_script()

Call update_today_field
' Show/Hide Toolbars
'       [ Show ]
Call clear_sort_indicator
DoCmd.Open acForm, "form"
DoCmd.Maximize
'       [ ResizeToFit ]
End Sub


Public Sub clear_sort_indicator()

tbl_assets__hilitesortedby_fld = ""
End Sub


Public Sub toggle_window()

DoCmd.Maximize
'       [ ResizeToFit ]
End Sub
```

(Declarations)
clear_sort_indicator
depreciation_report
export_images
find
go_to_form_layout
go_to_information_layout
go_to_list_layout
import_images
inventory_report
open_script
open_url_test
rf_convert_for_field__serial_number
rf_convert_for_table__asset_management2
sort_by_assigned_to
sort_by_assigned_to_ascending
sort_by_assigned_to_descending
sort_by_category
sort_by_category_ascending
sort_by_category_descending
sort_by_item
sort_by_item_ascending
sort_by_item_descending
sort_by_location
sort_by_location_ascending
sort_by_location_descending
sort_by_model
sort_by_model_ascending
sort_by_model_descending
sort_by_serial_number

To edit the converted FileMaker Pro scripts, open the _FmProConvertedScriptsVBA.bas file within the AccessDBFiles directory. These converted scripts will require further modification and are intended to be used as a template for the development of new Visual Basic code which will be used to replace the original FileMaker Pro script functionality.

In this screenshot, the Procedure drop down menu has been clicked, showing the list of scripts which have been converted into VBA Sub procedures.

Troubleshooting

There are several errors which could occur during the creation of the Forms and Reports.

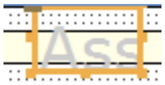
Procedure too large:
Microsoft Access has a 64K limit for procedures which can run within the database. Each of the procedures created by FmPro Migrator are designed to be smaller than this limit, by breaking up the creation of the forms and reports into procedures handling no more than 30 objects each. Complex layouts could potentially exceed this limit, and would require manual modification before they could be run.

Image sizing issues:

Different versions of Access may resize embedded form images differently. Access 2007 and higher versions more closely conform to the image sizing features built into FileMaker Pro. Therefore images may appear to be cut off by displaying only the upper left corner of the image when displayed in older Access database versions. If the converted database must run in Access 2000, manual changes may be required to the sizes embedded form images.

Post-Migration Tasks

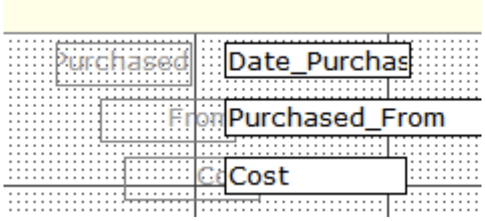
Re-size Text Labels



Text labels created using Visual Basic do not always expand wide or tall enough to view the entire text contained within the label. This situation seems to occur most often with text labels created at the larger text font sizes.

Note: This problem doesn't seem to occur very much anymore, since the ctlLabel.SizeToFit property has been added to the VBA text label creation code.

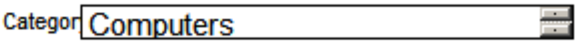
Re-position Text Labels



Some text labels may be mis-aligned from their original positions. In this screenshot, the "From" and "Cost" text labels need to be manually moved left with their right edges right aligned.

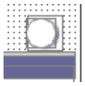
Text labels may also need to be manually configured with Font Weight and Underline styles.

Radio Button Sets



Text based checkbox sets on FileMaker Pro layouts are converted to multi-line selectable comboBox controls on Access Forms/Reports. These controls will often need to be resized to display taller in the vertical dimension so that the options can be observed and clicked by the user.

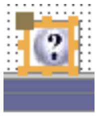
Overlapping Image Objects - Initial View



Some images are designed to overlap each other in order to create a visual effect on FileMaker Pro layouts. However there is no way to determine or set the correct top to bottom stacking level of the objects when creating the new Form/Report. The circle graphic is supposed to sit under the "?"

graphic, but they are reversed.

Overlapping Image Objects - Updated View



After selecting the white circle object, and pressing the Send to Back button on the Arrange toolbar, the image becomes visible as intended on the original FileMaker Pro layout.

Configure DBPix Active-X Control

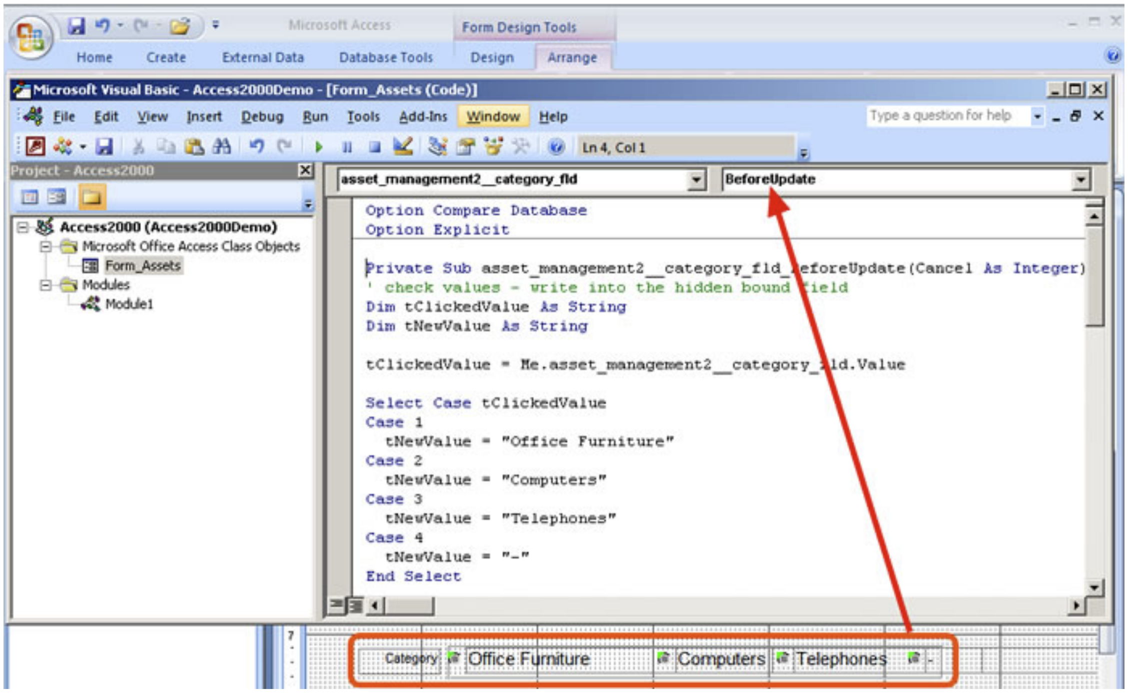
FileMaker Pro container fields are often used to store images for display on FileMaker Pro layouts. Container field image data is not directly converted into Access OLE Object fields for display within Access Forms/Reports. However you might have BLOB columns in SQL Server which are viewed via the Linked Tables feature in Access thru an ODBC link.

If you have trouble displaying those images within Access, you might consider using the DBPix control from Ammara.com. This plug-in is still available, and recent updates have been made available. A placeholder for the Ammara DBPix control is put onto the generated Form/Report for OLE Object fields. This placeholder object will need to be manually deleted and re-created to display the data in the OLE Object field.

Related Fields

Fields for a related table displayed on a Form/Report will require manual configuration of the field's control source.

Radio Button Form Event Code



```
Option Compare Database
Option Explicit

Private Sub asset_management2__category_fld_BeforeUpdate(Cancel As Integer)
' check values - write into the hidden bound field
Dim tClickedValue As String
Dim tNewValue As String

tClickedValue = Me.asset_management2__category_fld.Value

Select Case tClickedValue
Case 1
  tNewValue = "Office Furniture"
Case 2
  tNewValue = "Computers"
Case 3
  tNewValue = "Telephones"
Case 4
  tNewValue = "-"
End Select
```

FileMaker Pro radio buttons utilize text based value lists which are not directly compatible with Access radio button sets. The migrated data from the original FileMaker Pro database will also be in text form instead of the numeric values typically used for radio buttons within Microsoft Access.

FmPro Migrator generates a hidden field bound to the underlying database field along with Form BeforeEvent and Form Current Event code to translate the text based values back and forth between the radio buttons and the underlying database table. The Form Event code is automatically generated by FmPro Migrator but is commented out and will need to be manually pasted into each Form or Report where radio button sets are used.