

.com Solutions Inc.



FmPro Migrator - FileMaker to Access Migration Procedure

FmPro Migrator - FileMaker to Access Migration Procedure

1 Overview

- 1.1 Overview - FileMaker to Access Conversion 5
- 1.2 Licensing FmPro Migrator 6

2 Pre-Migration Preparation

- 2.1 Pre-Migration Preparation 11
- 2.2 Pre-Migration Task - Delete Unstored Calc Fields 20
- 2.3 Office 365 Compatibility 26

3 ODBC Driver Installation

- 3.1 Install FileMaker ODBC Driver - macOS (FileMaker 11+) 32
- 3.2 Install FileMaker ODBC Driver - Windows - FileMaker 11+ 48
- 3.3 Create Access ODBC DSN - Windows 60

4 Step 1

- 4.1 Step 1 - Get Info - FileMaker to Access 65

5 Step 2

- 5.1 Step 2 - Get Fieldsize 74

6 Step 3

- 6.1 Step 3 - Create Table - FileMaker to Access 78

7 Step 4

- 7.1 Step 4 - Transfer Data 83

8	Step 5	
8.1	Step 5 - Convert Data (Repeating Fields)	92
9	Step 6	
9.1	Step 6 - Create Table (Repeating Fields)	98
10	Step 7	
10.1	Step 7 - Transfer Data (Repeating Fields)	100
11	Migrating Relationships	
11.1	Migrating Relationships	103
12	Image Export	
12.1	Image Transfer to Microsoft Access	108
13	Migrating Layouts & Scripts to Access	
13.1	Importing FileMaker Layouts	115
13.2	Troubleshooting AppleScript GUI Automation [macOS]	125
13.3	About the FmPro Migrator WinAutomation Tester [Windows]	131
13.4	Troubleshooting WinAutomation GUI Automation [Windows]	136
13.5	Copy Script Workspace Scripts	142
13.6	Using Licensed Mode - FmPro to Access	145
13.7	Building Forms & Reports in Access	146
14	Post-Migration Tasks	
14.1	Post-Migration Tasks - FmPro to Access	154
14.2	Using the FmPro Code Conversion Workbench - FmPro to Access Conversions	157

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 7

4/14/2024

[Revision Notes: Updated for FmPro Migrator Platinum Edition 11.01.

Added new Licensing FmPro Migrator info, updated screenshots Added info about the Code Conversion Workbench script conversion feature.]

Licensing FmPro Migrator

As of FmPro Migrator v11, all FmPro Migrator downloads are fully functional with features unlocked via a single license key.

This section of the manual shows how to enter the license key to unlock the features within FmPro Migrator.

FmPro Migrator 11.01

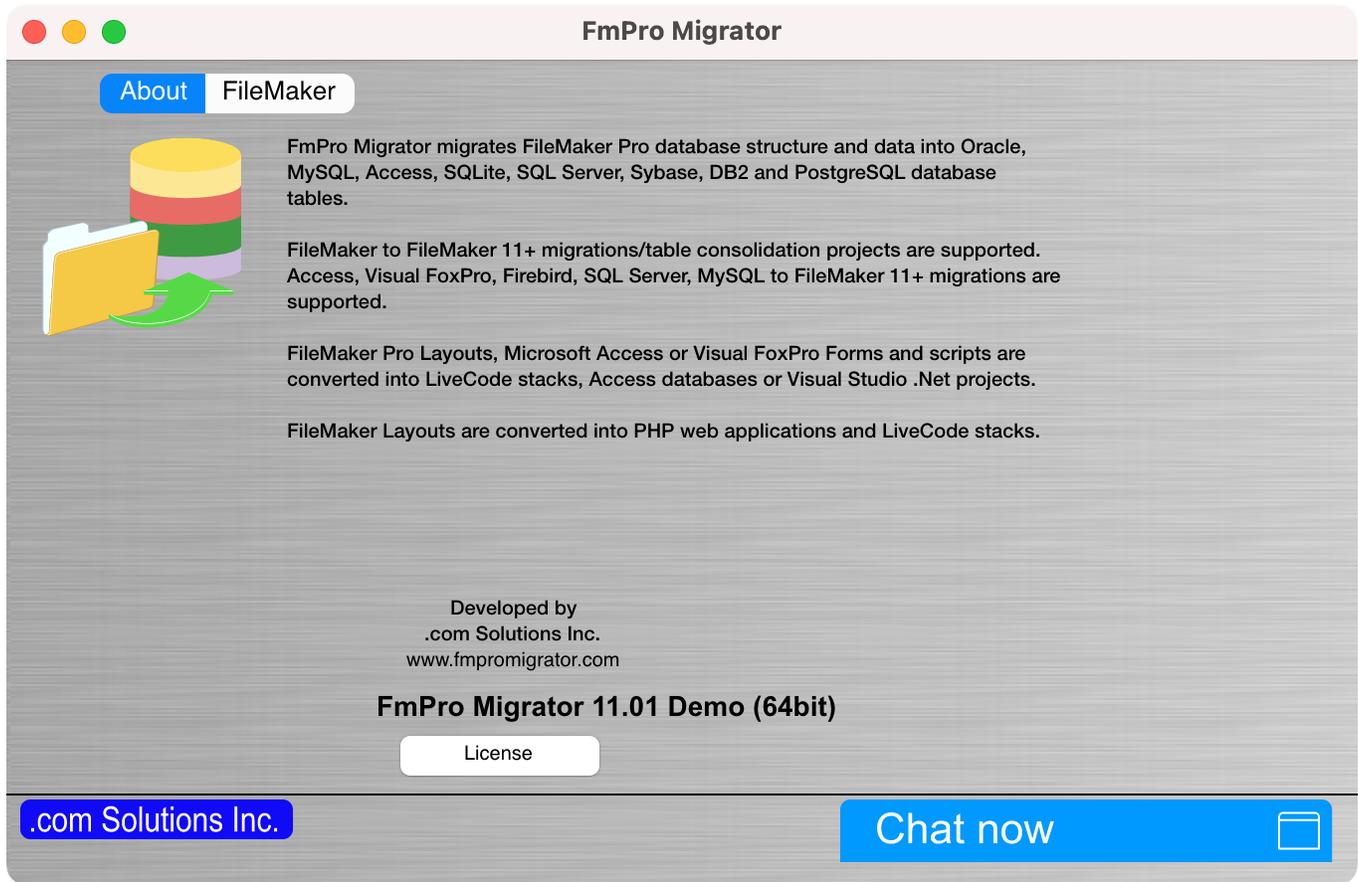
4/14/2024

Demo Edition Dialog



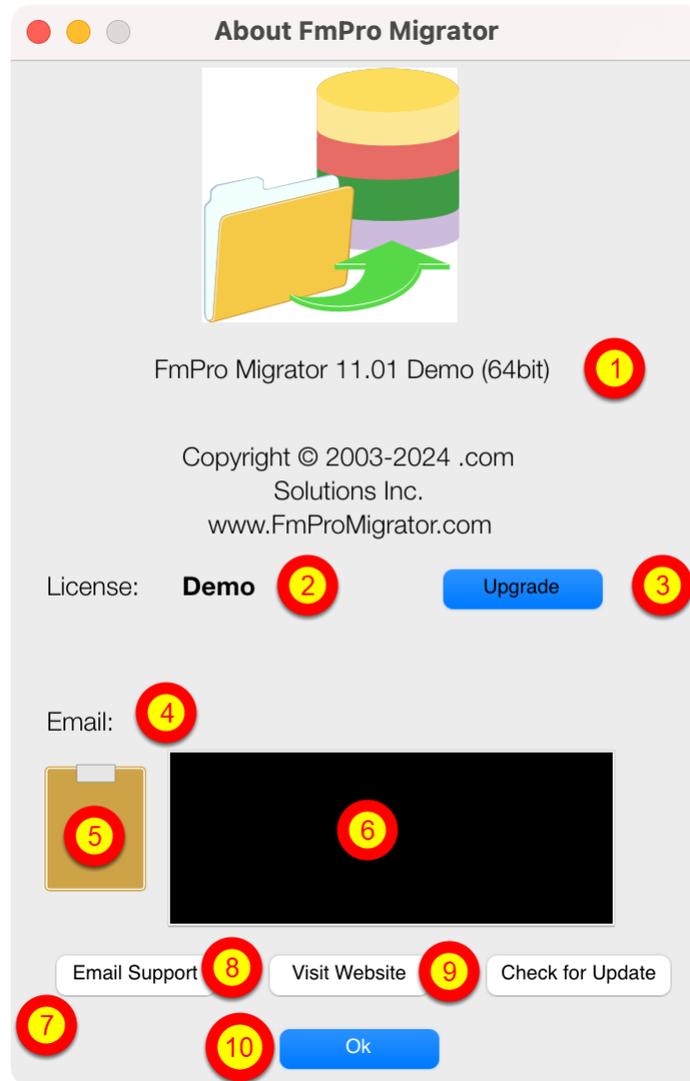
When launched the first time, FmPro Migrator will be running in Demo mode as shown in this screenshot. Clicking the Ok button opens the order page of the website.

In addition to transferring data for 5 database fields, the conversion features shown on the GUI tab of the Migration Process window will convert 5 layouts or forms/reports & scripts.



Clicking the "License" button on the About tab, or selecting the Help -> License/About menu items will open the About window where you can enter the license key.

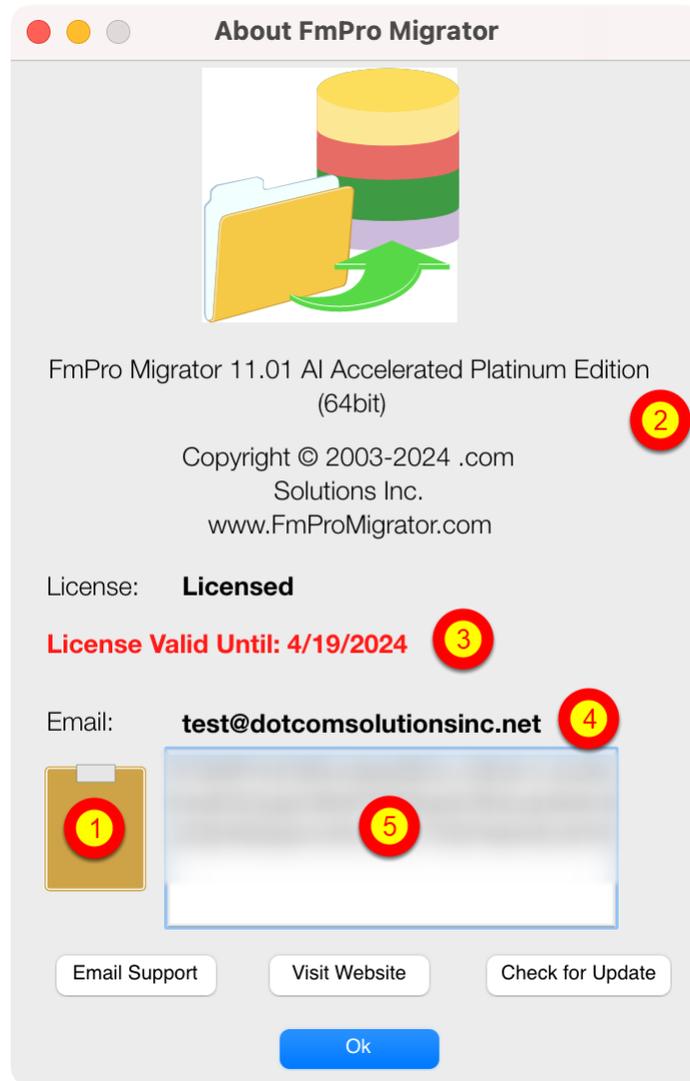
About/License Window



On the About FmPro Migrator window:

(1) Product name and version licensed, (2) Demo or Licensed will be displayed, (3) The Upgrade button opens the website order page, (4) Email address associated with your license key, (5) Clipboard button reads the license key from the clipboard, (6) License key field, (7) Email Support button opens the Contact form on the website, (8) Visit Website opens the order page in Demo mode or the product page in Licensed mode, (9) Check for Update opens the downloads page with the latest software version, (10) Ok button closes the About window.

Pasting License Key



Your product license key will be displayed in your web browser when your order has been completed. It will also be sent via email at the same time, please check your SPAM folder if the email doesn't arrive in a few minutes.

(1) Copy the license key to the clipboard and click the clipboard icon. Once validated (2) the product name will change, (3) the License validation date will be updated, (4) your email address will be displayed, and (5) the license key will be displayed in the field under the Email address when clicking inside the field. In this screenshot the license key has been obscured.

And that is all you need to do to license all of the features of FmPro Migrator.

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 CopyAs... -> 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 CopyAs... -> 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.

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 CopyAs... -> 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:

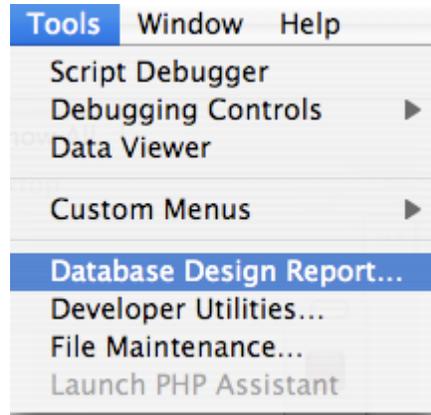
File -> Define/Manage Database...

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 1

Include fields from tables in selected file:

Include in report:

- Accounts
- Custom Menu Sets 2
- Custom Menus
- Data Sources
- Extended Privileges
- Functions
- Layouts

Report Format:

HTML

XML 3

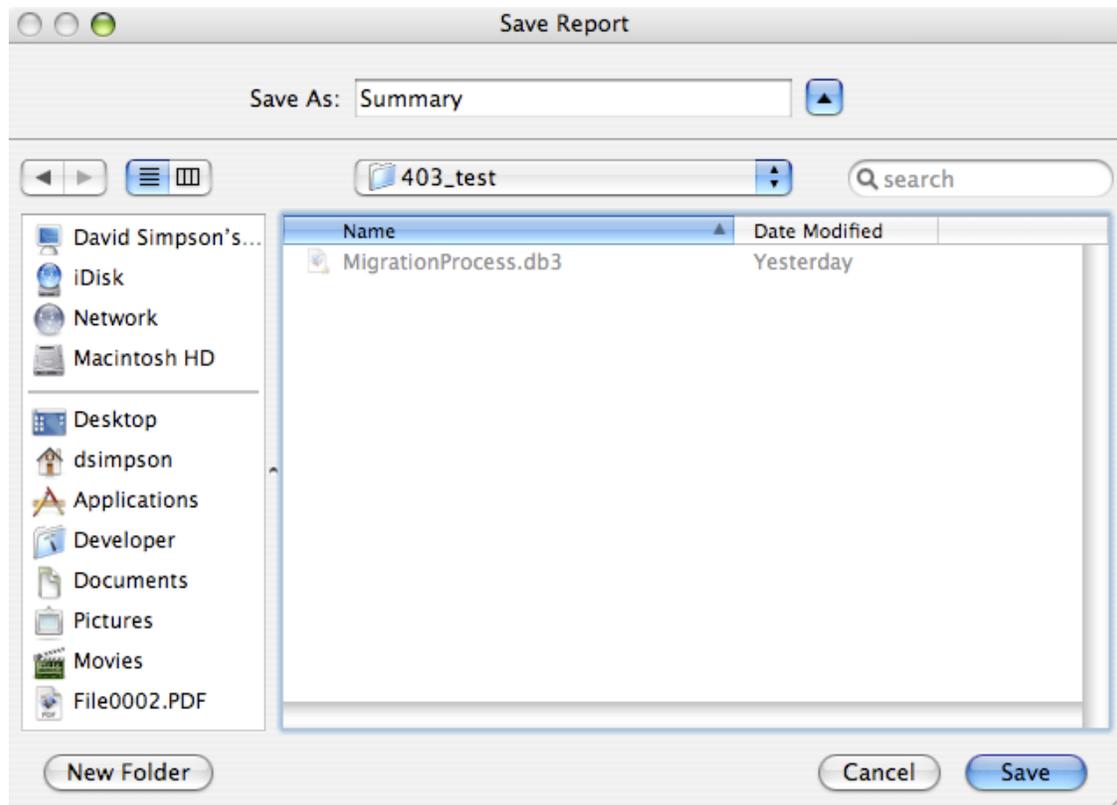
File Handling:

Automatically open report when done

4

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



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 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

The screenshot displays three windows from the FileMaker Pro Migrator software. The 'Migration Process' window shows a table list with 'tbl_Assets' selected. The 'Table Details' window shows the configuration for 'tbl_Assets', including the source and destination databases and the number of fields. The 'Manage Deleted Fields' window shows a table summary for 'tbl_Assets' with 67 fields, 15 marked for deletion, and 52 remaining fields. The 'Mark for Deletion' button is highlighted with a red circle.

Migration Process - Tables

Original Table	New Table	Source	Destination	Fields	Records
Companies	companies	FileMaker 11	SQLite	5	
Members	members	FileMaker 11	SQLite	17	29
tbl_Assets	tbl_Assets	FileMaker 11	SQLite	67	
tbl_Maintenance_Record	tbl_maintenance_record	FileMaker 11	SQLite	29	6
tbl_Value_List_Data_Ita		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

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

Manage Deleted Fields

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

Results:

- Mark for Deletion
- UnMark for Deletion
- Delete Fields
- Add Deleted Fields

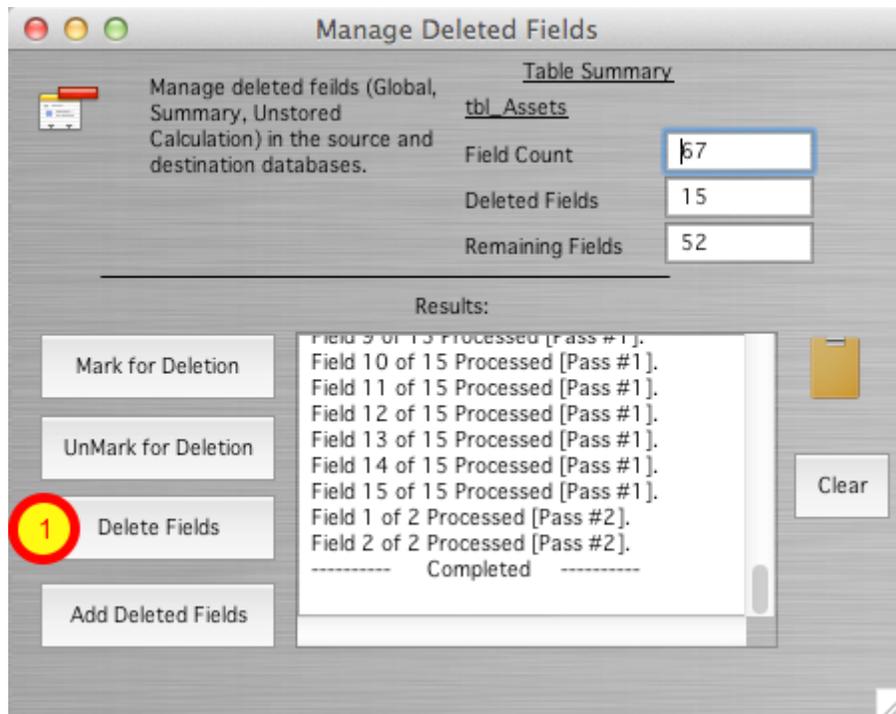
Once the FileMaker table definitions have been imported via the clipboard, the list of tables will be displayed within the Migration Process window.

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



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.

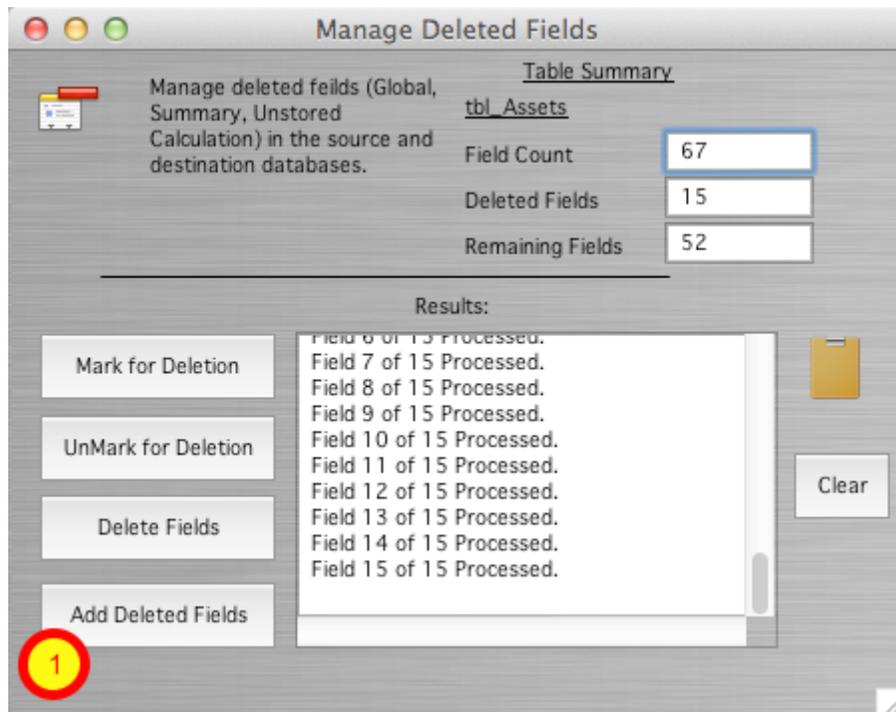
Note: It is common for some DROP COLUMN commands to fail as a result of Outstanding References by other fields. In order to resolve this issue, FmPro Migrator logs the failed DROP COLUMN commands and then re-sends those commands to the FileMaker Pro database again, in reverse order during Pass #2 processing. 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]



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

Previously, Microsoft Office 365 was 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. This problem is now resolved with Office 365 installations in 2023 when running the offline installer.

Installing Microsoft 365 Access Runtime - for ODBC Driver [Optional]

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.
3. Depending on your browser, select **Run**(in Edge or Internet Explorer), **Setup**(in Chrome), or **SaveFile**(in Firefox). If you see the User Account Control prompt that says, **Do you want to allow this app to make changes to your device?** select **Yes**. The install begins.

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

This step can probably be skipped as of 2023.

Testing being done in June 2023 shows that the Access 2016 32-bit and 64-bit ODBC drivers are being installed with the Office 365 offline installer: O365HomePremRetail.img (4.7GB). You only

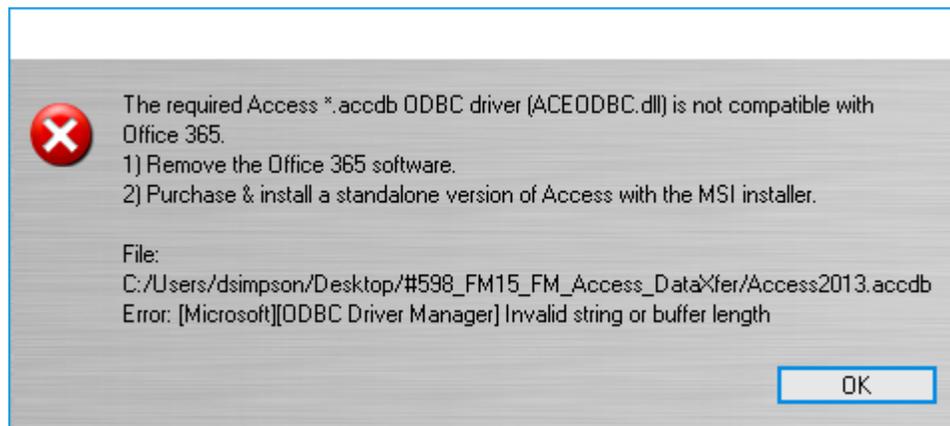
need to install this runtime if the installer didn't install the ODBC driver.

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:

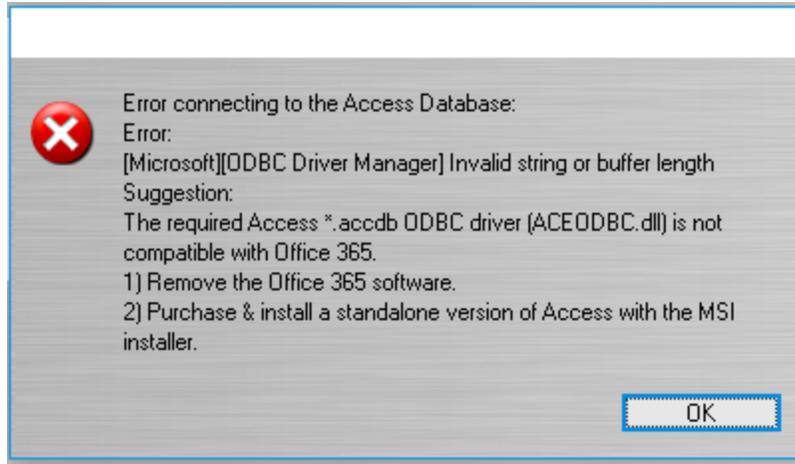
<https://support.microsoft.com/en-us/office/download-and-install-microsoft-365-access-runtime-185c>

FmPro Migrator Office 365 Error Dialog - Importing .accdb Files [Obsolete]

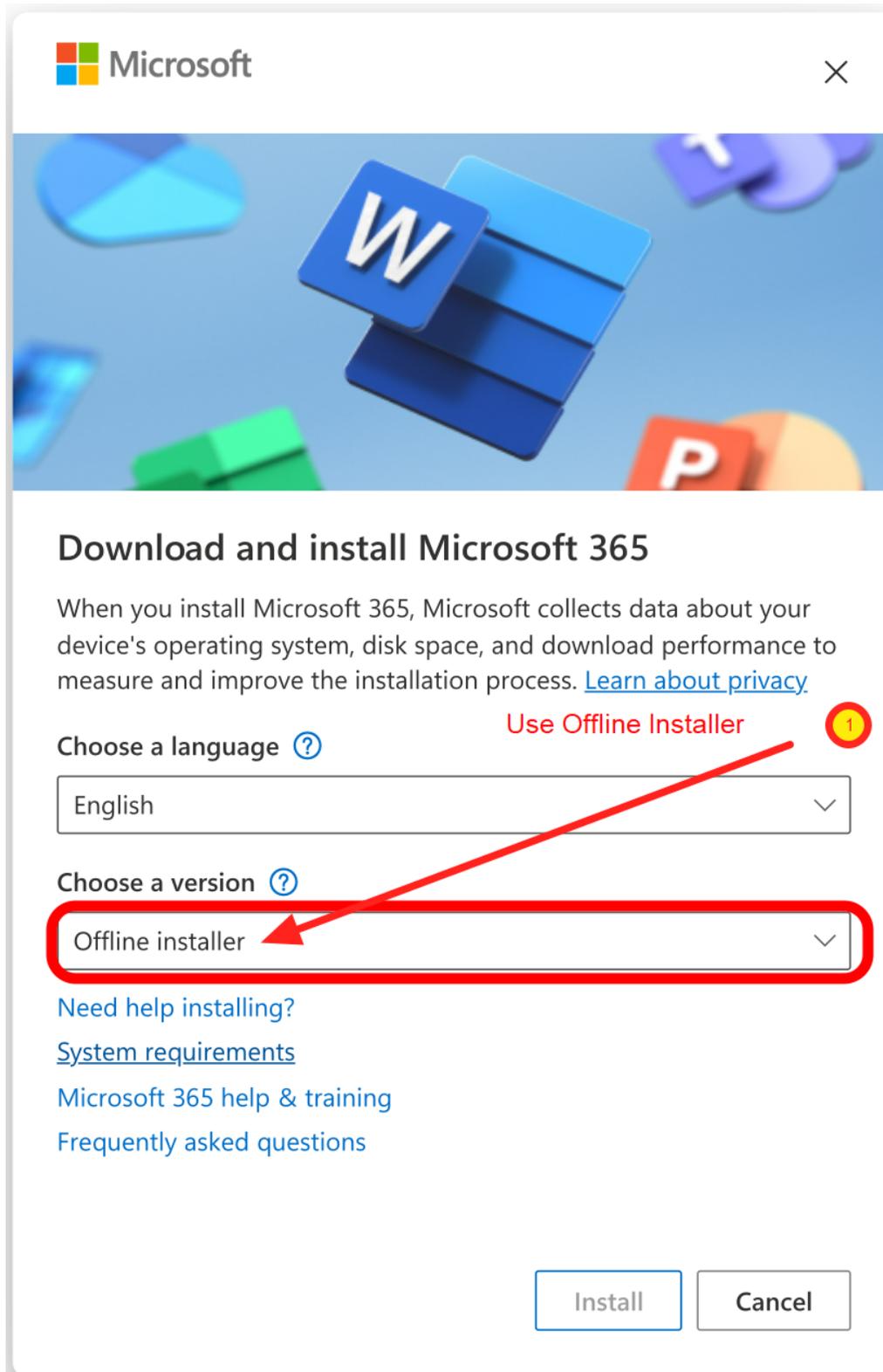


This error should no longer occur with new Office 365 installations. New installations are installing the ODBC driver and it works correctly on x86-64 hardware. This note in the manual is just for historical and documentation purposes - and should not occur with newer versions of Office 365 as of 2023.

If this error occurs, please send an email via the Contact page on the FmProMigrator website. The code which checks for this registry key still remains in FmPro Migrator, but should not get triggered anymore.



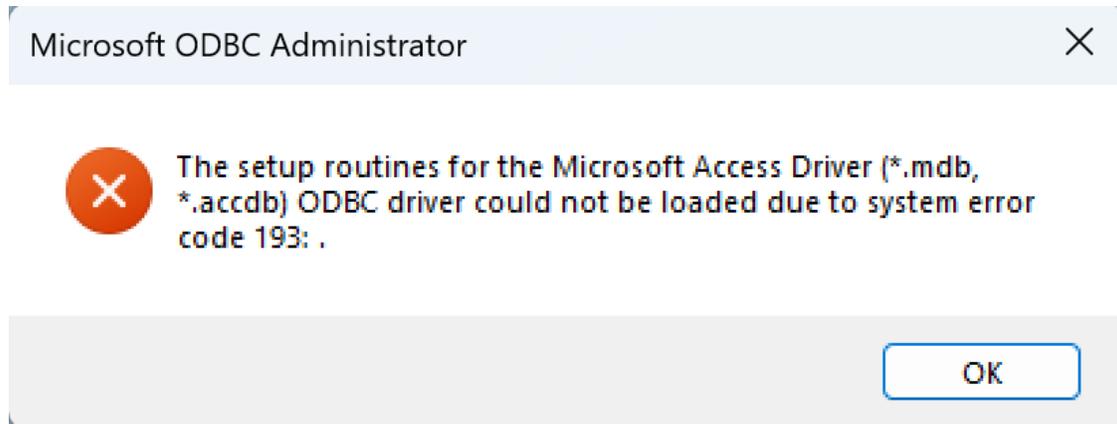
This error should no longer occur with new Office 365 installations as of 2023. Previously, this error could occur when creating a table or transferring data into an Access .accdb file.



By default, Microsoft will use the ClickToRun installer. You should download the offline installer. The offline installer includes both 32-bit and 64-bit versions of Office software.

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 installed. For instance, if the copy of FileMaker Pro is 64-bit, then all of the rest of the software should also be 64-bit.

Access ODBC Driver Error 193 - Windows 11 ARM



The remaining issues with Office 365 or Access 2016 ODBC functionality at this time, seems to be when running on Windows 11 ARM.

- 1) When creating/updating a 64-bit ODBC DSN, error 193 is displayed. Manually editing the registry entries has not solved this issue. A 32-bit DSN can be created and edited, but the .mdb driver doesn't support .accdb files. It may be possible to convert .accdb files into .mdb files, but features may be lost.
- 2) The system created links to the 32-bit and 64-bit ODBC Administrator EXE files are not created properly and need to be created manually for the Start menu.

These problems don't occur with Windows 11 on x86-64 CPUs.

ODBC Driver Installation

Install FileMaker ODBC Driver - macOS (FileMaker 11+)

Download ODBC Manager



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.

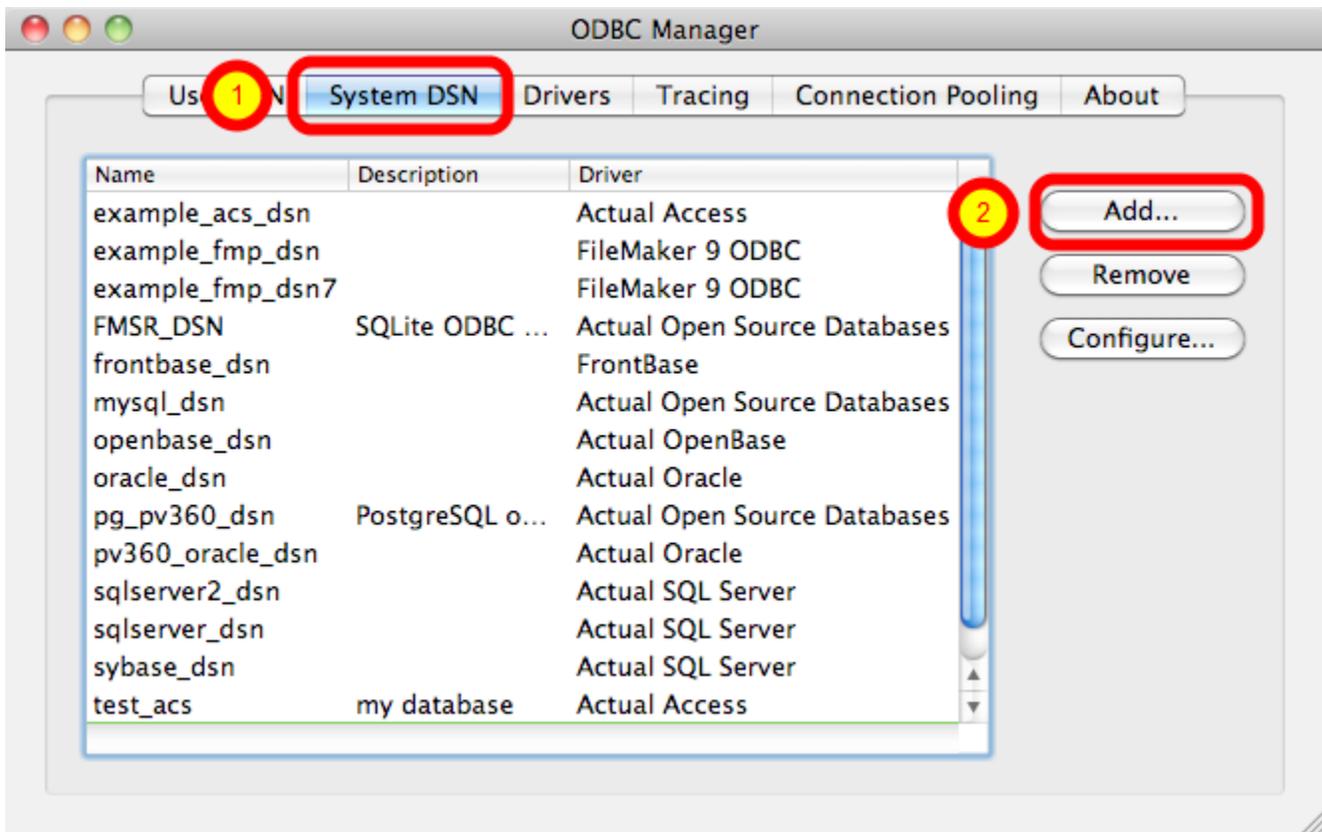
Install the FileMaker ODBC Driver



(1) Open the xDBC folder on the FileMaker install CD (or installer image DMG file).

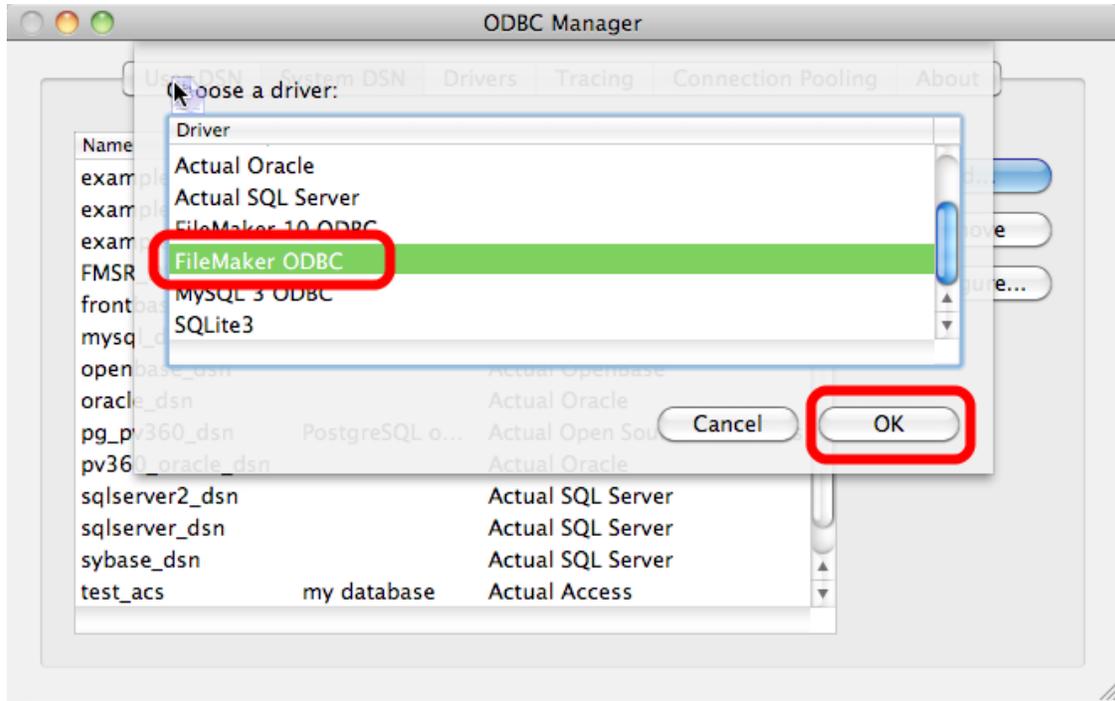
(2) Double-click the FileMaker ODBC.mpkg installer to install the FileMaker ODBC driver. Follow the insular prompts.

Add a System DSN

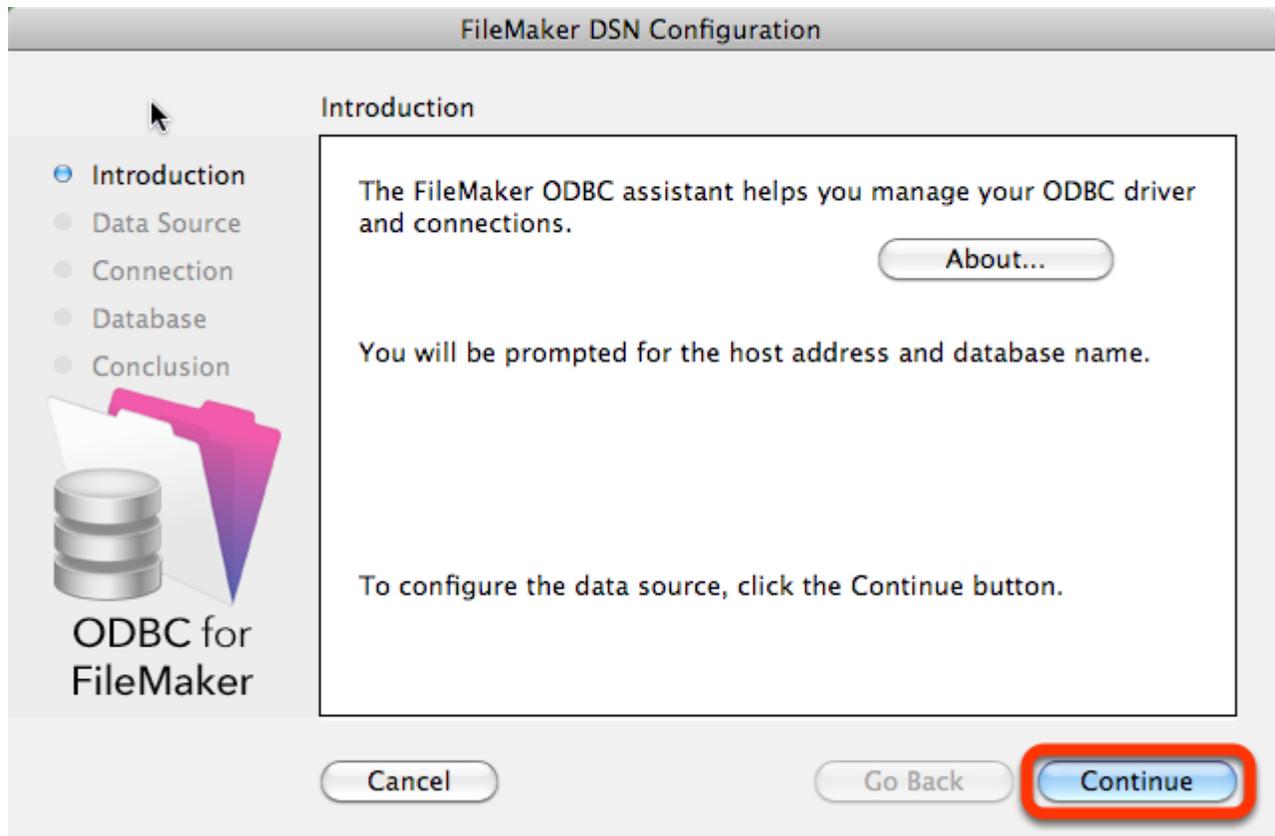


(1) Click the System DSN tab, then (2) click the Add button.

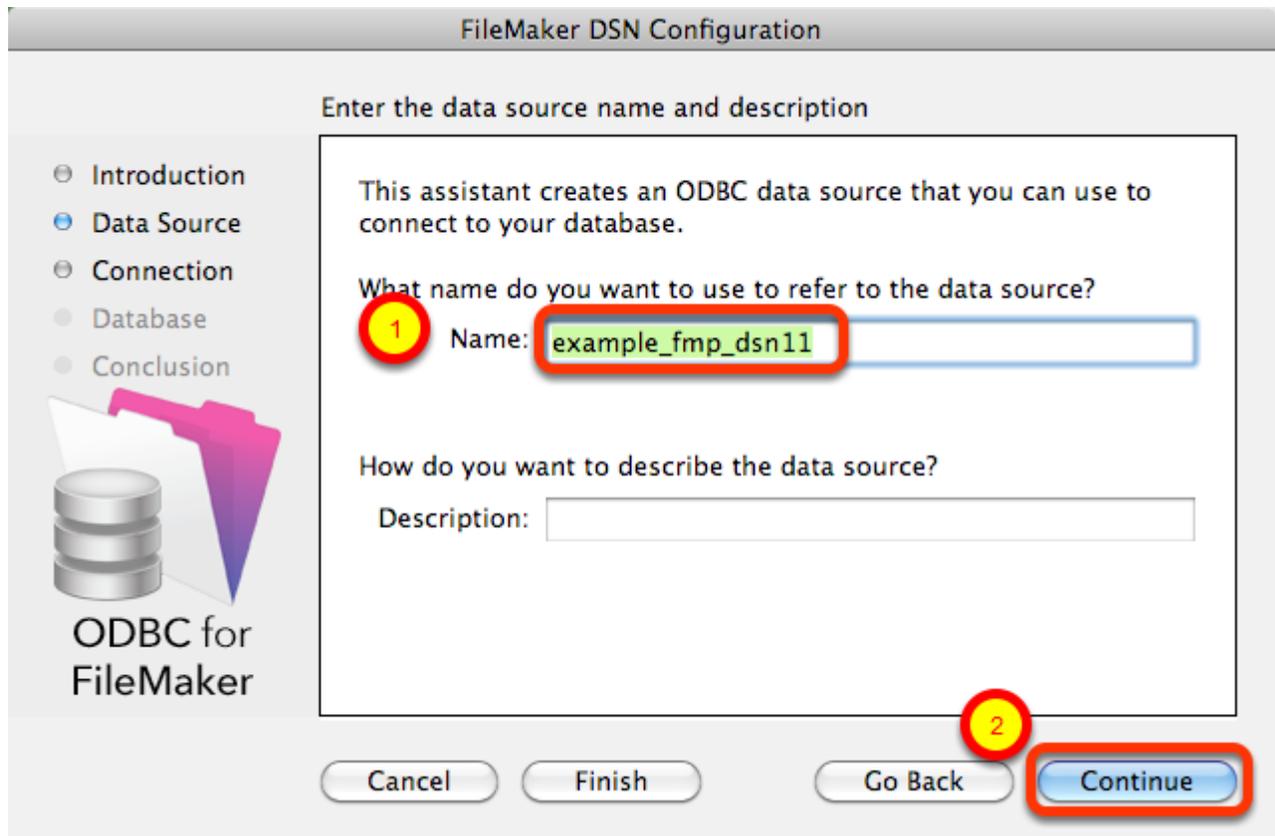
Choose the FileMaker ODBC Driver



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



Click the Continue button.

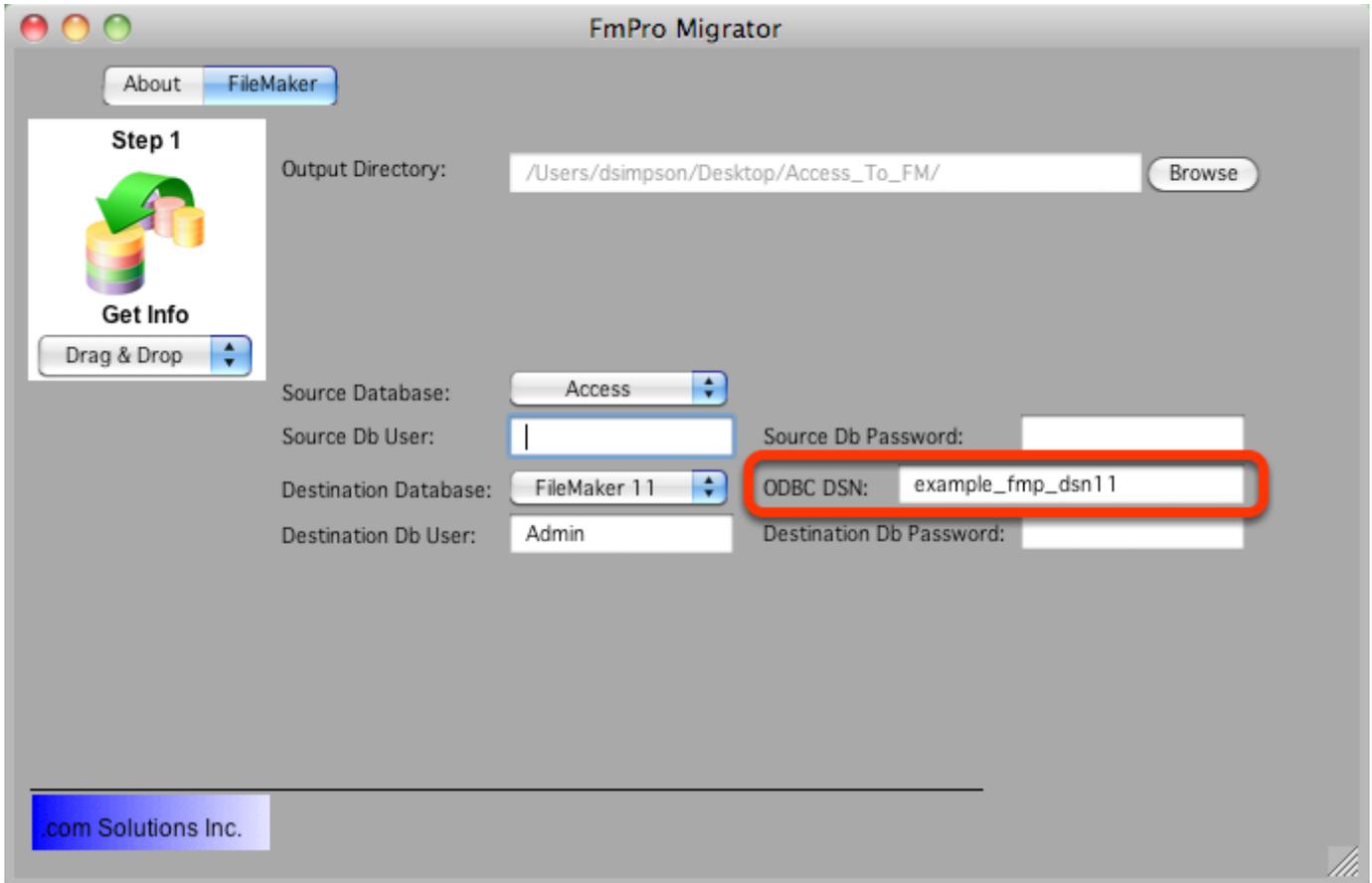


(1) Enter a name for the FileMaker ODBC DSN. The name used here needs to be exactly the same as the ODBC DSN Name you enter into FmPro Migrator.

(2) Click the Continue button.

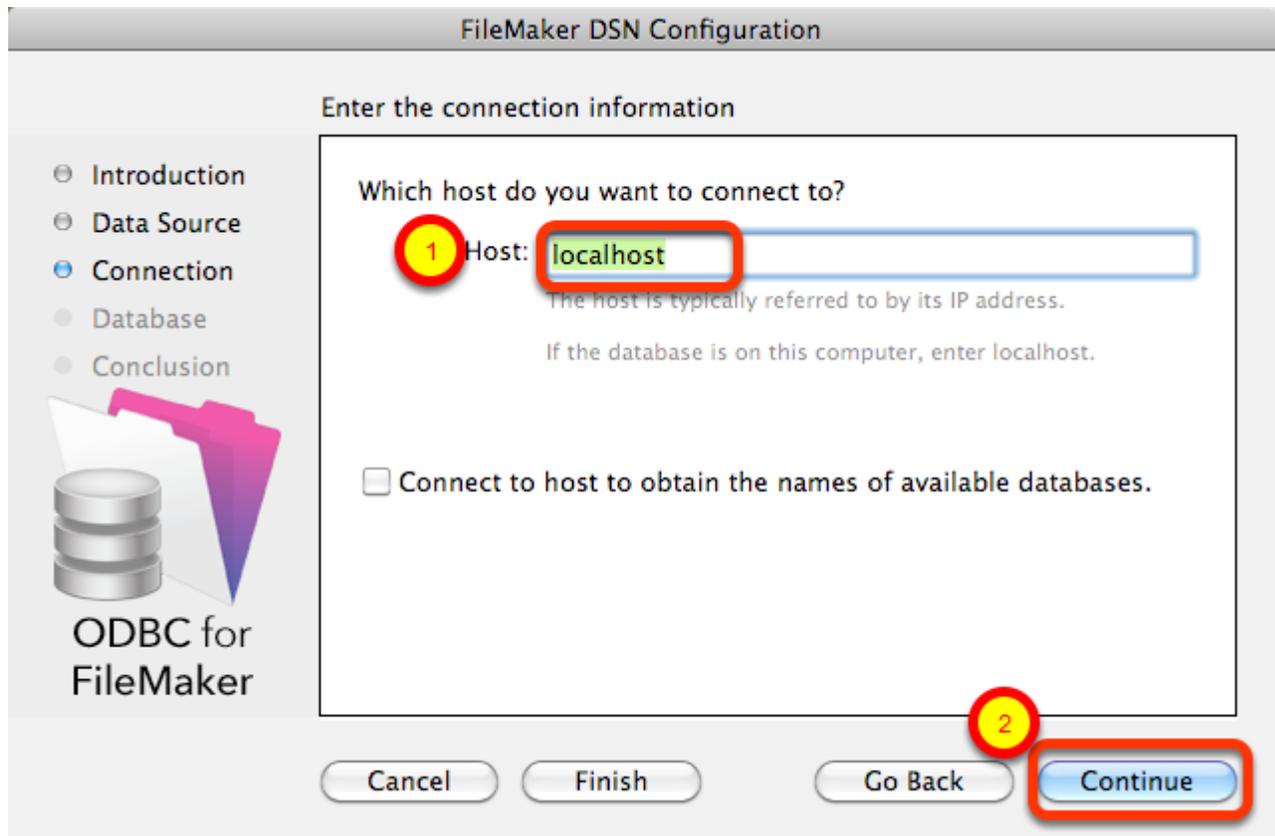
Note: You can pick a different name than you see here, just make sure that you enter exactly the same name into FmPro Migrator.

FileMaker ODBC DSN Configuration - ODBC DSN Field in FmPro Migrator



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).

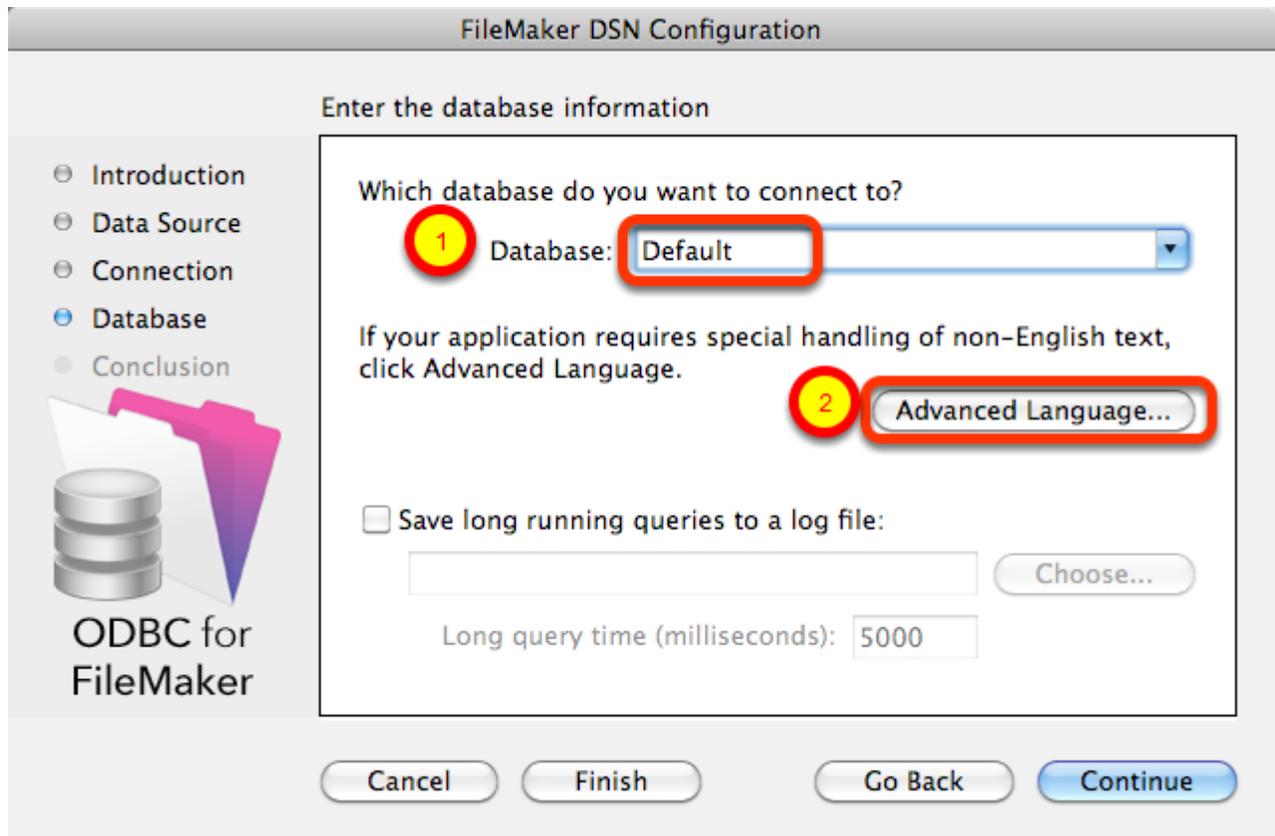


(1) Enter the hostname for the FileMaker database (default is localhost).

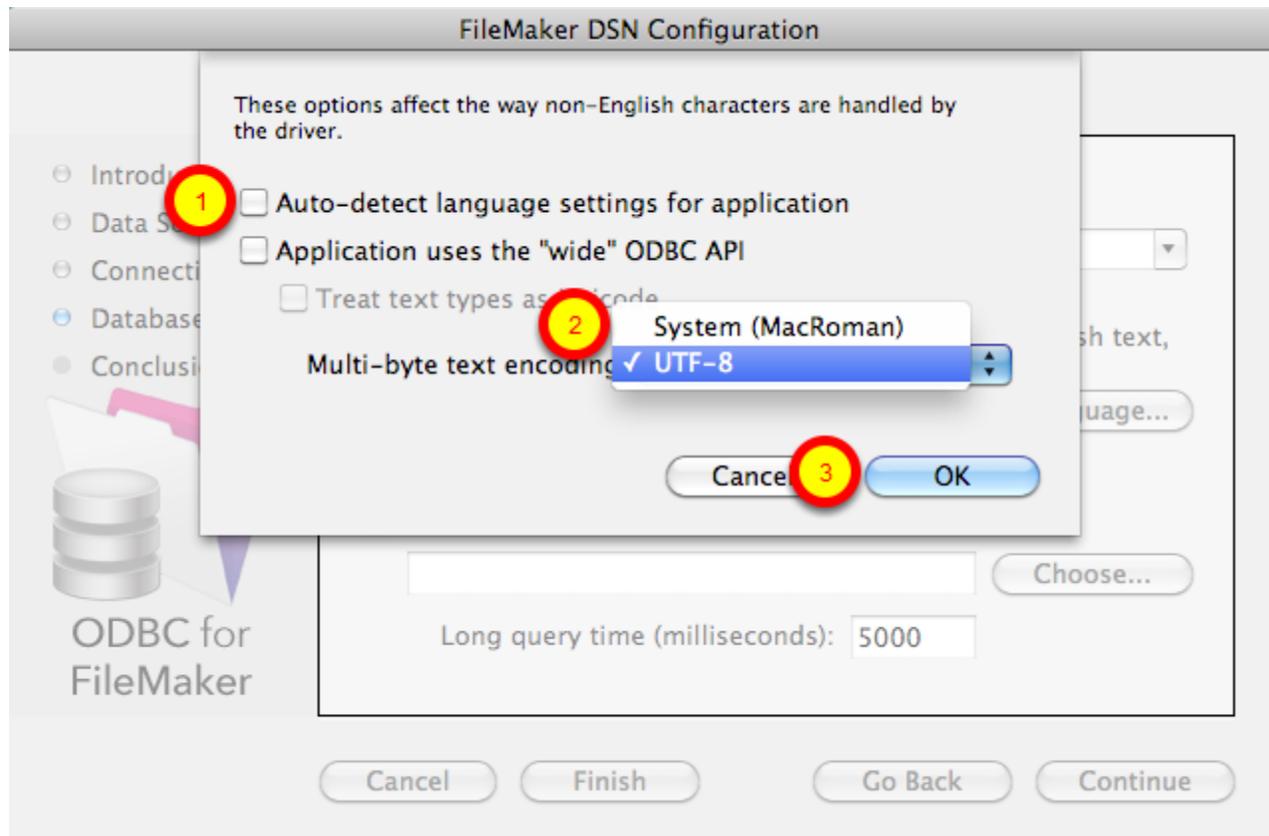
(2) Click the Continue button.

For a file which has been opened on your local desktop machine, leave the default setting of localhost.

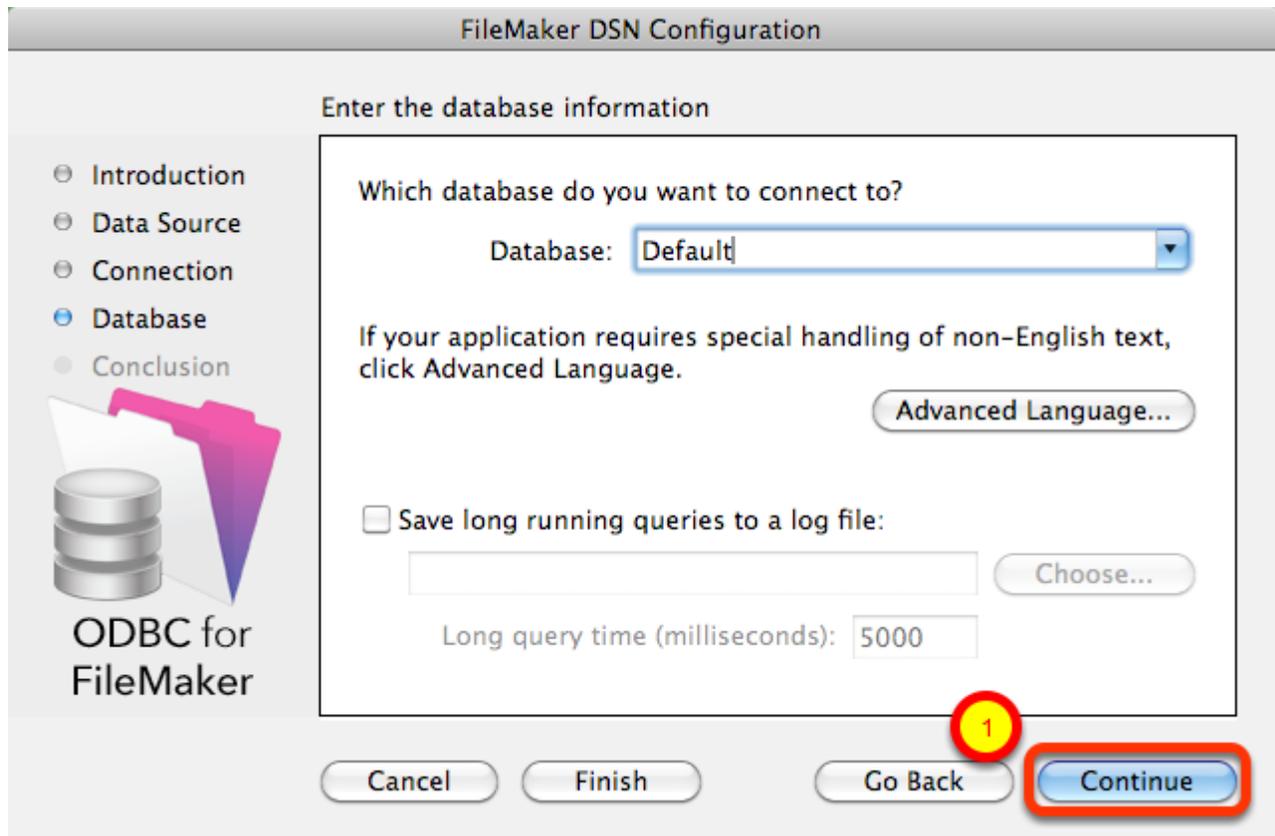
For a hosted file hosted by FileMaker Server Advanced, enter the IP Address or hostname of the FileMaker Server Advanced computer.



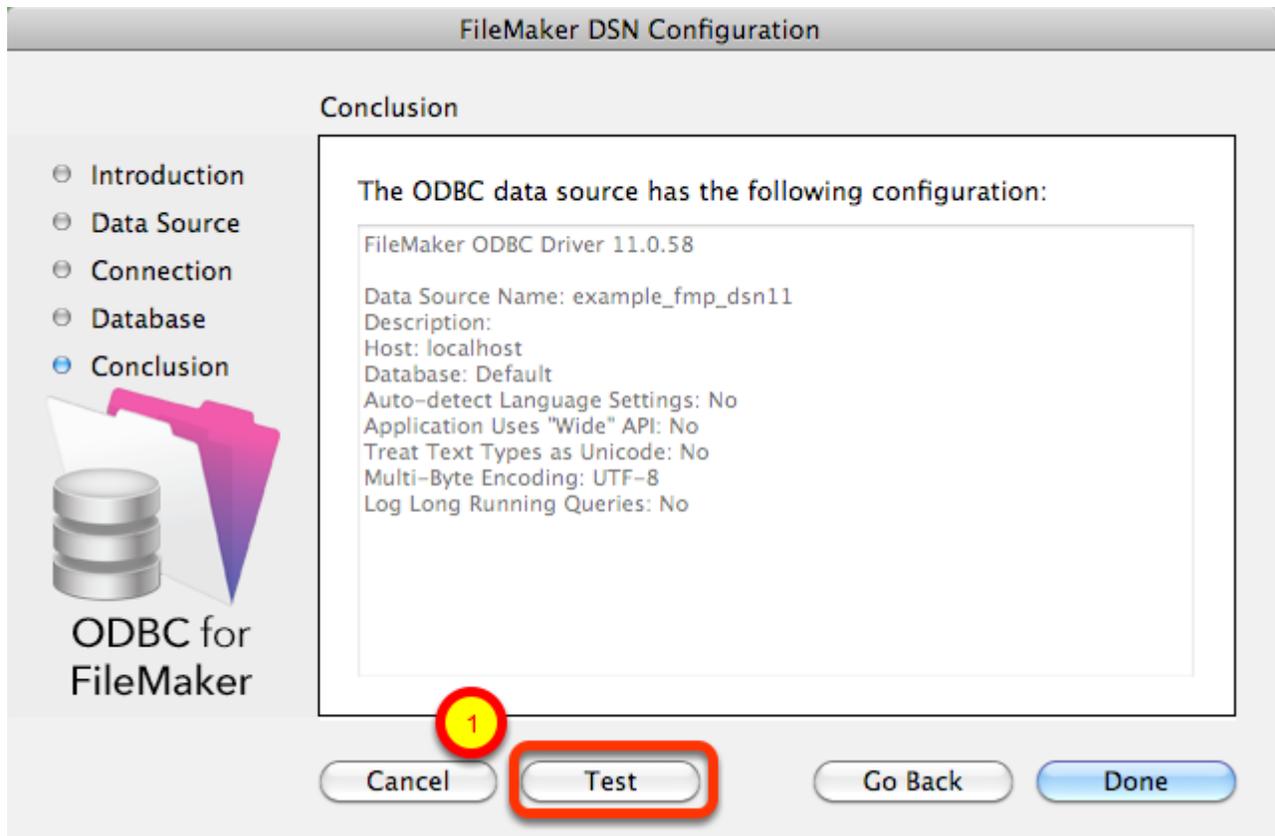
- (1) For most locally hosted database files, enter the text "Default" into the Database field, instead of selecting a database from the list. This way you can use any single database file without re-selecting the database name each time you open a new file. To use this feature, you must only have one 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.



- (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.



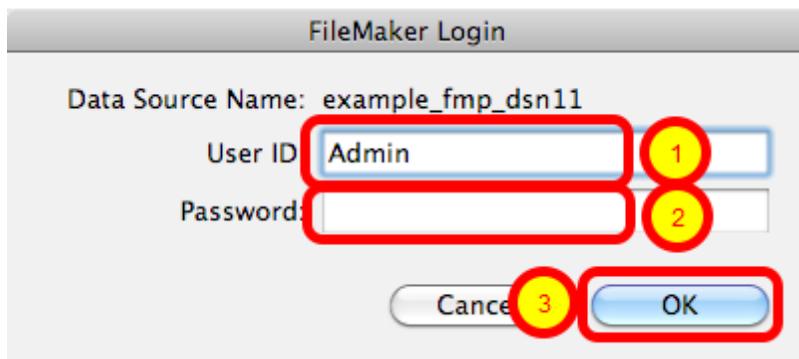
(1) Click the Continue button.



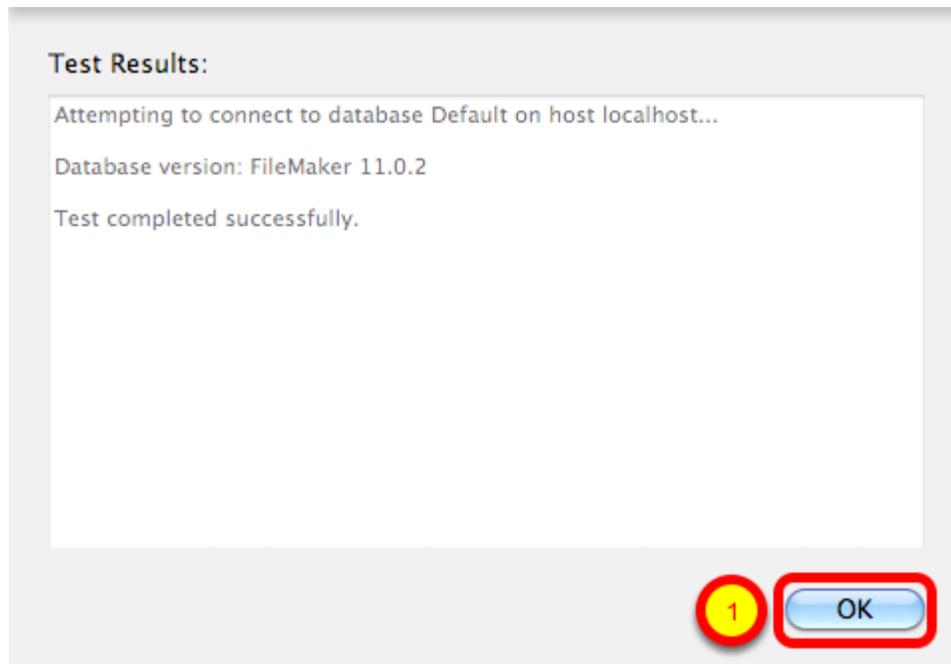
(1) Click the Test button.

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.



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

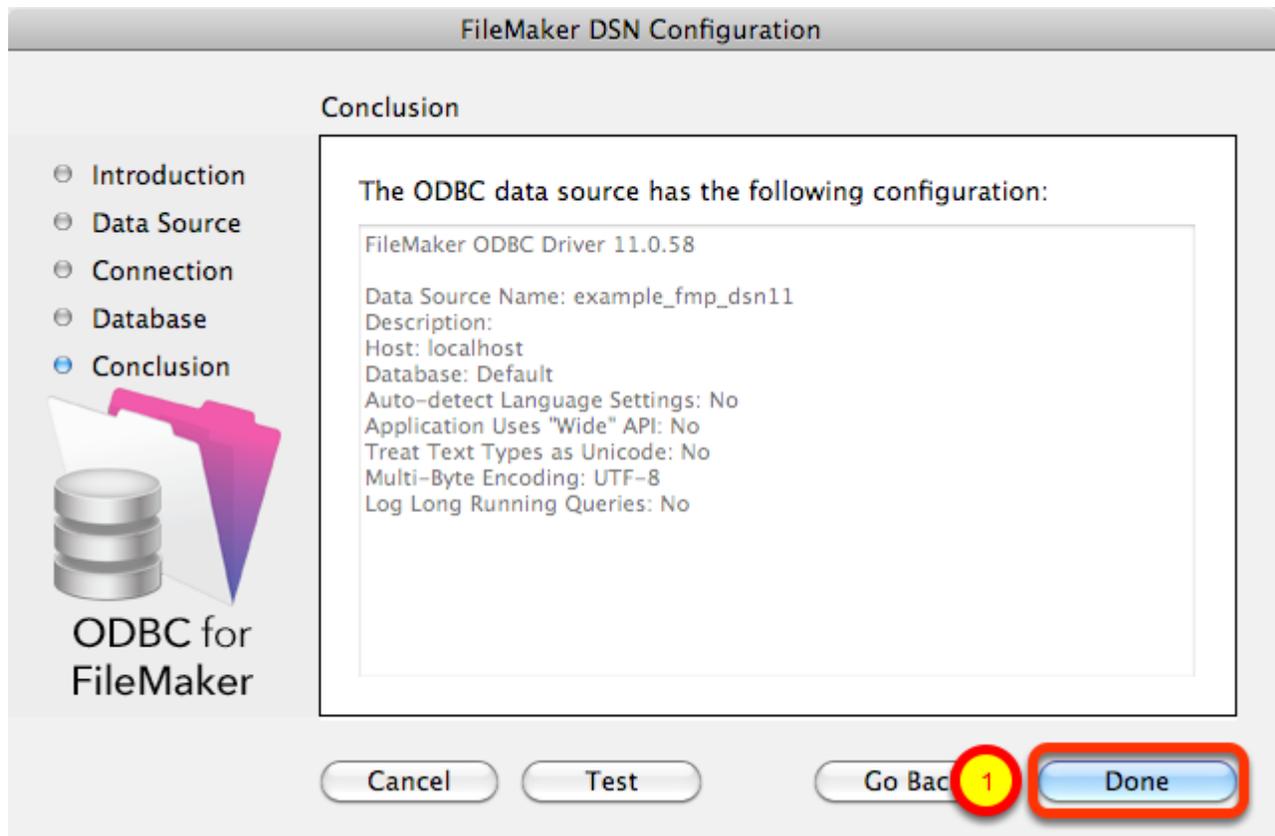


(1) Click the Ok button.

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.

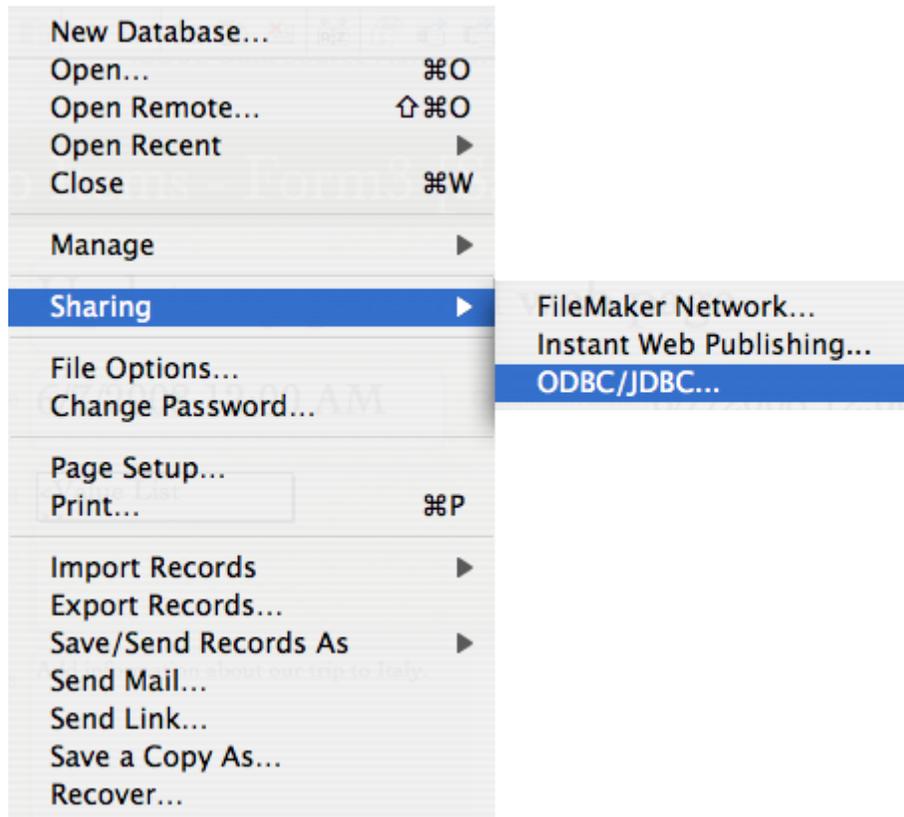
If the test fails:

- 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.



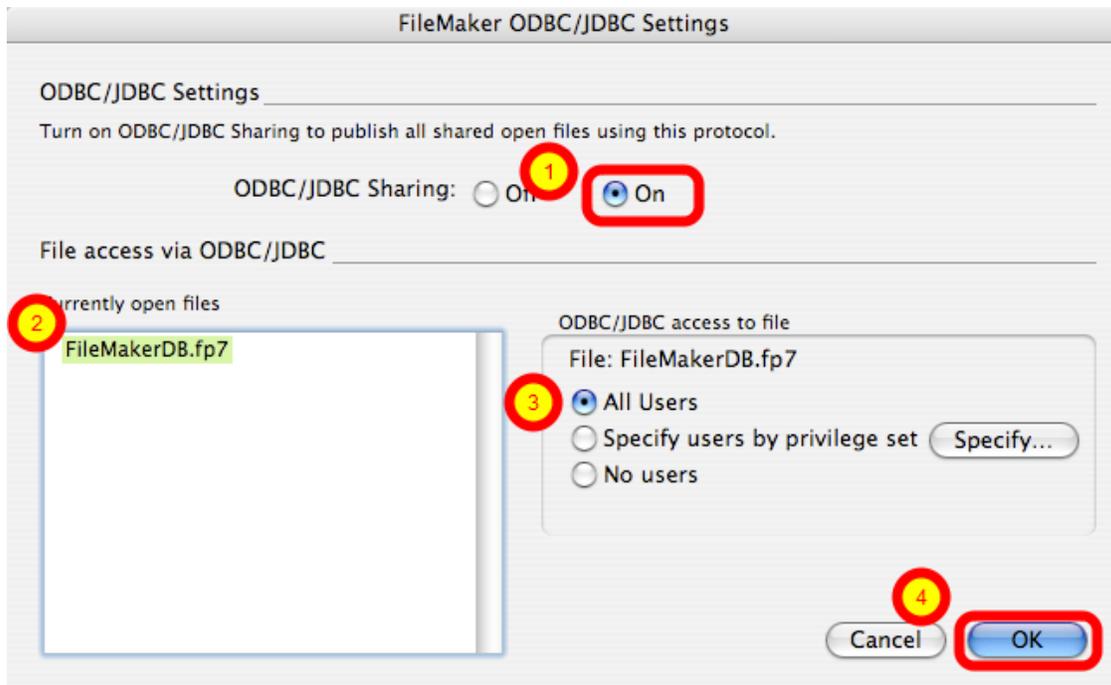
(1) Click the Done button to save the new ODBC DSN.

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



(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.

Downloading the FileMaker ODBC Driver

Software Update: FileMaker xDBC client drivers for FileMaker

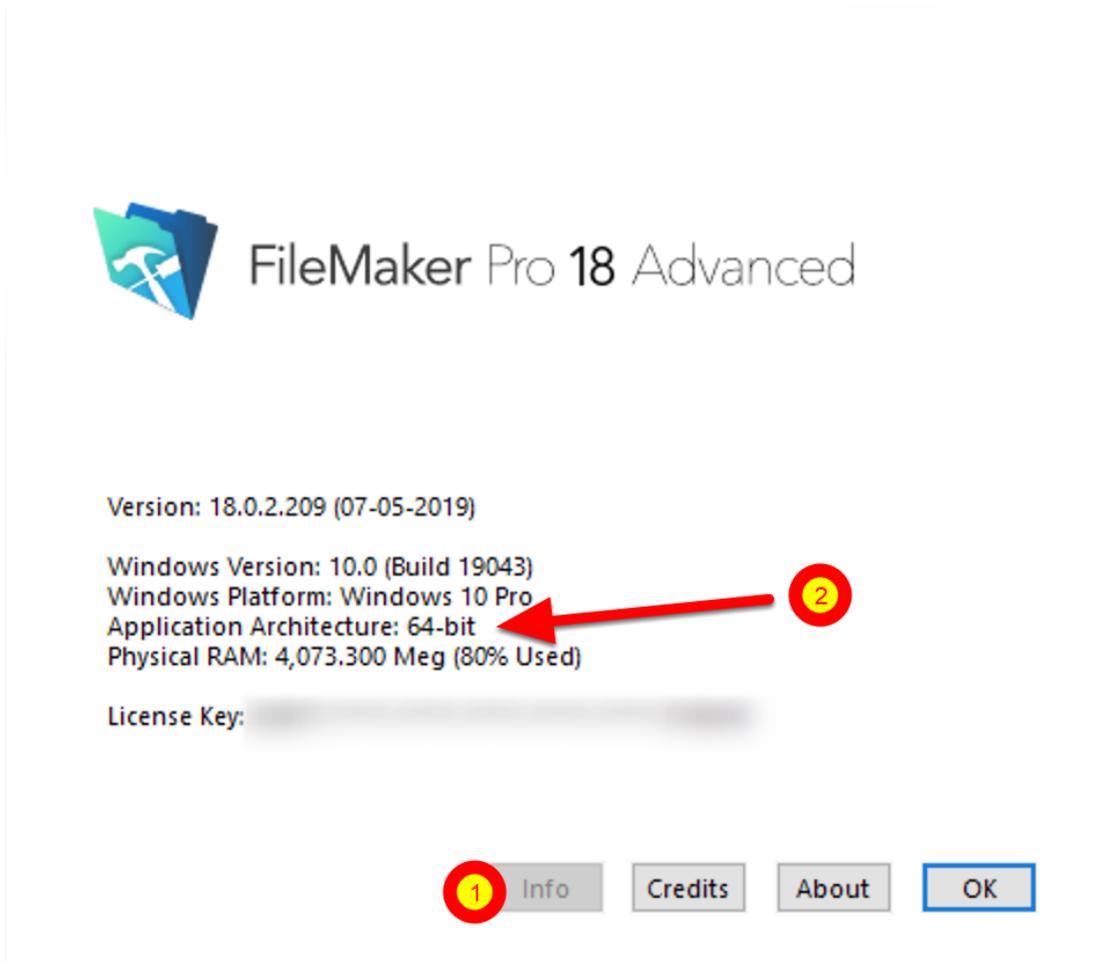
xDBC Client Drivers for FileMaker 19

Products	Windows download	Macintosh download
FileMaker Pro 19.4.1 FileMaker Server 19.4.1		 Requires 64-bit ODBC Manager
Products	Windows download	Macintosh download
FileMaker Pro 19.3.1 FileMaker Server 19.3.1		 Requires 64-bit ODBC Manager

The FileMaker ODBC driver is also available from the Claris support website:
<https://www.claris.com/resources/downloads/>

On this web page, look for the item labelled:
xDBC update for FileMaker Platform

This item will link to the page shown in this screenshot, the actual URL has a very specific document number, which might change - so it is best to start with the downloads page.

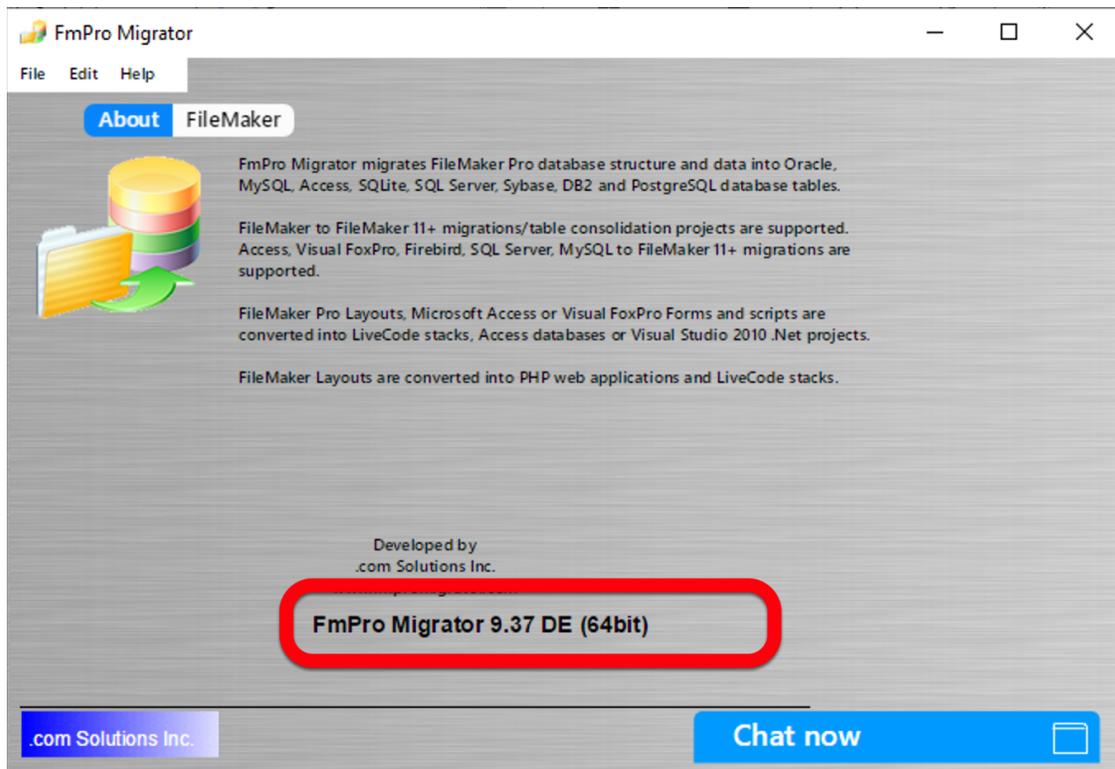


Starting with FileMaker Pro 19, only 64-bit versions are available for installation. Earlier versions of FileMaker may include both 32-bit and 64-bit versions. To find the version of FileMaker running on your computer:

1) Select the Help -> About FileMaker Pro Advanced menu. 2) Click the Info button as shown above to see the Application Architecture.

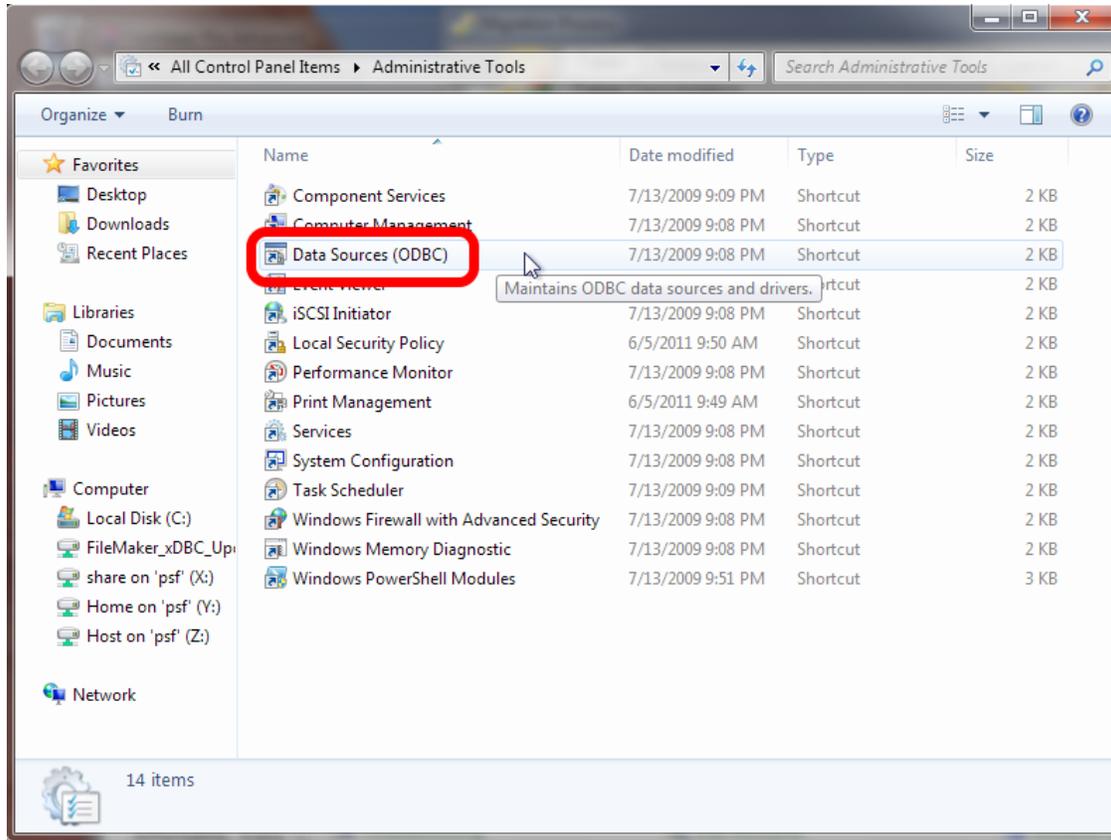
If you are running the 64-bit version of FileMaker Pro, then you want to be using the 64-bit version of FmPro Migrator and the 64-bit version of the ODBC Administrator to create the ODBC DSN.

FmPro Migrator - 64bit



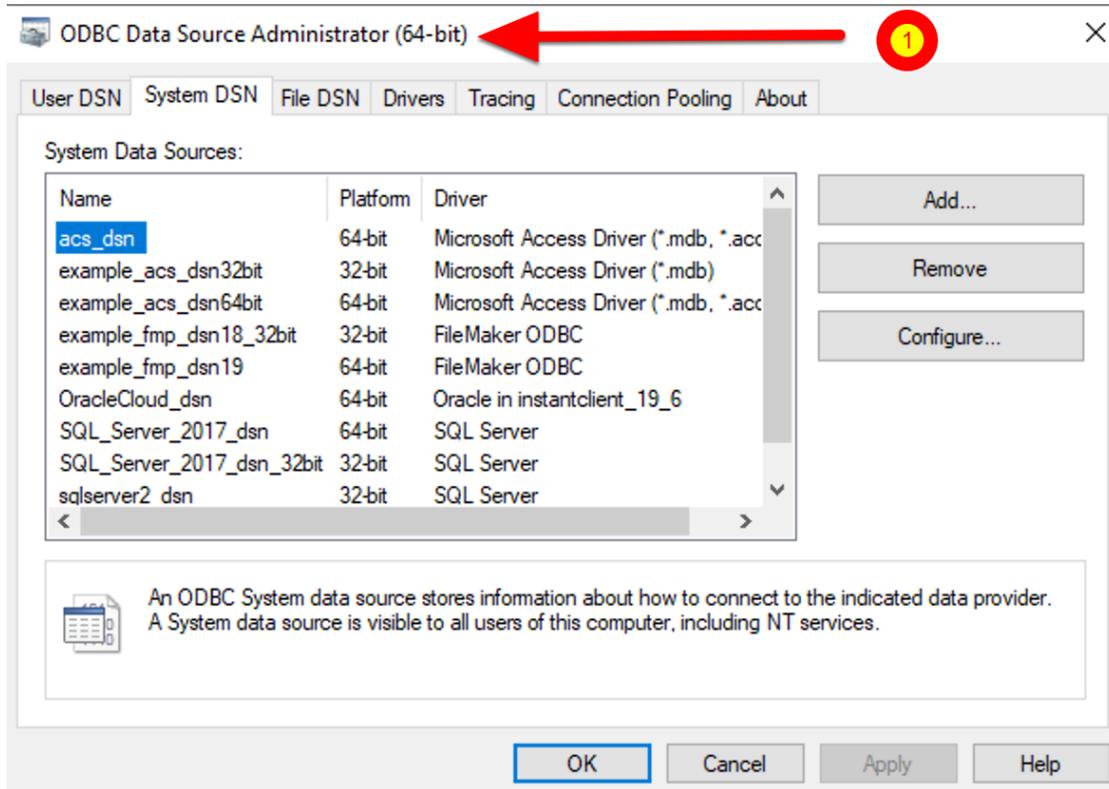
Both 32-bit and 64-bit versions of FmPro Migrator are available. The About screen shows which version you are running.

Create ODBC DSN - Windows



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

64-bit ODBC Data Source Administrator

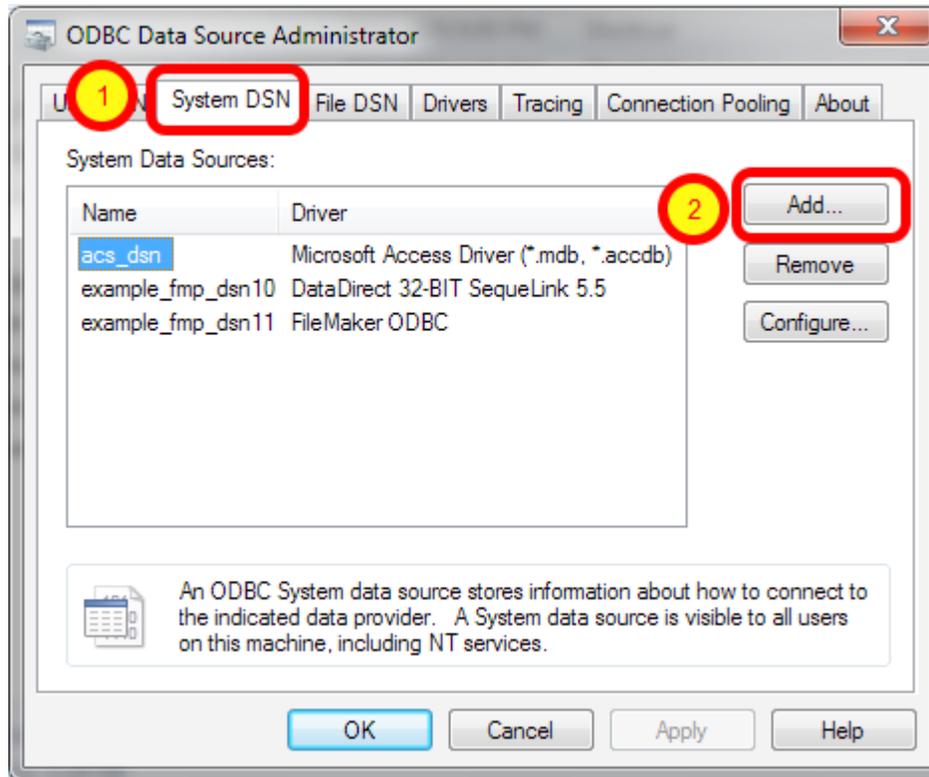


The 64-bit ODBC Administrator shown in this screenshot is available from the path:
%windir%\system32\odbcad32.exe

Both 32-bit and 64-bit ODBC DSNs are displayed in the ODBC Administrator, as shown in the Platform column of the listing.

Running under Windows 10, it shows (64-bit) in its window title when running.

32-bit ODBC Administrator

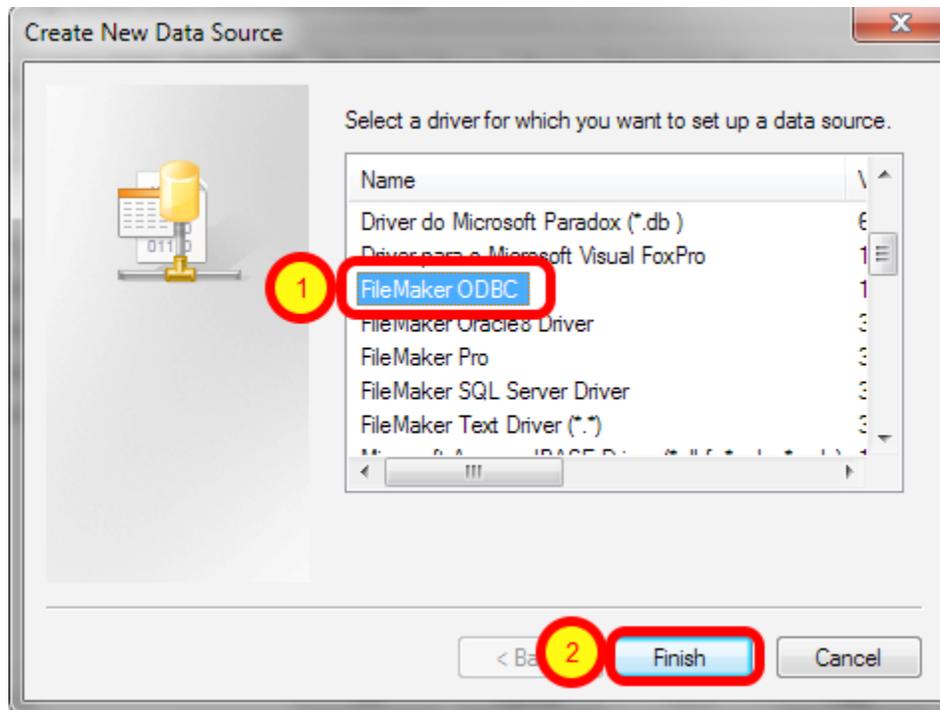


The 32-bit ODBC Administrator is available from the path:
%windir%\syswow64\odbcad32.exe

This screenshot is from an earlier version of Windows, which doesn't show the Platform type in the list of ODBC DSNs and it also doesn't show in its title whether it is running 32-bit or 64-bit.

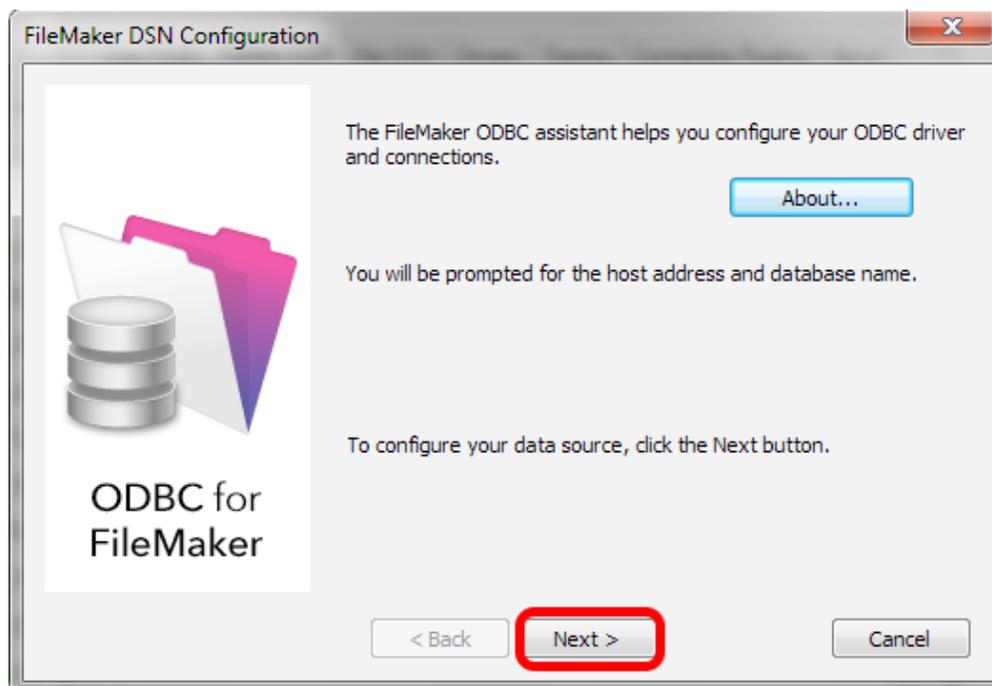
(1) Click the System DSN tab, then (2) click the Add button.

Select ODBC Driver - FileMaker 11 - Windows



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

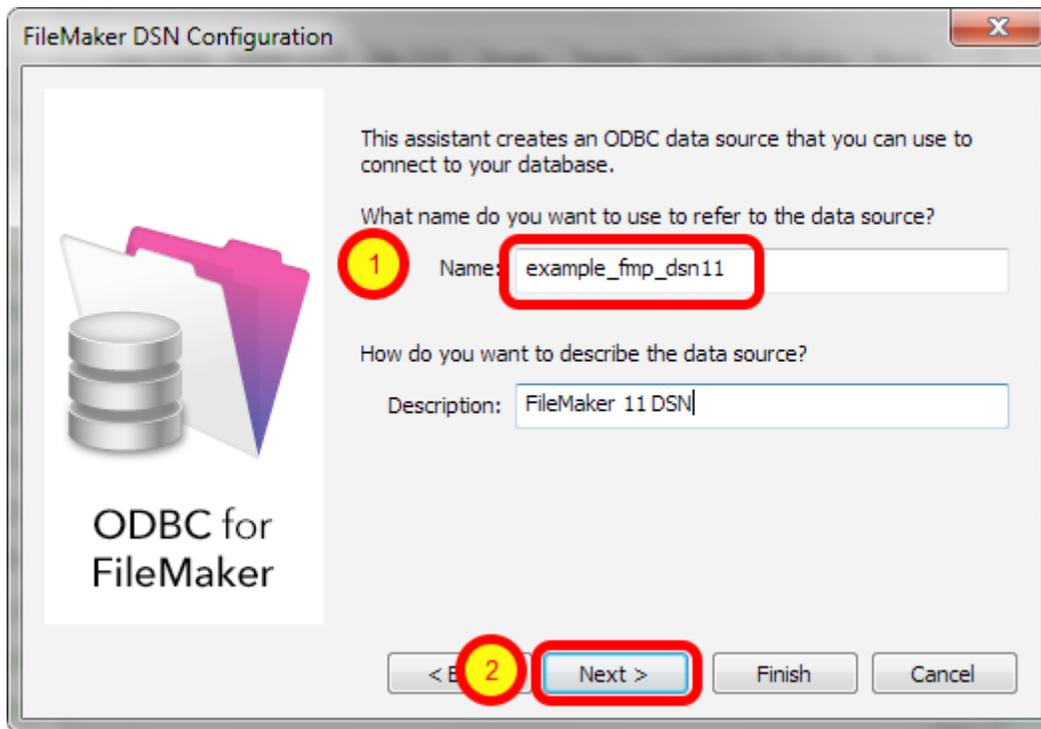
ODBC Driver Configuration - FileMaker 11+ - Windows (1)



Click Next button.

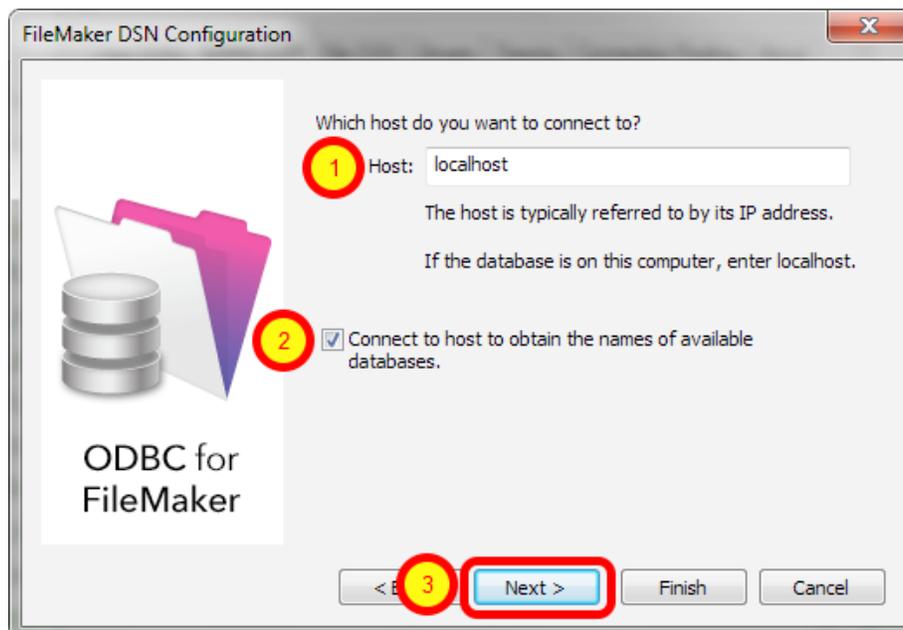
Click the About button if you need to find the version number for the driver.

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



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

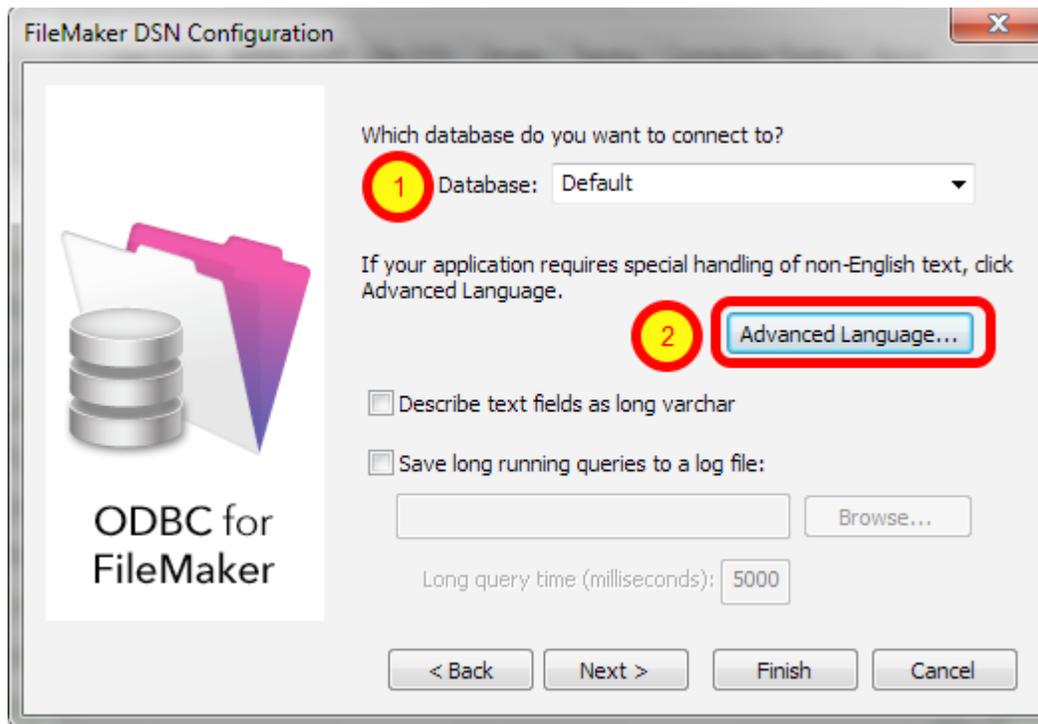
ODBC Driver Configuration - FileMaker 11+ - Windows (3)



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.

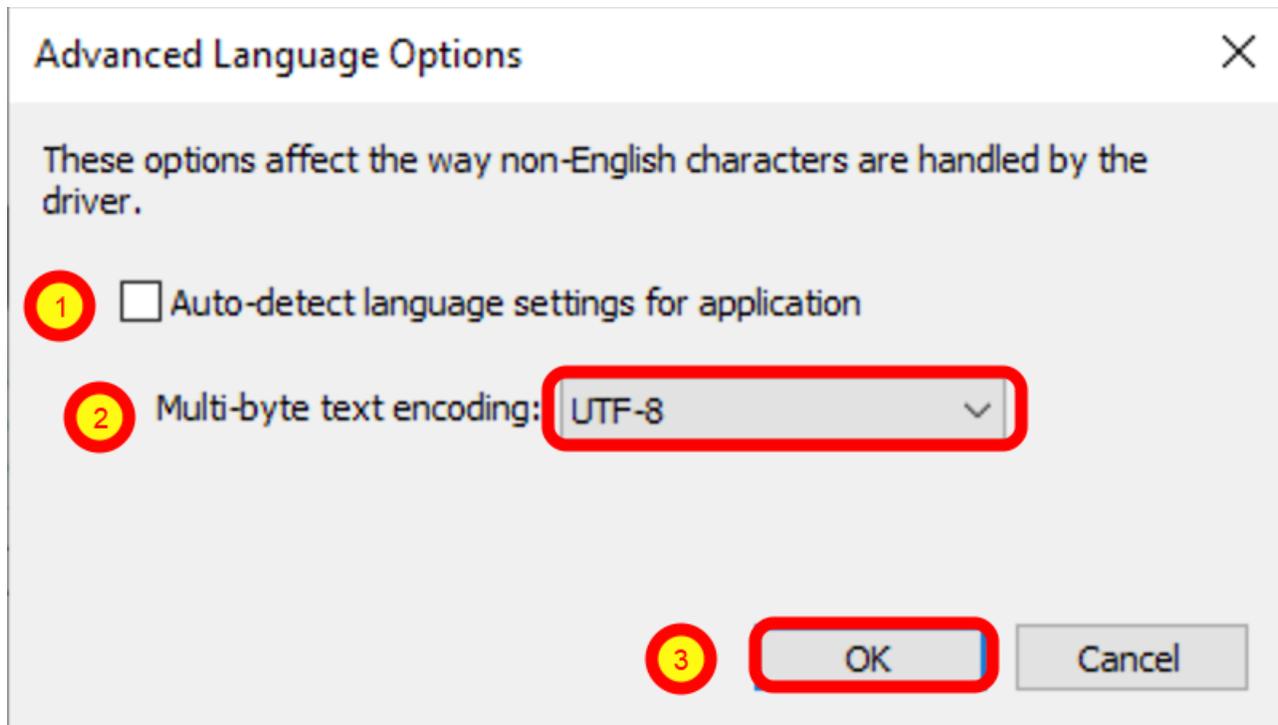
3) Click the Next button.

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



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.

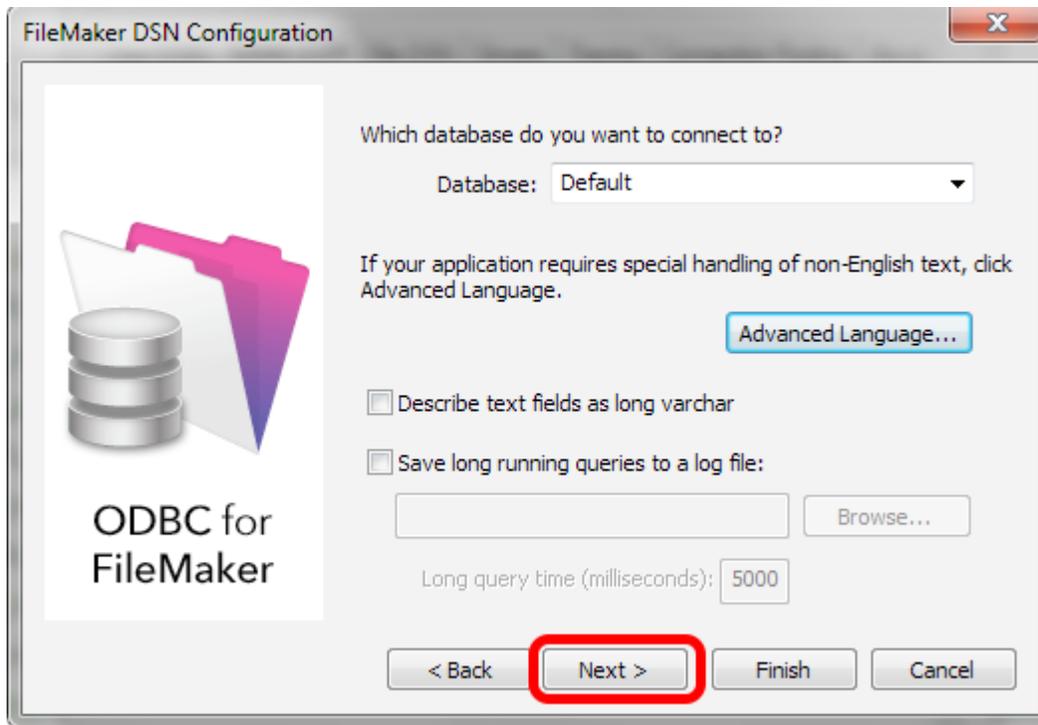


1) Uncheck the "Auto-detect language settings" button, 2) select System (WinLatin1) for the text encoding, 3) click the Ok button.

It has been found that for proper character encoding, the following configuration works well:

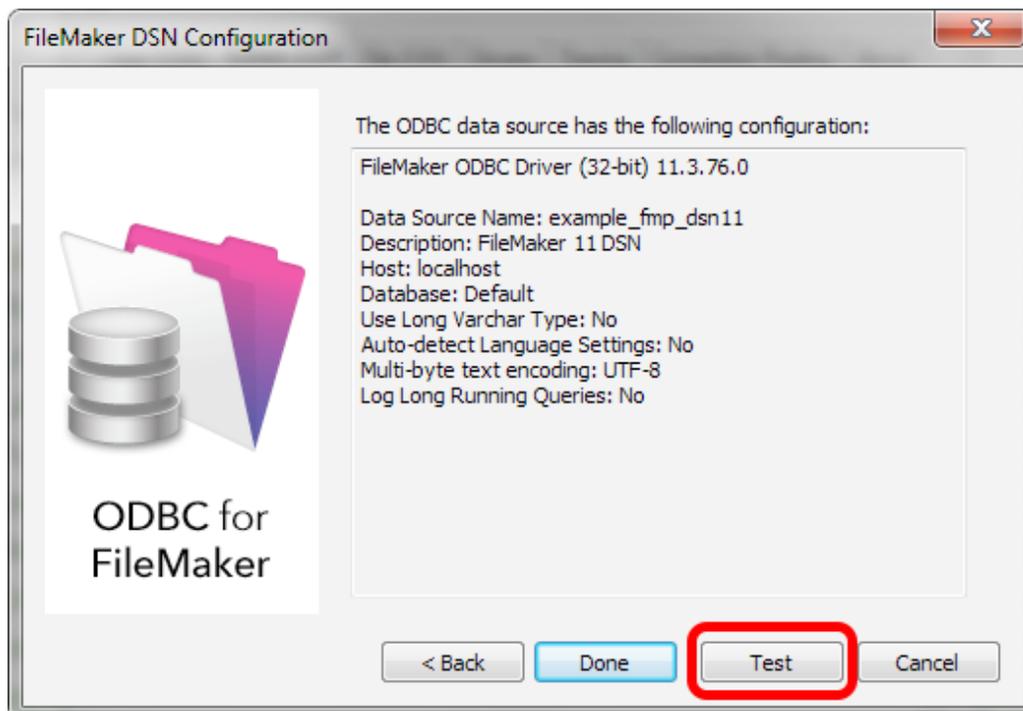
- 1) Install matching platform versions (64-bit or 32-bit) of the FileMaker database and FmPro Migrator on Windows.
 - 2) In the FileMaker ODBC driver, disable the Auto-Detect Language checkbox in the Advanced Language dialog.
 - 3) Set the Multi-byte Text Encoding menu to UTF-8 in the Advanced Language dialog.
- For SQL Server:
- 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)



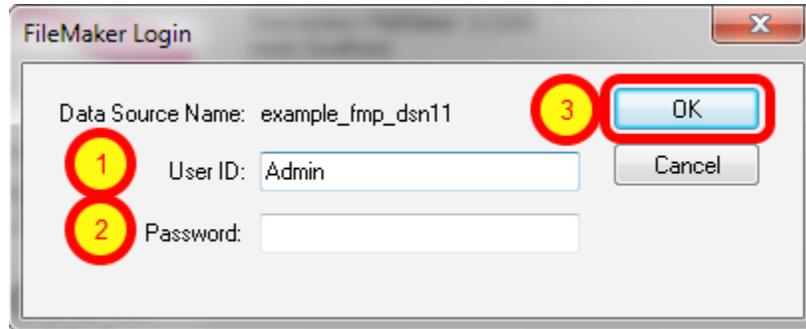
Click the Next button.

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



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.

ODBC Error Troubleshooting Steps:

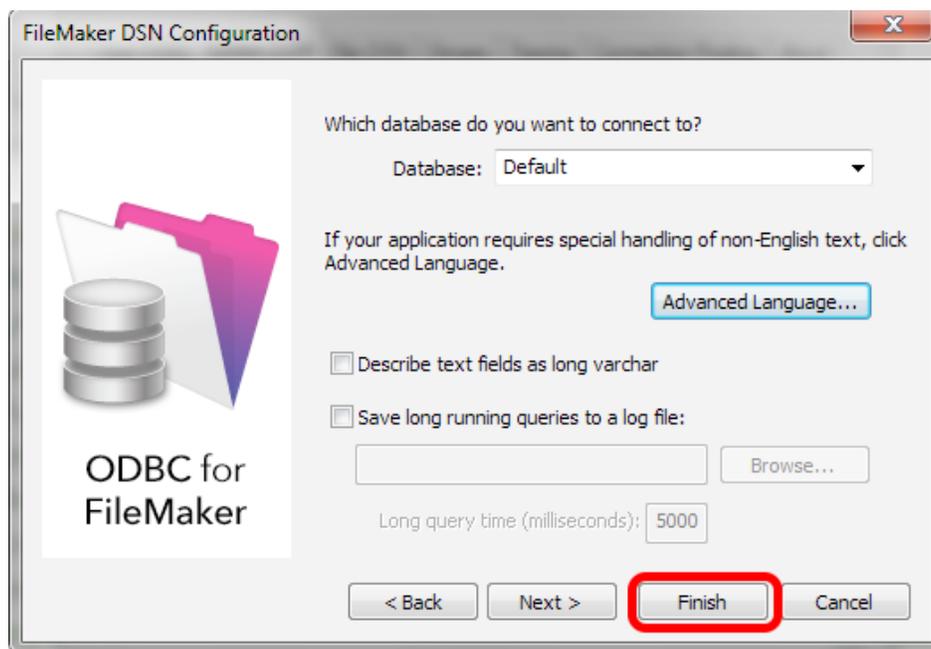
1) Verify that the FileMaker database file is open.

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)

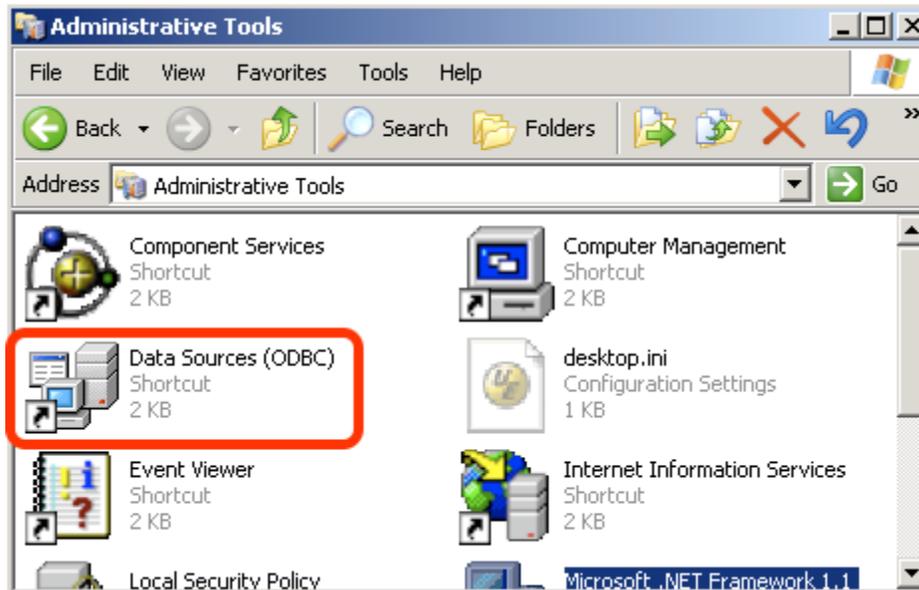


Click the Finish button.

Create Access ODBC DSN - Windows

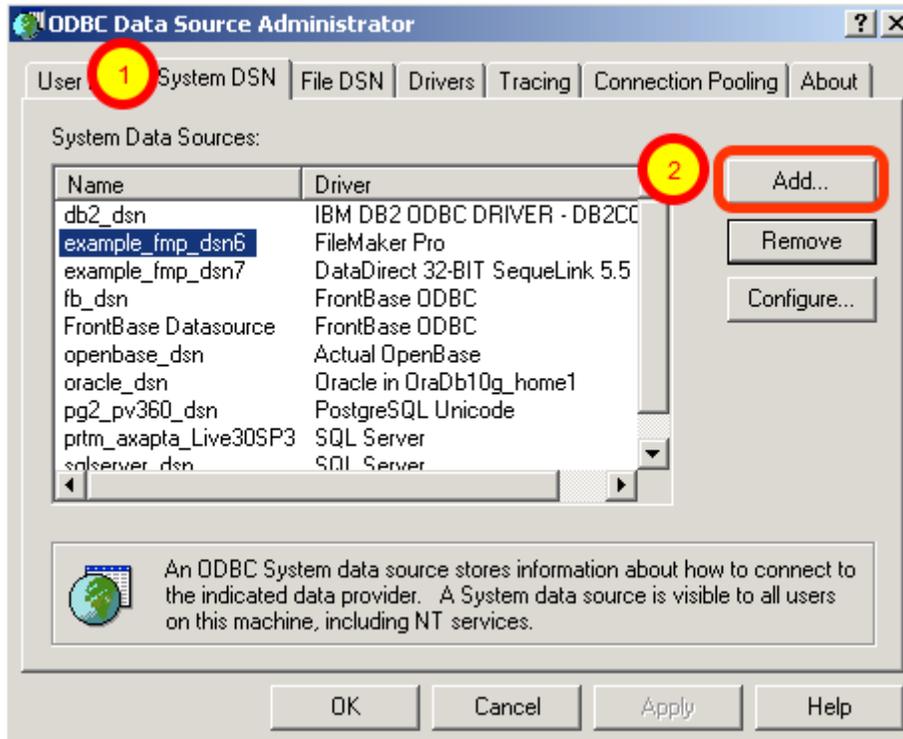
The Microsoft Access ODBC driver is installed automatically with every copy of the Windows operating system. This process shows how to create a System DSN to connect to an Access database (any version of Access).

Open ODBC Manager Control Panel



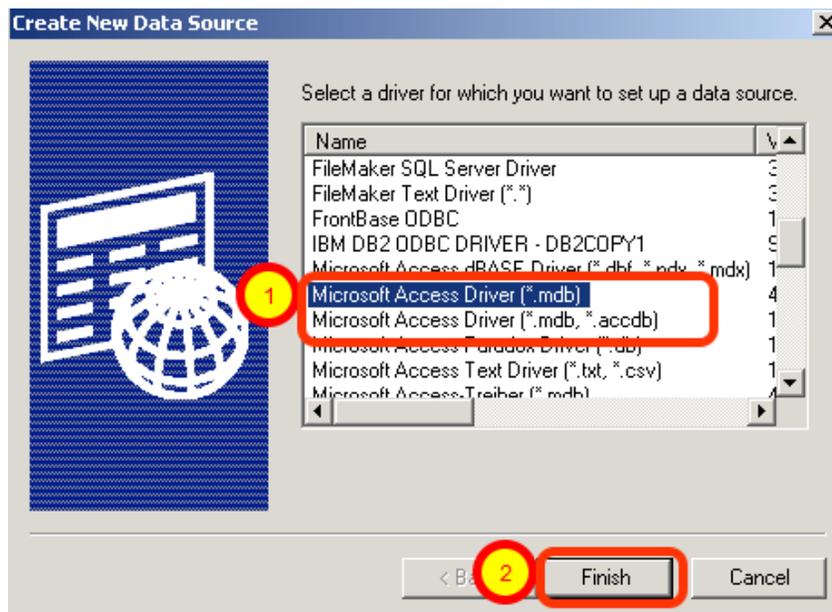
Open the ODBC Data Source Administrator Control Panel, located in Control Panel -> Administrative Tools.

Create New System ODBC DSN



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

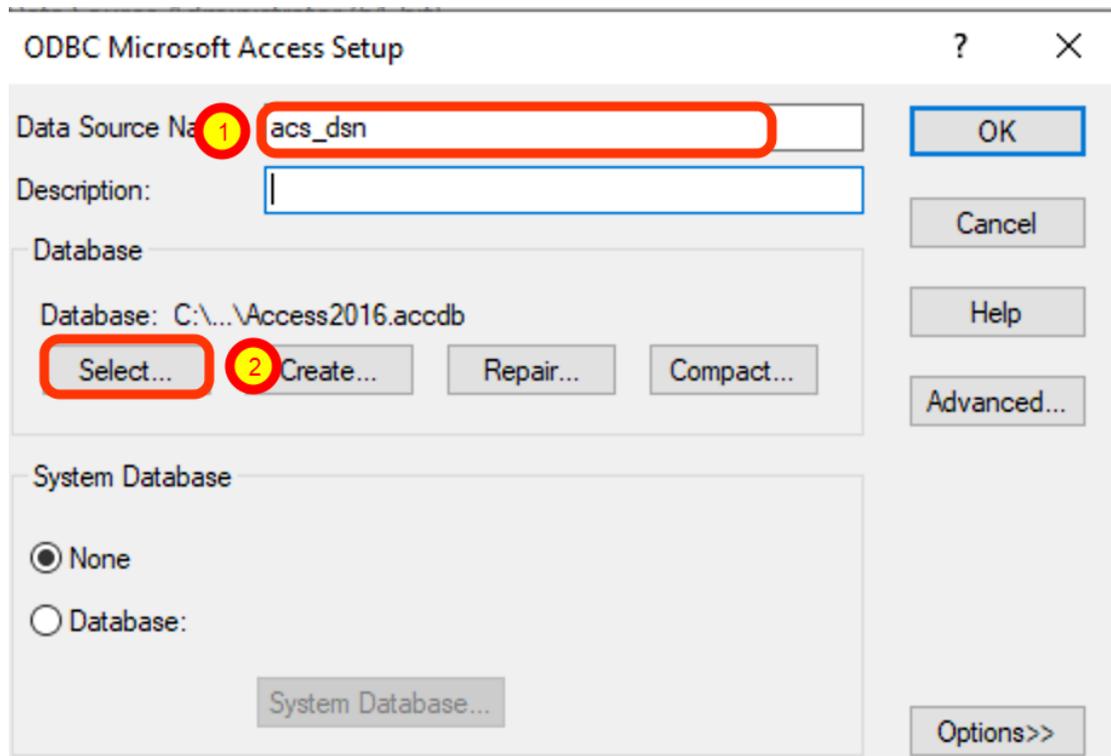
Select Microsoft Access Driver (*.mdb)



(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.

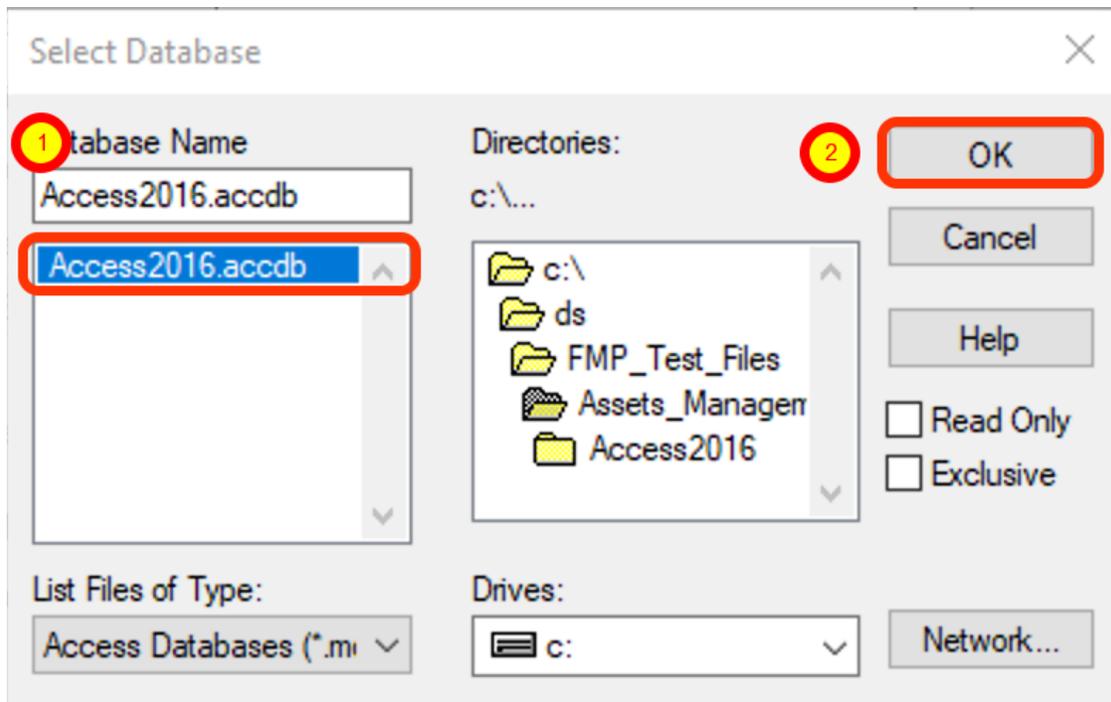
Enter ODBC DSN Name



(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



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

Step 1

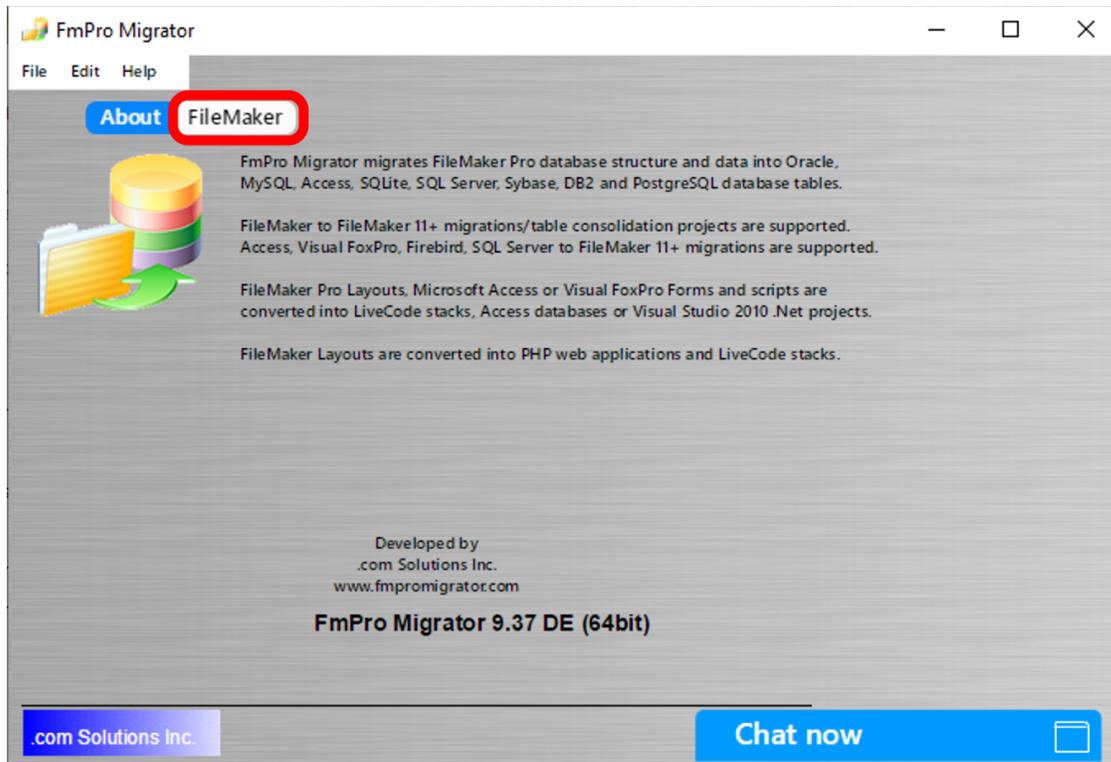
Step 1 - Get Info - FileMaker to Access

Step 1 - Install FileMaker ODBC Driver

FmPro Migrator uses the FileMaker ODBC driver provided by FileMaker Inc. to transfer data from FileMaker databases.

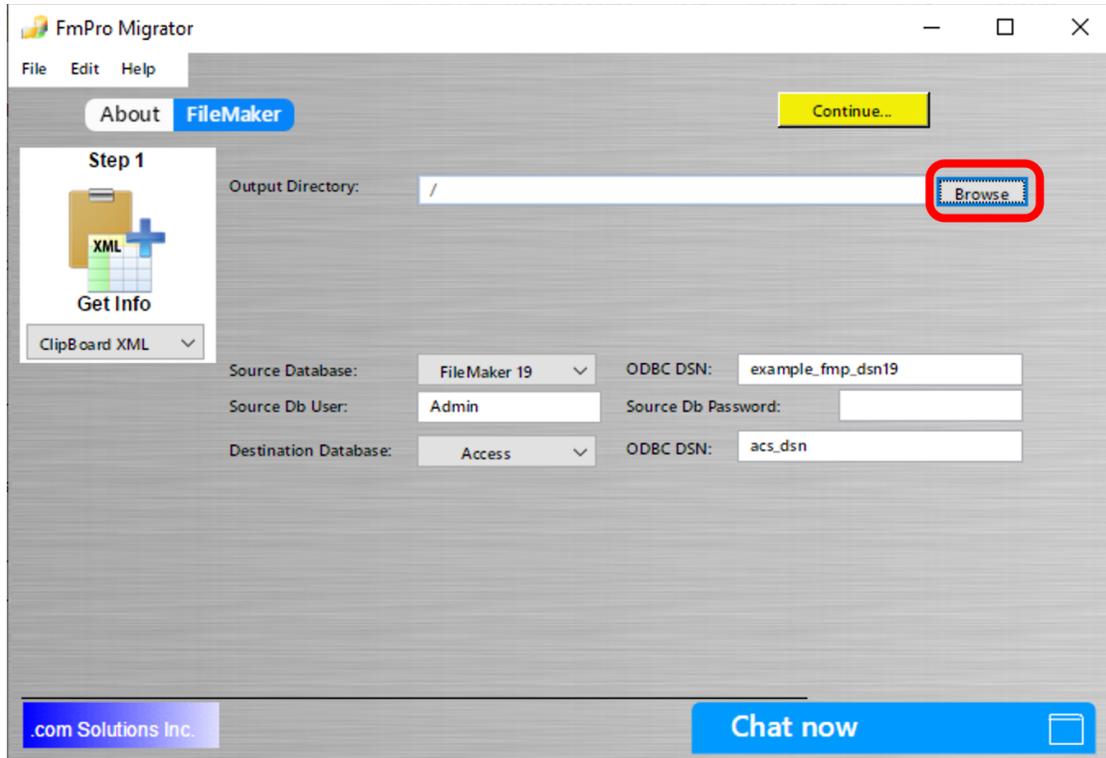
ODBC driver and ODBC DSN creation info is documented in previous chapters of this manual.

Step 1 - Get Info - Select FileMaker tab



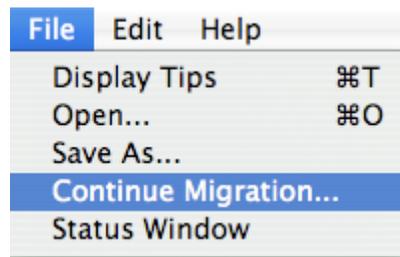
Launch FmPro Migrator, then click on the FileMaker tab at the top of the window.

Step 1 - Get Info - Select Output Directory



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.

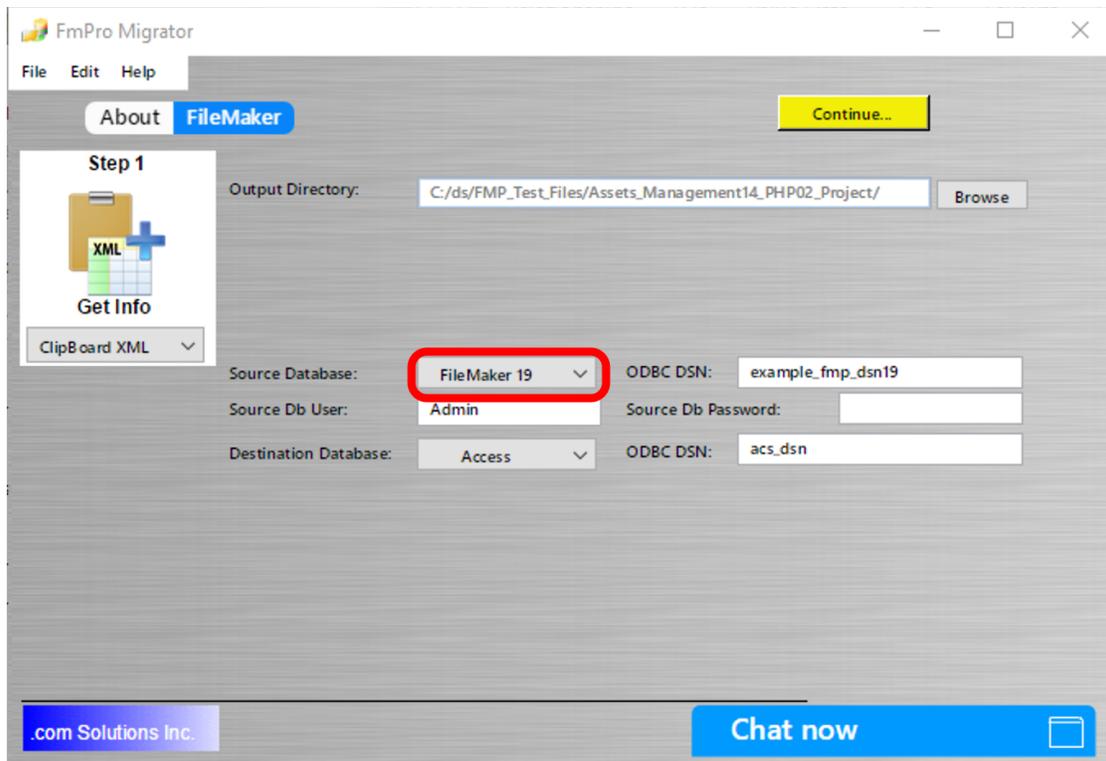
Continue Migration...



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



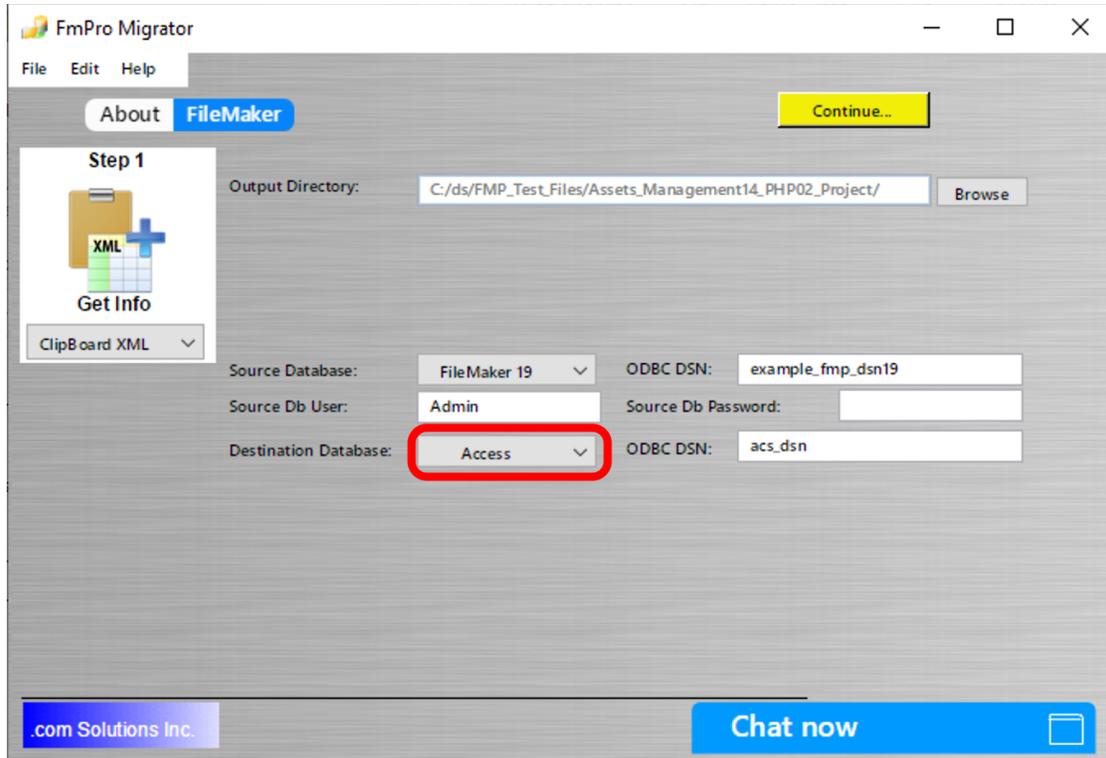
Select the type of source database from the Source Database menu.

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



(1) Select Access as the destination database type.

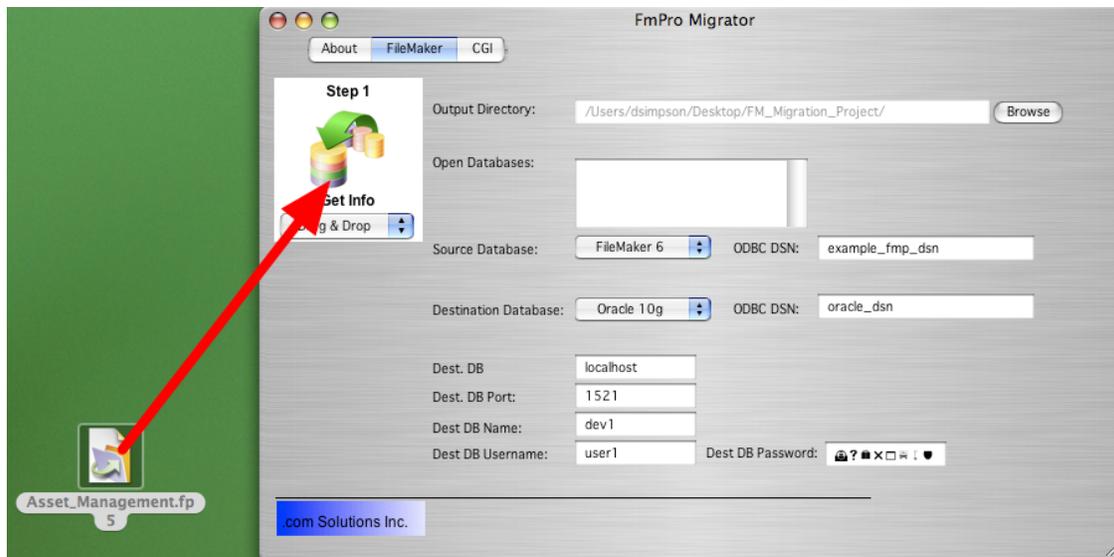
Step 1 - Get Info

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



The Drag & Drop Get Info method is only used for older FileMaker versions:

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).

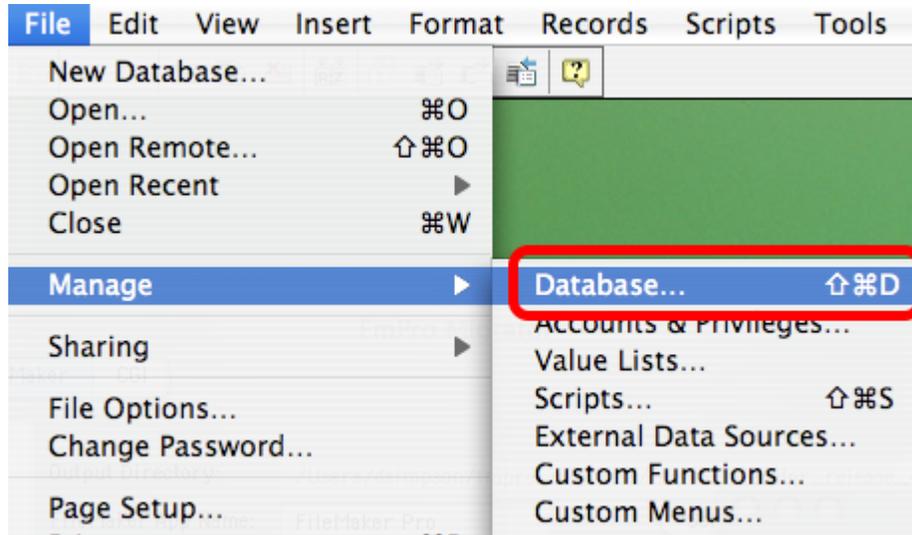
Drag & Drop the empty copy of the source database file onto the Step 1 Get Info icon.

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 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

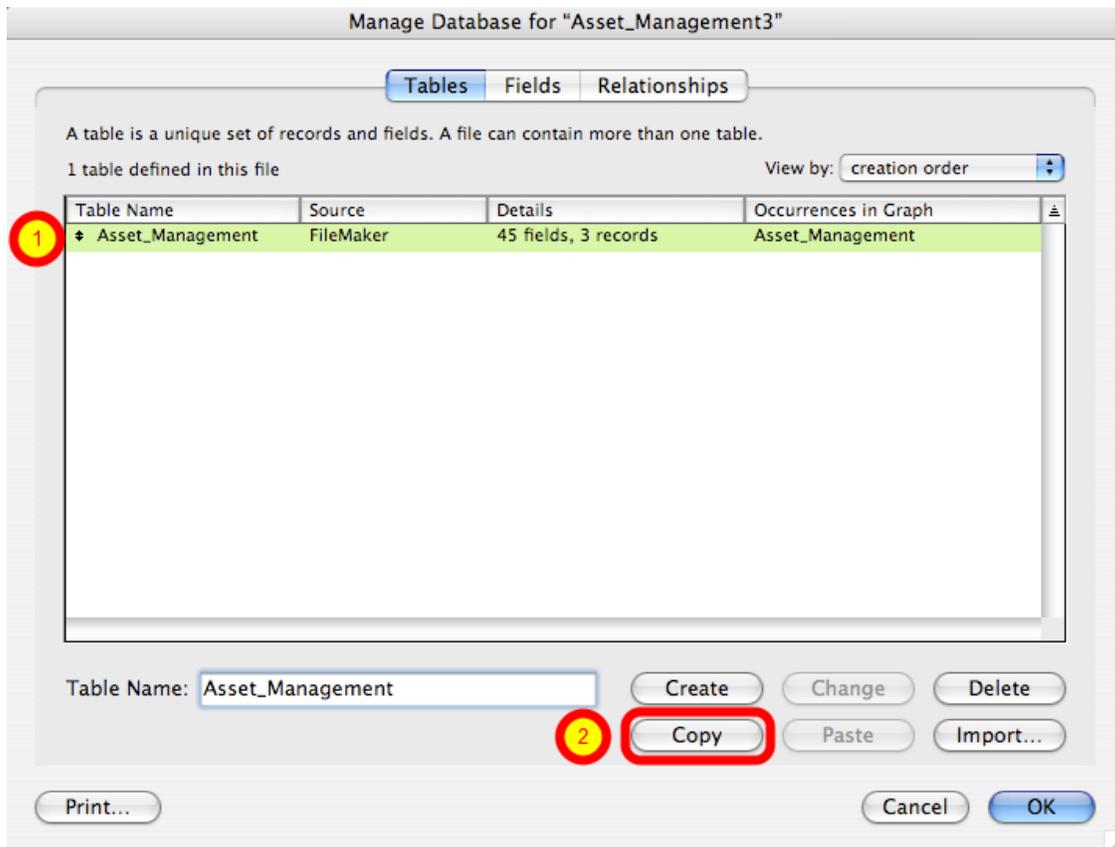


The most desirable method for obtaining info about FileMaker 7+ database files on macOS and Windows is thru copying the table structure using FileMaker Pro (version 17+). This method copies the table creation XML code from the clipboard and includes all of the info required to create the tables, fields,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

Open the FileMaker database file.

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

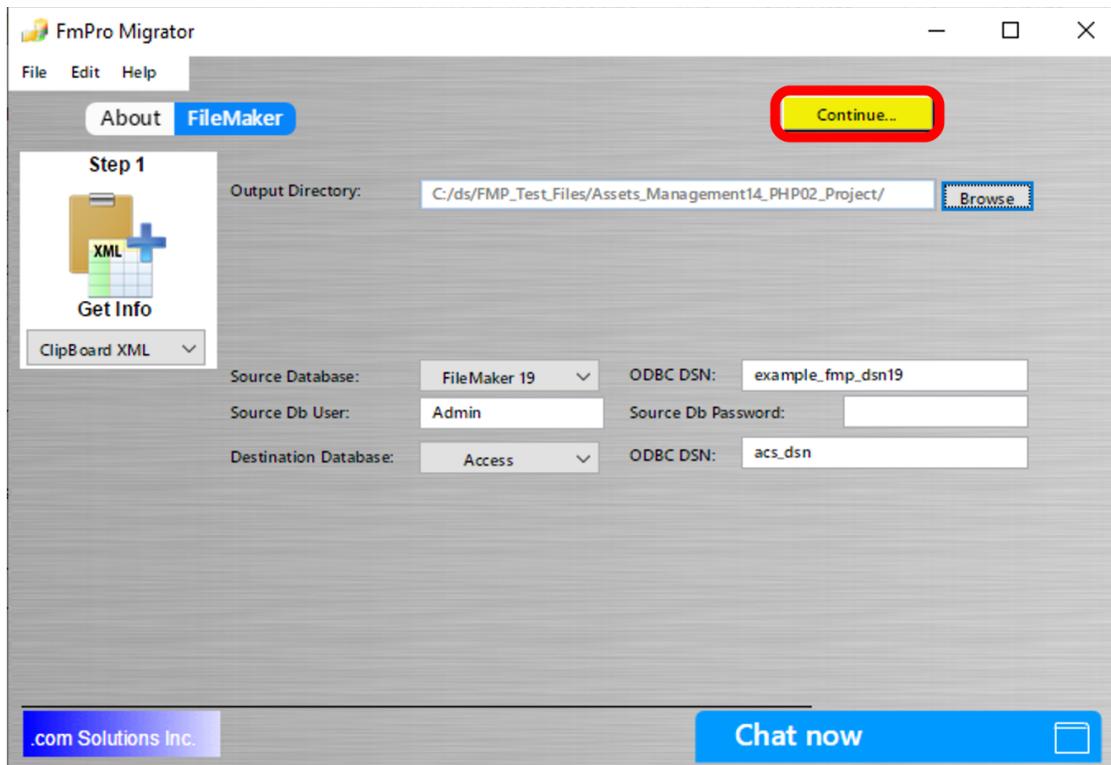
Step 1 - Get Info - Copy Table XML



(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

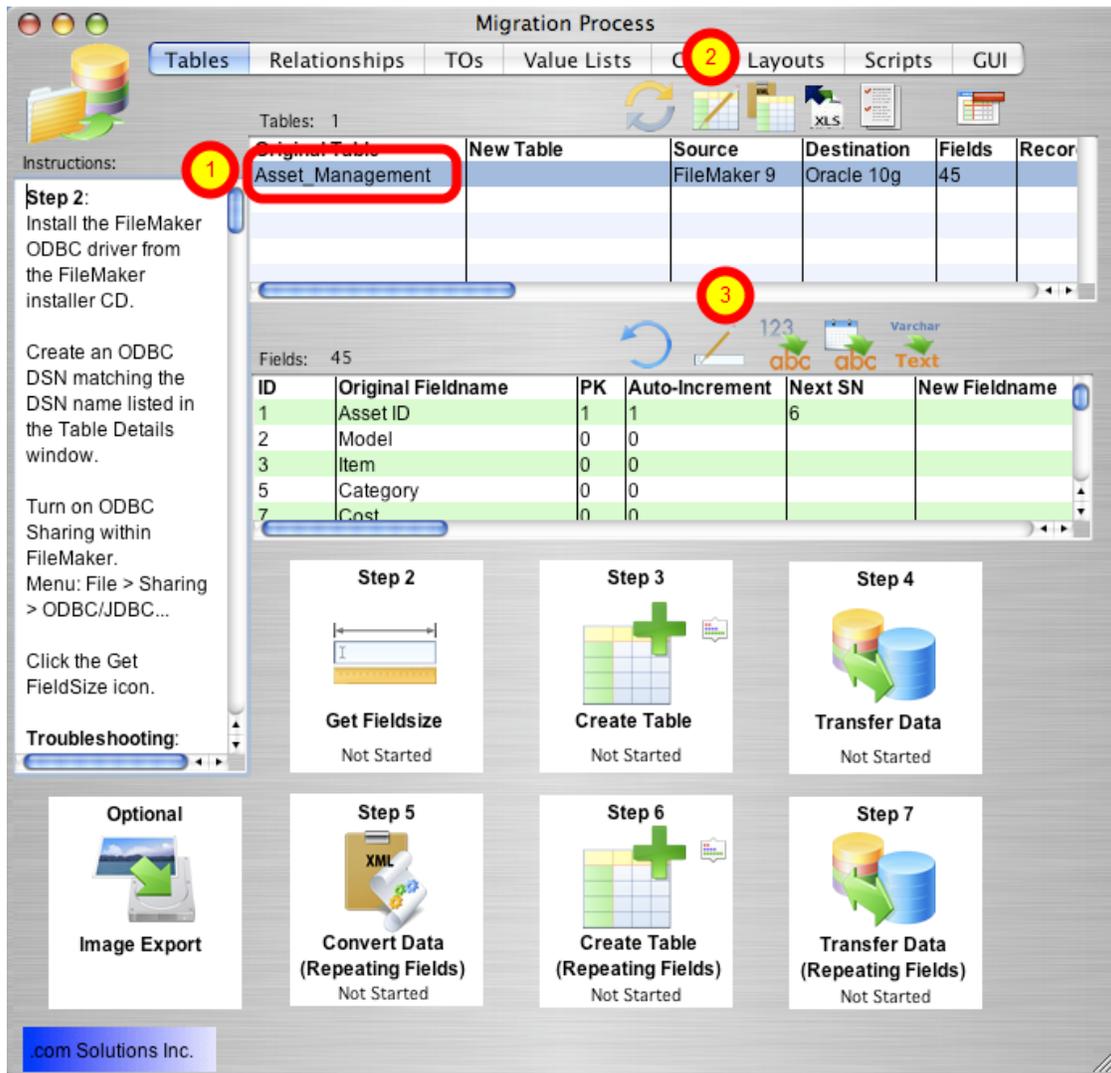


Within FmPro Migrator, click the Step 1 Get Info button (with the ClipBoard XML menu option selected). The yellow Continue... button will become visible once FmPro Migrator has completed processing the Table XML from the ClipBoard.

Step 2

Step 2 - Get Fieldsize

Step 2 - Get Fieldsize - Window Features



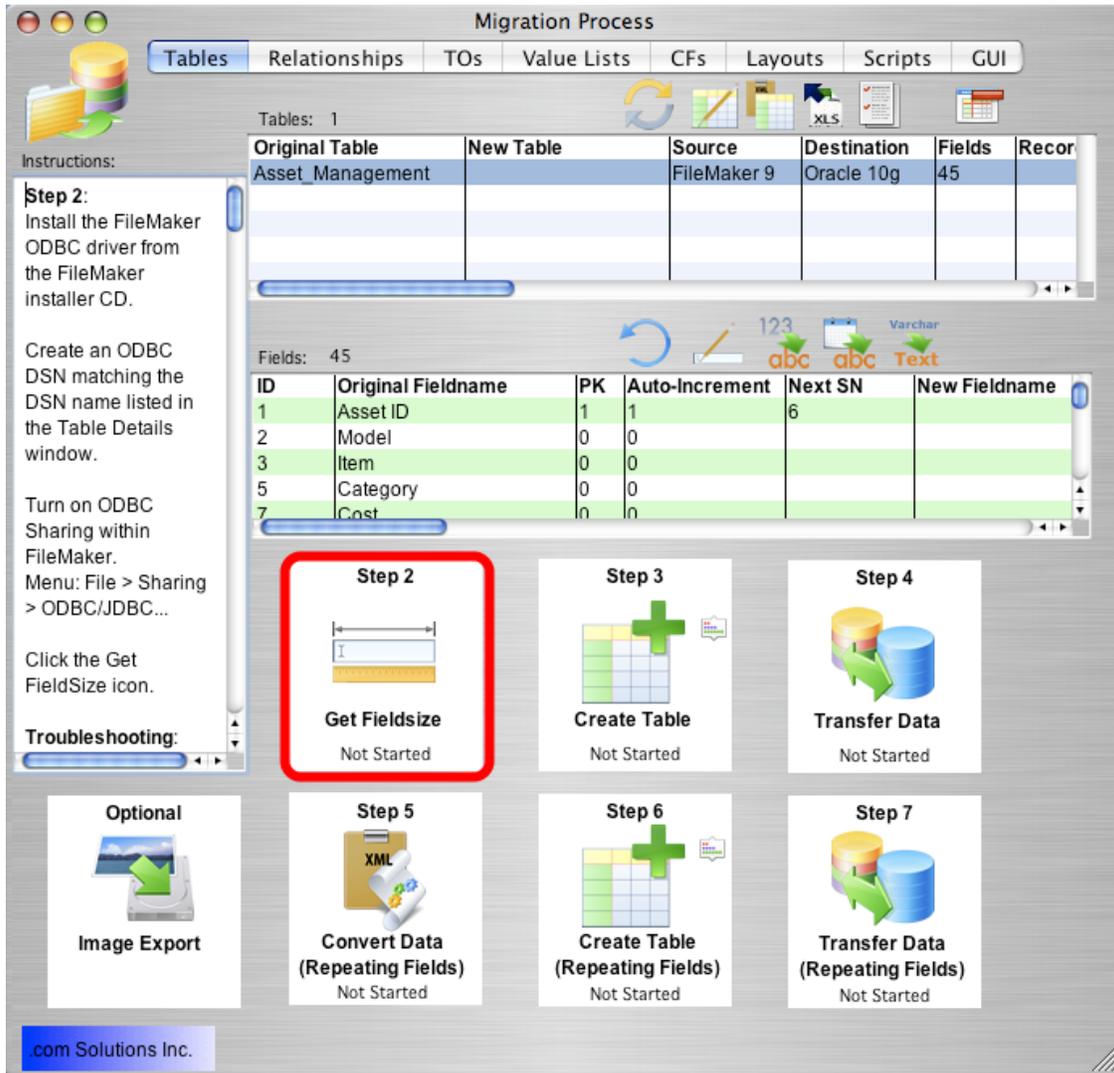
Note: Prior to clicking the Step 2 Get Fieldsize step, an ODBC DSN should have already been created and tested to make sure that FmPro Migrator can communicate with the FileMaker database. It is recommended that this process be done on the local computer to read the FileMaker database instead of running 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



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

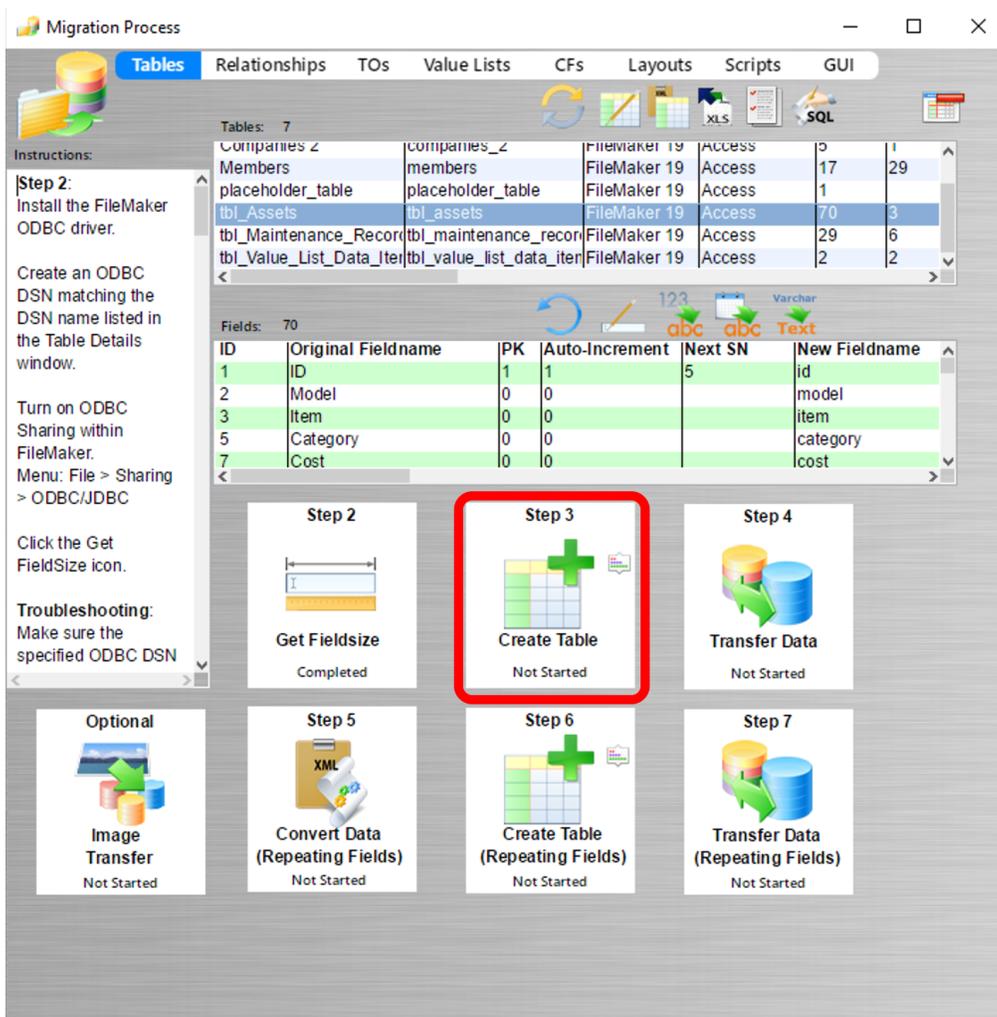
Step 3 - Create Table - FileMaker to Access

Once the Get Field size step has been completed, FmPro Migrator has enough information to create the SQL code defining the table in the destination database. The field size 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



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.

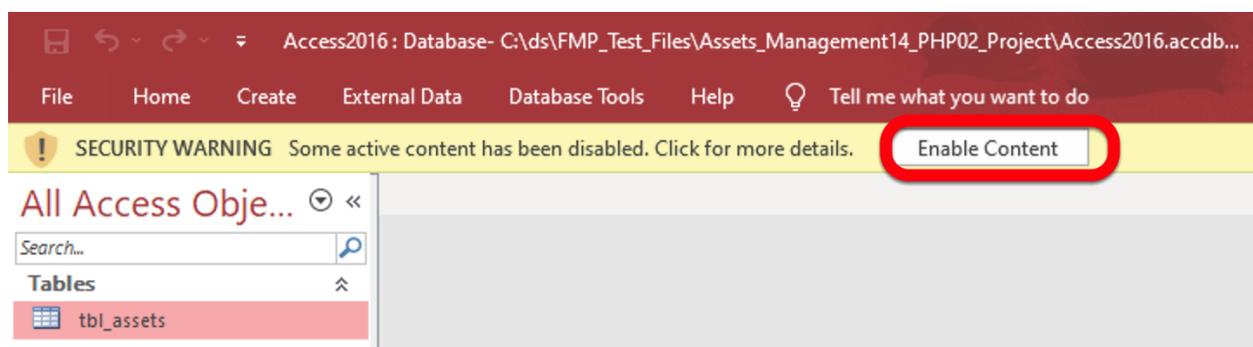
Access2016.accdb File in Output Directory

Name	Date modified	Type
 Access2016.accdb	10/4/21 10:51 AM	Microsoft Acc
 FileDSN002.dsn	10/4/21 10:51 AM	DSN File
 MigrationPreferences.dat	10/4/21 10:51 AM	DAT File
 MigrationProcess.db3	10/4/21 10:51 AM	DB3 File

The first time the Step #3 button is clicked, FmPro Migrator will automatically save the Access2016.accdb database file within the output directory. After the file has been saved, the table(s) will be created within this file.

Note: Don't rename this Access database file until the conversion process has been completed, or FmPro Migrator will create another one - and it will be confusing because some tables may end up in one file and the rest of the tables would be created within the newly created Access2016.accdb file.

Opening the Access2016.accdb File



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

Note: It is not a good idea to open the Access2016.accdb file until all of the tables have been

created and the data has been transferred. Access may lock the tables, preventing them from being used by an external program like FmPro Migrator. It is Ok to open the file, look at the results and then close the file before creating more tables or transferring data.

Step 3 - Table Creation SQL Details

Migration Process

Tables: 2

Original Table	New Table	Source	Destination	Fields	Records
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

Step 2: Get FieldSize (Completed)

Step 3: Create Table (Failed)

Step 4: Transfer Data (Not Started)

Optional: Image Export

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

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

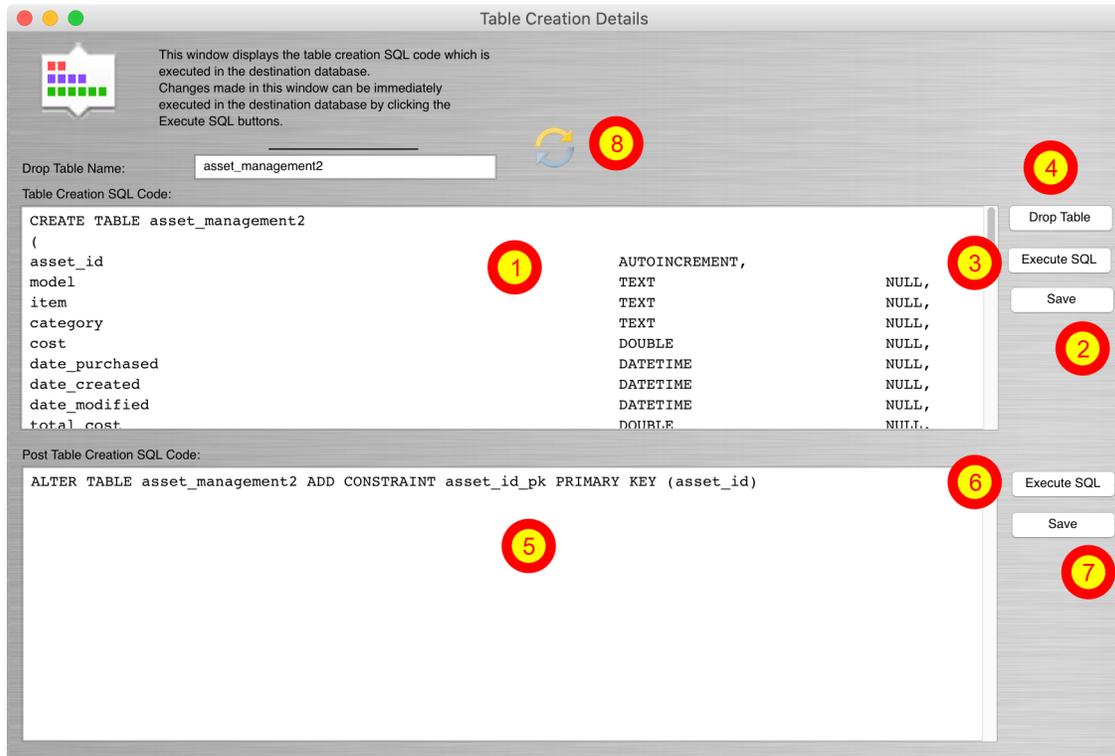
Step 7: Transfer Data (Repeating Fields) (Not Started)

123 abc abc Varchar Text

.com Solutions Inc.

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



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.

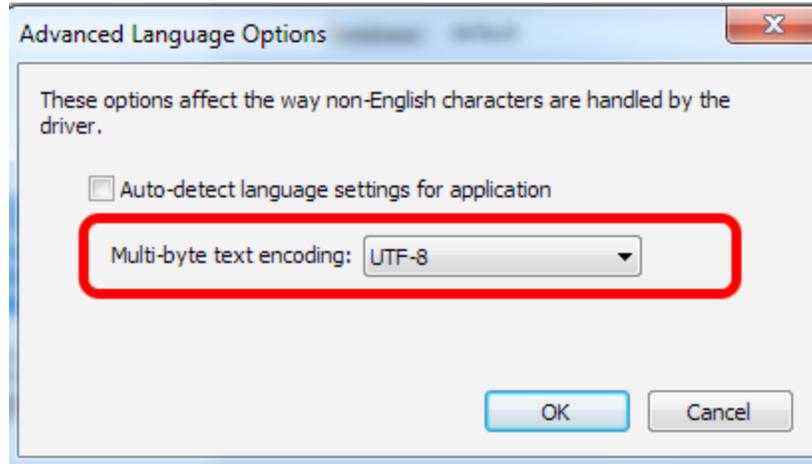
The Post Table Creation SQL code can also be (5) manually edited, (6) executed, and (7) saved.

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

Step 4 - Transfer Data - Unicode Characters



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

The screenshot shows the 'Migration Process' window with several components:

- Navigation Tabs:** Tables, Relationships, TOs, Layouts, Scripts, GUI.
- Tables:** 1 table selected: 'Asset_Management3' with 44 fields and 3 records. Source: FileMaker 9, Destination: MySQL.
- Fields:** 44 fields listed. The first few are: Asset ID (PK, Auto-Increment 1, Next SN 6, New Fieldname Asset_ID), Model, Item, Category, and Cost.
- Instructions Panel (Left):**
 - 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:** (Empty)
- Step Progress:**
 - Step 2: Get Fieldsize (Completed)
 - Step 3: Create Table (Completed)
 - Step 4: Transfer Data (Not Started)** - This step is highlighted with a red box.
 - 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)
- Footer:** .com Solutions Inc.

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

The screenshot displays the 'Migration Process' software interface. At the top, there are tabs for 'Tables', 'Relationships', 'TOs', 'Layouts', 'Scripts', and 'GUI'. The 'Tables' tab is active, showing a table with the following data:

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

Below the table, there are icons for 'abc', '123', and 'Varchar Text'. The 'Fields' section shows a list of fields:

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

On the left side, there is an 'Instructions' panel with the following text:

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:

The main interface shows a progress bar with seven steps:

- Step 2: Get FieldSize (Completed)
- Step 3: Create Table (Completed)
- Step 4: Transfer Data (Failed) - This step is circled in red.
- Step 5: Convert Data (Repeating Fields) (Not Started)
- Step 6: Create Table (Repeating Fields) (Not Started)
- Step 7: Transfer Data (Repeating Fields) (Not Started)

At the bottom left, there is an 'Optional' section with 'Image Export' and a 'com Solutions Inc.' logo.

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	Records
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 Fields Size (Completed)

Step 3: Create Table (Completed)

Step 4: Transfer Data (Failed)

Optional: Image Export (Not Started)

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.

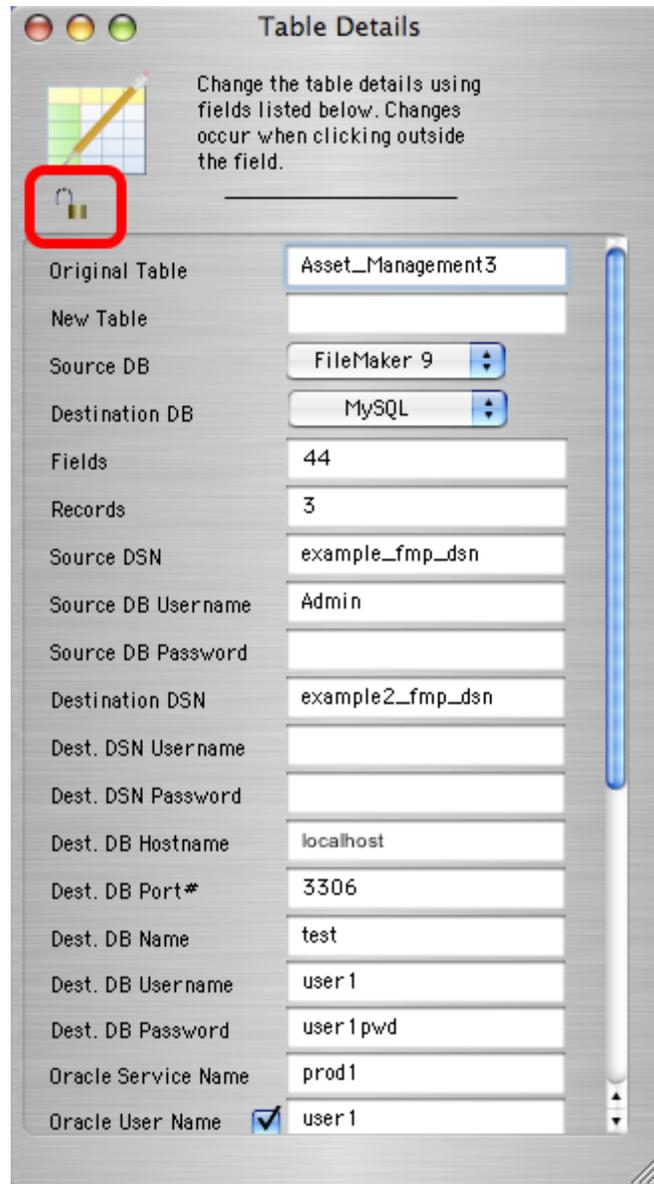
If this type of error occurs:

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



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.

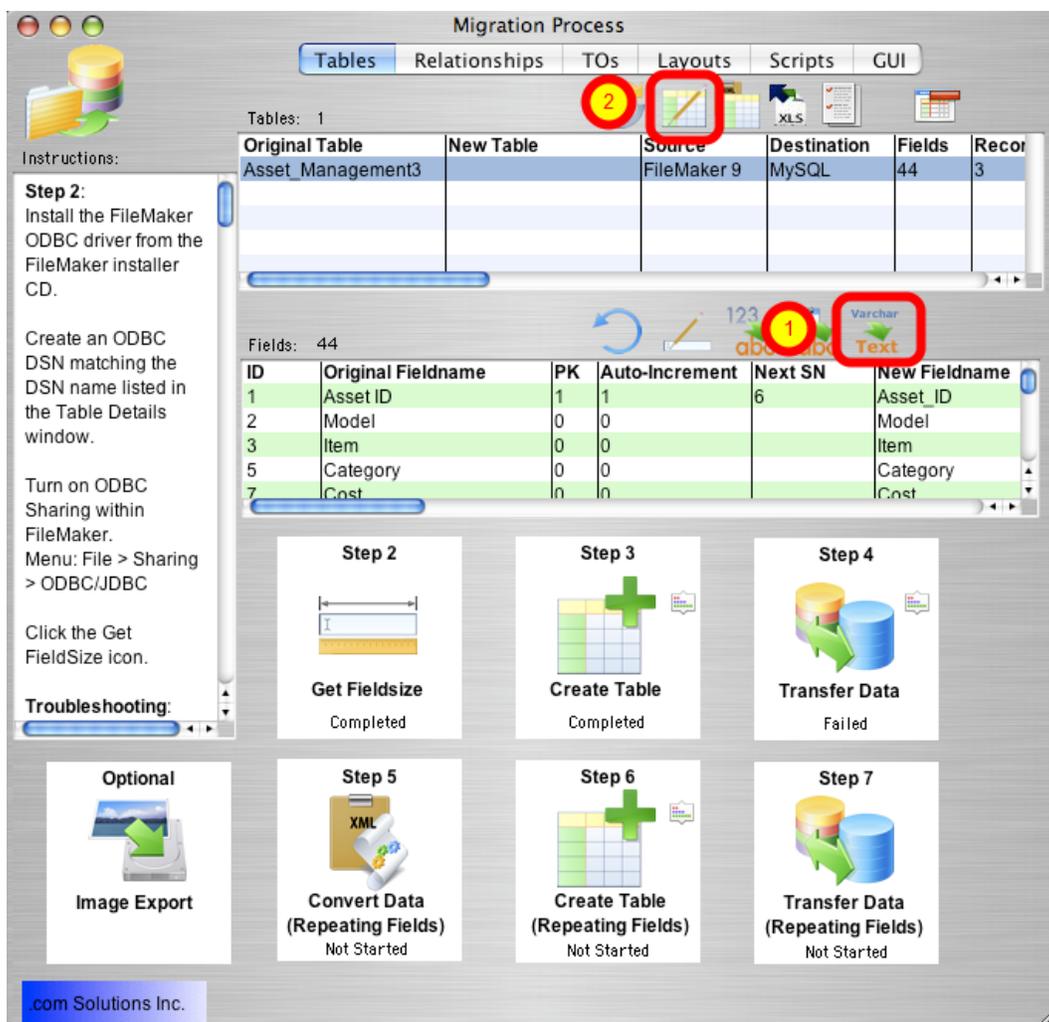
Note: You must generate table creation SQL code by clicking the Step 3 Create Table button at least once, before clicking the lock icon on the Table Details window.

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

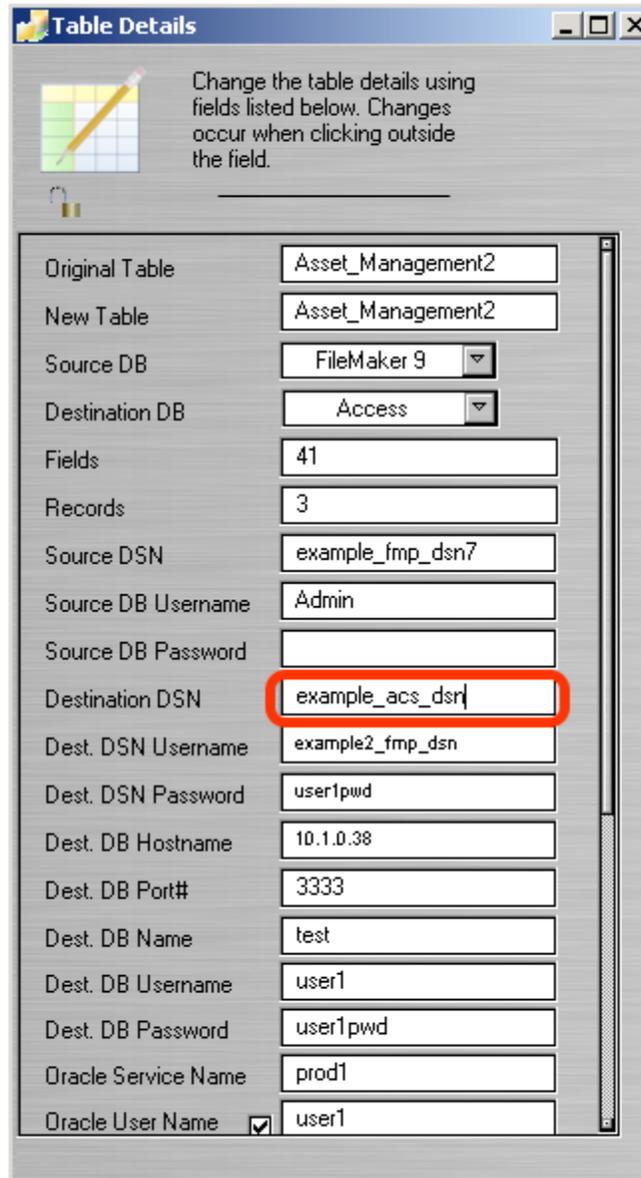


It is not usually necessary to change Varchar columns to TEXT column types in the destination database. But occasionally you might need to perform this task. If you wanted to perform a quick migration, without running the Get Fieldsize step, you could just convert all of the Varchar columns

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 <input checked="" type="checkbox"/>	user1

Error Message: Count Error (1).

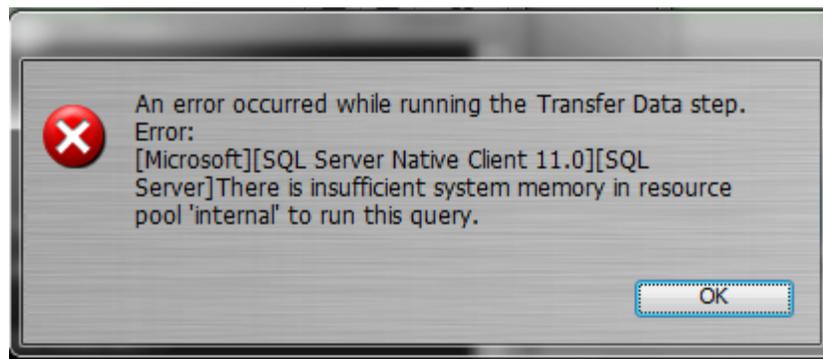
If an unexpected error occurs during the data transfer process, there is an alternate data transfer

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



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

The screenshot shows the 'Migration Process' window with the 'Tables' tab selected. The 'Tables' table lists 'Asset_Management3' with 44 fields and 3 records. The 'Fields' table below it lists fields with their PK status, and the 'Asset ID' field is highlighted with a red circle. The 'Step 5' button is visible and labeled 'Convert Data (Repeating Fields) Not Started'.

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

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 (Completed)
Optional: Image Export
Step 5: Convert Data (Repeating Fields) (Not Started)
Step 6: Create Table (Repeating Fields) (Completed)
Step 7: Transfer Data (Repeating Fields) (Not Started)

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.

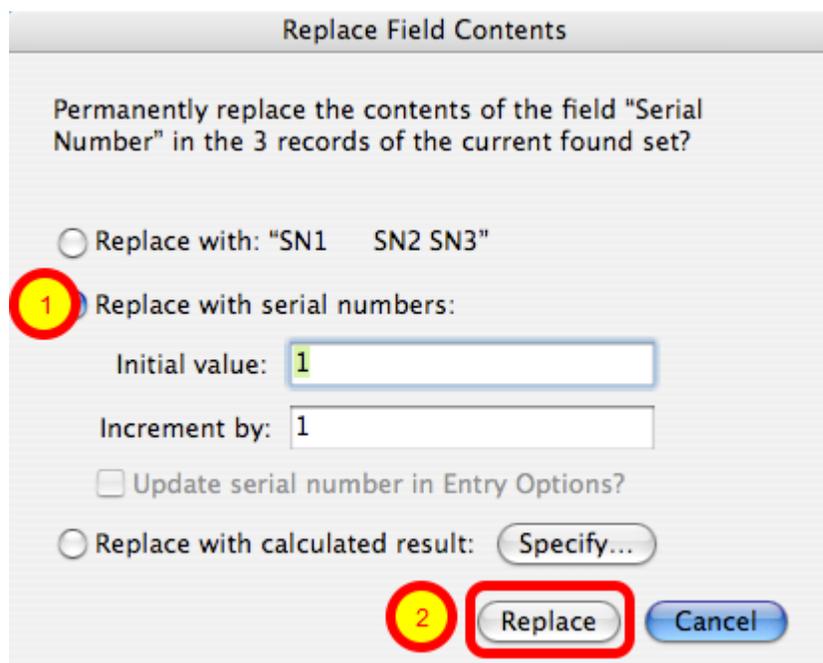
If the Primary Key field contains empty or non-unique values, then the data needs to be corrected, or a new Primary Key field needs to be defined. Only one field should be defined as the Primary Key, and this field should be defined as a numeric field, as SQL databases don't generally handle Auto-Incrementing 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



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

Migration Process

Tables: 1

Original Table	New Table	Source	Destination	Fields	Records
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:

Optional:
Image Export

Step 2: Get FieldSize (Completed)

Step 3: Create Table (Completed)

Step 4: Transfer Data (Completed)

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

Step 6: Create Table (Repeating Fields) (Completed)

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

com Solutions Inc.

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.

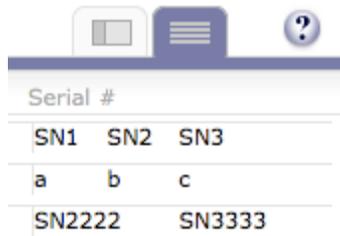
Manually change each of the repeating fields from Numeric or Date/Time format to Text within FileMaker. It is not necessary to re-gather info from FileMaker after making this change.

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



The image shows a FileMaker layout with a repeating field titled "Serial #". The field contains three columns of data: SN1, SN2, and SN3. The first row contains the values 'a', 'b', and 'c'. The second row contains 'SN2222' and 'SN3333'. The columns are separated by vertical lines, but no vertical lines are visible between the first and second rows, indicating that the data has been successfully converted into a single row.

SN1	SN2	SN3
a	b	c
SN2222	SN3333	

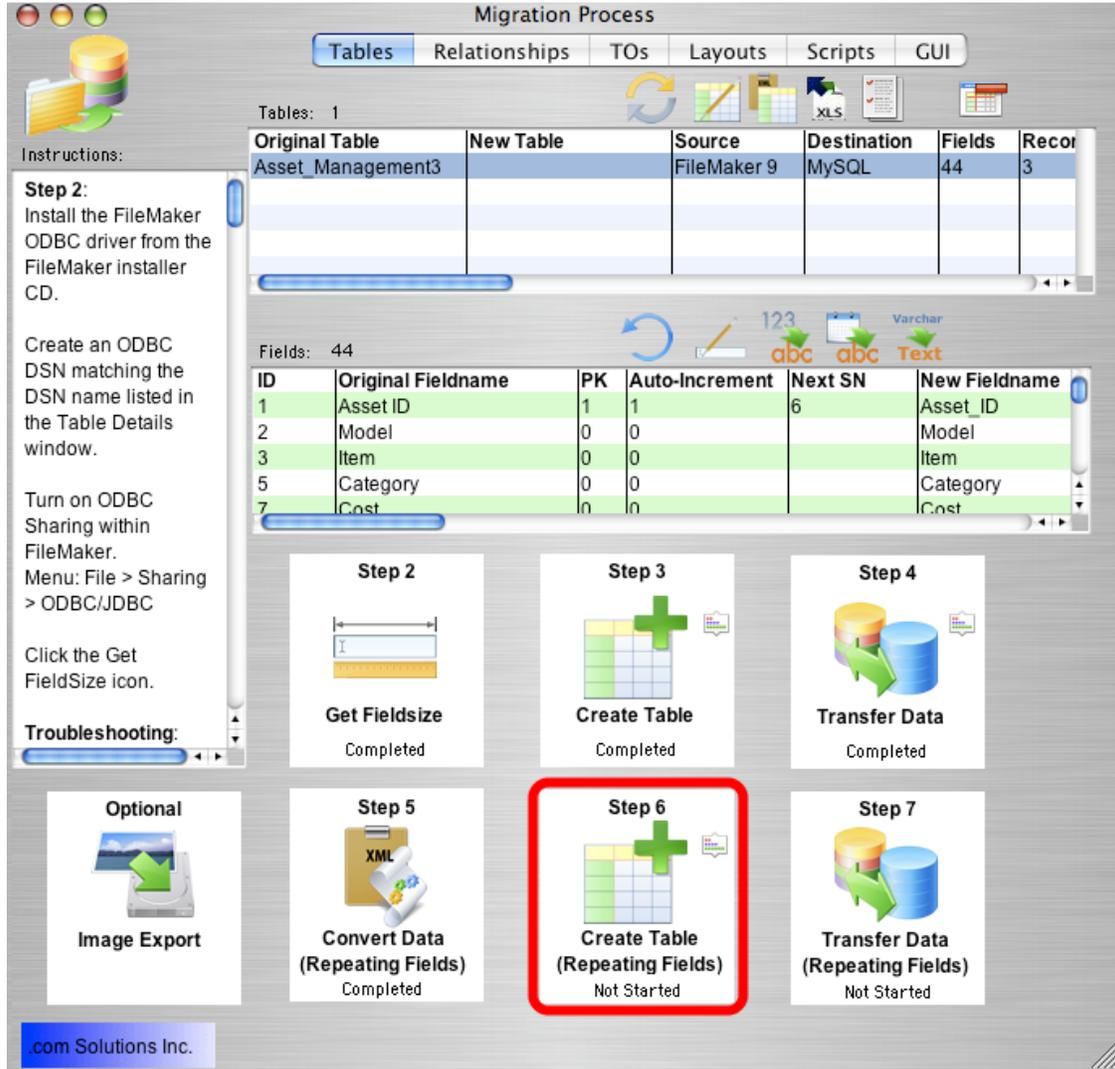
This image shows the contents of a repeating field shown on the FileMaker layout as "Serial #", which has been successfully converted with the RF Convert ScriptMaker script. If the RF convert script had not been run, then there would be vertical lines separating each repeating value. But since all of 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 6 - Create Table (Repeating Fields)



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)

Step 7 - Transfer Data (Repeating Fields)

The screenshot shows the Migration Process software interface. At the top, there are tabs for Tables, Relationships, TOs, Layouts, Scripts, and GUI. Below the tabs, there are two tables. The first table, titled 'Tables: 1', has columns for Original Table, New Table, Source, Destination, Fields, and Records. The second table, titled 'Fields: 44', has columns for ID, Original Fieldname, PK, Auto-Increment, Next SN, and New Fieldname. Below the tables, there are seven step buttons: Step 2 (Get FieldSize), Step 3 (Create Table), Step 4 (Transfer Data), Step 5 (Convert Data (Repeating Fields)), Step 6 (Create Table (Repeating Fields)), and Step 7 (Transfer Data (Repeating Fields)). Step 7 is highlighted with a red border. The status of each step is shown below the button: Step 2, 3, 4, 5, and 6 are 'Completed', while Step 7 is 'Not Started'. On the left side, there are instructions for Step 2 and a Troubleshooting section. At the bottom left, there is an 'Optional Image Export' button and the text 'com Solutions Inc.'

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

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
Completed

Optional
Image Export

Step 5
Convert Data (Repeating Fields)
Completed

Step 6
Create Table (Repeating Fields)
Completed

Step 7
Transfer Data (Repeating Fields)
Not Started

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

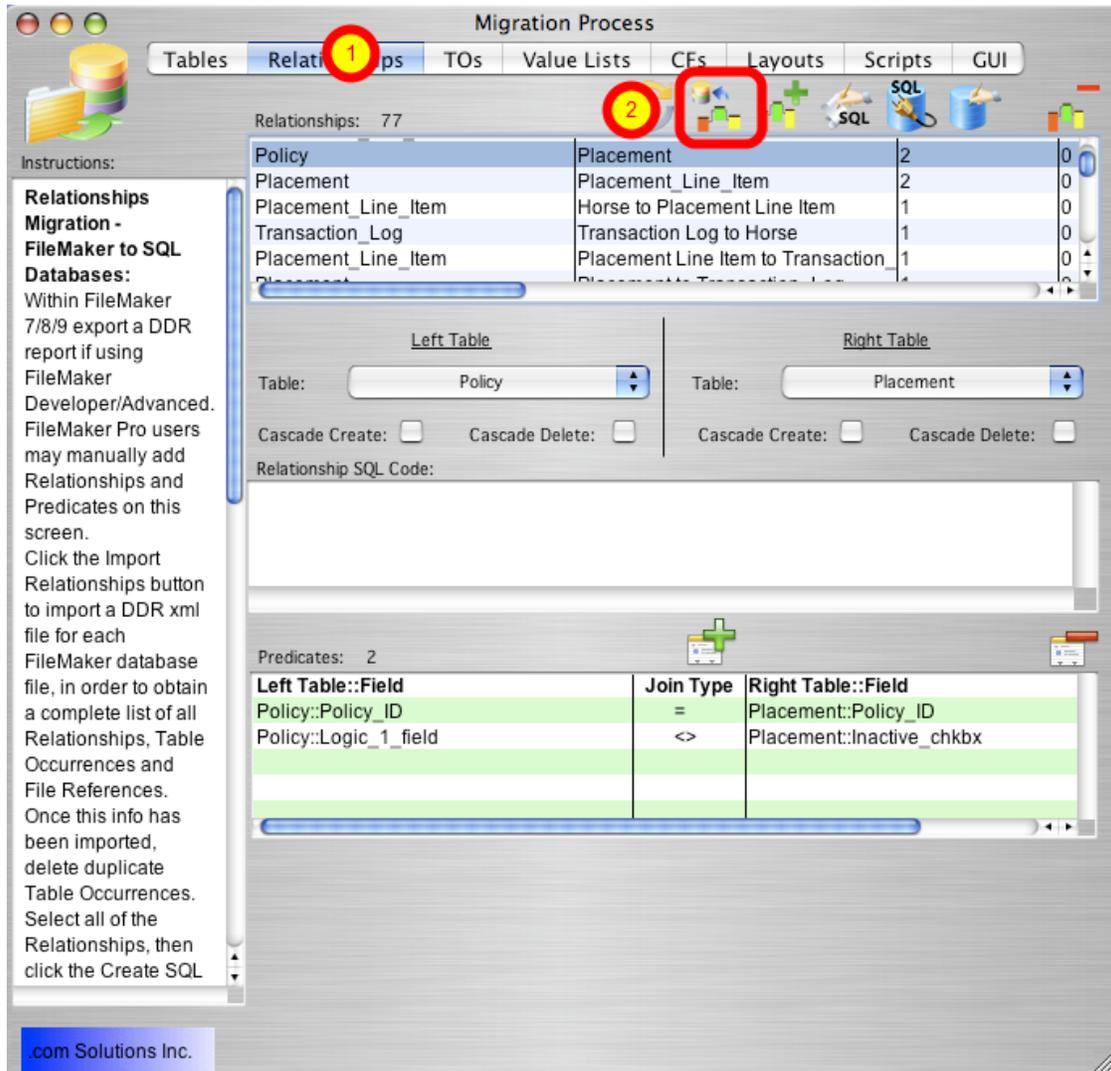
instance, if only 1 repeating value was transferred to the destination table, then this could mean that the RF Convert script wasn't run or the contents of the source table fields weren't converted into Text prior to processing. If either of these problems occur, it is a simple matter to go back and 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.

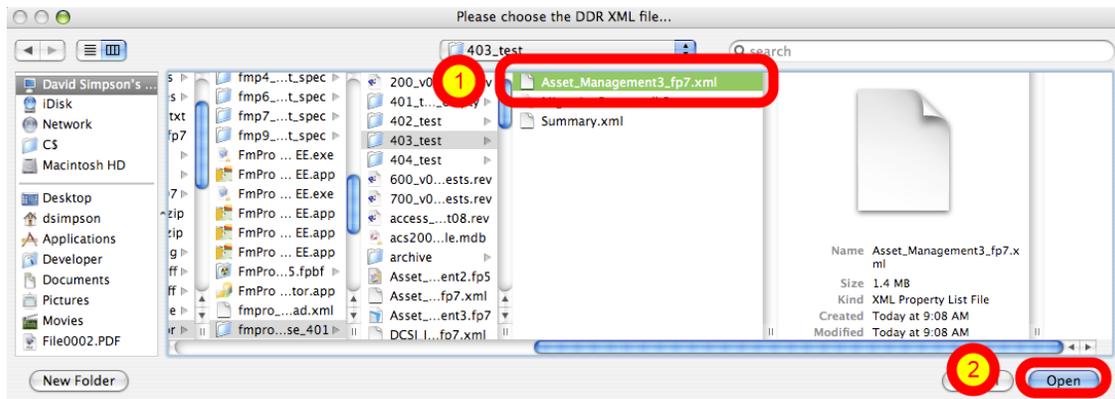
Migrating Relationships - Import DDR XML File



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

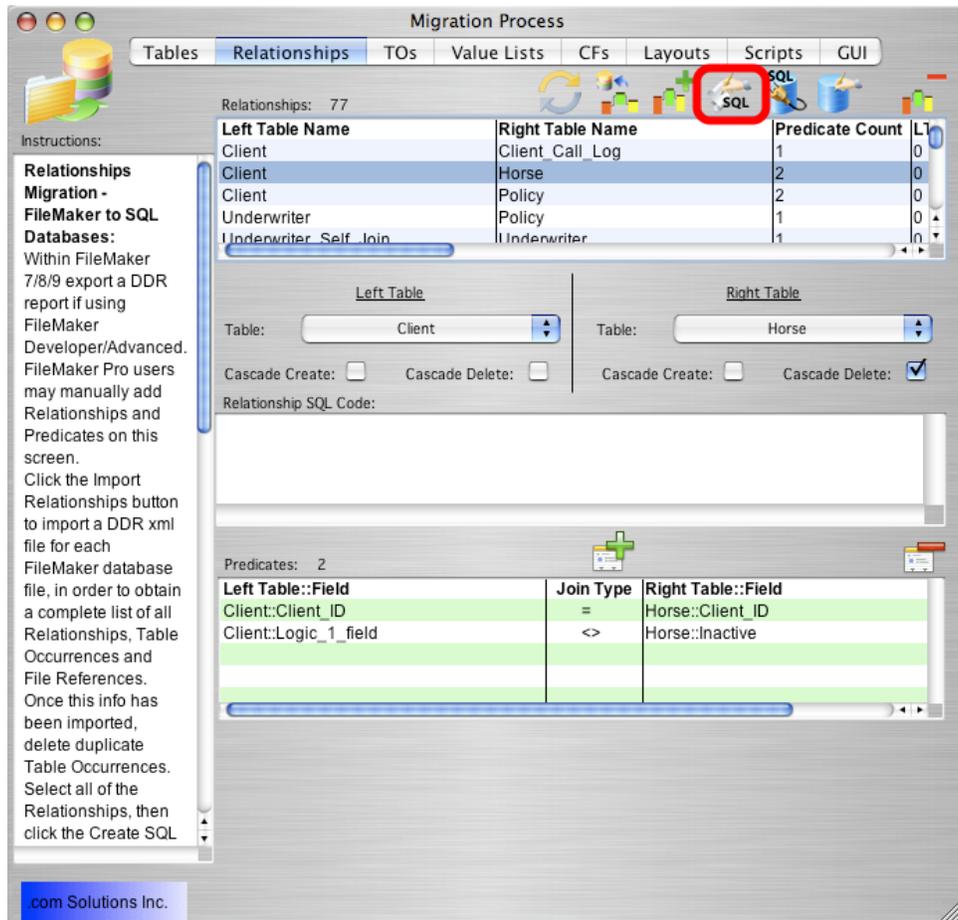
Note: Relationships will only be imported correctly if all of the base tables have already been created within the Tables tab.

Migrating Relationships - Import DDR XML File - Select File



(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



Select one or 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

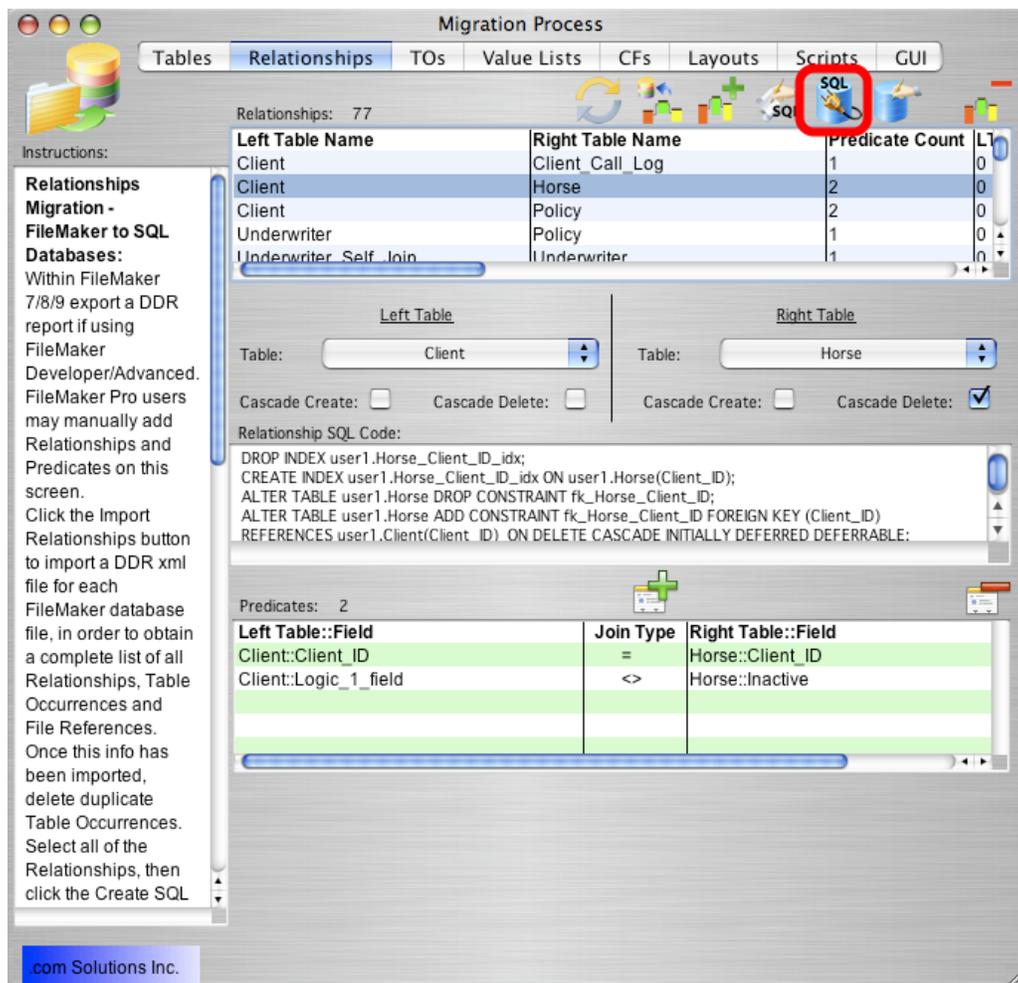
displayed within the Relationship SQL Code field. Otherwise, the generated SQL code will be displayed in the Relationship SQL Code field.

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



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

Note: 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.

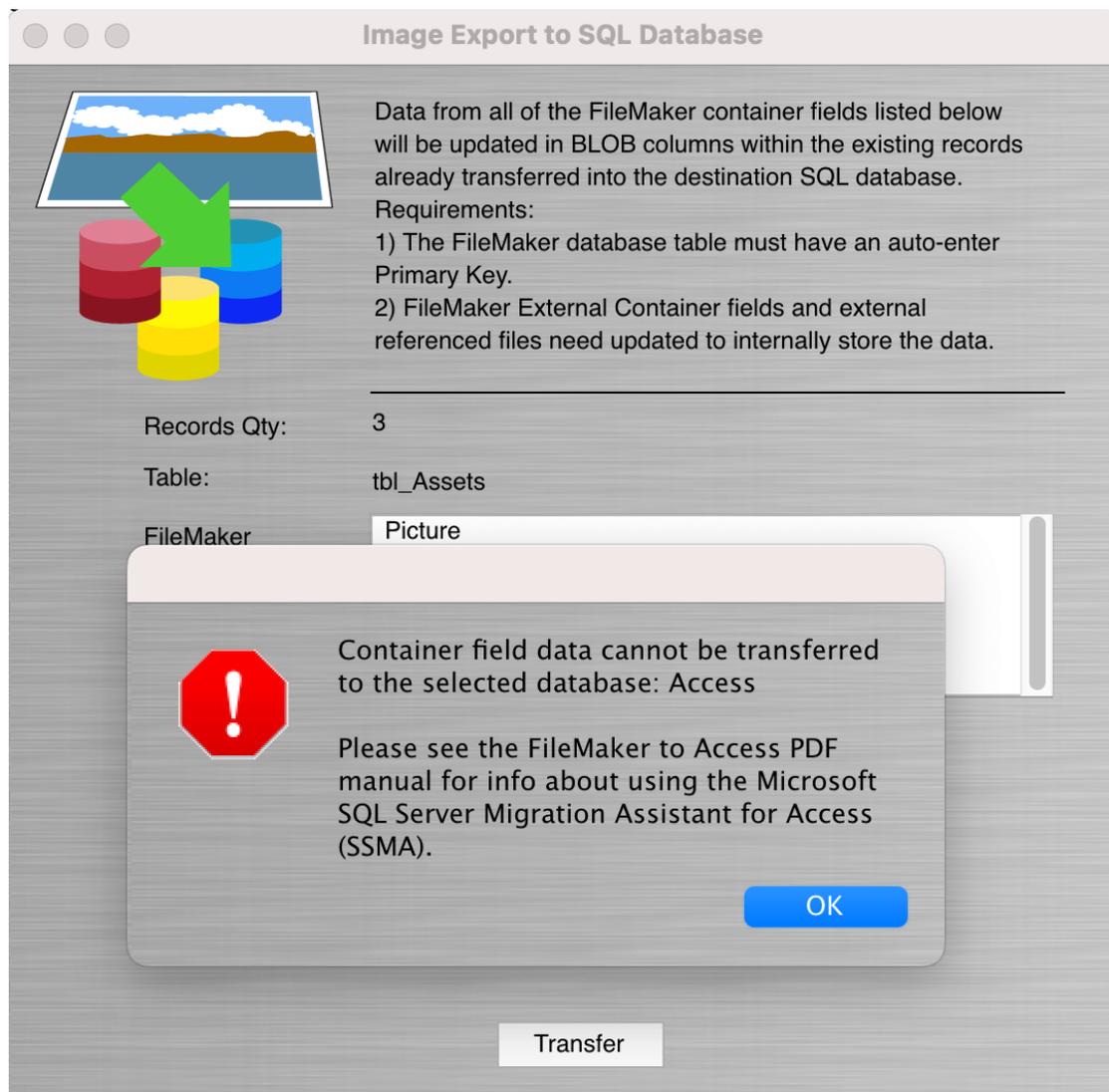
Image Export

Image Transfer 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



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

Download

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)

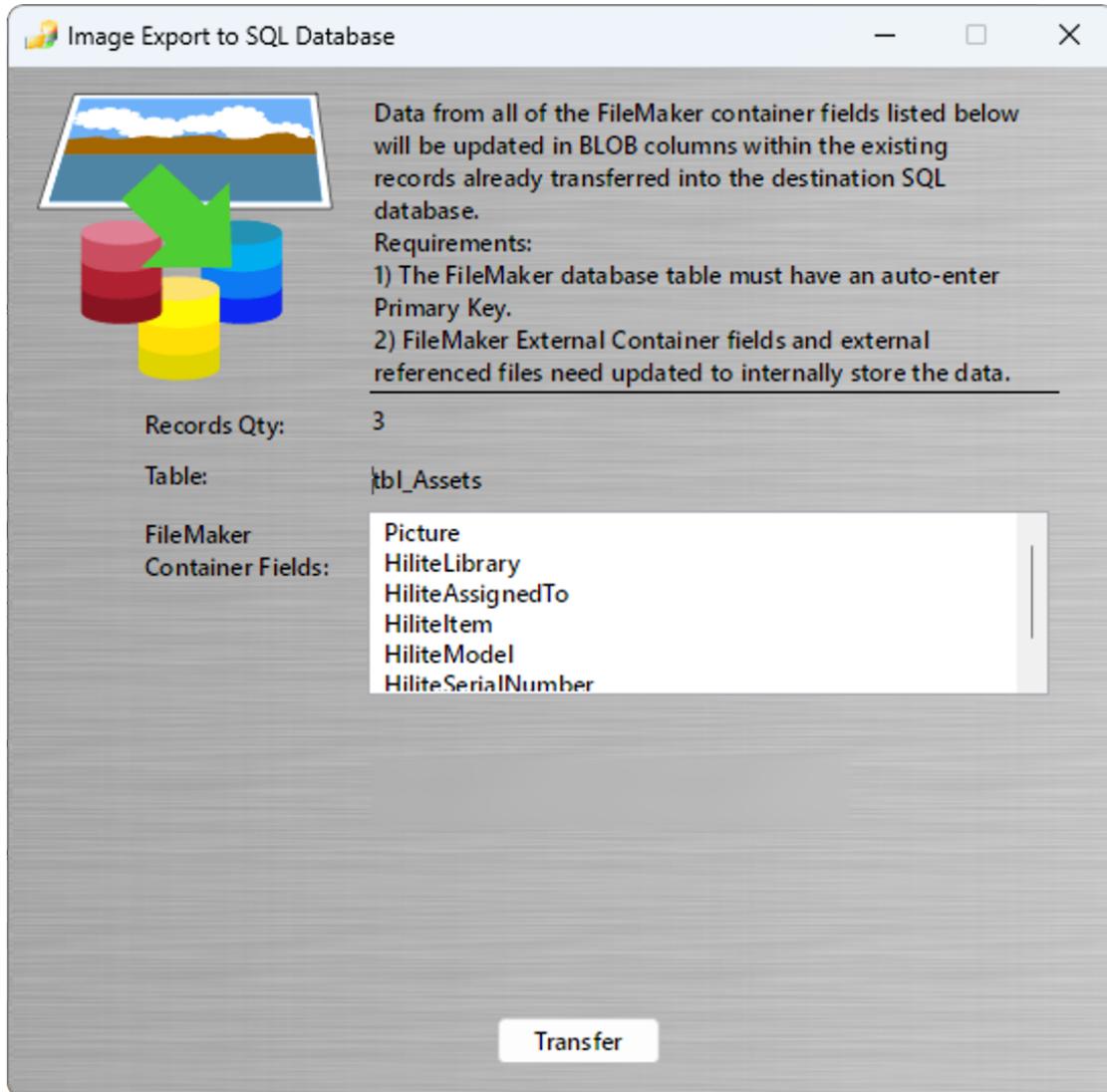
10/10/2019 • 2 minutes to read •       +4

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-acces>

Transfer Images into SQL Server with FmPro Migrator



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>

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:



0

```
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 TablImmaginiMenu (name, imagine, estensione, status, DescrizImma, datainser)
values ('ImageName.bmp', (SELECT * FROM OPENROWSET (BULK N'd:\Images\LogoName.bmp ',
go
```

```
INSERT INTO TablImmaginiMenu (name, imagine, estensione, status, DescrizImma, datainser)
values ('ImageName.bmp', (SELECT * FROM OPENROWSET (BULK
N'd:\Images\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, "TablImmaginiMenu ")
```

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

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

and this is the function

```
Public Function ExtractImmaginiForms(InFKID 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(InFKID 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

The screenshot shows the 'Migration Process' window with the 'Layouts' tab selected. The interface includes a navigation bar with tabs for Tables, Relationships, TOs, Value Lists, CFs, Layouts, Scripts, and GUI. Below the navigation bar, there are buttons for 'Add Layout', 'Batch Import', and 'XLS'. The main area displays a list of layouts with columns for Name, Record Count, and other details. A table below shows portal details for Portal1. A thumbnail image of the layout is also visible.

Layout Name	Record Count
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	267,120	0	0	

Portal Name	Fill Method	Field Titles
Portal1	On Record Advance	

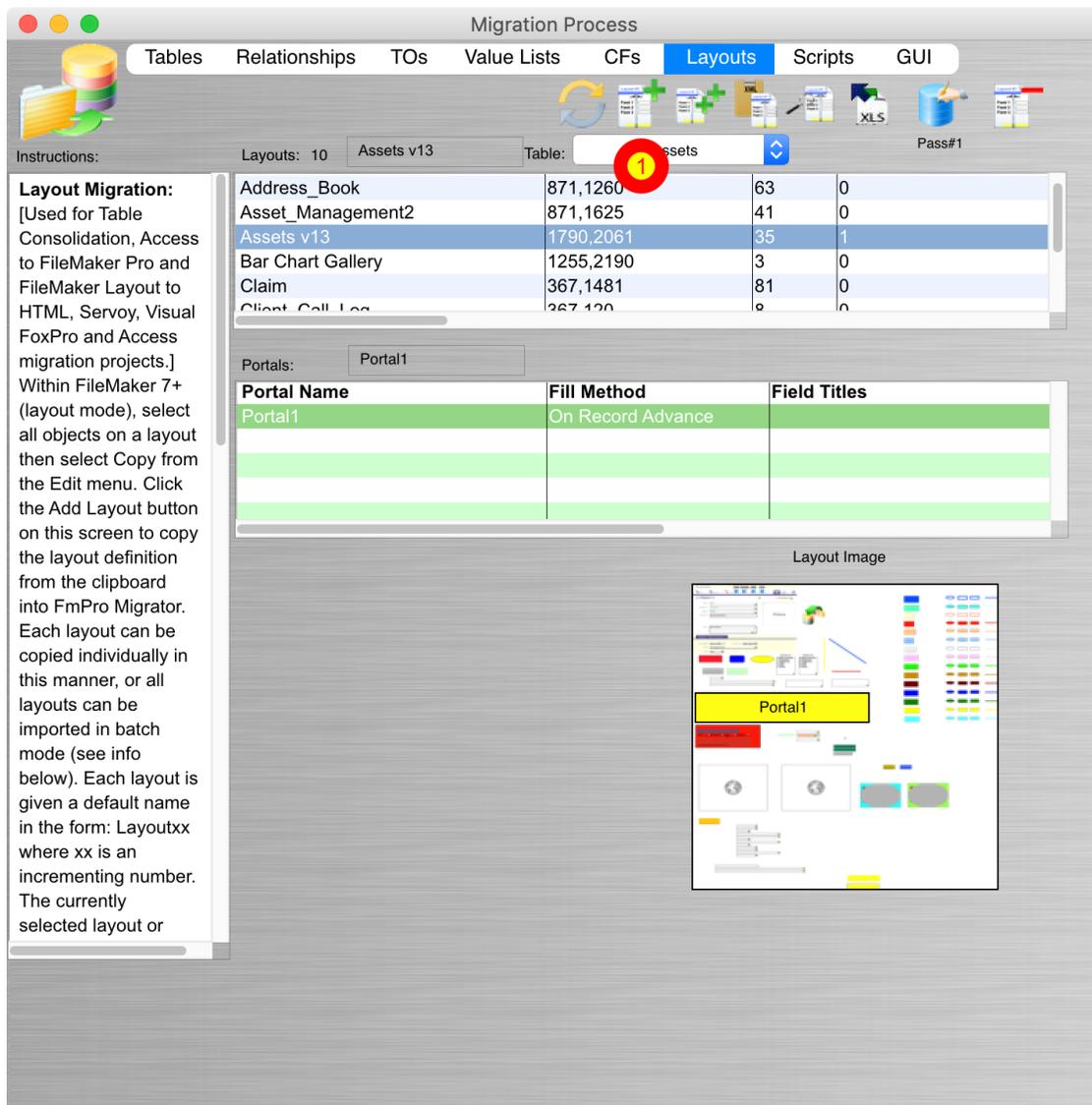
Layout Image

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



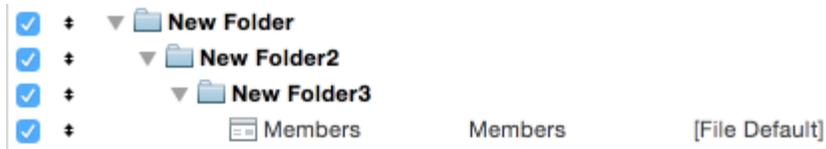
For small databases having under 20 layouts, it is easy to import layouts one at a time. To copy a layout via the clipboard:

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.

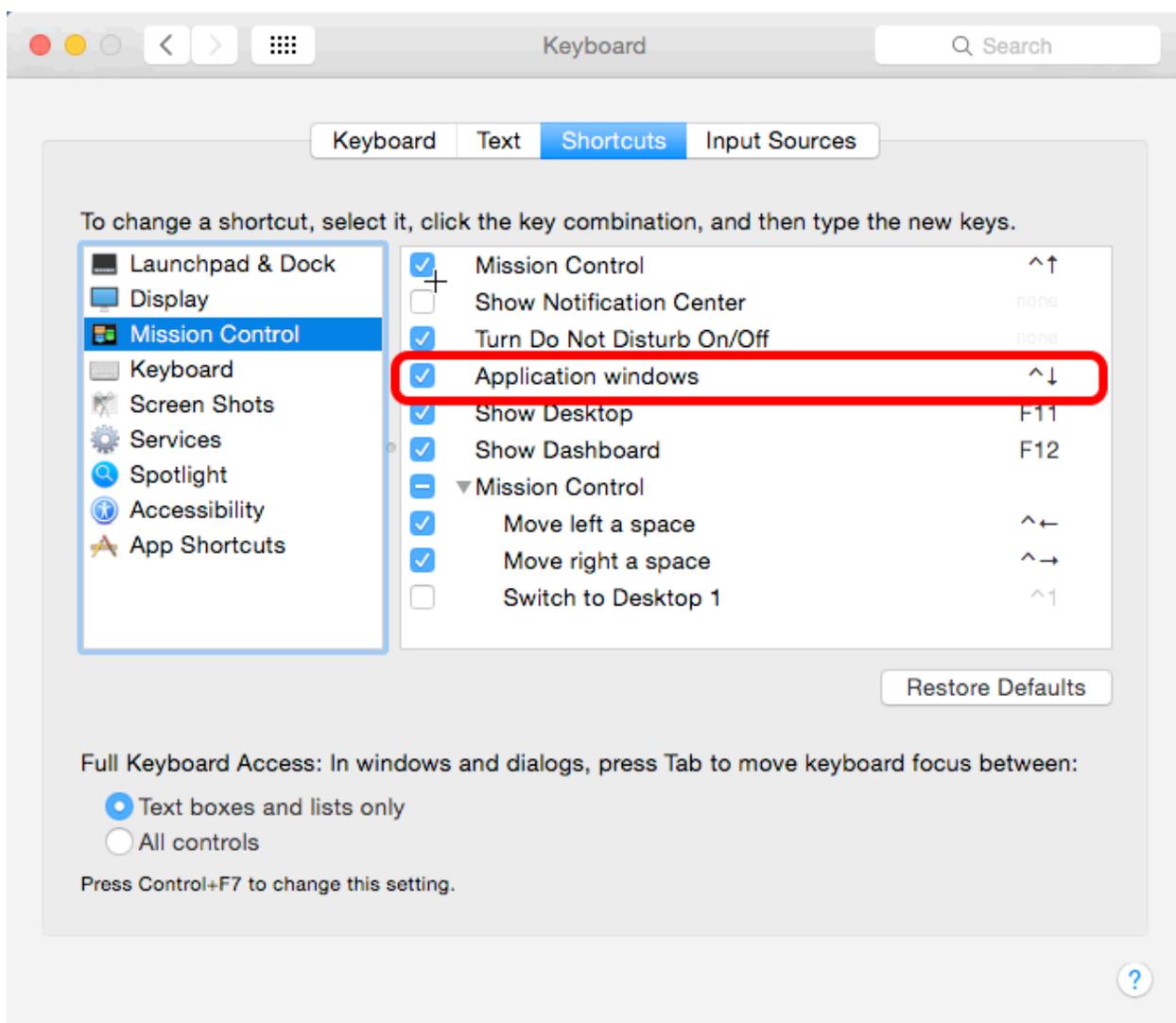
Then select the name of the Table associated with the layout and enter a name for the layout.

Layouts Within Nested Folders



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)



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 exists. 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

The screenshot shows the 'Migration Process' application window. The 'Layouts' tab is selected, and the 'Import All Layouts' button is circled in red. The interface includes a table of layouts, a portals table, and a layout image preview.

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

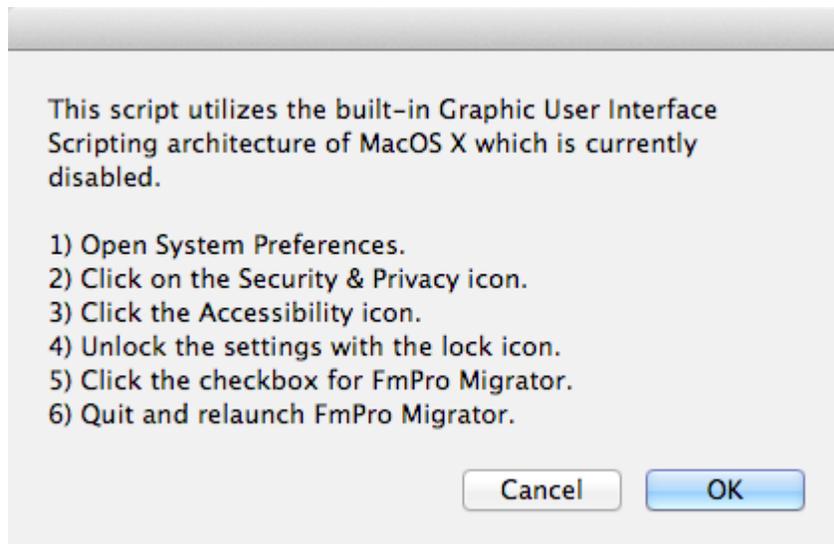
Layouts:	Assets v13	Table:	tbl_Assets	Pass#1
Address_Book	871,1260	63	0	
Asset_Management2	871,1625	41	0	
Assets v13	1790,2061	35	1	
Bar Chart Gallery	1255,2190	3	0	
Claim	367,1481	81	0	
Client_Call_Log	267,120	8	0	

Portals: Portal1

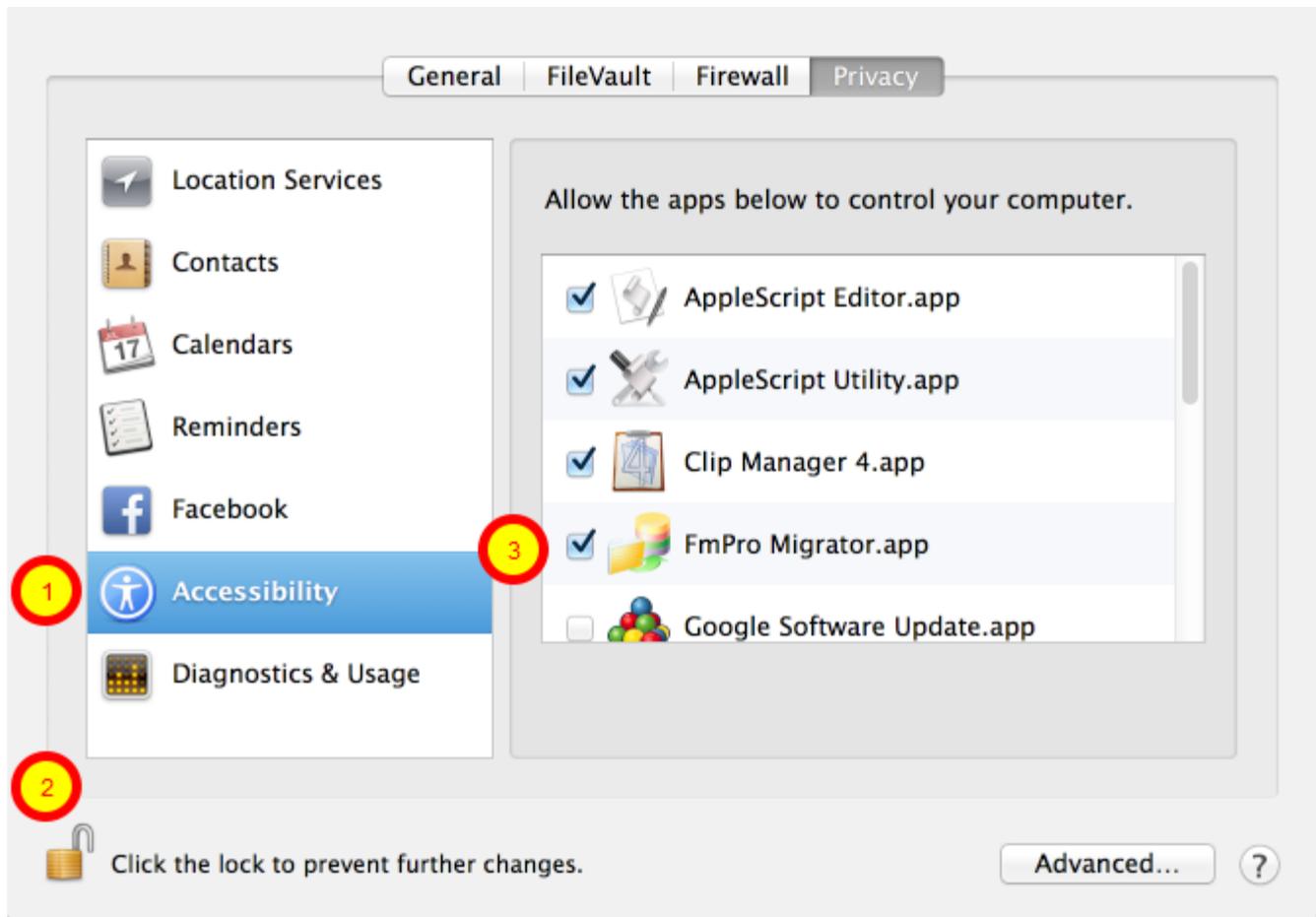
Portal Name	Fill Method	Field Titles
Portal1	On Record Advance	

Layout Image

The ImportAll Layouts button starts the process of importing all of the layouts.



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 AccessibilityAPI 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.

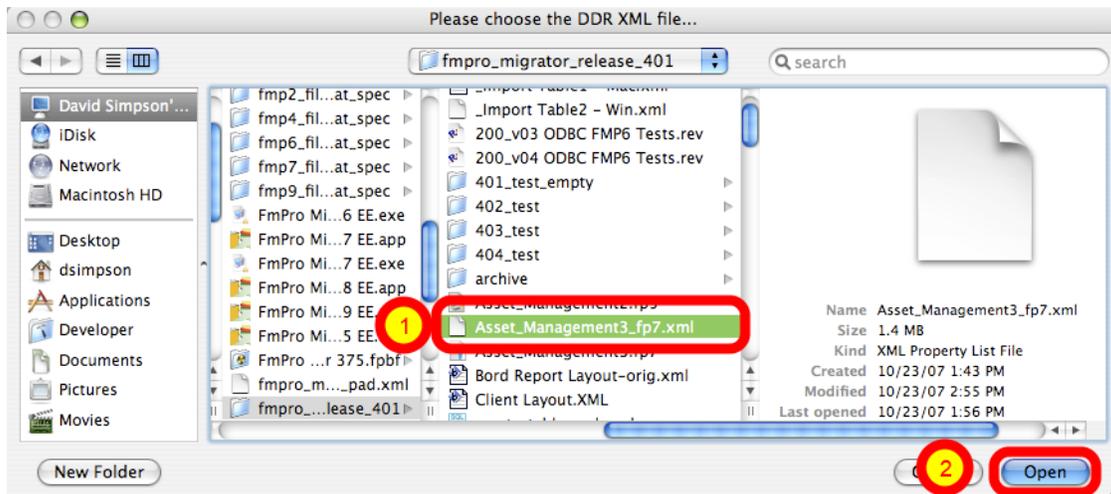


In the macOS System Preferences, click the Security & Privacy icon, (1) click the Accessibility item in the list, (2) click the lock button to allow changes, (3) click the checkmark next to the FmPro Migrator application.

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



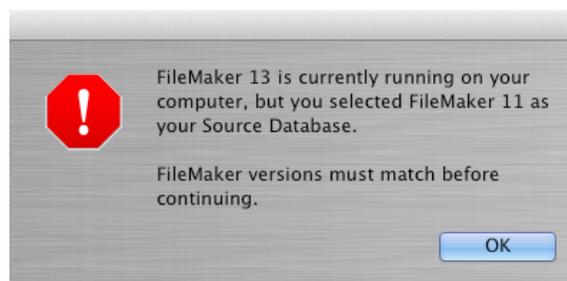
(1) Select a DDR XML file, (2) click the Open button.

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]

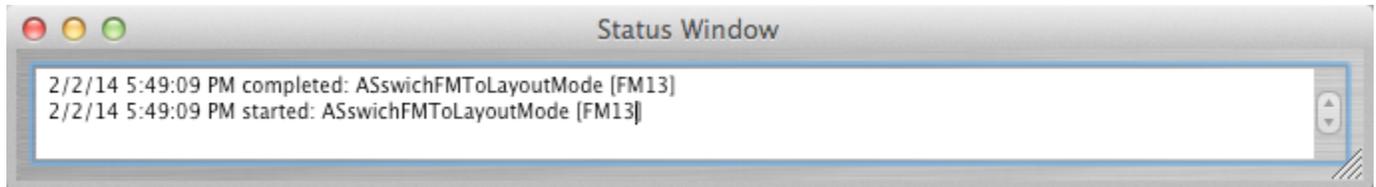


FmPro Migrator needs to have an accurate idea of which version of FileMaker Pro is currently running on your computer in order to use the correct AppleScript code to perform each task. This means that it is important to select the correct Source Database and Destination Database on the

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]



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 ASswitchFMTToLayoutMode 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)

The screenshot shows the Migration Process software interface. The 'Layouts' tab is selected, and the 'Assets v13' table is chosen. A dialog box titled 'Get FileMaker Window Name.' is open, providing instructions for batch importing layouts. The dialog box contains the following text:

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.

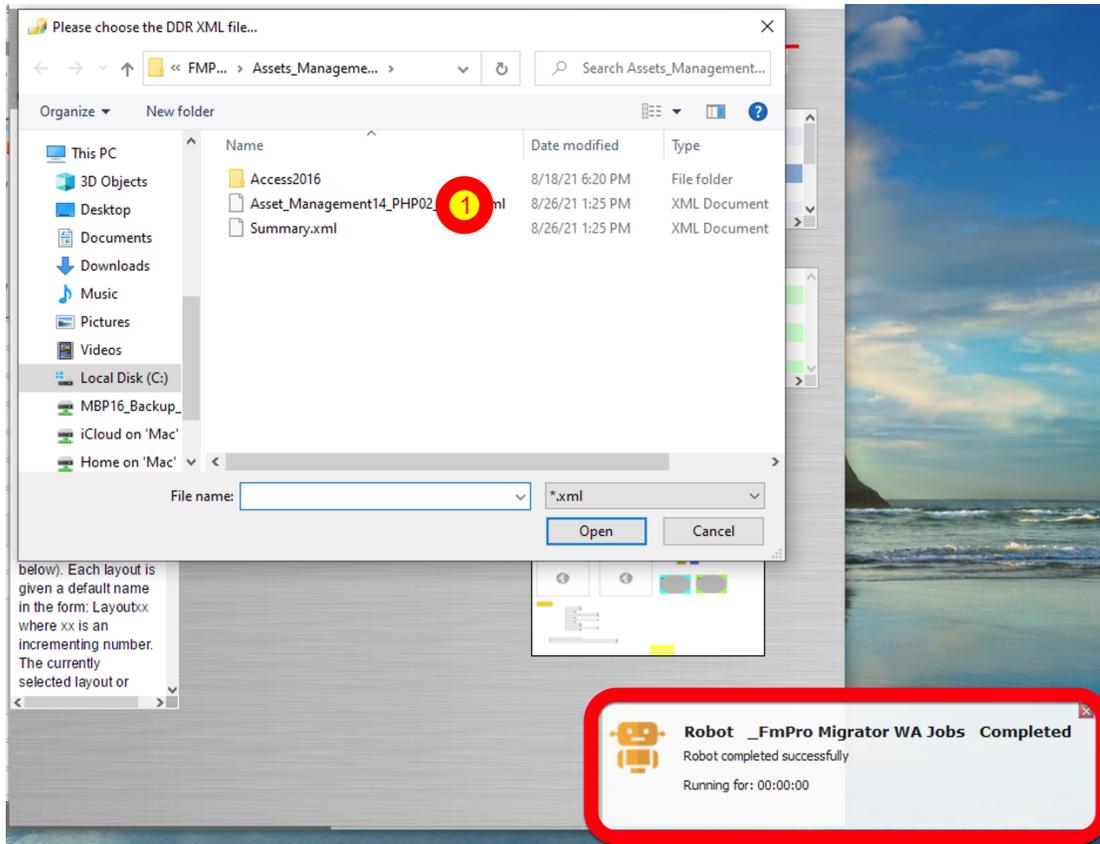
The dialog box has 'Cancel' and 'Ok' buttons. A red circle with the number '2' is placed over the 'Ok' button. In the background, the 'Layouts' table is visible with the following data:

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	10

The 'Portals' table is also visible with the following data:

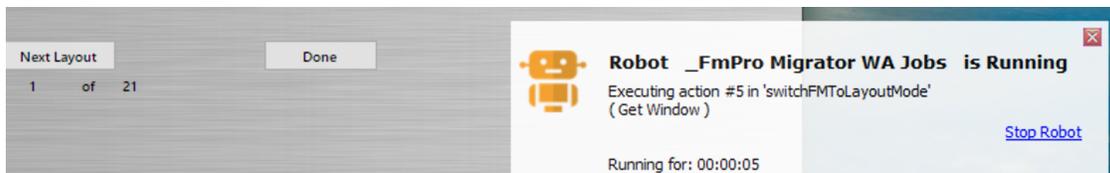
Portal Name	Fill Method	Field Titles
Portal1	On Record Advance	

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



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

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

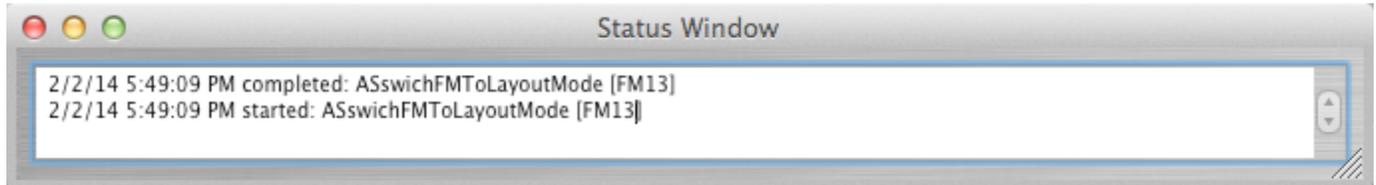


FmPro Migrator will work thru the list of layouts which were read from the DDR XML file and will show the progress at the bottom of the Layouts tab. During processing, multiple WinAutomation job actions will be displayed during the processing.

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]



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 ASswichFMToLayoutMode 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.

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

ASopenFMEditRelationshipWindow - 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.

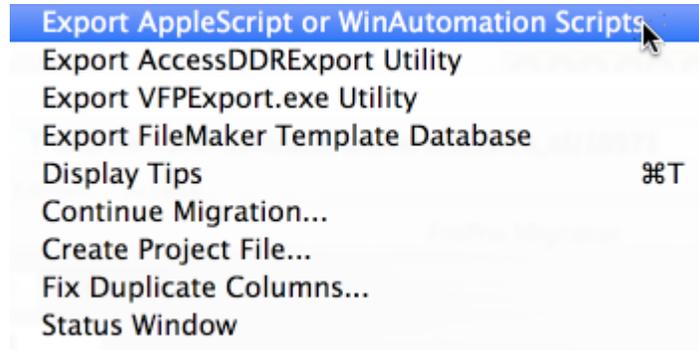
ASsaveFMRelationship - Sends keystroke return key to click the Ok button of the Edit Relationship window to save the current relationship.

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



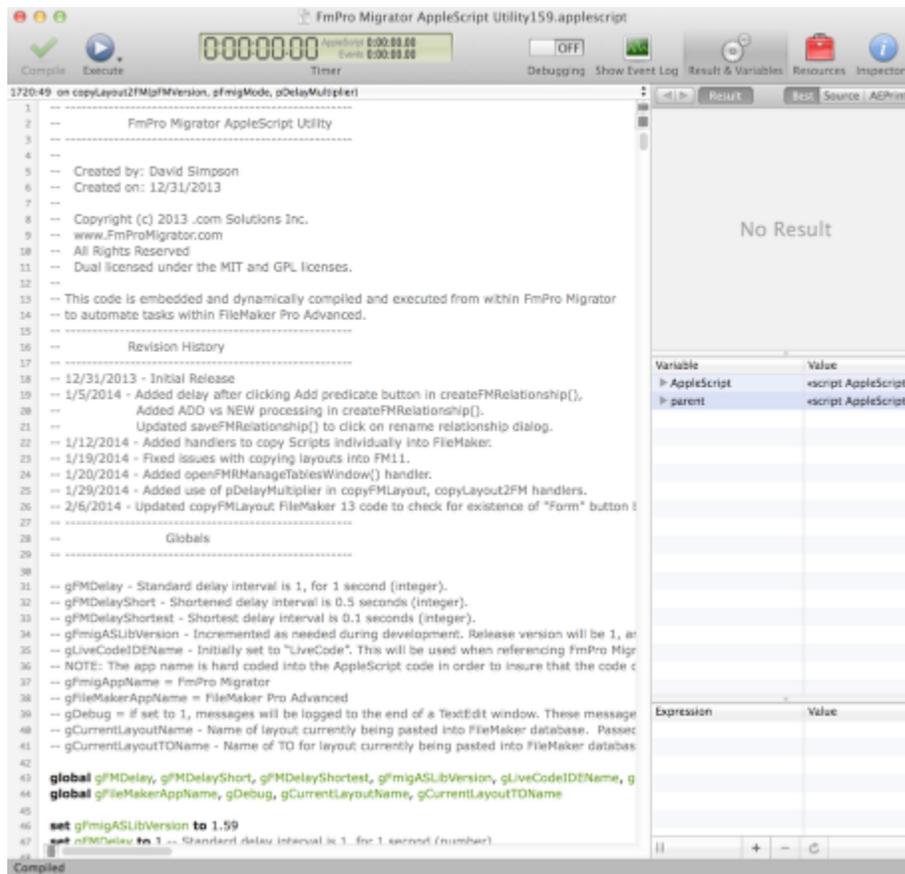
On macOS, the AppleScript code is embedded within FmPro Migrator therefore no installation is required.

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.

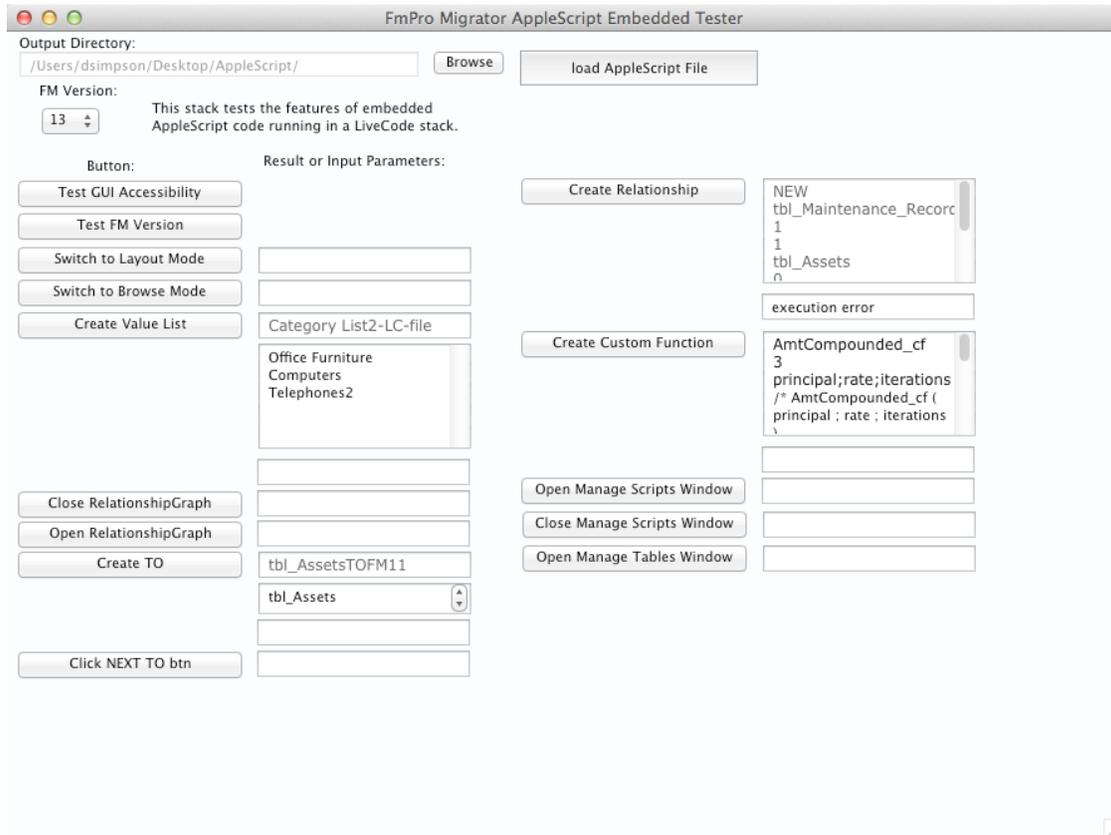
These files have been released as open source for use by the FileMaker community.

FmPro Migrator AppleScript Utility???.applescript File



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



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

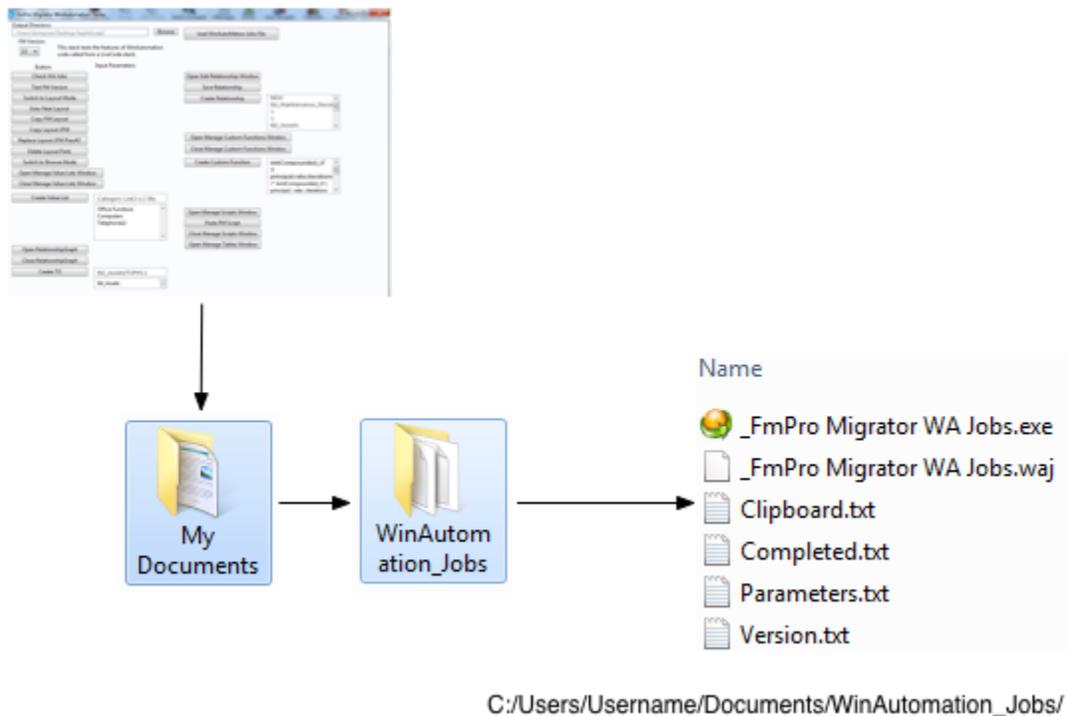
10/13/2020

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



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 WAJobs.exe), and then enters a loop, waiting for completion of

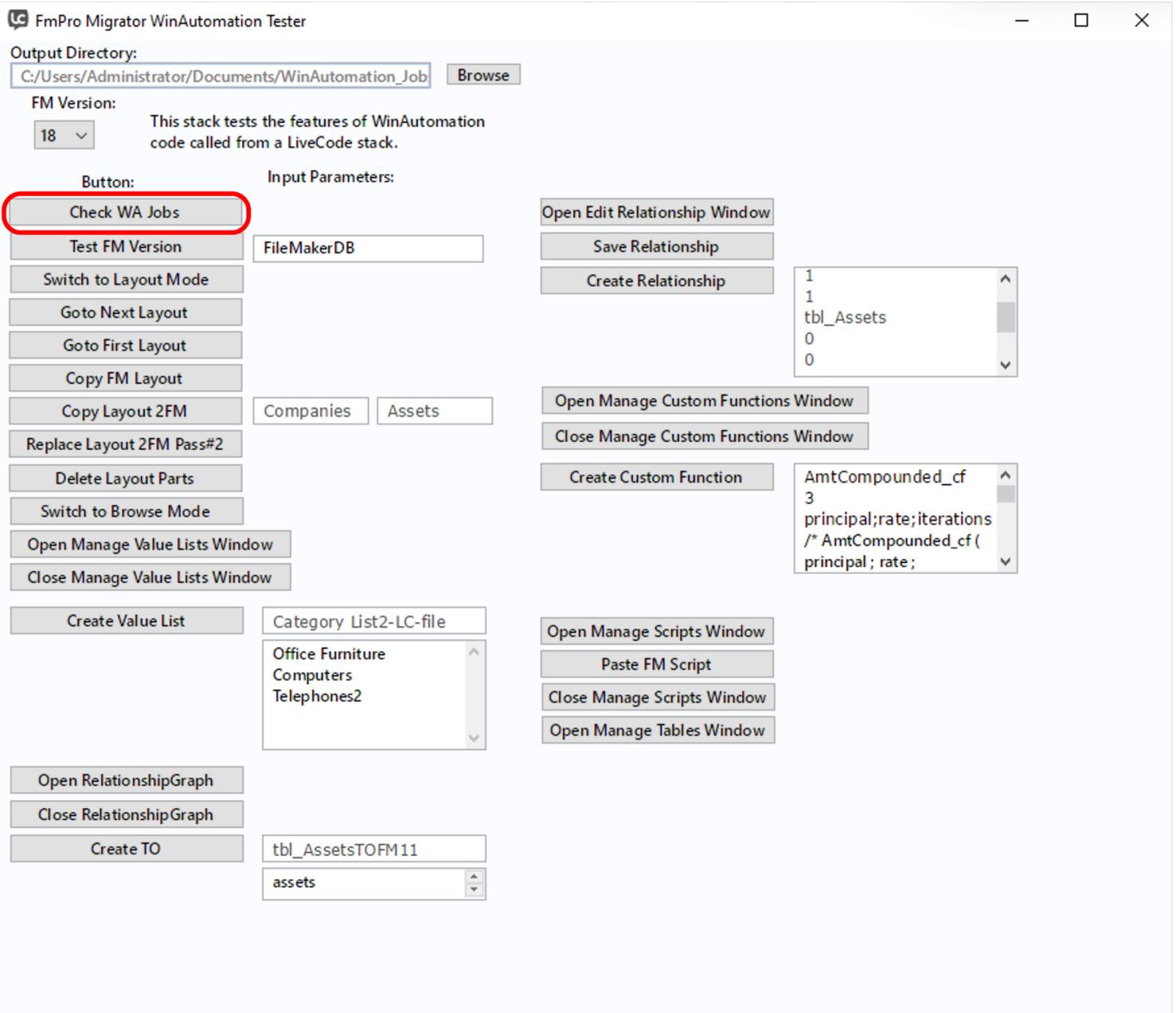
the job.

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.

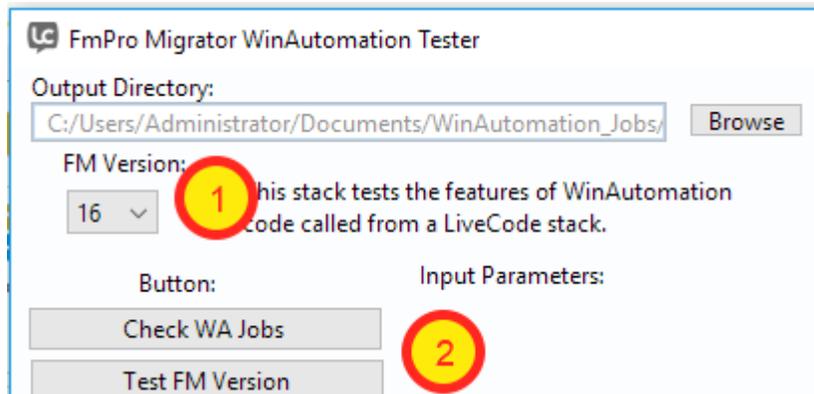
Setup



Clicking the Check WAJobs button creates the WinAutomation_Jobs folder within the Documents directory. The _FmPro Migrator WAJobs.exe and _FmPro Migrator WAJobs.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

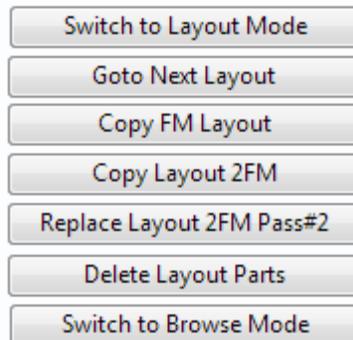


Launch the FileMaker Pro Advanced application. The FileMaker Pro Advanced application needs to already be running in order for any of these tasks to complete.

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



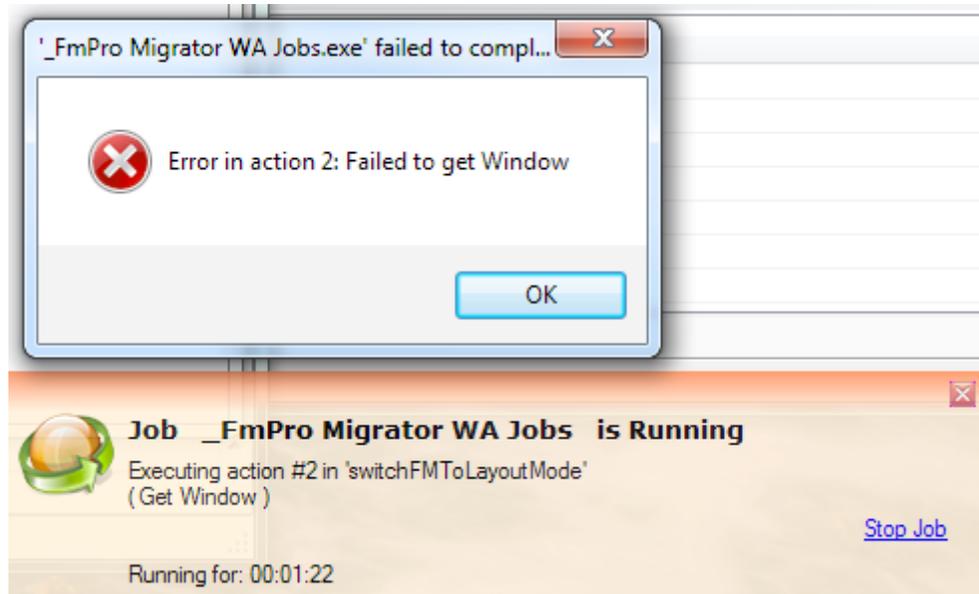
One of the simplest tasks to test is switching between Browse and Layout modes. These tasks require no setup prior to running.

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



The compiled WinAutomation Jobs file is compiled with the option to display Notifications and Error Message Dialogs, for troubleshooting purposes.

This error in the switchFMTtoLayoutMode 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



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

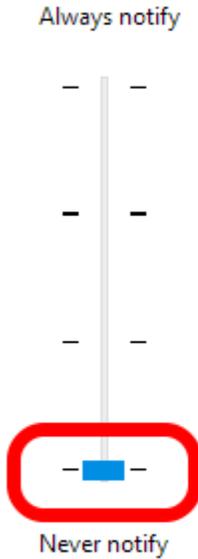


Choose when to be notified about changes to your computer

User Account Control helps prevent potentially harmful programs from making changes to your computer.

[Tell me more about User Account Control settings](#)

Always notify



Never notify

Never notify me when:

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

i Not recommended.

 OK

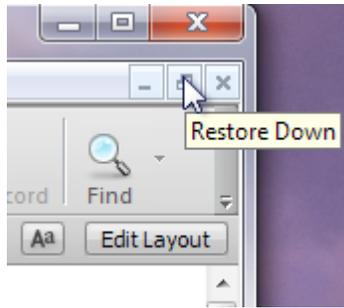
Cancel

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.

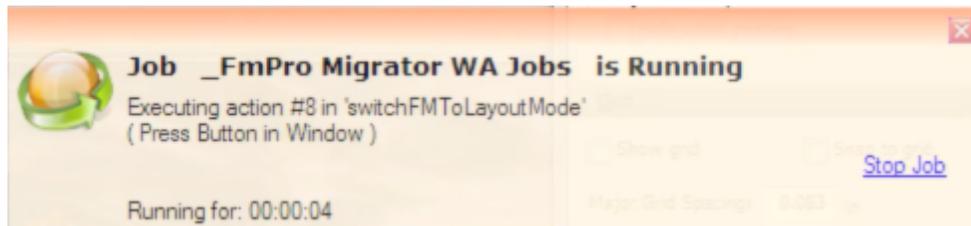
Maximized Window Problem



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



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

In this screenshot:

`_FmPro Migrator WAJobs` - 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.

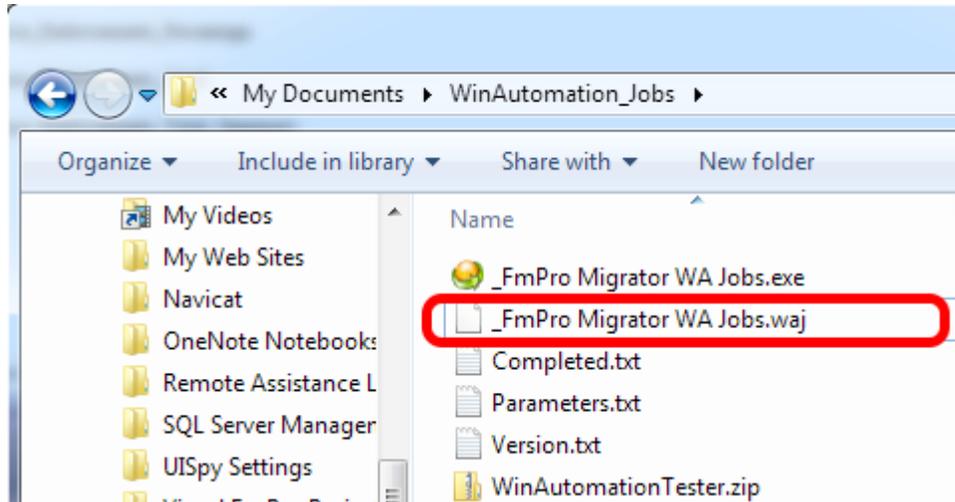
WinAutomation - Wrong Job Executing

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 WAJobs.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.

openFMEditRelationshipWindow - 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.

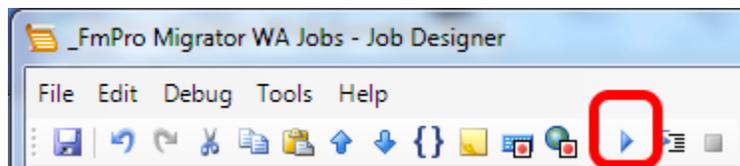
replaceLayout2FMPPass2 - Selects the contents of the current layout in Layout mode, deletes the contents, pastes the layout contents already on the clipboard. Used during Pass #2 processing during Table Consolidation projects.

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 WAJobs.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

Migration Process

Tables Relationships TOs Layouts **Scripts** GUI

Scripts: 4

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:

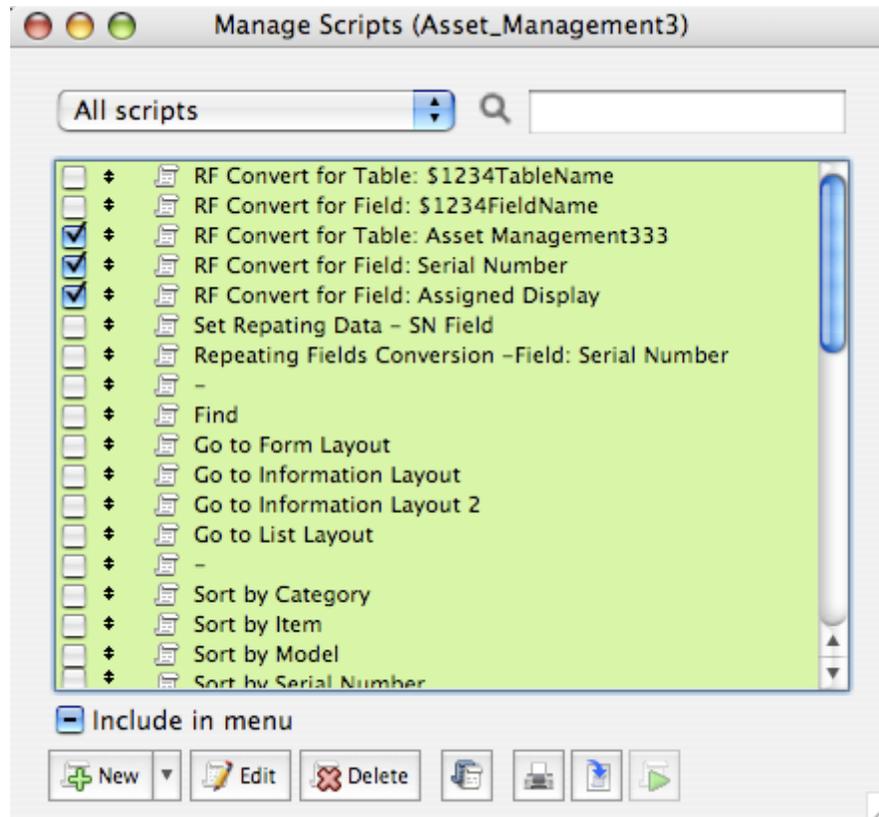
```
This script imports records from the AB2FM FileMaker file which was used to import re
requires: AB2FM.fp7 file in same directory as this INS database file, for importing purp
# Access - Admin only
# 11/4/2006 - David Simpson - .com Solutions Inc. - www.fmpromigrator.com - Initial Rele
#
#
Go to Layout [ "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
```

.com Solutions Inc.

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.

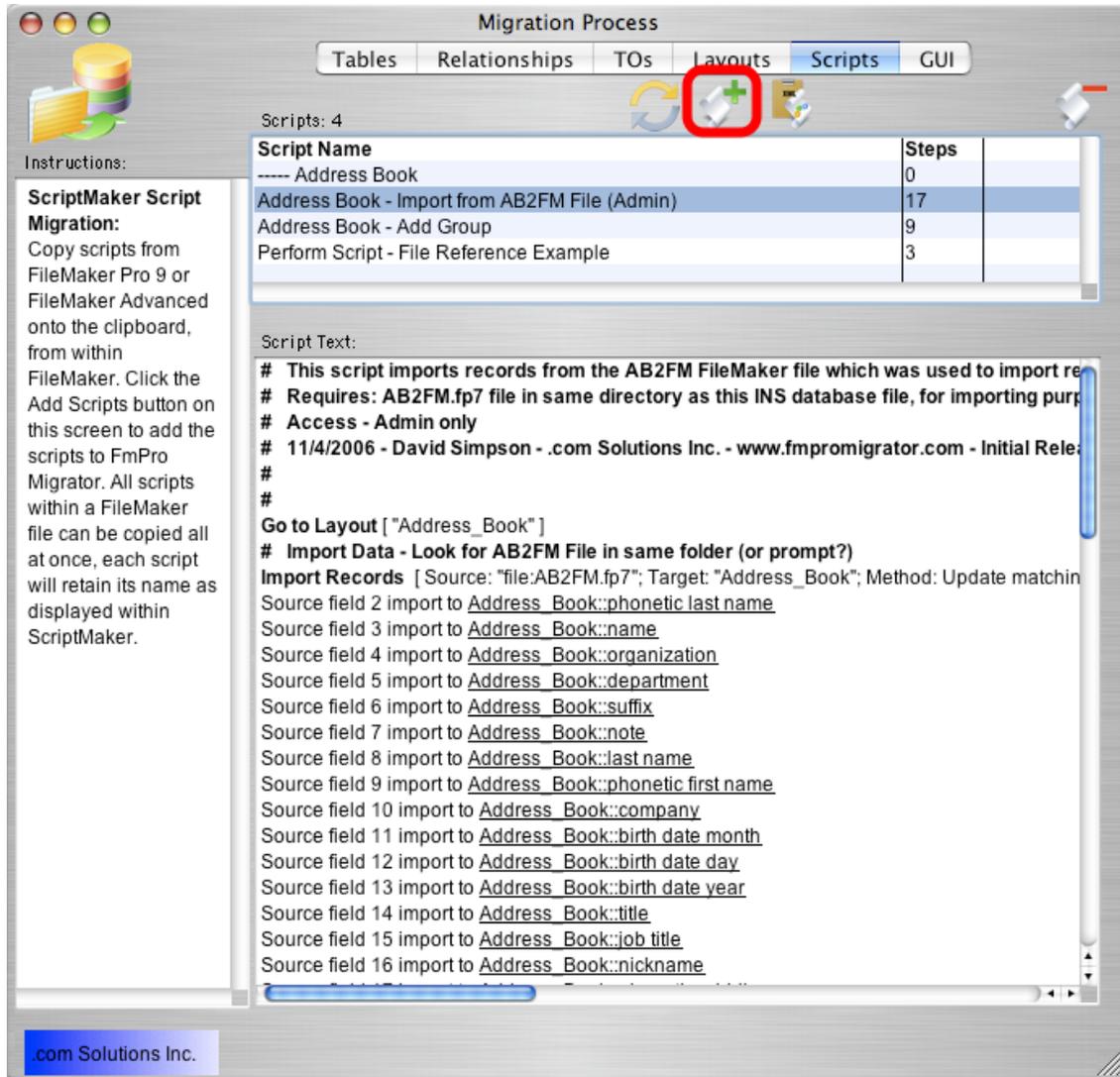
Scripts can also be viewed within FmPro Migrator by clicking each script and viewing the text version of the script within the Script Text field [2]. Duplicate or unneeded scripts can also be deleted by selecting the scripts and clicking the Delete button [3].

Copy Script Workspace Scripts From FileMaker to ClipBoard



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



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

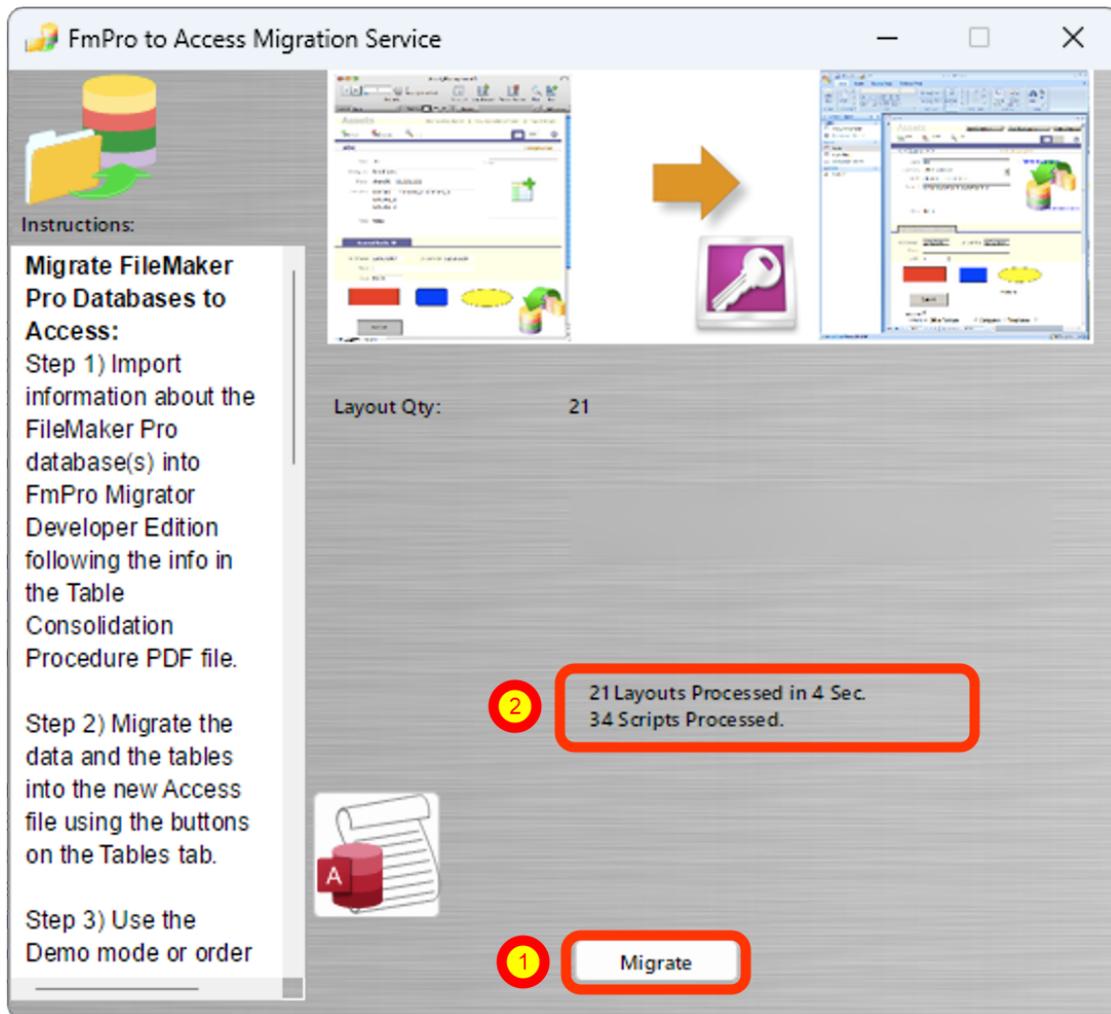
You will perform this task for every source database file which contains scripts to be migrated.

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.

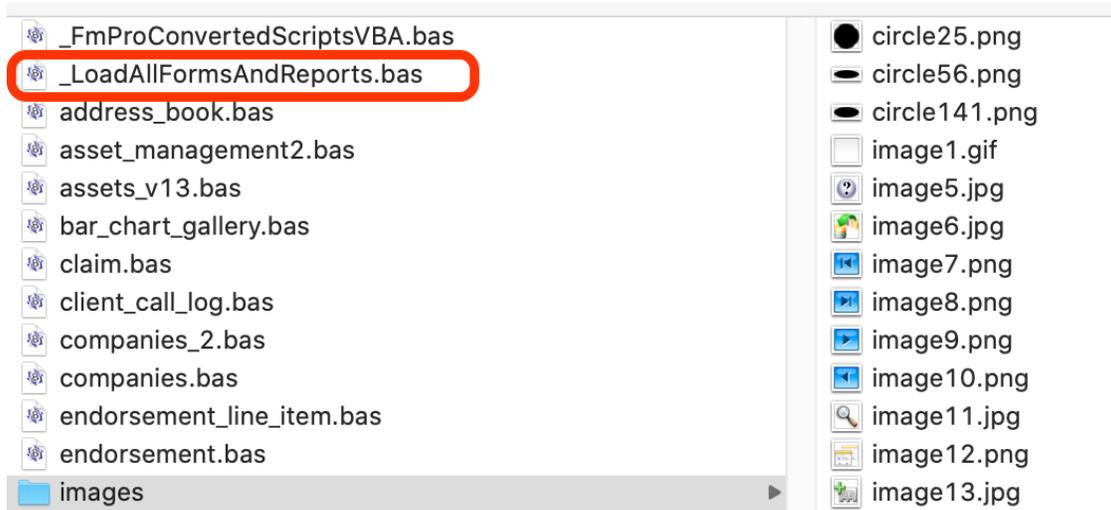
Migration Statistics - Licensed Mode



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

Building Forms & Reports in Access

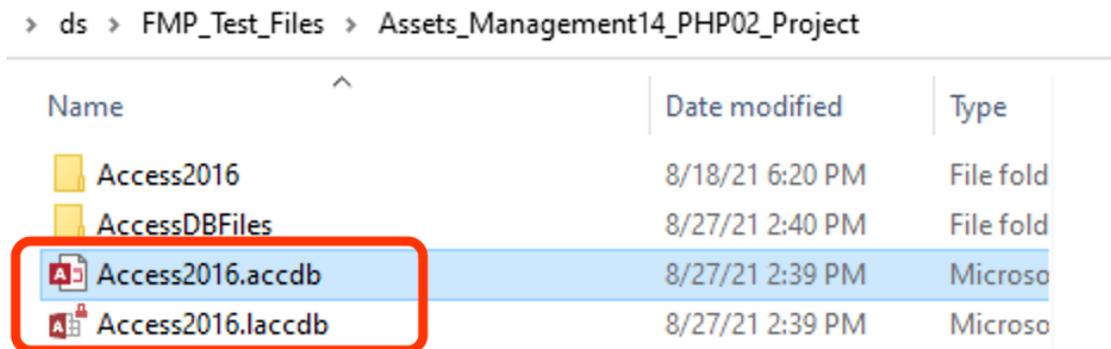
Generated Files



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

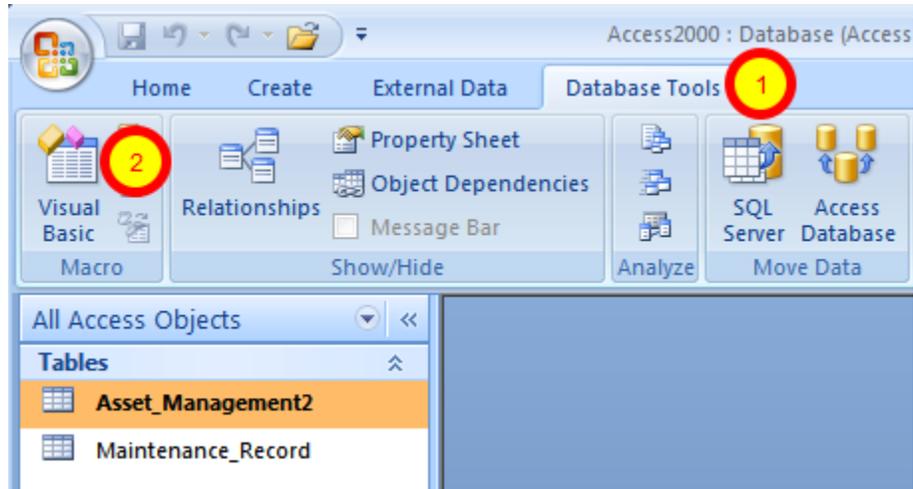
The `_LoadAllFormsAndReports.bas` file will load and run each of the 10 Visual Basic files in order to re-create each of the layouts as new forms, subforms and reports within the Access database.

Close FmPro Migrator



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.

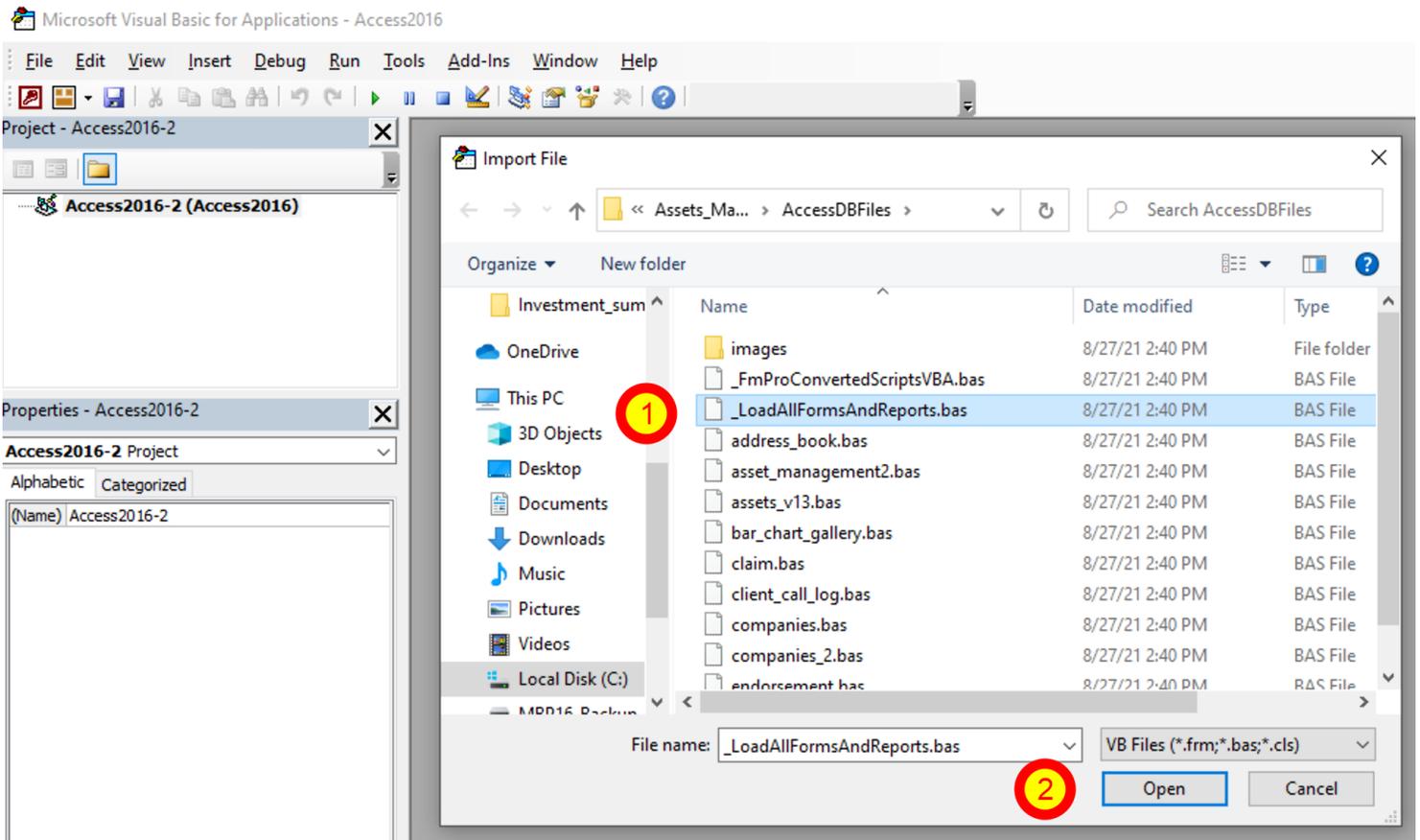
Open the Access2016.accdb file



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.

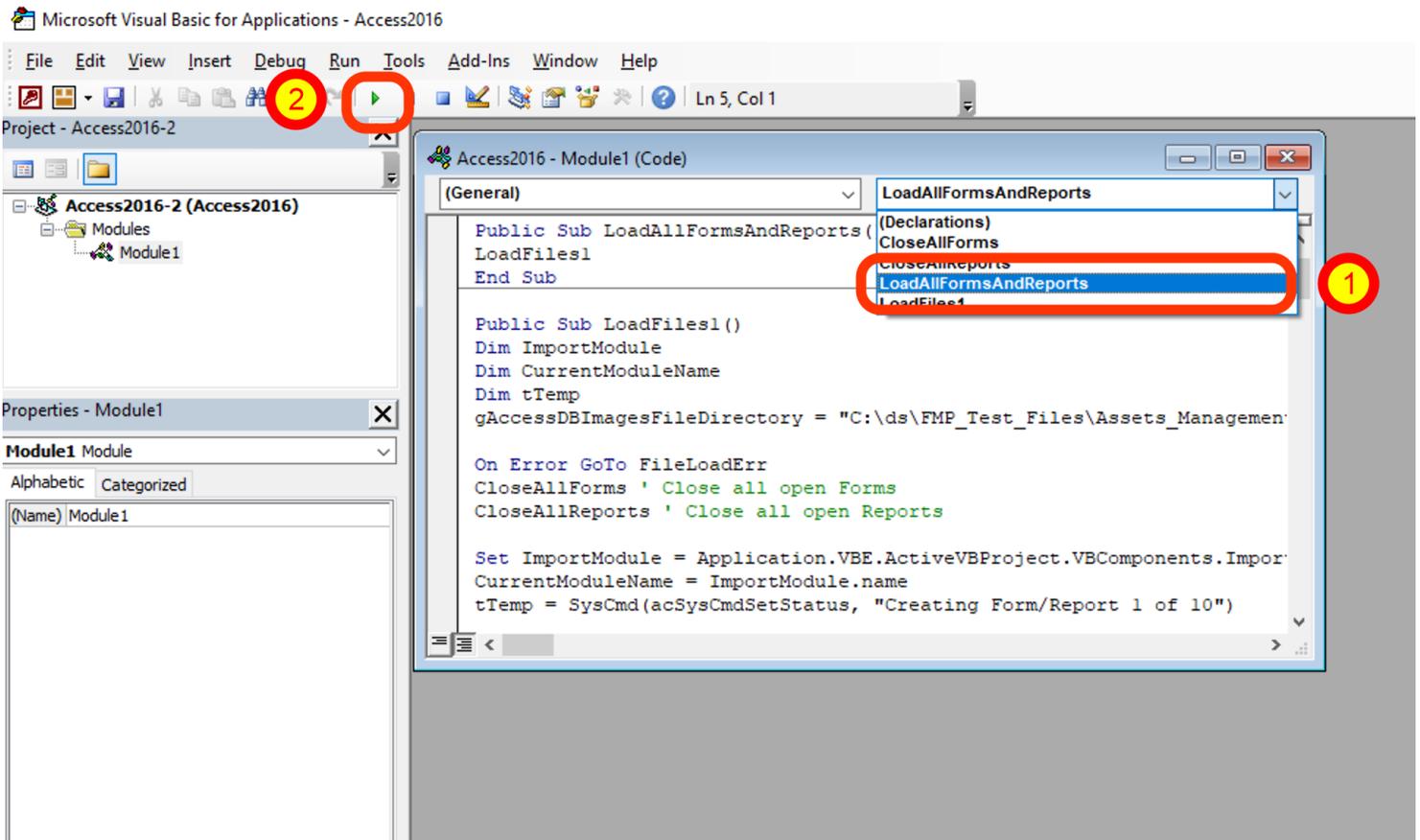
(1) Click the Database Tools tab, (2) click on the Visual Basic icon to open the Visual Basic editor.

Import File... _LoadAllFormsAndReports.bas



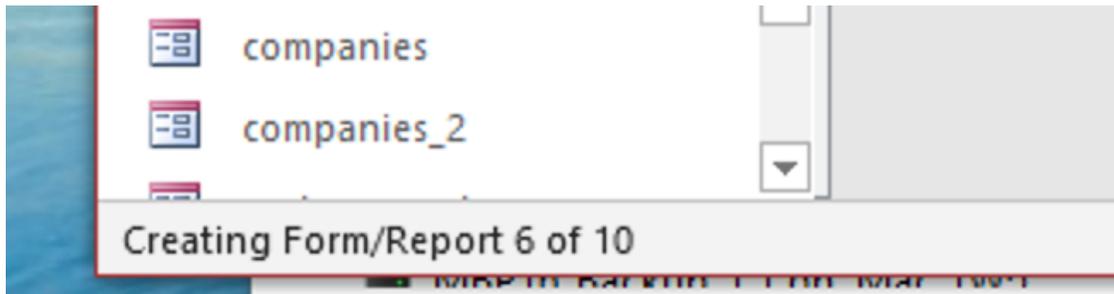
From within the Visual Basic editor window, select Import File..., and (1) select the _LoadAllFormsAndReports.bas file, (2) click the Open button.

Run LoadAllFormsAndReports Procedure



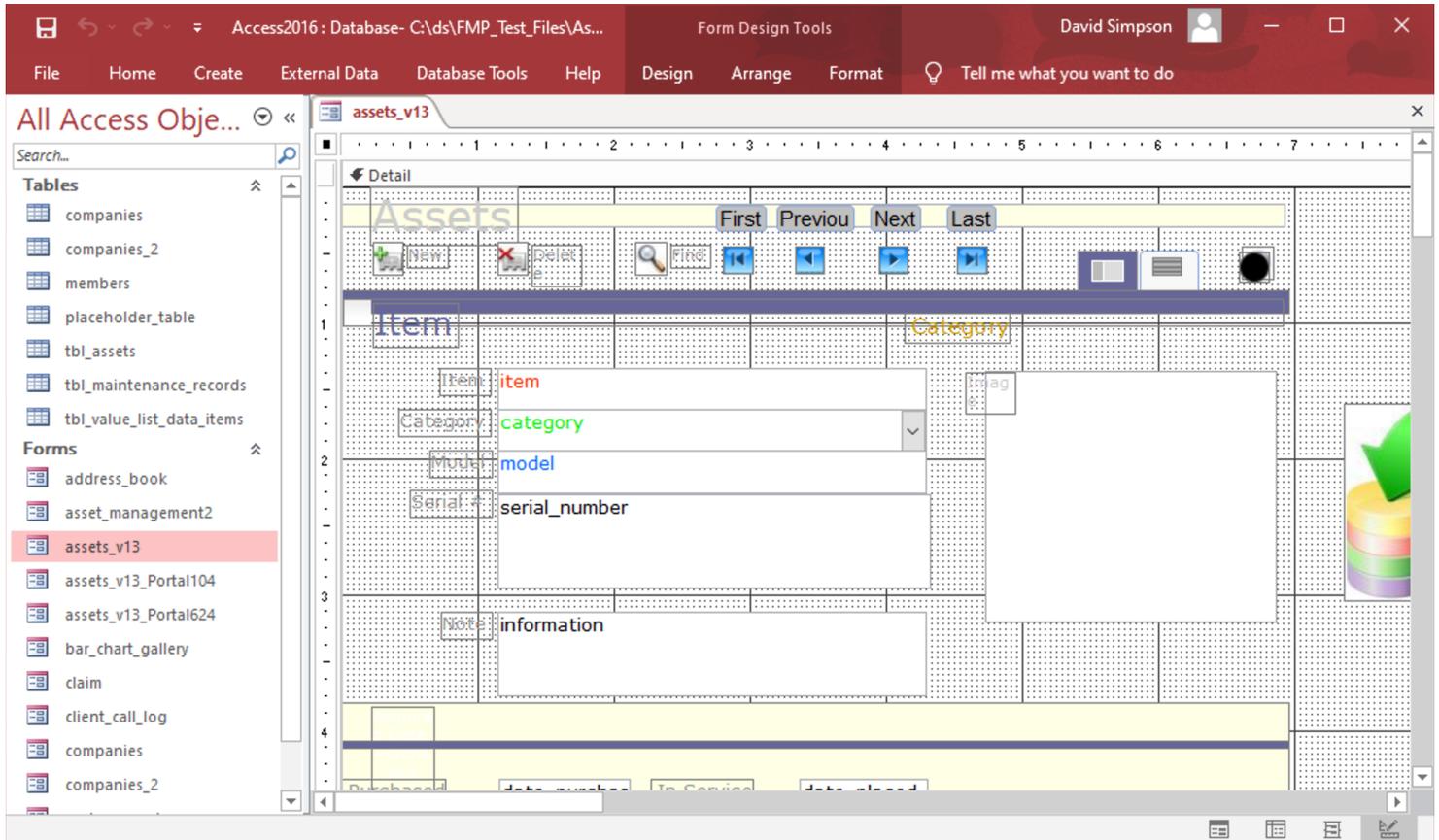
(1) Select the procedure named LoadAllFormsAndReports from the procedure menu, then click on the Run button.

Progress Display



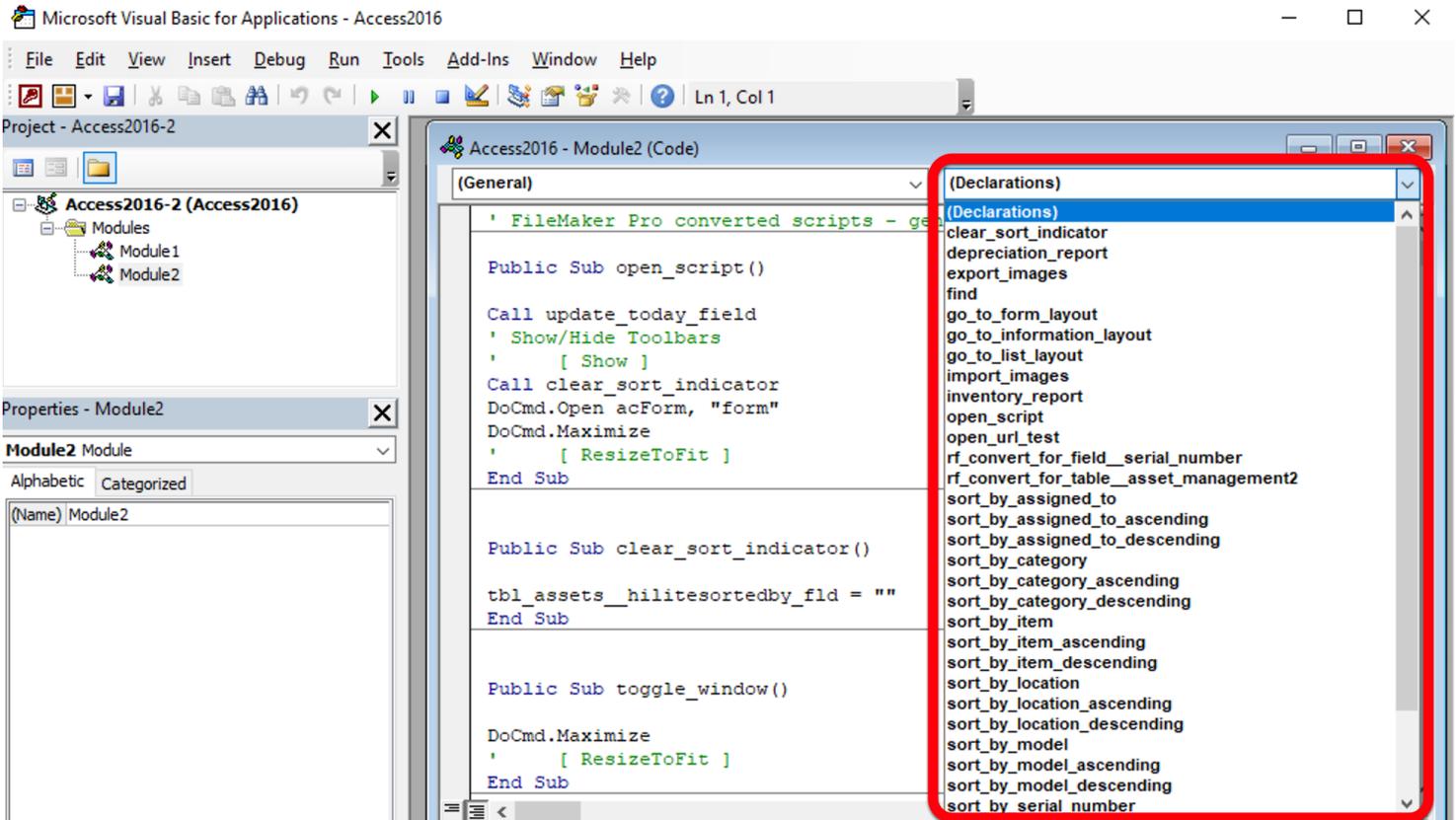
As the Forms, SubForms and Reports are being created, a progress message will be displayed in the lower left corner of the main Access database window.

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



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

Post-Migration Tasks - FmPro to Access

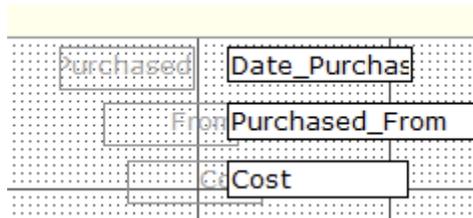
Re-size Text Labels



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



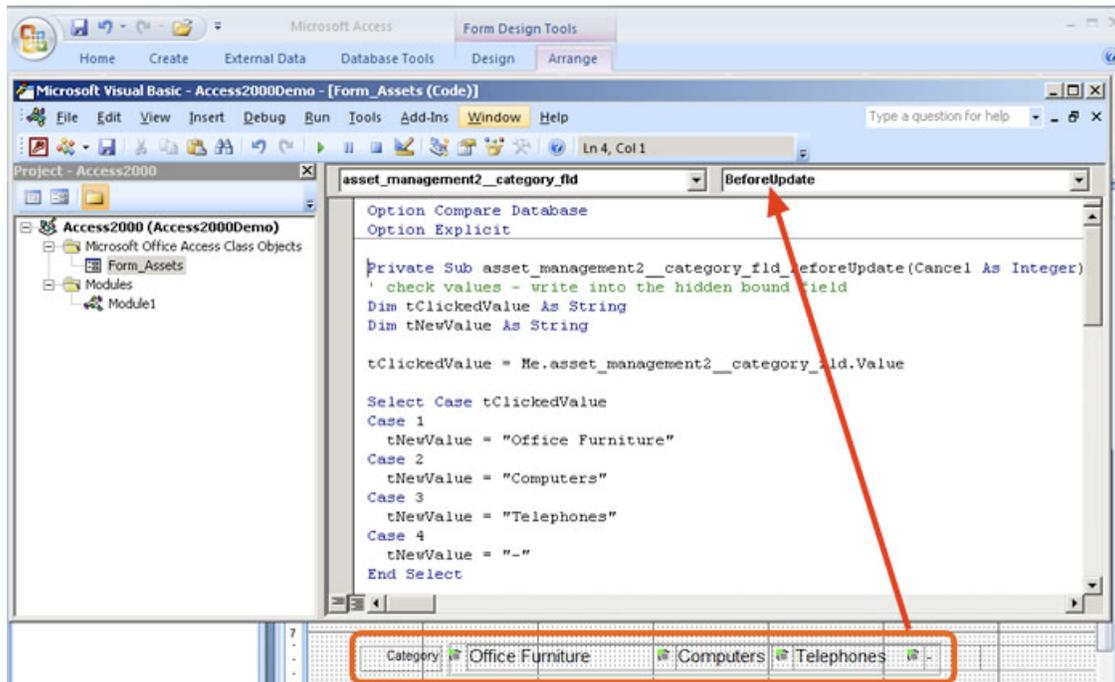
FileMaker Pro container fields are often used to store images for display on FileMaker Pro layouts. Container field image data is not directly converted into Access OLE Object fields for display within Access Forms/Reports. However you might have BLOB columns in SQL Server which are viewed via the Linked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



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.

Using the FmPro Code Conversion Workbench - FmPro to Access Conversions

This chapter shows how to use the FmPro Code Conversion Workbench. Before performing the code conversion with the FmPro Code Conversion Workbench, you should have already imported the FileMaker Pro database into FmPro Migrator, including tables, relationships, value lists, layouts and scripts.

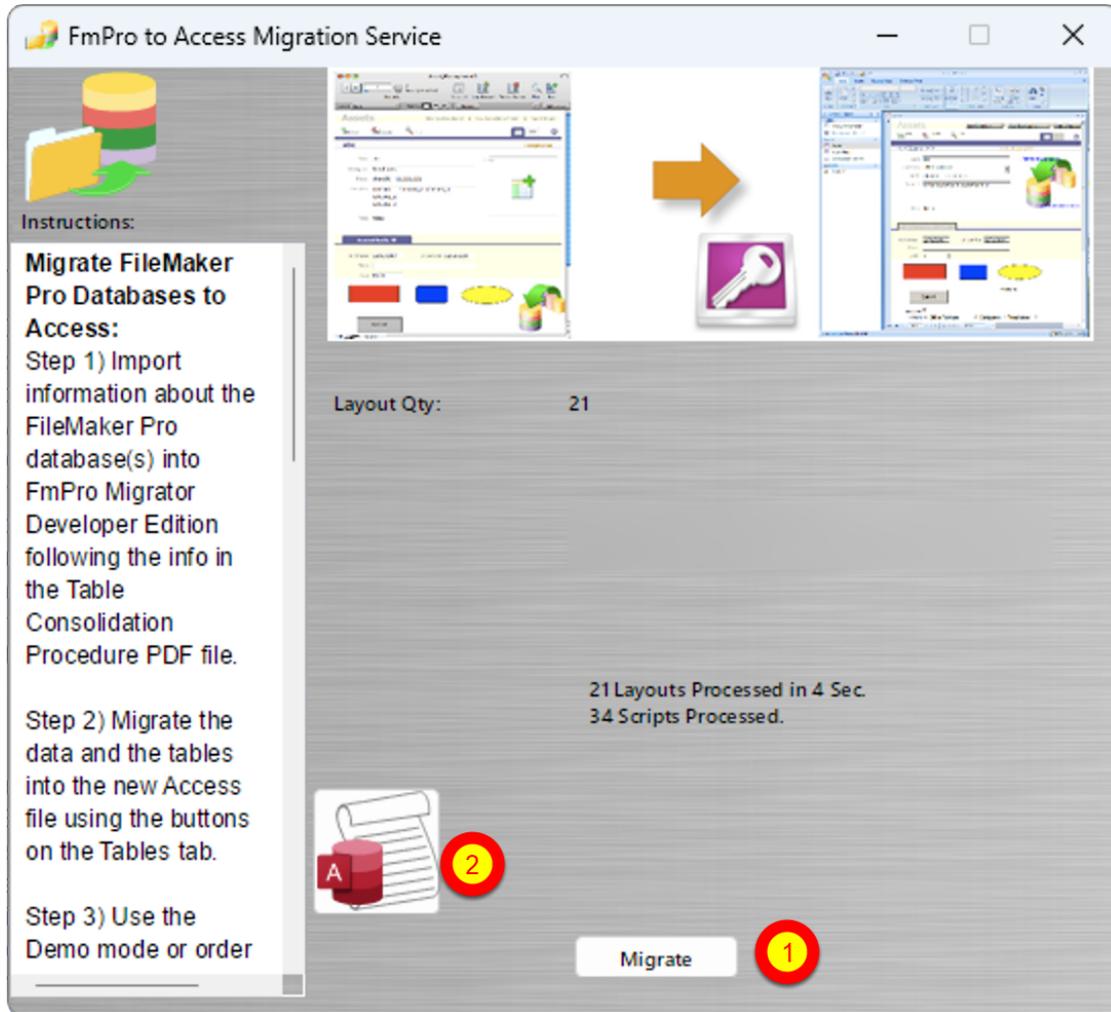
FmPro to Access Migration Button - GUI Tab



FmPro to Access Migration

Click the [FmPro to Access Migration](#) button on the GUI tab of the Migration Process window of FmPro Migrator.

FmPro to Access Migration Window



Prior to reaching this step, you should have already imported the FileMaker Pro database into FmPro Migrator, (1) clicked the Migrate button to create the new Access database and run the `_LoadAllFormsAndReports.bas` script within the `AccessDBFiles` folder. After you have built the new Access database with all of the new forms/reports converted from the FileMaker Pro database you are ready (2) to click the Code Conversion Workbench button.

Note: Using the FmPro Code Conversion Workbench replaces the need to run the existing `_FmProConvertedScriptsVBA.bas` VBA script.

FmPro Code Conversion Workbench Window

Completed: 2 of 6 Scripts Load Data

Status	ID	Size	Script Name
<input type="checkbox"/> Not Started	1	175	Open Script
<input type="checkbox"/> Not Started	2	46	Clear Sort Indicator
<input type="checkbox"/> Not Started	3	35	Toggle Window
<input checked="" type="checkbox"/> Completed	4	675	Find
<input type="checkbox"/> Not Started	5	59	Go to Form Layout
<input type="checkbox"/> Not Started	6	66	Go to Information Layout
<input type="checkbox"/> Not Started	7	66	Go to List Layout
<input type="checkbox"/> Not Started	8	354	Sort by Category
<input type="checkbox"/> Not Started	9	333	Sort by Item
<input type="checkbox"/> Not Started	10	339	Sort by Model
<input type="checkbox"/> Not Started	11	381	Sort by Serial Number
<input type="checkbox"/> Not Started	12	369	Sort by Assigned To
<input type="checkbox"/> Not Started	13	355	Sort by Location
<input type="checkbox"/> Not Started	14	228	Sort by Category Ascending
<input type="checkbox"/> Not Started	15	229	Sort by Category Descending
<input type="checkbox"/> Not Started	16	188	Sort by Item Ascending
<input type="checkbox"/> Not Started	17	189	Sort by Item Descending
<input type="checkbox"/> Not Started	18	190	Sort by Model Ascending
<input type="checkbox"/> Not Started	19	191	Sort by Model Descending

Source Script (Editable): Size: 675

```
Set Error Capture [ On ]
If [ Get(WindowMode) = 0 ]
# We're in Browse Mode: Enter Find Mode
Enter Find Mode [ ]
[ Pause ]
Else
# We're already in Find Mode. Let the Find button function the same as the
Continue button.
End If
Perform Find [ ]
If [ Get( LastError) = 400 ]
Show Custom Dialog [ Title: "Message"; Message: "No find criteria were entered."

```

Command (Editable):
Translate this FileMaker Pro code into Microsoft Access VBA code

Converted Script:
Microsoft Access VBA
On Error Resume Next
If Application.CurrentObjectName = "" Then
We're in Browse Mode: Enter Find Mode
DoCmd.RunCommand acCmdFind
If Err.Number = 2501 Then
No find criteria were entered. All records will be displayed.
MsgBox "No find criteria were entered. All records will be displayed.",
vbInformation, "Message"
DoCmd.RunCommand acCmdShowAllRecords
End If
Else
We're already in Find Mode. Let the Find button function the same as the
Continue button.
DoCmd.RunCommand acCmdFind
If Err.Number = 2501 Then
No records match this request.
MsgBox "No records match this request.", vbInformation, "Message"
DoCmd.RunCommand acCmdShowAllRecords
ElseIf DCount("", Application.CurrentObjectName) = 1 Then
Only one record found
DoCmd.RunCommand acCmdGoToRecord, , acGoTo, 1

Source: FileMaker Pro
Output Language: Microsoft Access VBA
Vendor: OpenAI
API Key: dcsi
Model: gpt-3.5-turbo
Tokens Used Today: 1,073
Output Filename: Find.bas
Need Help?

By default, the FmPro Code Conversion Workbench window opens with the (1) scripts listed in the grid, (2) FileMaker Pro selected as the Source Database Type and (3) Microsoft Access VBA selected as the Output Language.

Just because the options are selected a certain way when the window opens, doesn't mean you can't change them. You can easily change the default settings on this window for Output Language, Vendor, API Key Type, or AI Model - and the changes will be saved into the project or application preferences for FmPro Migrator.



+



FMPPro Code Conversion Workbench Demo

If you forgot to enter your license key or didn't have a license for the AI Accelerated version of FmPro Migrator, then the FmPro Code Conversion Workbench will open in Demo mode.

A pre-converted set of sample scripts will be displayed in the grid instead of the scripts from your own FileMaker Pro database. Clicking any of the scripts will display the text "This demo only includes a few scripts." in the Source Script field.

[At the present time, there aren't any FileMaker Pro to Access demo scripts.]