FileMaker 7

Converting FileMaker Databases from Previous Versions

FileMaker, Inc.
5201 Patrick Henry Drive
Santa Clara, California 95054

All persons and companies listed in the examples are purely fictitious and any resemblance to existing persons and companies is purely coincidental.

Credits are listed in the Acknowledgements document provided with this software.

For more information, visit our web site at www.filemaker.com.

Edition: 01

Contents

Chapter 1

Planning for database conversion

Chapter 2

Converting your database files

Chapter 3

Troubleshooting file conversion

Chapter 4

Changes that occur during conversion

Chapter 5
Defining accounts and privilege sets for files protected by the Web Security Database

Chapter 6
Considerations for associated files and interfaces

Chapter 1
Planning for database conversion

The file format of FileMaker® Pro has changed to support new features, such as multiple database tables per file, changes to internal data storage, and changes in how scripts and functions work. To use your existing files from previous versions with FileMaker 7 products, you must convert these files to the new file format used in FileMaker Pro 7.

About this guide

This document contains conversion-related information for FileMaker Pro, FileMaker Developer, and FileMaker Server.

Online Help provides comprehensive step-by-step instructions on FileMaker product features. You can open Help from the Help menu in FileMaker products.

Important You can download PDFs of FileMaker 7 documentation from www.filemaker.com/downloads. Any updates to this document are also available from the web site. For more information on upgrade, conversion, and migration considerations, see www.filemaker.com for TechInfo Knowledge Base articles and other resources.

Conversion overview

The method you use for updating your database for use with FileMaker Pro 7 depends on the complexity of your database and whether or not you can benefit by implementing new features available in FileMaker Pro 7.

Converting simple files

If your database solution is fairly simple, you should be able to convert your files using the instructions in "Converting your database files" on page 17 or as described in the *FileMaker Pro User's Guide*, test your converted files, and use them. For example, simple files that were built from the templates provided with your previous version of FileMaker Pro should convert accurately without a lot of manual adjustment afterwards.

Converting complex databases

If you're converting a multi-file relational database created with features such as custom features such as complex scripts, complex calculations, or a database that's business-critical, you should plan your conversion based on the following seven-step process.

Note The "Conversion checklist for complex databases" on page 8 provides detailed tasks for the six general conversion steps listed below. You might find it helpful use the checklist as a guide for the conversion process.

1. Plan for conversion.

   Determine which parts of your database might require special consideration and post-conversion verification.

2. Prepare for conversion.

Set up an environment in which you place converted files and store them as you work through the conversion process. Clean up current files, removing obsolete information and structures, such as scripts and layouts that are no longer used. Think about how you want to implement database security in FileMaker Pro 7. Information about what changes in your file during conversion is detailed in chapter 3, so you might want to browse through it before you convert your files.

3. Convert files.

   Using the procedures described in "Converting your database files" on page 17, convert your files to FileMaker Pro 7 format.

4. Verify conversion changes.

   Some databases need manual modification after conversion. Use "Verifying conversion changes" on page 12 to make sure that all aspects of your files have been converted properly and will produce intended results. Many items on the checklist won't apply to all users, but it's a good idea to review the entire list to make sure you don't miss anything.

5. Test the converted database.

   Before you start using converted databases, it's very important that you test them thoroughly to make sure every aspect has been converted successfully. You can use a subset of your data for testing, but make sure the data you test is representative of all functionality included in the database. For example, make sure that fields contain data that allows you to validate every calculation in the database. Be sure to consider calculations that return results from fields, calculations used to control which script steps are performed in a script, and calculations that are used to control which records are displayed using privileges. In addition, be sure to test user accounts, passwords, and privileges.

6. Populate the converted database with current data.
7. Delete test data and import data from converted copies of your current database.
8. Deploy the converted database

When you have completed the post-conversion tasks, tested the converted database, and are confident that it is ready for use, start using it.

Migrating data to a database designed in FileMaker Pro 7

Because there are extensive changes and improved features in FileMaker Pro 7, it might make sense to re-create your database in FileMaker Pro 7 rather than converting it in its current design and structure. For example, when you convert a multi-file database, the files are not combined into a single file with multiple, related tables. You may prefer to create a single new file in FileMaker Pro 7 with all of the tables and relationships that you want, and then import data from your converted copies of your original files.

By designing your database in FileMaker Pro 7, you can remove obsolete functionality and get the full benefit of the improvements made to FileMaker Pro.

Conversion checklist for complex databases

Use the following checklist to plan and complete your file conversion. If the task doesn't apply to your solution, check N/A (Not Applicable).

Planning for conversion

Done	N/A	Task
□	□	Learn about how file sharing has changed in FileMaker 7. See “Access privileges” on page 29. In addition, if your database is: ▪ hosted by FileMaker Server, familiarize yourself with the new features of FileMaker Server 7 and plan how all of your databases currently hosted by FileMaker Server will be converted or migrated. ▪ published to the web, see “Changes to databases that are published to the web” on page 63 and the *FileMaker Instant Web Publishing Guide* or the *FileMaker Server Advanced Custom Web Publishing Guide*.
□	□	Determine whether you have the correct hardware and software to support the FileMaker products you plan to use: ▪ Check the system requirements in the *Installation and New Features Guide for FileMaker Pro 7 and FileMaker Developer 7* and make sure you have adequate resources on the computer you plan to place the database (and all client computers in a multi-user environment). ▪ Make sure you have enough disk space for converted files. Converted files can require more disk space than the original files if the same images are used in container fields and on layouts. Files that contain many graphics that are not in JPEG or GIF format could grow by as much as 50% to 75%. ▪ If the database is shared by FileMaker Server, make plans to upgrade the server computer to FileMaker Server 7 or FileMaker Server 7 Advanced and all client computers to FileMaker Pro 7. Check the system requirements in the *FileMaker Server Administrator’s Guide* and the *FileMaker Server Advanced Web Publishing Installation Guide*, if you plan to use web publishing to host the database on the web. **Note** Versions of FileMaker Pro previous to FileMaker Pro 7 can’t communicate with FileMaker Server 7. If you are not planning to upgrade all databases to FileMaker Pro 7, you need to acquire a separate server computer to host the databases you are converting. ▪ If the database is very large, consider acquiring a separate computer (or set of computers) for conversion and testing activities.
□	□	If you’re not planning to convert all shared databases to FileMaker Pro 7 before deployment, consider how you plan to make multiple versions of FileMaker Pro and FileMaker Pro Server available to users. ▪ Only one version of FileMaker Pro or FileMaker Server can host files on a single computer. Therefore, a second computer for hosting files converted to version 7 (either peer-to-peer using FileMaker Pro 7 and FileMaker Networking or FileMaker Server 7) is required. ▪ If you are planning to run your previous version of FileMaker Server, FileMaker Developer, or FileMaker Pro simultaneously with version 7, see www.filemaker.com for licensing restrictions.
□	□	If the database is relational, account for all related files in your conversion plan. Files from previous versions of FileMaker Pro will not interact with converted files.

Done	N/A	Task
□	□	Determine if your database uses any associated files, such as third-party plug-ins, that need to be verified or updated for use with FileMaker Pro 7.
□	□	Determine whether your should convert the existing files or migrate the database by creating a new design in FileMaker Pro 7 and then import data. For information about file conversion, review "Changes that occur during conversion" on page 29. For information about enhancements and improvements made in FileMaker Pro 7, see the *Installation and New Features Guide for FileMaker Pro 7* and the *FileMaker Pro User's Guide*.
□	□	If your database has container fields that store Windows Metafile images (and certain other images stored internally by FileMaker in a metafile format, such as OLE, SLD, or TIFF metafile), perform the conversion on a Windows computer that has QuickTime installed to preserve the images. If you perform the conversion on a computer running Mac OS, you do not lose any data, but you cannot immediately drag, drop, copy, or paste the images to another container field on a computer running Mac OS: you must first open the database on a Windows computer and then view each record with a container field to convert the metafile images.

Important FileMaker, Inc. does not warrant the functionality of database solutions that rely on undocumented behavior of FileMaker products. Databases that have exploited unintended behaviors may not convert and function correctly in the FileMaker Pro 7 and FileMaker Server 7 environments. It is the responsibility of the database developer to verify and test the function of converted solutions.

Preparing for conversion

Done	N/A	Task
□	□	If you are converting a database that contains a great number of records, consider cloning it before conversion. See "Saving files" in FileMaker Pro Help. **Tip** Converting a clone of your database is quicker than converting a database that contains a lot of data. You can make modifications to the layouts, calculations, scripts in the converted clone, convert the original database, and then import the data into the clone.
□	□	Document your current database definitions to use as a reference as you review and modify converted files. ▪ Print your current field definitions, script definitions, and layouts ▪ If you have FileMaker Developer 6, consider creating a Database Design Report. This report details the structure of your database and could be more useful than printing database information feature by feature.
□	□	If your database is extremely complex or business-critical, diagram its structure and the flow of information through it identifying all layouts and reports. This diagram can serve as your conversion roadmap during verification and testing.

Done	N/A	Task
□	□	Prepare for security changes that will be required after conversion. The security model of FileMaker Pro 7 is different and greatly improved, so existing security settings must be redefined during conversion. See “Access privileges” on page 29 and the *FileMaker Security Guide*. ▪ Create a description of the security model used in the current database. ▪ Determine if each user of your file should have a separate password or if passwords can be shared.
□	□	Prepare your conversion environment. ▪ Create backups of the files you plan to convert and place them in a separate folder away from the existing database that will continue to be used during conversion. ▪ If you are converting multiple files that exist in a nested folder structure, create destination folders in the same structure. See “Converting multiple files at once” on page 18. ▪ Set up dedicated folders for testing the converted database. ▪ Consider setting up a separate computer (a self-contained environment) to use during conversion.
□	□	Examine the existing database and remove any obsolete definitions or structures, such as passwords, scripts, layouts, and fields that are no longer used. **Tip** If you have FileMaker Developer, creating a Database Design Report for your original database can help identify areas that might benefit from pre-conversion clean-up.
□	□	Review data for consistency and printer settings (Windows) in existing databases before conversion. Some areas that benefit from pre-conversion clean-up are: ▪ **Two-digit dates.** If pre-1900 or post-1999 years are entered in two-digit format, change them to four-digit format. See “Two-digit date conversion” on page 37. ▪ **Date separators.** If dates use an alphanumeric character or the time separator as a date separator, consider replacing date separators before conversion. For more information, see “Date and time fields” on page 42. ▪ **Text in number fields.** Determine whether text in numeric fields should be removed before conversion. See “Text functions recognize text characters in number fields” on page 52. ▪ **Calculation parameters from number fields.** If parameters in calculations are number fields, was the expectation that alphabetic data returns a blank value or do Boolean calculations expect “T,” “Y,” “F,” or “N”? See “Numeric expressions with text input” on page 54. ▪ **Fonts.** Windows: If layouts use non-TrueType fonts, consider switching to TrueType fonts before conversion. For more information, see “Modified text styles and font display on layouts” on page 36. ▪ **Printer settings.** Windows: Printer settings might be lost during file conversion. See “Printer settings (Windows)” on page 40.

Done	N/A	Task
□	□	If the database uses associated files, such as plug-ins, that need to be updated for compatibility with FileMaker Pro 7: ▪ For databases protected by passwords, determine how the associated files will be updated to support the account name and password model in FileMaker Pro 7. Remember to review account and password settings in associated files, if you modify accounts and passwords after file conversion. ▪ For third-party plug-ins, contact the plug-in provider and verify that plug-ins have been certified for use with FileMaker Pro 7. Obtain updated versions of plug-ins, if needed. ▪ For databases that are published to the web and will be hosted by FileMaker Server 7 Advanced, obtain new web enabled plug-ins. (Plug-ins written to work with FileMaker Pro 7 do not automatically support the new web publishing engine or server plug-in environments.) ▪ For other associated files that interact with FileMaker Pro, such as Apple events, ActiveX, or ODBC/JDBC, verify and test files developed in-house and by third-parties. For more information, see "Apple event changes in FileMaker Pro 7 (Mac OS)" on page 75.

Converting files

Done	N/A	Task
□	□	If you are converting a FileMaker Pro 2.x or 1.x file, convert the file to FileMaker Pro 6 as described in "Converting FileMaker Pro 1.x and 2.x databases" on page 21.
□	□	If you are converting a large database with an extensive index, consider skipping index conversion. See "Indexes" on page 39.
□	□	Convert single-file databases using the instructions for "Converting a single file" on page 17.
□	□	Convert a relational database or a group of files using the instructions for "Converting multiple files at once" on page 18.

Verifying conversion changes

Done	N/A	Task
□	□	Review the Conversion.log file located in the folder that contains the converted files and determine areas of the converted database that might need special attention. See "Troubleshooting file conversion" on page 23.
□	□	Using the security model description you created as you prepared for conversion, review the accounts and privilege sets in the converted database and change passwords as needed to improve security. "Access privileges" on page 29.
□	□	Review converted file references and change any asterisks in remote file references to domain names or IP addresses. See "Wild card characters in converted file references" on page 34.

Done	N/A	Task
□	□	Using the database structure diagram that you created as you prepared for conversion, eliminate redundant relationships in the relationships graph.
□	□	If the database uses a user-defined dictionary, make sure it is being accessed after conversion. See “Preserving user dictionary information” on page 35.
□	□	If the database uses external functions, make sure they will evaluate properly. See “Accelerated calculation evaluation” on page 50.
□	□	If records might contain alphabetic characters in number fields, determine if they need to be modified. For example, if some records contain data with a leading T or Y to indicate “true” or a leading F or N to indicate “false,” the calculations that use these fields must be modified manually. See “Number fields” on page 41.
□	□	Determine if you want FileMaker Pro 7 to ignore case-sensitivity when sorting. See “Accented characters” on page 38.
□	□	Determine if summary reports need to be manually adjusted. See “Sorted data and found sets” and “Sorting records by subsummary values” on page 38.
□	□	Determine if the database contains Get functions (converted from Status functions) that contain calculations in parameters. See “Status functions changed to Get functions” on page 44 and “Change to rules for allowed parameters in Get functions” on page 47.
□	□	If the database includes calculations that use the Status(CurrentFilePath) function, you might need to make adjustments to handle changes in local file path syntax. For example, any If statements that evaluate the string returned from the Status(CurrentFilePath) function need to be adjusted to handle local files correctly. See “Status functions changed to Get functions” on page 44.
□	□	If the database uses the Mod function in a calculation, be aware that the definition of this function has changed in FileMaker Pro 7. See “Changes to the Mod function” on page 53.
□	□	If the database contains Japanese functions, determine whether you need to examine and modify parameter specifications. See “Japanese functions that are converted” on page 46.
□	□	Determine whether you want filename extensions to appear in calculation results and make adjustments, if necessary. See “Functions that return filename extensions” on page 53.
□	□	Determine if the database contains calculations that might be dependent on the context of the current window. See “Functions that are dependent on context” on page 52.
□	□	If the calculations in the database contain textual parameters when numeric parameters are expected, review calculation results after conversion. See “Numeric expressions with text input” on page 54.
□	□	Determine if any script steps have text specified in ancillary dialog boxes. See “Some script step options convert to calculations” on page 62.
□	□	Determine if the database uses the Get(UserName) function to track user activities. See “Get(UserName) function” on page 55.

Done	N/A	Task
□	□	Determine if certain database fields should be indexed. ▪ If you converted the index, review it and make necessary adjustments to it. See “Indexes” on page 39. ▪ If you didn’t convert the index, consider creating one.
□	□	If the database uses associated ActiveX files, reset the ActiveX library reference. See “ActiveX library renamed” on page 75.
□	□	Review and adjust settings for ODBC/JDBC. See “LDAC and RDAC changed to ODBC/JDBC and FileMaker Server Advanced” on page 74.
□	□	Verify third-party plug-ins are working correctly.
□	□	Mac OS: Review and adjust settings for Apple events. See “Apple event changes in FileMaker Pro 7 (Mac OS)” on page 75.
□	□	If you plan to publish this database to the web, update settings to accommodate the new web publishing model in FileMaker Pro 7. See “Changes to databases that are published to the web” on page 63, “Defining accounts and privilege sets for files protected by the Web Security Database” on page 67.
□	□	Determine if there are design changes you want to make to eliminate redundant or obsolete items or implement new features before you begin testing the database.

Testing the converted database

Done	N/A	Task
□	□	If you cloned the database before conversion, import sample data that will allow you to exercise all functionality in the database. ▪ Whether you cloned the database or not, consider testing a subset of your data, but make sure the sample you test fully represents your data and allows you to exercise every aspect of the database. ▪ Disable all accounts except the Admin account and turn off indexing during testing.
□	□	Set up a back-up plan to save changes that you make to the database during testing to let you revert to the last saved version without losing too much work, if you make mistakes.
□	□	Test every aspect of the database. For example, test: ▪ Access privileges for shared files (via FileMaker Server, FileMaker Network, Web Publishing, or ODBC/JDBC) ▪ Layouts (check all fonts and graphics) ▪ Value lists ▪ Buttons ▪ Data entry fields (manual, auto-entry, and validation settings) ▪ Calculations (trigger all calculations, including calls to external functions) ▪ Reports (both online appearance and printed output) ▪ Scripts

Done	N/A	Task
□	□	If you converted a multi-user database, consider having a “test day” so many users can test the converted database simultaneously.

Populate the converted database with current data

Done	N/A	Task
□	□	After testing is completed, delete all test data by deleting records or saving as a clone. **Tip** Saving as a clone is a better method because the resulting files are more compressed. Remember to save the clone to the correct destination folder, then remove the test file and remove “<space> clone” from the clone’s filename.
□	□	Shut down the pre-FileMaker Pro 7 database to ensure users make no changes as you move data to the verified copy of the database or clone.
□	□	Convert the database that contains current data. You can’t import data directly from a previous version of FileMaker Pro, so you must convert the current database and transfer the data to the verified copy of the database or clone. **Tip** Skipping index conversion when you convert the database can save time.
□	□	Import current data from converted copies of your current databases. ▪ Determine if you want auto-enter values to be copied from the database or have the auto-enter calculation triggered when data is imported **Important** Do not convert auto-enter values in a relational database. Doing so in a relational database can disassociate the related fields.
□	□	Reset automatically entered serial number options for fields based on current data.
□	□	If you converted auto-enter values, populate new fields that contain auto-enter values.
□	□	Adjust back-up schedule to normal operation settings, if necessary.

Deploy the converted database

Done	N/A	Task
□	□	Make sure all database users have FileMaker Pro 7 installed on their computers and that user-defined dictionaries are accessible. See “Preserving user dictionary information” on page 35.
□	□	Distribute new account name and password information for the converted database, if necessary.
□	□	Verify that associated files, such as plug-ins and Apple events, are communicating with the “live” database.
□	□	Announce that the converted database is available and online.
□	□	Closely monitor activity to make sure you haven’t overlooked anything during testing.

Special considerations for converting business-critical databases

If the database you are converting is essential to your business and you can’t take it off-line during the time it takes to convert and test it, the following guidelines can help you plan for conversion. For more information on upgrade, conversion, and migration considerations, see www.filemaker.com for TechInfo Knowledge Base articles and other resources.

- Use the checklists in “Conversion checklist for complex databases” on page 8 to prepare for database conversion. Acquire a computer for conversion and a second dedicated server environment for testing, and plan an extremely aggressive back-up schedule (every 15 to 30 minutes) so you can easily revert manual changes made after conversion.
- If it is essential that you get the converted database online as quickly as possible, do not plan design changes or feature enhancements at this time. Convert the database to FileMaker Pro 7 and verify it as described in this chapter before making it available to users. If you have flexibility in the conversion and test period, consider whether this is a good time to make design changes that take advantage of new features in FileMaker Pro 7.
- If you have FileMaker Developer 7, generate a Database Design Report, which can help you identify areas that might need manual manipulation as described in this document. This can be especially helpful if you are converting a database that you didn’t design and develop yourself.
- Ensure that when the converted database goes online, it contains current data, including any changes or additions made during the conversion period.

Chapter 2
Converting your database files

FileMaker Pro 7 can convert files created with versions 3.x, 4.x, 5.x, and 6.x. If you have files created using FileMaker Pro version 2.x or earlier, you must first convert them to one of the supported formats. Then you can convert the files to FileMaker Pro 7 format as described in this guide. For more information, see “Converting FileMaker Pro 1.x and 2.x databases” on page 21.

Converting files

When you convert files, FileMaker Pro 7 always preserves your original files and creates new, converted files in FileMaker Pro 7 format. The original files are not modified, and you can open them in the previous version of FileMaker Pro that created them. The converted files can be opened only in FileMaker Pro 7.

You can convert a single file or convert multiple files at once.

- Use the single-file conversion method for stand-alone database files that don’t display related data from other files. For more information, see the next section.
- Use the multiple-file conversion method to convert all the files in a relational database. For more information, see “Converting multiple files at once” on page 18.

Converting a single file

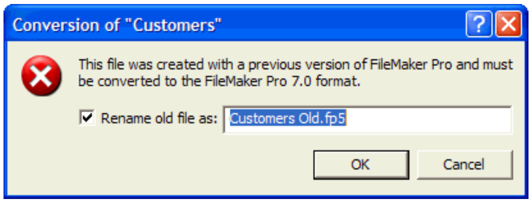
To convert and open a single file:

1. If the file you want to convert is currently open in the previous version of FileMaker Pro, close the file.

   Files that are open won’t convert properly.
2. Start FileMaker Pro 7.
3. Do one of the following:
   - If you see the Open New or Existing File dialog box, continue with step 4.
   - If you see the New Database dialog box, select Open an existing file, and then click OK.
   - If you don’t see any dialog box, choose File menu > Open.
4. In the Open dialog box, select the file to convert, and then click Open.

   Note You can also begin conversion by dragging a file onto the FileMaker Pro 7 application icon.

   The Conversion dialog box appears, where you can rename the existing file prior to conversion. FileMaker Pro automatically adds “Old” to the filename.



This file was created with a previous version of FileMaker Pro and must be converted to the FileMaker Pro 7.0 format.

☑ Rename old file as: Customers Old.fp5

5. Do one of the following:

   - Skip renaming the old file by clearing the checkbox named Rename old file and clicking OK. It's not necessary to rename the old file if it has an.fp5,.fp3, or.fmj extension because your converted file will have an .fp7 extension. In Windows, the filename extension .fp7 is added to all converted files.
   - Rename the old file by entering the name you want and clicking OK. Do this if you're using the Mac OS, the file does not have an extension, and you want to continue using the existing filename for the converted file.

   The Name Converted File dialog box appears.

6. Enter the filename that you want for the new file.

   If you renamed the old file in step 5, the default filename for the converted file is the original filename (with a .fp7 extension, if the original file had an extension). If you did not rename the old file, the default filename is *filename* Converted.

7. Click Save to start the conversion.

   If the file you're converting is large, the File Format Conversion dialog box appears and shows the progress of the conversion. You can save time when converting a large file by clicking Skip when the index is being converted, which postpones index creation until later.

   In most cases, FileMaker Pro converts the file and opens it. If conversion doesn't work or error messages appear, see "Solving conversion problems" on page 23.

8. Using a text editor such as Notepad (Windows) or TextEdit (Mac OS), open the Conversion.log file located in the folder that contains your converted database.

   Tip You can drag the Conversion.log file onto the FileMaker Pro 7 application to open it in FileMaker Pro.

The conversion log contains conversion status messages and errors, which indicate whether you need to make manual corrections to the converted file before using it. For more information about the log file, see "Using the conversion log file" on page 24.

Converting multiple files at once

FileMaker Pro 7 can convert multiple files at the same time. You should use the multi-file conversion method when you need to convert a database composed of multiple related files, or if you simply have many FileMaker Pro files that you want to quickly convert without being prompted to individually rename each file.

When you convert multiple files at the same time, FileMaker Pro prompts you to specify a folder where you want the converted files to be saved, leaving your original files unchanged. FileMaker Pro automatically names each converted file without prompting you, usually with the same name as the original file except with a .fp7 extension added (Windows) or if the original file has an extension (Mac OS). FileMaker Pro won't overwrite any existing files in the folder. Instead, it adds a number to the end of the filename, and increments that number until a unique name is found.

There are several ways that you can select the files that you want to conver. You can drag and drop the files (or a folder containing the files) onto the FileMaker Pro 7 application icon, or you can select multiple files in the FileMaker Pro Open dialog box.

To convert multiple files at once:

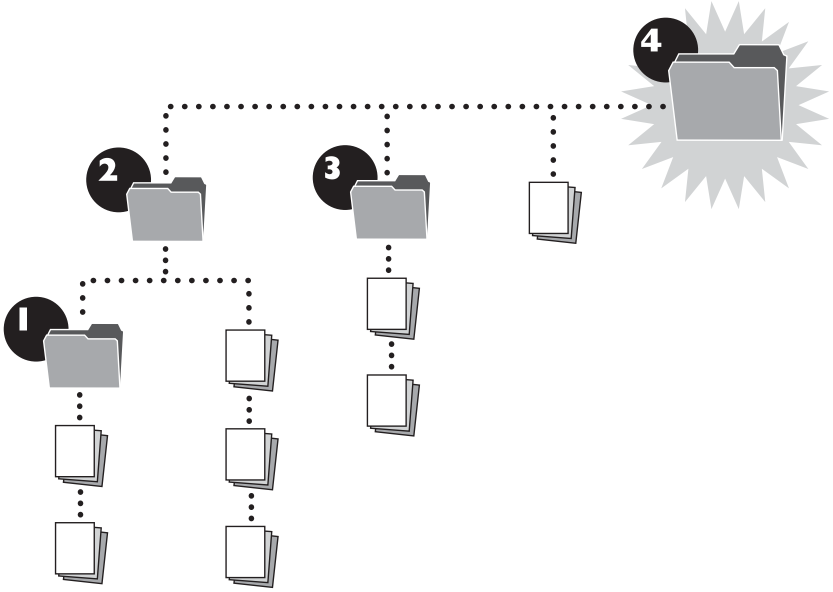
1. If the files you want to convert are currently open in the previous version of FileMaker Pro, close the files.

   Files that are open won't convert properly.

2. In Windows Explorer or in a Finder window (Mac OS) create a new, empty folder for the converted files.

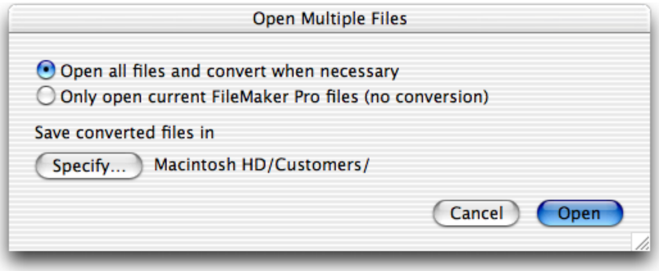
3. Do one of the following to select and open the files you want to convert:

   - **Files method:** In Windows Explorer or in a Finder window (Mac OS) select the files you want to convert, and then drag them onto the FileMaker Pro 7 application icon.
   - **Folder method:** If a folder contains all of the files you want to convert, drag the folder onto the FileMaker Pro 7 application icon. If you are converting a multi-file relational database in which files are organized in subfolders, you need to convert each folder separately. First create new folders in the same structure for the converted files. Then convert files in subfolders at the lowest level first by dragging the subfolder onto the FileMaker Pro 7 application icon and indicating the correct new destination folder when prompted. When all related files in subfolders have been converted, use the "Files method" described above to convert files located in the original parent folder.



   - **Open dialog box method:** In FileMaker Pro 7, choose File menu > Open, and select the files you want to convert in the Open dialog box by holding down Ctrl (Windows) or ⌘ (Mac OS), and clicking each file. (You can also select a range of adjacent files by clicking the first file and then Shift-clicking the last file in the range.) Then click Open.

The Open Multiple files dialog box appears.



Open Multiple Files

● Open all files and convert when necessary
○ Only open current FileMaker Pro files (no conversion)

Save converted files in

Specify... Macintosh HD/Customers/

Choose	To
Open all files and convert when necessary	Open all FileMaker Pro files or files in a supported file format (such as tab-delimited text or a Microsoft Excel file) in the folder. All other files are ignored.
Only open current FileMaker Pro files (no conversion)	Open only FileMaker Pro 7 files found in the folder. If files from previous versions are found in the folder, they are ignored. If you choose this option, skip to step 6.

4. Click Specify, choose the folder where you want to save the converted files, and click OK (Windows) or Choose (Mac OS).

   Important Do not choose a folder that contains any of your original files. If you do, then converted files may be renamed (by appending a number to the filename) to prevent the converted files from replacing the original files. This renaming of files could cause file references to convert improperly, which affects relationships and other features that rely on file references.

   If necessary, you can create a new, empty folder by clicking Make New Folder (Windows) or New Folder (Mac OS).

5. Click Open in the Open Multiple files dialog box.

   If the files you're converting are large, the File Format Conversion dialog box appears and shows the progress of each conversion. You can save time when converting a large file by clicking Skip when the index is being converted, which postpones index creation until later.

   FileMaker Pro 7 creates converted files in the folder you specified and opens them, leaving your original files unchanged. In Windows, the filename extension .fp7 is added to all converted files. In the Mac OS, the .fp7 extension is added to all converted files that previously had an.fp3,.fp5, or.fmj extension.

6. Using a text editor such as Notepad (Windows) or TextEdit (Mac OS) open the Conversion.log file located in the folder that contains your converted database files.

The log file contains conversion status messages and errors, which help determine whether you need to make manual corrections to the converted files before using them. For more information about the log file, see "Using the conversion log file" on page 24.

Note If you later need to convert the same files again, make sure you convert the files into an empty folder; move or delete any files from the previous conversion attempt, or create a new folder for the next conversion. This prevents converted files from being renamed during the next conversion which could cause file references to convert improperly.

Converting FileMaker Pro 1.x and 2.x databases

FileMaker Pro 7 cannot convert databases created with FileMaker Pro 1.x or 2.x. To use these files with FileMaker Pro 7, you must convert them using FileMaker Pro 3.x, 4.x, 5.x, or 6.x, and then convert them again using FileMaker Pro 7.

Previous versions of FileMaker Pro for Windows can convert only 2.x files. If you need to convert a 1.x file, convert the file on a Mac OS computer using FileMaker Pro 3.x, 4.x, 5.x or 6.x.

If you don’t have a previous FileMaker Pro application with which to do the conversion, you can download a trial version of FileMaker Pro 6 from www.filemaker.com and use it to convert files.

Chapter 3
Troubleshooting file conversion

Solving conversion problems

FileMaker Pro 7 can't open my file

FileMaker Pro 7 can't convert databases that were created with FileMaker Pro 1x or 2x. For more information, see "Converting FileMaker Pro 1.x and 2.x databases" on page 21.

My file didn't convert properly

- Make sure the file is closed before you attempt to convert it.
- If you are converting a copy of a file, make sure that the original file was closed before you copied it. Otherwise, the copy will not convert correctly.
- Check the Conversion.log file located in the conversion destination folder. See "Using the conversion log file" below.
- Try recovering the file first using a previous version of FileMaker Pro and then convert the file again. See "Recovering files" in FileMaker Pro Help.

I received a consistency check or auto-repair message when I opened my original file using my previous version of FileMaker Pro

Try recovering the original file using your previous version of FileMaker Pro, then open it using the same version of FileMaker Pro before trying to convert with FileMaker Pro 7.

I'm not sure if the conversion was successful

Consult the Conversion.log file located in the same folder as the converted file and the "Conversion checklist for complex databases" on page 8.

Note If you convert a database that used the Expense Report template, you might find two blank layouts in your file. You can delete these layouts.

The converted file won't accept my password

Passwords are case-sensitive in FileMaker Pro 7. Make sure you enter your password exactly as it was created in a previous version of FileMaker Pro. If you are the owner of the original file or the database administrator, open the original file using a previous version of FileMaker Pro and look at the password in the Define Passwords dialog box. If you are not the owner of the file or the database administrator, consult your database administrator for password information.

If you need to change a password, see "Editing existing accounts" in the FileMaker Pro 7 Help.

The converted file opens very slowly

The converted file could contain file references that point to external locations, which could be changed to local paths, or wild card characters could be used in FileMaker Network (FMNET) paths. For more information, see "Poor performance when opening files" and "Wild card characters in converted file references" on page 34.

Using the conversion log file

The conversion log reports conversion status and possible problems found during conversion, such as fields with invalid names. In most cases, you can correct problems in converted files. If the log file indicates that there is a problem that would be complicated and time-consuming to fix after conversion, you may want to try correcting the problem in the original file using a previous version of FileMaker Pro, and then convert the file again.

The log file is created in the same folder as your converted files. If you use the multi-file conversion method to convert more than one file at a time, conversion status information for all files appears in one log file. Conversion information is appended to the log as each file in the folder is converted.

Log entries appear in the following format:

```
date time, filename, error_number, conversion status/error message
```

Note If the status or error message refers to a condition in a file that was renamed during conversion, the new filename (not the original filename) is reported in the log.

The following sections describe the status messages that may appear in the conversion log. Not all status messages are reported for every file. For example, if there are no container objects in the file you are converting, there are no messages about container fields in the log file.

Tip You can import the conversion log into a database to run reports on logged information for further analysis.

Status of items that are removed from or changed in the converted file

Status message	Action required
The “Today” function was replaced with a “Today” field and a startup script to update it.	None. For more information, see “Today function no longer supported” on page 51.
“Status(CurrentWebSharing)” was converted to “(Position(Get(ApplicationVersion), "Web", 1, 1) > 0).”	Depending on your usage of Status(CurrentWebSharing), you might need to modify your calculation. For more information, see “Changes to databases that are published to the web” on page 63.
A sort order based on a summary field was updated to try to match old behavior.	Verify report results. For more information, see “Sorting records by subsummary values” on page 38.

Reports of damaged objects in converted files

Error message	Action required
Field damaged: *field_name*.	Inspect the field in the converted file for lost attributes. For example, a data entry option or the entire field could be missing in the converted file.

Error message	Action required
Relationship damaged: *name_of_relationship*.	Inspect the relationship and table occurrences in the relationships graph in the converted file for lost attributes. (Relationship names become table occurrence names during conversion.) For information about how to display the relationships graph, see “Converted relationships” on page 37 and “Viewing the relationships graph” in the FileMaker Pro 7 Help.
Layout damaged: *name_of_layout*.	Inspect the layout in the converted file for lost attributes. For example, one or more objects on the layout or the entire layout could be missing in the converted file.
Script damaged: *script_name*.	Inspect the script in the converted file for lost attributes. For example, one or more script steps or the entire script could be missing in the converted file.
Calculation damaged: *calculation_field_name*.	Inspect the calculation in the converted file for lost attributes. The calculation could be “commented out” due to syntax changes in FileMaker Pro 7.
Access privilege damaged: <Unknown>.	Inspect accounts and privilege sets in the converted file for lost attributes. For security reasons, specific account and privilege set information is not logged.
Value list damaged: *name_of_value_list*.	Inspect the value list in the converted file for lost attributes.
File reference damaged: *file_reference_name*.	Inspect the file reference in the converted file for lost attributes.

Invalid object names in converted files

Error message	Action required
Poor field name: *field_name*.	None, but consider inspecting calculations that reference the field in the converted file. The calculations may have been modified to allow the use of an reserved character or word in a field name. See “Invalid characters and words in field names used in calculations” on page 47.
Poor relationship name: *name_of_relationship*.	None, but consider inspecting table occurrences in the relationships graph in the converted file. (Relationship names become table occurrence names during conversion.) For more information, see “Converted relationships” on page 37.
Poor script name: *script_name*.	None, but consider inspecting the script in the converted file. The script name might have been shortened. See “Script names have been shortened” on page 60.
Invalid password name: <Unknown>.	Verify the account and password in the converted file. For security reasons, password information is not logged.

Duplicate object names in converted files

Status message	Action required
Duplicate field: *field_name*.	Inspect the field in the converted file. Duplicates might have been renamed. You can rename the field manually in FileMaker Pro 7 after conversion, if you want. For information about objects that might be renamed during conversion, see “Reserved words” on page 47.
Duplicate relationship: *name_of_relationship*.	Inspect the relationship in the relationships graph in the converted file. Duplicates might have been renamed. (Relationship names become table occurrence names during conversion.) For more information, see “Converted relationships” on page 37 and “Viewing the relationships graph” in the FileMaker Pro 7 Help.
Duplicate privilege: *privilege_set_nameX*.	Inspect privilege sets in the converted file. Two identical privilege sets were created based on the group settings in the original file. In the converted file, they share the same name but one has a number appended to it (for example, GroupA/GroupA2).
Duplicate file reference: *file_reference_name*	Inspect the file reference in the converted file. Duplicates might have been renamed (see “File references” on page 33 and “Miscellaneous error messages” below). You can rename the file references manually in FileMaker Pro 7 after conversion, if you want.

Account status messages

Status message	Action required
Guest account disabled.	Review account settings in the converted file. For more information see “Access privileges” on page 29 and “About accounts, privilege sets, and extended privileges” in the FileMaker Pro 7 Help.
Default user account missing.	Review account settings in the converted file. For more information see “Access privileges” on page 29 and “About accounts, privilege sets, and extended privileges” in the FileMaker Pro 7 Help.

Miscellaneous error messages

Error message	Action required
Record damaged: *record_number*.	Use your previous version of FileMaker Pro to inspect the record identified in the original file, then compare it to the same record in the converted file. The record number is the current record number displayed next to the book in the status area and is based on the complete data set. To find the correct record in the original database, make sure that all records are displayed in the original database (choose Records menu > Show All Records) and that data is not sorted. Then use the bookmark to navigate to the record identified in the Conversion.log file.
Unknown Error: *error_text*.	Record the error text in case you need to contact Technical Support at FileMaker, Inc.
Skipped out-of-range field repetition: *x*.	Inspect data in this repetition in the original file. It was identified as damaged and was not converted.
Skipped unknown field key.	Ensure that all fields were converted.
Calculation parsing error: *field_name*.	Inspect the calculation in the converted file for errors. Possibilities are calculation validation, auto-enter, or calculation field errors.
Conversion aborted.	None, if you cancelled the conversion manually. If not, inspect the conversion log for possible conversion problems.

Miscellaneous status message

Status message	Action required
Deleting unneeded file reference: *file_reference_name*.	None. When a duplicate file reference is deleted, the reference is pointed to the original file reference. For example, if the original file had two file references (FileRef1 and FileRef2) that pointed to the same file, FileRef2 is deleted during conversion and FileRef1 is used anywhere FileRef2 was originally used.
Verify Windows Print Setup settings: *script_name*	Printer settings may not convert properly in Windows. After conversion, check settings in scripts that use the Print Setup and Print script steps. For more information, see “Printer settings (Windows)” on page 40.
Verify Windows Print settings: *script_name*	Printer settings may not convert properly in Windows. After conversion, check settings in scripts that use the Print Setup and Print script steps. For more information, see “Printer settings (Windows)” on page 40.

Chapter 4
Changes that occur during conversion

In many cases, files convert completely and are ready to use with no additional modification. Many of the changes discussed in this chapter are for information purposes only and are documented to help you recognize areas that change during conversion. However, you might have to make modifications to your files to make your database function as it did with previous versions of FileMaker Pro.

Important You can download PDFs of FileMaker 7 documentation from www.filemaker.com/downloads. Any updates to this document are also available from the web site. For more information on upgrade, conversion and migration considerations, see www.filemaker.com for TechInfo Knowledge Base articles and other resources.

Access privileges

FileMaker Pro 7 introduces many changes and improvements to database access and security.

In previous versions of FileMaker Pro	In FileMaker Pro 7
Passwords restrict access to a file and define what can be done in the file (for example, create, delete, or print records).	**Accounts** restrict access to a file by requiring a user to enter an account name and (usually) a password. Valid account information is required to open a protected file.
Groups define more specific access to the file, such as which layouts or fields can be viewed.	**Privilege sets** specify a level of access to a file, such as which layouts are viewable, which menus are available, and whether printing is permitted. **Extended privileges** determine the data sharing options that are permitted by a privilege set, such as whether a privilege set permits users to open a shared file or view a database in a web browser.

During the conversion of a previous file, access privileges convert as follows:

- Each password converts into an account.
- Passwords and groups convert into the appropriate number of privilege sets needed to provide the same levels of access that the passwords and groups provided in the previous file.
- FileMaker Pro 7 adds certain default accounts and privilege sets during the conversion process.

The three sections that follow provide details on the conversion of access privileges.

Note If you used the Web Security Database in a previous version of FileMaker Pro to control access to web-published databases, see “Defining accounts and privilege sets for files protected by the Web Security Database” on page 67 for more information on converting these access settings.

Passwords

During conversion, passwords are converted to accounts in FileMaker Pro 7. Because an account requires both an account name and a password in FileMaker Pro 7, for each converted account, both the account name and the password in the converted file are initially set to the password from the original file.

Important The conversion process makes all passwords visible to anyone who has privileges to open the Define Accounts & Privileges dialog box. For security reasons, you should either change each account name so that it no longer matches its password, or change each password so that it no longer matches its account name. (You could also set an option for each account to require a password change by the user upon the next login.)

Also note the following about converted passwords:

- If the file you're converting uses the Try default password document preference to automatically open the file with a certain password, this password is retained in the converted file as a Log in using account name and password in the File Options dialog box. If you later change the account name or password of this account, make sure you also update the Log in using settings in the File Options dialog box. Otherwise, the file will prompt for account information when the file is opened.
- Previous versions of FileMaker Pro permit defining a blank "(no password)" password, which allows users without a password to open a database with limited privileges. If a blank password exists in a file created with a previous version of FileMaker Pro, the Guest account will be enabled in the converted file. If a blank password does not exist, the Guest account is present but disabled in the converted file.
- Passwords in FileMaker Pro 7 are case-sensitive. In previous versions of FileMaker Pro, passwords are not case-sensitive. When a password such as "Them" is converted, you must enter it as "Them" in order for it to work. The previous version would have accepted "them" and "THEM," but FileMaker Pro 7 will not.

If you convert a file that does not contain any passwords (an unprotected file), the converted file initially contains two default accounts:

- **Admin:** this account is created with an account name only (no password) and is assigned the Full Access privilege set. The file will automatically open to the highest level of account access. You can restrict access to the file by defining accounts and disabling the Log in using option in the File Options dialog box.
- **Guest:** this account is present but disabled in unprotected files.

Groups

During conversion, privilege sets are created based on how passwords and groups are defined in the original file. The converted file will contain one privilege set for each unique combination of password/group settings in the original file. Although this provides the appropriate number of privilege sets for the same levels of access available in the previous file, the number of privilege sets most likely won't match the number of groups. Depending on how groups are assigned to passwords in the original file, your converted file may contain a greater or fewer number of privilege sets.

The converted privilege set names are based on the group names in the file being converted:

- If the original file includes passwords to which a single group is assigned, then some privilege set names will be identical to group names in the previous file.
- If the original file includes passwords to which multiple groups are assigned, then some privilege set names will combine the names of several group names in the original file.

- If the original file includes passwords to which no group is assigned, then the converted file will contain one or more generic privilege sets named Privilege Set 1, Privilege Set 2, and so on.

Some groups may not appear in the converted file. This can occur when:

- Two or more groups in the original file have identical settings. When this occurs, only the first occurrence of the group is retained in the converted file, and the duplicates are ignored.
- A group has no passwords assigned to it (other than the master password). Unused groups do not convert to privilege sets in the converted file.

Also note the following about the conversion of passwords and groups to privilege sets:

- Three default privilege sets are always added to the converted file: Full Access, Data Entry Only, and Read-Only Access.
- If the file you're converting is configured to be shared using FileMaker Network sharing set to Multi-User or Multi-User(Hidden)), the Access via FileMaker Network extended privilege is enabled for all privilege sets in the converted file. When FileMaker Network sharing is enabled and the converted file is open, any client with valid account information can open the shared file. (If the Guest account is enabled, then account information would not be needed to open the file.) You may want to disable the Access via FileMaker Network extended privilege and the Manage extended privileges option for certain privilege sets. In addition, if the file was set to Multi-User(Hidden) the FileMaker Network Sharing option Don't display in Open Remote File dialog is set in the file.
- All the other default extended privilege settings (Access via Instant Web Publishing, Access via ODBC/JDBC, and Access via FileMaker Mobile) are always disabled in the converted file, even if the file you're converting is configured for sharing on the web, via ODBC, or with FileMaker Mobile. You'll need to enable these extended privileges in each appropriate privilege set.
- For FileMaker Developer, stand-alone runtime solutions created with a previous version of the Developer Tool can be converted into FileMaker Pro 7 files. (The converted files will no longer be bound as a runtime solution, so all users will need to have FileMaker Pro 7 licenses after conversion.) If the Developer Tool was set to create the runtime solution with the Permanently prevent modification of database structure option enabled, then converted files won't include an account assigned the Full Access privilege set, and it won't be possible to make modifications that require Full Access privileges.

Example conversion of passwords and groups to accounts and privilege sets

The following table shows the password and group settings in an example FileMaker Pro database. There are nine passwords and two groups. Each password is assigned one group, multiple groups, or no groups. Every password has unique settings except for "User 2a" (identical to "User 2") and "User 3a" (identical to "User 3").

Password	Group(s) assigned to each password
dbadmin (master password)	This password can access the entire file
User1	Group A
User 2	Group B
User 2a	Group B

Password	Group(s) assigned to each password
User 3	Group A Group B
User 3a	Group A Group B
User 4	Group A Group B
User 5	No group assigned
User 6	No group assigned

After conversion to FileMaker Pro 7, the database contains ten accounts and eleven privilege sets:

Account	Assigned privilege set	Comments
dbadmin	Full Access	Every master password in the previous file converts to an account that is assigned the Full Access privilege set.
User1	Group A	
User 2	Group B	Because the “User 2” and “User 2a” passwords have identical settings and are assigned the same group, the converted file creates and assigns the same privilege set to them.
User 2a	Group B	
User 3	Group A/Group B	Because the “User 3” and “User 3a” passwords have identical settings and are assigned the same groups, the converted file creates and assigns the same privilege set to them.
User 3a	Group A/Group B	
User 4	Group A/Group B 2	Even though the “User 4” password is assigned the same groups as “User 3” and “User 3a,” the converted file contains another privilege set that includes the unique settings of the “User 4” password.
User 5	Privilege Set	Because no groups are assigned to the “User 5” and “User 6” passwords and these passwords have different settings, the converted file contains two generic privilege sets based on the unique settings in each password.
User 6	Privilege Set 2	
	Data Entry Only	Three default privilege sets are added to every converted file: Full Access (dbadmin account), Data Entry Only, and Read-Only Access (Guest).
Guest	Read-Only Access	

Sharing a database stored on a CD-ROM

In previous versions of FileMaker Pro, a single user database stored on a CD-ROM could be opened and hosted as a multi-user database. To share a similar database in the current version of FileMaker Pro, you must enable sharing before copying a file to a CD-ROM.

File references

FileMaker Pro uses *file references* to locate database information such as external files, tables, scripts, or value lists. File references were not visible in previous versions of FileMaker Pro, but in FileMaker Pro 7 you have greater control over them.

The following table describes how existing file references are converted:

If the source file and target of the file reference are located	Before conversion, the file reference stores	After conversion, the file reference stores
On the same drive (Windows), volume (Mac OS), or FileMaker host and Save relative path only was specified for the file reference	The relative path	The relative path **Example** file:invoices
On the same drive (Windows) or volume (Mac OS) and Save relative path only wasn't specified for the file reference	Both the relative path and the full path	Both a relative and full path **Example** file:/C:/databases/sales/current_sales; file:../sales/current_sales
On a different drive (Windows) or volume (Mac OS) regardless of the Save relative path only setting	The full path	The full path **Example** file:/D:/databases/sales/current_sales; file:../sales/current_sales
On a FileMaker host and the Save relative path only wasn't specified for the file reference	The network path	The network path **Example** fmnet: /10.0.0.7/current _sales; file:../ sales/current_sales

In previous versions of FileMaker Pro, the Save relative path only checkbox is available:

- in all dialog boxes that have a Host button, when the file is being added to a database solution
- when you specify a file from the Import Records, Open, Close, or Recover script step
- when you specify a value list from another file or a relationship for a file

Note A converted file reference might point to the same file in multiple ways (for example, different locations in which the file existed at different times before conversion).

File reference naming

Because file reference names must be unique, if multiple references to the same file are encountered during conversion, numbers are appended to file reference names (for example, the file references named fileA, fileA2, fileA3, could all refer to one file (<*path*>/fileA)).

Duplicate file references

Some duplicate file references are deleted automatically during conversion and others are not. For example, file references indicating relationships, value lists that reference external files, and certain script steps (such as the Import Records and Export Records script steps) convert to *named file references*. Named file references, which appear in the Define File References dialog box, are duplicated during conversion.

If you decide to delete duplicates after conversion, do so with care. A Database Design Report created using FileMaker Developer is helpful in locating and verifying file references before you delete them. For more information about working with file references, see "Defining file references" in the FileMaker Pro 7 Help.

Poor performance when opening files

If it takes a long time to open a converted file, it's possible that file references are pointing to external locations. You can increase the speed with which files open by resetting file references to local paths, if possible, or enter the IP Address or domain name of the computer on which the remote file resides.

Mac OS: FileMaker Pro 7 no longer stores an Alias Record with a file reference, and therefore some alias-related features (such as automatic searching for moved or renamed files) are no longer supported. The resulting behavior is more consistent with user expectation when working in databases.

Wild card characters in converted file references

File references that contain an asterisk (*) wild card are converted with the asterisk in place, which might reduce the speed with which FileMaker Pro can resolve network file paths.

The format of the converted path is dependent on the way it was originally specified. For example, if the file you are converting is shared through FileMaker Networking or is hosted by FileMaker Server and a network file path is specified in a script or value list, the format of the converted file reference will change according to how the path was originally specified.

- If you selected the file from the file list in the Open File dialog box after specifying the IP address of the host computer, the path is in the following format:

  ```
  fmnet:/<IP_address>/SalesReports
  ```

  File references in this format can be left as is.

- If you selected the file from the file list in the Open File dialog box without specifying the IP address of the host computer, the path is in the following format:

  ```
  fmnet:/*/SalesReports
  ```

  If you find file references in this format after conversion, you should replace asterisks with the domain name (for example, `fmserver1.mycompany.com`) or IP address of the host computer. If the referenced file is located on the same server as the referencing file, it's best to change the file reference to a relative path. For example:

  ```
  file:SalesReports
  ```

When you use a relative path for files located on the same computer and later move the database to another computer, you won't have to update file references.

Document preferences

During conversion, document preferences are converted to file options. For more information, see "Setting file options" in FileMaker Pro 7 Help.

The following changes have been made to document preference options that existed in previous versions of FileMaker Pro:

- The Store compatible graphics preference is no longer available. FileMaker Pro 7 copies each graphic in its original format and, if necessary, automatically creates an additional copy of the graphic in a cross-platform compatible format, which is stored alongside the original graphic. Depending on the number of graphics stored in your database, duplicate graphic storage could increase file size, but might not.
- The Spelling Dialog location preference is no longer available because the spelling checker in FileMaker Pro 7 retains window location information during a session.
- Mac OS: If you use the Mac OS version of FileMaker Pro 7 to convert a file that was created with a previous version of FileMaker Pro for Windows, the Flash menu bar spelling option is converted to Beep when spelling is questioned.
- The Try default password document preference is converted into a Login using file option that uses the old password as both the account name and password in FileMaker Pro 7. This process makes this password visible in the File Options dialog, which could be a security risk. You should change both the account name and password to fully secure the database. For more information, see “Passwords” on page 29.

Hidden files

In previous versions of FileMaker Pro, a file that had an underscore in its name just before the file extension (for example, mydatabase_.fp3) was hidden in the list of hosted files in the Hosts dialog box. This behavior is no longer supported in FileMaker Pro 7. To hide a converted file, you must manually select Don’t display in Open Remote File dialog in the FileMaker Network Settings dialog box.

Note If the original file was set to Multi-User(Hidden), the option Don’t Display in the Open Remote File dialog is selected in the converted file. See “Groups” on page 30.

Preserving user dictionary information

During conversion, the reference to a user dictionary is handled differently depending on whether you are using the default user dictionary or a custom user dictionary.

- **Default user dictionary:** When you convert a database file that uses the default user dictionary (user.upr), the converted file points to the user.upr file in the new FileMaker Pro 7 folder. Therefore, you should copy the user.upr file from the previous FileMaker Pro application folder to the FileMaker Pro 7 folder, or export information from the user.upr file to the FileMaker Pro 7 user.upr file before using the converted database. You can copy the file or export its contents before or after conversion. For more information, see “Editing user dictionaries” in the FileMaker Pro 7 Help.
- **Custom dictionary:** When you convert a database file that references a custom dictionary, the converted file continues to look for the dictionary in its previous location. If you move or delete the dictionary from that location, the converted file will no longer have access to information in the custom dictionary. This could happen accidentally if the original custom dictionary is located in the previous FileMaker Pro folder during conversion and you later uninstall the previous version of FileMaker Pro.

To preserve dictionary information, move the existing custom user dictionary to the FileMaker Pro 7 folder either before or after conversion. After conversion, open the file and specify that it use your custom dictionary. For more information, see “Creating and selecting spelling dictionaries” in the FileMaker Pro 7 Help.

Note If you have added words to your user dictionary in FileMaker Pro 3.x or earlier or in a Claris product, you can add those words to your FileMaker Pro user dictionary. First export the words from the previous user dictionary to a text file, then import the text file into the FileMaker Pro user dictionary.

Changes in layouts

Modified text styles and font display on layouts

The following issues could cause font display problems in converted files:

- If you used small point-sized fonts or thin, Serif-based fonts in layouts, they might look different after conversion due to the anti-aliasing technology now used in FileMaker Pro 7. Anti-aliasing generally improves the appearance of fonts, but some might appear blurry.
- FileMaker Pro 7 might have trouble converting fonts that were originally defined in a version of FileMaker Pro previous to version 1.x. If the original font can’t be found, the font that was used when the file was converted to FileMaker Pro 1.x or later is used.
- Windows: If your file uses non-TrueType fonts they will be substituted with TrueType fonts during conversion. When creating new files in FileMaker Pro 7, always use TrueType fonts, which are supported on both Windows and the Mac OS and are therefore cross-platform.

  Note Font substitution may still occur if a font specified on one platform is unavailable on the other.

  Tip Third-party utilities are available for converting fonts from other formats to TrueType format.
- Mac OS: Outline and Shadow text styles change to Plain text style during conversion.

Changes to line display

FileMaker Pro 7 improves the smoothness of objects on layouts, such as diagonal lines, curves, and rotated objects. However, straight vertical and horizontal lines on layouts might appear slightly thicker than they did in original files and colored lines might look slightly faded.

Changes in the behavior of fields in header and footer parts

FileMaker Pro 7 allows data entry into fields in headers and footers. If a field has previously been defined as a button, this change can have the unexpected consequence of placing the user’s cursor within the field, rather than activating the button. After file conversion, you can recreate the behavior of the previous version of FileMaker Pro by explicitly disabling data entry for any fields in headers/footers for which data entry should not be allowed. (In Layout mode, select the field and choose Format menu > Field Behavior, and set the appropriate options for Allow field to be entered.)

Converted relationships

FileMaker Pro 7 allows multiple tables per file, so relationships can now exist between tables in a single file. However, when you convert relational databases composed of several files to FileMaker Pro 7, the separate files are not combined into a single file with multiple tables.

Relationships

If you named relationships in a file created with a previous version of FileMaker Pro, after conversion the relationship name is the name given to the table occurrence in the relationships graph. The characteristics of the relationship (settings allowing records to be created or deleted and sorting requirements for the relationship) are the characteristics defined for the converted relationship. The related file might not appear in the relationships graph by name and won't appear in the Tables tab in the Define Databases dialog box. To display the name of the source table and the name of its related file, move the pointer over the arrow to the left of the table name in a table occurrence.

Note When you first display the relationships graph after file conversion, the root table will be centered at the top of the graph. Tables that are related to the root table will be drawn in groups, one key field at a time. Within a group, the related tables will be alphabetized from top to bottom, A to Z.

Self-joins

When related files that contain self-joins are converted, duplicate table occurrences are created in the relationships graph for each self-join defined. Names of duplicate tables (fully qualified names) are based on the name of the relationship in the original file.

Portal behavior

In FileMaker Pro 7, portals no longer scroll back to the first row when you exit the portal. Instead, the status and position of the portal is retained as long as the user remains in a record.

Change to how word breaks are determined

The way in which FileMaker Pro 7 determines word breaks when creating indexes, performing find requests, and evaluating word-based text functions (such as LeftWords, MiddleWords, and so on) has changed. For example, the hyphen is now considered a word break, so "father-in-law" is considered three words. Therefore, results of features that depend on word-break logic (such as Find requests and some calculations) might differ slightly from those in previous versions of FileMaker Pro.

Two-digit date conversion

If your database contains pre-1900 or post-1999 date data in two-digit year format, you should change it to four-digit format before conversion. Two-digit year dates from 1900 to 1999 will convert correctly.

For more information about date conversion, see "Conversion of dates with two-digit years" in the FileMaker Pro 7 Help. If you need details on how previous versions of FileMaker Pro handle two-digit year dates, see the "Year 2000 Readiness Disclosure" document, available at www.filemaker.com.

Changes to sorting behavior

The following changes can affect the order in which records sort in converted files.

ASCII changed to Unicode

During conversion, the ASCII indexing and sorting language choice is changed to Unicode. For more information, see “Choosing a language for indexing or sorting” in the FileMaker Pro 7 Help.

Accented characters

In previous versions of FileMaker Pro, accented and non-accented characters (such as Ç and C) were sorted based on record creation order; accent marks were not considered. In FileMaker Pro 7, sorting is based on the Unicode Collation Algorithm, which does consider accent marks. For more information, see “Choosing a language for indexing or sorting” in the FileMaker Pro 7 Help.

After conversion, fields that contain accented characters may sort differently.

Sorted data and found sets

If the file you are converting was sorted or contained a found set, this information is not retained. After conversion, data appears in the file in creation order, which can cause existing summary reports to be incorrect. To display the pre-conversion sort order, sort or perform a find request to narrow the data that is considered in summary reports.

Sorting records by subsummary values

Previous versions of FileMaker Pro allowed sorting by only one summary field based on the subsummary parts defined in a layout. In FileMaker Pro 7, you can specify a summary field as part of the sort criteria and reorder the groups by sorting on a non-summary field. For more information, see “Sorting records by subsummary values” in the FileMaker Pro 7 Help.

In previous versions of FileMaker Pro, the summary sort order was dependent on the state of the window at the time the sort was performed. FileMaker Pro 7 can’t always convert summary sort reports accurately, because it can now display multiple windows per file. During conversion, FileMaker Pro 7 applies the summary sort to the last entry in the list of non-summary fields from the previous sort criteria. For example, suppose your sort criteria was defined as follows:

Field name	Sort option setting
Salesperson	Ascending
Region	Descending
Total Sales (a summary field)	Ascending, with Include summary fields selected

When you convert this file, the sort order converts to:

Field name	Sort option setting
Salesperson	Ascending
Region	Ascending, reordered by Total Sales (a summary field)

In this case, the resulting sort order change is based on the setting of the summary field "Total Sales" in the original sort definition.

In this example, the sort order of a report (Report A) in the converted file has a subsummary part based on the state of the layout currently being viewed. However, if a layout also exists that only summarizes by Salesperson but uses the same sort criteria that was used in Report A, a second report (Report B) will not sort the same as it did before conversion. To achieve the sort order of Report A, you must create different sort criteria for Report B that reorders Salesperson data by the summary field Total Sales rather than by the Region field.

Indexes

If you are converting a large database file, you might consider skipping index conversion as it can take a long time. (During conversion, the Cancel button changes to Skip when index conversion begins.) After conversion, you can:

- create a new index in the converted file
- use the automatic indexing option, which indexes fields when a user performs a search on a field with this setting or the field is used as a match field in a relationship

If you choose to convert an existing index, the type of index created in the converted file is dependent on the type of field indexed and the indexing settings in the Storage tab of the Options for Fields dialog box in the original file.

	Index settings in the original file		**Index settings after conversion to FileMaker Pro 7**		
Type of field	**"Indexing" setting is:**	**"Automatically turn indexing on if needed" setting is:**	**Indexes created during conversion**	**"Indexing" setting is**	**"Automatically create indexes as needed" setting is:**
All fields	Off	Off	None	None	Off
	Off	On	None	None	On
Non-text fields	On	Off	Value	All	Off
	On	On	Value	All	On
Non-Japanese text fields	On	Off	Value and word	All	Off
	On	On	Value	Minimal	On
Japanese text fields	On	Off	Value, word, and additional special-use indexes	All	Off
	On	On	Value	Minimal	On

For information about different types of indexes and usage tips, see "Defining field indexing options" in the FileMaker Pro 7 Help.

If you convert your index, you'll notice the following changes:

- Previous versions of FileMaker Pro formatted index entries with initial capital letters. (cat entered in a text field displayed as Cat when you viewed the index.) Index entries in FileMaker Pro 7 appear in the case of the first indexed record encountered. For example, if the first instance of "cat" encountered when the index is created is lowercase, then "cat" appears in lowercase in the index.
- In Roman language indexes, previous versions of FileMaker Pro didn't index non-alphanumeric characters but FileMaker Pro 7 does.
- During conversion, the following field indexing language settings are changed:

Language setting before conversion	Language setting after conversion
Spanish (New Style)	Spanish (Modern)
Czech/Slovak	Czech
ASCII	Unicode

Exporting in the Lotus character set

FileMaker Pro 7 does not allow you to export data in the Lotus International Character Set (LICS). During conversion, any scripts or export orders that specify LICS are changed to Windows (Windows) or Macintosh (Mac OS). If the Windows or Mac OS character sets are not valid for the export format specified, data is exported in the file's default character set.

Printer settings (Windows)

Printer settings, such as a non-default paper size specification, might be lost during conversion. When printer settings are lost, default settings are used in converted files until you redefine settings.

To ensure proper printing of converted files, record printer settings in your original files before conversion by writing settings down or taking a screen shot of various options in the Print dialog box. (To take a screen shot, display printer settings in the Print Setup or Print dialog box, press Alt+PrintScreen, then paste the screen shot into a document for future reference. Repeat for each tab in the Print Setup or Print dialog box.) Then verify printer settings in the converted file and make corrections, if necessary.

Important You must commit a record in the database to save changes printer settings. Simply opening the database, changing printer settings, and closing the database will not save settings. To save printer settings, change settings in the Print Setup or Print dialog box, close the dialog box, and make a change to a field or add a new record before closing the file.

Print Setup script step stored setup options (Windows)

The Print Setup script step is able to store and use custom paper options that can be recalled when you run a script. This allows you to use a script to print a particular layout with different paper options — for example, to print labels or print on a pre-printed form.

FileMaker Pro 7 uses a new format to create and store options in the Print Setup script step. The new format is not compatible with the format used in previous versions of FileMaker Pro, and FileMaker Pro 7 cannot completely convert stored setup options in previous files to the new format.

However, in files that you convert from previous versions, FileMaker Pro 7 initially maintains any stored setup options in the previous format. Depending on how the Print Setup script step is configured, the script may continue to work properly.

But if you attempt to change the setup options by opening the Print Setup dialog box, FileMaker Pro 7 may not be able to maintain some of the stored settings in the dialog box, and you may have to re-specify the paper options that you want. Common settings like paper size and orientation are usually maintained, but less-common settings like a particular paper tray or duplex setting may not be maintained.

The following table summarizes what may occur when you run or edit scripts that contain Print Setup script steps with stored setup options.

If you	The result is
Run a script that contains this script step: ▪ Print Setup [Restore, No dialog]	The script step usually will work properly because it does not open the Print Setup dialog box.
Run a script that contains this script step: ▪ Print Setup [Restore]	When you run the script, some of the stored setup options may not appear properly in the Print Setup dialog box. To correct the problem, edit the Print Setup script step so that it stores the setup options that you want.
Edit either of the following script steps in the Edit Script dialog box: ▪ Print Setup [Restore, No dialog] ▪ Print Setup [Restore]	When you click Specify to edit the setup options for the script step, some of the stored setup options may not appear in the Print Setup dialog box, and you will have to re-specify the options. To correct the problem, check the options carefully as you are editing the Print Setup script step. Make sure the script step is storing the setup options that you want.

In essence, if you want to edit stored setup options for the Print Setup script step, you may need to remember what the settings were so that you can enter them again properly. You may want to view the options in the previous, unconverted file so that you can enter them correctly.

Global fields

The global field type no longer exists in FileMaker Pro 7 because "global" is now a storage option. During conversion, any global fields in a file are converted into the field type of the data they contain and the global storage option is selected. For more information about the global storage option for field definitions, see "Defining global fields (fields with global storage)" in the FileMaker Pro 7 Help.

Number fields

FileMaker Pro 7 does not recognize previously supported alphabetic characters in a number field as numbers. If you have used Y or T to mean "Yes" or "True" or "N" or "F" to mean "No" or "False" in number fields, Boolean calculations relying on these characters will not return results consistent with previous versions of FileMaker Pro.

To preserve the behavior from previous versions of FileMaker Pro, you can use the following calculation to evaluate alphabetic characters in Boolean calculations:

```
Let (char1 = Left(NumberField;1); Case (char1 = "Y" or char1 = "T"; 1;
char1 = "N" or char1 = "F"; 0; NumberField))
```

Note You can use Edit menu > Find/Replace in FileMaker Pro 7 to update the contents of a field replacing "Y" with "1" and "F" with "0," and so on.

Date and time fields

Because Japanese emperor year dates are now supported, FileMaker Pro 7 no longer allows alphabetic characters, the time separator : (colon), or the + (plus sign) to be entered as a date separator. During conversion, existing dates that were entered using alphabetic characters or the time separator remain unchanged. For example, 3:21:04 is now considered to be time data because the colon is a time separator. Dates that use obsolete separators won't work properly in converted databases.

Any calculation that uses dates with obsolete separators will not evaluate properly when it is recalculated by FileMaker Pro 7. However, results of calculations evaluated and stored before conversion will display correctly after conversion.

Data that contains obsolete date or time separators is indexed as invalid and can be found by specifying `?` in Find requests.

To avoid these issues, use one of the following methods to manually edit Date data that uses alphabetic characters or the time separator:

- Edit menu > Find/Replace
- Substitute function. For example:

Before conversion, use: `Substitute(DateField,″:″,″/″)`

After conversion, use: `Substitute(DateField;″:″;″/″)`

Important To avoid introducing errors in data, correct date separators only in fields and in records that contain unsupported separators rather than making global changes to date fields.

For more information, see the topics on finding and replacing data or the Substitute function in FileMaker Pro Help. For more information on upgrade, conversion and migration considerations, see www.filemaker.com for TechInfo Knowledge Base articles and other resources.

Container fields

In the previous version of FileMaker Pro, both Windows Metafile and PICT images could be stored in a single container field if the option Store compatible graphics was selected in the database file's document preferences.

When files with these images are converted to FileMaker Pro 7 on Windows computers that do not have QuickTime installed, only the Windows Metafile is preserved.

When these files are converted on a computer running Mac OS, only the PICT image is preserved. You do not lose any Windows Metafile data, but you cannot immediately drag, drop, copy, or paste the images to another container field on a computer running Mac OS: you must first open the database on a Windows computer and then view each record with a container field to convert the images.

To preserve both types of images in the same file, perform the conversion on a Windows computer that has QuickTime installed.

Calculation fields

During conversion, all stored results from existing calculations are converted and retain integrity. However, due to the many changes in the way calculations are evaluated in FileMaker Pro 7, it's possible that some existing calculations could produce different results if they are recalculated after conversion.

The following sections describe changes during conversion that can affect calculation fields.

Functions that are renamed in FileMaker Pro 7

Text functions

The following text function names have changed. For more information about how to use these functions, see "Text functions" in the FileMaker Pro 7 Help.

Function name in previous versions of FileMaker Pro	New function name
TextToDate	GetAsDate
TextToNum	GetAsNumber
TextToTime	GetAsTime
NumToText	GetAsText
DateToText	GetAsText
TimeToText	GetAsText

Additional circumstances affect the GetAsNumber function. For more information, see "Text functions recognize text characters in number fields" on page 52 and "GetAsNumber function" on page 54.

Design functions

The design functions RelationIDs and RelationNames have been renamed in FileMaker Pro 7. If these functions are used in the file you are converting, the names are changed during conversion, which has no affect on the calculations in which they are used.

Function name in previous versions of FileMaker Pro	Function name in FileMaker Pro 7
RelationIDs(fileName)	TableIDs(fileName)
RelationNames(fileName)	TableNames(fileName)

Note TableIDs(fileName) returns the table IDs for each table in the database. Table IDs are numbered consecutively, beginning with the value 1065089.

Status functions changed to Get functions

The following Status functions have new names in FileMaker Pro 7. For more information about how to use these functions, see "Get functions," in the FileMaker Pro 7 Help.

Function name in previous versions of FileMaker Pro	Function name in FileMaker Pro 7
Status(CurrentAppVersion)	Get(ApplicationVersion)
Status(CurrentDate)	Get(CurrentDate)
Status(CurrentError)	Get(LastError)
Status(CurrentFieldContents)	Get(ActiveFieldContents)
Status(CurrentFieldName)	Get(ActiveFieldName)
Status(CurrentFilePath)	Get(FilePath)
Status(CurrentFileName)	Get(FileName)
Status(CurrentFileSize)	Get(FileSize)
Status(CurrentFoundCount)	Get(FoundCount)
Status(CurrentGroups)	Get(PrivilegeSetName)
Status(CurrentHighContrast)	Get(HighContrastState)
Status(CurrentHighContrastColor)	Get(HighContrastColor)
Status(CurrentHostName)	Get(HostName)
Status(CurrentLanguage)	Get(SystemLanguage)
Status(CurrentLayoutAccess)	Get(LayoutAccess)
Status(CurrentLayoutCount)	Get(LayoutCount)
Status(CurrentLayoutName)	Get(LayoutName)
Status(CurrentLayoutNumber)	Get(LayoutNumber)
Status(CurrentMessageChoice)	Get(LastMessageChoice)
Status(CurrentMode)	Get(WindowMode)
Status(CurrentModifierKeys)	Get(ActiveModifierKeys)
Status(CurrentMultiUserStatus)	Get(MultiUserState)
Status(CurrentNetworkChoice)	Get(NetworkProtocol)
Status(CurrentODBCError)	Get(LastODBCError)
Status(CurrentPageNumber)	Get(PageNumber)
Status(CurrentPlatform)	Get(SystemPlatform)
Status(CurrentPortalRow)	Get(PortalRowNumber)
Status(CurrentPrinterName)	Get(PrinterName)
Status(CurrentRecordAccess)	Get(RecordAccess)
Status(CurrentRecordCount)	Get(TotalRecordCount)
Status(CurrentRecordID)	Get(RecordID)

Function name in previous versions of FileMaker Pro	Function name in FileMaker Pro 7
Status(CurrentRecordModificationCount)	Get(RecordModificationCount)
Status(CurrentRecordNumber)	Get(RecordNumber)
Status(CurrentRepetitionNumber)	Get(ActiveRepetitionNumber)
Status(CurrentRequestCount)	Get(RequestCount)
Status(CurrentScreenDepth)	Get(ScreenDepth)
Status(CurrentScreenHeight)	Get(ScreenHeight)
Status(CurrentScreenWidth)	Get(ScreenWidth)
Status(CurrentScriptName)	Get(ScriptName)
Status(CurrentSortStatus)	Get(SortState)
Status(CurrentStatusArea)	Get(StatusAreaState)
Status(CurrentSystemVersion)	Get(SystemVersion)
Status(CurrentTime)	Get(CurrentTime)
Status(CurrentUserCount)	Get(UserCount)
Status(CurrentUserName)	Get(UserName)
Status(CurrentView)	Get(LayoutViewState)
Status(CurrentWebSharing)	Position (Get(ApplicationVersion); "Web"; 1; 1) > 0 See "Changes to databases that are published to the web" on page 63.

In addition, you'll notice the following behavioral changes between Status functions in previous versions of FileMaker Pro and Get functions in FileMaker Pro 7.

- Get functions only accept valid parameter entries. Previous versions of FileMaker Pro would allow you to enter a calculation as a parameter, for example, but FileMaker Pro 7 does not. For more information, see "Change to rules for allowed parameters in Get functions" on page 47.
- Get (RecordModificationCount) returns continuous numbers and no longer wraps numbers after 32767.
- Get(RecordID) returns contiguous values. In previous versions of FileMaker Pro, Status(CurrentRecordID) did not return values in the following ranges:
  - 128 to 32767
  - 491152 to 12582911
  - 14680062 to 3758096383

  Note During conversion, record IDs from the original file are retained. Record IDs for records that are deleted or exported after conversion are reserved. If you create a new record after deleting or exporting records, the record ID for the new record is the next new ID available, not the ID for the first record in the set that was deleted or exported.

- If a database uses the Status(CurrentGroups) function to check if a user was a member of a specific group, the same check using the new Get(PrivilegeSetName) function may not work as expected in FileMaker Pro 7 because the group name may have been deleted or modified during conversion. Duplicate groups are combined into a single privilege set during conversion. If calculations reference groups that were deleted or whose names were modified, these calculations must be manually modified so they test for the remaining privilege set name. For more information about how groups are affected during conversion, see “Groups” on page 30 and “Example conversion of passwords and groups to accounts and privilege sets” on page 31.
- In FileMaker Pro 7, local paths returned from the Get(FilePath) function begin with a single slash rather than a double slash.

Local path format in previous versions of FileMaker Pro	Local path format in FileMaker Pro 7
file://folder/filename	file:/folder/filename

In FileMaker Pro 7, Get(FilePath) returns full path information (double-slash) for remote files. If your database contains an If statement using Get(FilePath) that looks for a double-slash in a local path, it will not evaluate correctly. These calculations should be edited manually after conversion to reflect the new path syntax.

Results of Status(CurrentFilePath) function calls that are stored in a file before conversion are not changed, but subsequent calls to Get(FilePath) after conversion are returned in the new format.

Japanese functions that are converted

The following Japanese functions are no longer supported and are converted as shown in the following table:

Japanese function in previous versions of FileMaker Pro	Function in FileMaker Pro 7
Leftb	Left
Lengthb	Length
Middleb	Middle
Positionb	Position
Replaceb	Replace
Rightb	Right

You need to evaluate and possibly modify the parameters in converted Japanese functions that refer to a starting position, ending position, or count of bytes as this information depended in previous versions on the type of data (one-byte or two-byte) being evaluated, and these byte differences no longer exist as of FileMaker 7.

Invalid characters and words in field names used in calculations

In FileMaker Pro 7, certain characters and words can't be used as field names. If reserved characters or words are used in field names that are specified in calculations, during conversion the entire field name is enclosed in ${ } characters in the calculation. Only field names used in calculation definitions are enclosed in ${ } characters. This change has no effect on calculation evaluation.

Note Text strings in calculations enclosed in quotation marks are not changed during conversion and are treated as text literals.

Invalid characters and words

- leading or trailing spaces
- + - * / ^ & = ≠ < > ≤ ≥ (, ;) [ ] \" :: characters
- AND, OR, XOR, NOT, TRUE, FALSE reserved words

In addition, field names can't begin with a digit or a period or have the same name as a function, parameter, or keyword.

Reserved words

Reserved words in FileMaker Pro 7 fall into the following categories:

- Text parameters in calculation functions, such as "Cyrillic" in `TextFont("Plaid";"Arial";${Cyrillic})`
- Extended privilege keywords
- Function names, such as "Pi" and "Random"

Note Although SQL keywords are not reserved, avoid using them as field names, especially if those field names will be referenced in calculations.

Parameter separators in calculations

FileMaker Pro 7 no longer uses commas to separate parameters in calculations. During conversion, comma separators are changed to semicolons. If FileMaker Pro 7 is set to a language locale that uses the comma as the list separator character (for example, English), you can type commas as parameter separators when you are defining new calculations. The commas are changed to semicolons when you save a calculation.

Change to rules for allowed parameters in Get functions

In previous versions of FileMaker Pro, calculations could be entered as parameters in Status functions (converted to Get functions in FileMaker Pro 7), even if there was no logic behind the calculation. If calculations were used as parameters in the database you are converting, after conversion the entire calculation containing the function is commented out in the Specify Calculation dialog box. To return results consistent with previous versions of FileMaker Pro, you must redefine the calculation.

FileMaker Pro 7 converts field types as needed before processing calculations

Before processing, data is converted to the data types specified in the following tables. After processing, calculations return the data type selected in the Specify Calculation dialog box. When the calculation result is set to Text, non-textual results, such as dates and times, are set to the data types listed in the following tables.

Previous versions of FileMaker Pro

The following table summarizes the field type of results returned from calculations when the operands are fields of different type:

Any mathematical operator	Text field	Number field	Date field	Time field
Text field	Text	Number	Number	Number
Number field	Number	Number	Number	Number
Date field	Date	Date	Date	Date
Time field	Time	Time	Time	Time

FileMaker Pro 7: Arithmetic calculations

+/-	Text field	Number field	Date field (days since 1/1/0001)	Time field (seconds since midnight)	Timestamp field (seconds since 1/1/0001)	Container field
Text field	Number	Number	Date[1, 2]	Time[1, 2]	Timestamp	Number[4]
Number field	Number	Number	Date[1, 2]	Time[1, 2]	Timestamp	Number[4]
Date field (days since 1/1/0001)	Date	Date	Number[1, 5]	Timestamp[6]	Time[1, 3, 6]	Number[4]
Time field (seconds since midnight)	Time	Time	Timestamp[6]	Time	Timestamp	Number[4]
Timestamp field (seconds since 1/1/0001)	Timestamp	Timestamp	Time[1, 3, 6]	Timestamp	Time[1, 3]	Number[4]
Container field	Number[4]	Number[4]	Number[4]	Number[4]	Number[4]	Number[4]

Notes:

[1] The field type returned has changed from previous versions of FileMaker Pro. For example, in FileMaker Pro 7, Date(3, 21,2003)+2 returns 3/23/2003 in date format. In previous versions of FileMaker Pro, the same calculation returns 731297, which is the number of days since 1/1/0001.

[2] FileMaker Pro 7 returns the numeric representation of dates and times.

[3] Values are retained as time.

[4] A container value converted to a number value always returns 0 (zero). If you want to determine the size of an object stored in a container field, include the Length function in the calculation.

[5] Data is returned in number format to indicate number of days added or subtracted.

[6] Dates are converted to seconds since 1/1/0001 when used with time and timestamp data.

FileMaker Pro 7: Comparison calculations

=, < >, <, >, <=, >=	Text field	Number field	Date field (days since 1/1/0001)	Time field (seconds since midnight)	Timestamp field (seconds since 1/1/0001)	Container field
Text field	Text	Text[6]	Text[6]	Text[6]	Text[6]	Text[3, 6]
Number field	Text[6]	Number	Number	Number	Number	Number[4]
Date field (days since 1/1/0001)	Text[6]	Number	Number	Number[1]	Number[1]	Number[4]
Time field (seconds since midnight)	Text[6]	Number	Number[1]	Number	Invalid[5]	Number[4]
Timestamp field (seconds since 1/1/0001)	Text[6]	Number	Number[1]	Invalid[5]	Number	Number[4]
Container field	Text[3, 6]	Number[4]	Number[4]	Number[4]	Number[4]	Container[2]

Notes:

[1] When comparing a date value with a time or timestamp value, the date value is converted to seconds since 1/1/0001, the time value is converted to seconds since midnight and the timestamp value is converted to seconds since 1/1/0001 before the comparison is made. In other comparisons involving date, time, and timestamp values, the date value is converted to days since 1/1/0001, the time value is converted to seconds since midnight, and the timestamp value is converted to seconds since 1/1/0001before the comparison is made.

[2] For = and < > comparisons, the Cyclic Redundancy Code (CRC) of the two containers are compared. For other comparisons, the total size of the item stored in the container objects is compared. This behavior differs from previous versions of FileMaker Pro.

[3] When a container object is compared with a text value, the container object is converted to text format. When the container object is imported as a reference, the external path information of the container is used in the comparison. Otherwise, the container is converted to a "?" (for security reasons), which is used in the comparison.

[4] A container value converted to a number value always returns 0 (zero). If you want to determine the size of an object stored in a container field, include the Length function in the calculation.

[5] When a time value is compared to a timestamp value, FileMaker Pro 7 returns Invalid, because it isn't useful to compare an elapsed time quantity such as a time with the moment-in-time quantity such as a timestamp.

[6] When one field type in a comparison operation is text and the other isn't, values are compared in text format because it allows FileMaker Pro to compare as much data as possible.

Automatic conversion of date, time, and timestamp field types

FileMaker Pro 7 automatically converts data in date, time, and timestamp fields when they are used together in a calculation so that durations can be determined. This allows you to return the number of days, for example, between a date and a timestamp.

The following table summarizes how FileMaker Pro 7 automatically fills in data when field types are converted to and from date, time, and timestamp formats when a calculation includes the functions noted in the table.

Format of data provided	Hour function Minute function Second function	Day function Month function Year function	GetAsTimestamp function
Time	Auto-fill not needed	invalid	Auto-fill 1/1/0001
Data	Auto-fill midnight	Auto-fill not needed	Auto-fill midnight
Timestamp	Auto-fill not needed	Auto-fill not needed	Auto-fill not needed

Note When a value is automatically filled in, time is the number of seconds since midnight, a date is the number of days since 1/1/0001, and a timestamp is the number of seconds since 1/1/0001.

Operator precedence in calculations

In previous versions of FileMaker Pro the +, - (minus), and & operators all had the same level of precedence when calculations or expressions were evaluated. In FileMaker Pro 7, the + and - operators have higher precedence than the & operator. During conversion, parentheses are added to existing calculations or expressions so that the evaluation order from previous versions of FileMaker Pro is retained.

Decimal precision in validated fields

Field and validation values must be identical to successfully validate the contents of a field. If the Validated by calculation option is used, the calculation results must exactly match the contents of the field being evaluated. For example, if a field contains .3333333333333333 and is set to validate by the calculation "`field = 1/3`," the data won't pass validation in FileMaker Pro 7.

If you are validating fields by calculation that return fractional results, use the Round function to limit calculation results to an expected number of digits.

Accelerated calculation evaluation

To enhance performance, FileMaker Pro 7 now evaluates only the portion of a calculation required to determine a result. Functions that take advantage of accelerated evaluation are If, Case, and Choose. Operators that use accelerated evaluation are AND and OR.

Example 1

```
If(2+2=4; "true"; "false")
```

In previous versions of FileMaker Pro, both result1 and result2 are evaluated. If external plug-in functions calls are included in the "true" and the "false" conditions, evaluation of the results of both calls are part of the evaluation of the full calculation statement (in other words, both calls to the plug-in function are performed whether or not the test evaluates to be true or false).

In FileMaker Pro 7, such calls to the external plug-in functions are executed based on the results of the test; so in this example the “true” condition results in the call to the plug-in function, and the “false” condition is skipped. This can result in unexpected behavior when the plug-in function is not executed.

Accelerated evaluation will save time when calculations operate on very large datasets, especially if either or both clauses deal with related fields. However, it could produce errors in files created with previous versions of FileMaker Pro if the database design expects a plug-in function to be called as the result of the evaluation of the entire calculation.

Example 2

Consider the following example with calls to two different external functions:

```
If (External("Func1") and External("Func2"); "true"; "false")
```

In FileMaker Pro 7, if Func1 returns false then Func2 is not called. The second call is made only when the first condition evaluates true. In previous versions of FileMaker Pro, Func2 would be called regardless of the result of Func1.

New rounding rules

Previous versions of FileMaker Pro used a floating-point method when rounding numbers, which yielded results up to approximately 15 digits. FileMaker Pro 7 uses a fixed-point method, which allows up to 400 digits each to the left and right of the decimal point. The following examples compare sample results from different versions of FileMaker Pro:

Expression	Result in previous versions of FileMaker Pro	Result in FileMaker Pro 7
126.96 - 119.29	7.66999999999999	7.67
4.1 - Int(4.1)	.0999999999999996	.1
408.80 - 400.20	8.60000000000002	8.6
10000000000000000 + 10	10000000000000000	10000000000000010

New syntax for nested substitutions

The syntax for nesting Substitute functions in previous versions of FileMaker Pro was:

```
Substitute(Substitute(Substitute(…)))
```

FileMaker Pro 7 introduces the following new syntax, which evaluates faster and is easier to read:

```
Substitute(TextField; [“a”;”x”];[“b”;”y”];[“c”;”z”])
```

During conversion, nested Substitute function calls are changed to the new syntax, when possible. However, when nested substitutes can’t be converted and syntax is left unchanged, they will continue to evaluate properly in FileMaker Pro 7.

Today function no longer supported

The Today function has been removed in FileMaker Pro 7. During conversion, files that contain the Today function will be changed as follows:

- If Today is used in an unstored calculation field or script, it is changed to Get(CurrentDate).

- If Today is used in a stored calculation, a new Date field named "Today" is created and the stored calculation will reference it. The Today field has modified field options so that the creation date is automatically entered in it. To achieve the same behavior that was provided by the Today function in previous versions of FileMaker Pro, a script is created during conversion named "Update Today Field," which will become part of the start-up script in the file and run when the file is opened.

Text functions recognize text characters in number fields

In previous versions of FileMaker Pro, text functions ignored alphabetic data in number fields, so you couldn't define a calculation that could access textual data in a numeric field. In FileMaker Pro 7, alphabetic data is recognized, so results of new calculations defined after conversion might differ from what you expect.

Example 1

Suppose you have a number field that contains 123aV67. The function Middle(NumberField, 1, 6) returns:

Previous versions of FileMaker Pro	FileMaker Pro 7
12367	123aV6

During conversion, functions are modified to return the same result that a previous version of FileMaker Pro would provide. In this example, the function call above would convert to Middle(GetAsNumber(NumberField); 1; 6). The difference in evaluation between versions is pointed out here so you know to make adjustments to calculations you create in FileMaker Pro 7.

Example 2

Suppose a number fields contains "1 apple costs 20 cents." In this case, the function MiddleWords(NumberField, 1, 3) returns:

Previous versions of FileMaker Pro	FileMaker Pro 7
120	1 apple costs

Functions that are dependent on context

In previous versions of FileMaker Pro, the *context* (the starting point or perspective from which calculations and scripts are begun) in which a calculation or script evaluated was static; only one window could be open at a time, so the current context was always the open window.

In FileMaker Pro 7, which supports multiple open windows, calculations and scripts are dependent on the context in which they evaluate. If a fixed context is assigned to a calculation in the Specify Calculation dialog box or script definition, the calculation returns results based on that context. If the same calculation is used in a script, as a script parameter, as a button parameter, in a Replace By calculation, and so on, calculation results are dependent on the context of the current window.

This change in behavior could cause errors when existing calculations or scripts are converted because they might execute in the wrong context (for example, the found set in an incorrect window). Therefore, you might have to edit scripts and calculations to specify the window in which they execute. Note that the Select Window, New Window, and Close window script steps affect the context of execution for both scripts and calculations.

Window context could affect calculations in converted databases that call the following Get functions:

Get(RecordID)
Get(RecordModificationCount)
Get(RecordAccess)
Get(RecordNumber)
Get(FoundCount)
Get(UserCount)
Get(TotalRecordCount)
Get(SortState)
Get(AccountName)
Get(ExtendedPrivileges)
Get(FilePath)
Get(FileName)
Get(FileSize)
Get(HostName)
Get(MultiUserState)
Get(PrivilegeSetName)

Calculations that are fixed in FileMaker Pro 7 and unaffected by context include calculation fields, auto-entry by calculations, field validation by calculations, and custom table privileges (Record Level Access by calculation).

For more information about context and calculation evaluation, see “Defining calculation fields” in the FileMaker Pro 7 Help.

Functions that return filename extensions

The following functions no longer return filename extensions:

- DatabaseNames
- Get(FileName)

Results from converted records are not changed (filename extensions are displayed), but filename extensions will not appear in any new records created after conversion.

Tip After conversion, you can change Get(FileName) calls in FileMaker Pro 7 as follows to make filename extensions appear in results.

```
Let(possibleExtension = right(get(FilePath), 4), Get(FileName) &
if(left(possibleExtension, 1) = “.”, possibleExtension, “”))
```

Changes to the Mod function

In FileMaker Pro 7, the definition of the Mod function has changed. The new definition is:

```
Mod(x; 0) = x; Mod(x; y) = x - (y * Floor(x/y))
```

This definition gives the same results when both inputs are positive or both are negative.

Use the following syntax to achieve Mod results consistent with previous versions of FileMaker Pro:

```
Case( y = 0; 1/0;
         (x < 0) = (y < 0); Mod(x;y);
         (x < 0); - Mod(Abs(x); y);
         (y < 0); Mod(x; Abs(y)))
```

When the second parameter is positive, this reduces to:

```
if (x < 0; - Mod(Abs(x); y); Mod(x;y))
```

Numeric expressions with text input

In previous versions of FileMaker Pro, when you specified textual parameters to functions that expected numeric parameters, the result returned was blank. In FileMaker Pro 7, the result is 0 (zero). During conversion, previously stored results remain blank, but any new results calculated after conversion are 0. You may need to modify the calculation to get results consistent with the previous version of FileMaker Pro.

Boolean calculations

In FileMaker Pro 7, the following changes have been made to the way Boolean calculations evaluate:

- The logical operators AND, OR, XOR, and NOT return only Boolean results.
- Boolean calculations return False when results are blank or 0 (zero), or do not resolve into a number. All other results return True.
- Boolean calculations that expect "T," "Y," "F," or "N" to evaluate to 1 or 0 (zero), true or false will not evaluate properly. For more information, see "Number fields" on page 41.

GetAsNumber function

In FileMaker Pro 7, if you reference a number, date, or time field as the parameter of a text function, the full contents of the field are returned, including any text in the field. Therefore, in FileMaker Pro 7 you need to use the GetAsNumber function in the calculation specification to change the field type of the parameter to number field type.

Example 1

Syntax from previous versions of FileMaker Pro	Syntax after conversion to FileMaker Pro 7
Left(NumberField,6)	Left(GetAsNumber(NumberField); 6)

During conversion, the GetAsNumber function is added automatically to the following functions when they contain a number, date, time, or timestamp field as the parameter:

Left
Middle
Right
LeftWords
RightWords

MiddleWords
Exact
Length
Lower
Upper

Proper
PatternCount
Position
Replace
Substitute

Trim
WordCount
TrimAll
NumToJText

The GetAsNumber function is applied to the first parameter of these functions except the Exact, Replace, and Substitute functions in which it is applied to multiple parameters.

Example 2

Syntax from previous versions of FileMaker Pro	Syntax after conversion to FileMaker Pro 7
Exact (numberField1, numberField2)	Exact (GetAsNumber(numberField1); GetAsNumber(numberField2))

Note If additional parameters are not number fields, the GetAsNumber function is not applied to the Replace or Substitute functions.

If comparison operators (<, =, >, ≤, ≥, < >) are used in a calculation that requires the GetAsNumber function, GetAsNumber is always added to the first and second parameters during conversion.

Get(LastError) function

In FileMaker Pro 7, the following error codes have been reassigned to a different error condition or made obsolete. If the file you are converting uses the Get(LastError) function to capture any of these codes, you might get unexpected results after conversion.

Error codes that have been redefined

Error code	Definition in previous version of FileMaker Pro	Definition in FileMaker Pro 7
302	Record is in use by another user	Data table is locked by another user
303	Paper size is in use by another user	Database schema is locked by another user
304	Password definitions are in use by another user	Layout is locked by another user

Error codes that are no longer used in FileMaker Pro 7

- 305 Relationship or value list definitions are locked by another user
- 411 Cannot perform delete because related records cannot be deleted
- 512 Field failed single value validation test
- 701 Data Access Manager can't find database extension file
- 702 The Data Access Manager was unable to open the session
- 703 The Data Access Manager was unable to open the session; try later
- 704 Data Access Manager failed when sending a query
- 705 Data Access Manager failed when executing a query
- 712 XTND version is incompatible
- 713 Couldn't initialize the XTND system
- 818 FileMaker Pro cannot network for some reason

Get(UserName) function

In previous versions FileMaker Pro, the Get(UserName) function uses settings from the Application Preferences (Preferences, in FileMaker Pro 7). This value can easily be modified by a user, so it can’t reliably be used to report activity associated with individual users.

In FileMaker Pro 7, the Get(AccountName) function is now available for files with accounts and passwords. This function returns the account name authenticated when the user logs in to the database with a valid account and password.

If your file uses `Get(UserName)` to track user activities, you may want to change Get(UserName) to Get(AccountName) after conversion to more accurately associate user activities with particular accounts. For more see “Get(AccountName) function” and “About accounts, privilege sets, and extended privileges” in the FileMaker Pro 7 Help.

Rotated fields

When you edit data in a rotated field, FileMaker Pro temporarily displays the field unrotated while you edit the data. FileMaker Pro 7 includes two changes to this behavior:

- The location of the temporary unrotated field is now at the top-left corner of the field object, which more clearly indicates the active field. In previous versions of FileMaker Pro, the location of the unrotated field is at the vertical center of the object.
- When you edit a rotated repeating field, FileMaker Pro 7 only unrotates the single repetition that is being edited. Earlier versions of FileMaker Pro would temporarily unrotate the entire field (all the repetitions), which could cause display problems in certain cases.

In some converted files, you may need to make adjustments to rotated field objects in Layout mode.

Scripts

Many changes have been made to ScriptMaker™ in FileMaker Pro 7. The following sections describe changes during conversion that can affect scripts in files created using previous versions of FileMaker Pro.

Conversion of script steps

During conversion, if any of the following script steps exist in scripts defined in your file, they are converted as described in the following table.

Script step in previous versions of FileMaker Pro	Script step equivalent in FileMaker Pro 7
Close	Close File
Copy All Records	Copy All Records/Requests
Copy Record	Copy Record/Request
Exit Record/Request	Commit Records/Requests
Insert from Last Record	Insert from Last Visited
New	New File
Omit Multiple	Omit Multiple Records
Open	Open File
Open Application Preferences	Open Preferences
Open Define Fields	Open Define Database
Open Define Relationship	Open Define Database
Open Document Preferences	Open File Options
Open Hosts	Open Remote

Script step in previous versions of FileMaker Pro	Script step equivalent in FileMaker Pro 7
Quit Application (Mac OS)	Exit Application
Recover	Recover File
Relookup Contents	Go to Field plus Relookup Field Contents
Replace Contents	Go to Field plus Replace Field Contents
Send Apple Event	Send Event
Send Message (Windows)	Send Event
Show Message	Show Custom Dialog
Show Omitted	Show Omitted Only
Sort	Sort Records
Toggle Status Area	Show/Hide Status Area
Toggle Text Ruler	Show/Hide Text Ruler
Toggle Window	Adjust Window

Script step behavior

The following sections describe changed behavior in specific script steps and changes that take place during file conversion.

Go To Layout script step

- The Refresh option has been removed from the Go To Layout script step. During conversion, the Refresh Window script step is added to scripts that previously included this option.

Open script step

The Open script step (changed to Open File during conversion) no longer opens a remote file unless you have specified the file reference in FileMaker network path (FMNET) format (fmnet:/*host IP address/filename*). If you have used the Open script step to open remote files, change the file reference to FMNET format or substitute the Open Remote script step for the Open script step after conversion.

- The Open File script step does not convert files created using a previous version of FileMaker Pro to FileMaker Pro 7 format. Use the Convert File script step to open files created in FileMaker 3.x, 4.x, 5.x or 6.
- In previous versions of FileMaker Pro, a script that contained only an Open script step activated the file that was opened. If any script steps followed the Open script step, the file that called the script became active.

  In FileMaker Pro 7, the file opened by the Open File script step remains active while the script runs. The following calculation will simulate the behavior of previous versions of FileMaker Pro, bringing the window that called the script forward:

  ```
  Open File [“<filename>”]
  Select Window [Current Window]
  ```

  Be sure to verify that the proper layout is activated when you test your script.

- Windows: The Open Minimized option in the Open File script step has been changed to Open Hidden.

Perform Find script step

During conversion, options for the Perform Find script step are changed. The new options differ depending on whether or not the scripts stores found sets.

When a script stores found sets, options are changed as follows:

Option name in previous versions of FileMaker Pro	Option name in FileMaker Pro 7
Perform Find [Restore, Replace Found Set]	Perform Find [Restore]
Perform Find [Restore, Constrain Found Set]	Constrain Found Set [Restore]
Perform Find [Restore, Extend Found Set]	Extend Found Set [Restore]

When a script doesn’t store find results, options are changed as follows:

Option name in previous versions of FileMaker Pro	Option name in FileMaker Pro 7
Perform Find [Replace Found Set]	Perform Find [ ]
Perform Find [Constrain Found Set]	Constrain Found Set [ ]
Perform Find [Extend Found Set]	Extend Found Set [ ]

Perform Script script step

- The Perform sub-script option in the Perform Script step has been removed from FileMaker Pro 7. When you use the Perform Script step in FileMaker Pro 7, any included subscripts are always executed.
- When necessary, previous versions of FileMaker Pro would automatically change the active window after running a script. This automatic windowing behavior is no longer supported because FileMaker Pro 7 supports a new Select Window script step, which script writers can use to set the active window.

  To emulate the behavior of previous versions, during file conversion FileMaker Pro 7 automatically inserts Select Window script steps after certain Perform Script and Go to Related Record script steps. However, to achieve the windowing behavior you actually want, you may need to insert additional Select Window script steps, or even remove some of the Select Window script steps that were inserted automatically during file conversion.

Go to Related Record script step

- During conversion, the relationship name specified for the Go To Related Record script step is replaced with the related table occurrence that was also created during conversion. Current layout is always set as the option for the Show records using layout parameter. If the previous relationship pointed to a table occurrence in another file, then Use external table’s layouts is also selected. If Show only related records was set in the original file, it is also set in the converted file.
- After each Go to Related Record script step that refers to an external layout and is not followed immediately by another external Go to Related Record step, FileMaker Pro 7 either:

- adds a Select Window[Current Window] script step, or
- adds a Select Window["name of external file"] script step if this is the last step in the script (not counting Comment, End If, and End Loop).

Refresh Window script step

The Bring To Front option in the Refresh Window script step has been removed from FileMaker Pro 7. During conversion, the Select Window script step is added to scripts that previously had this option specified.

Revert Records/Requests script step

Changes to related records are committed (saved) differently in FileMaker Pro 7 than they were in previous versions of FileMaker Pro, so reverting behavior has changed as well. See the FileMaker Pro 7 Help for more information on committing records.

Save a Copy As script step

During conversion, the Compressed (smaller) option is changed to Compacted (smaller). Compacted (smaller) removes unused sections of files, which can include deleted records. Activities due to normal operation of a file may reclaim the space that was previously used for deleted records.

Toggle Text Ruler script step

The Refresh Window option has been removed from the Toggle Text Ruler script step. During conversion, the Refresh Window script step is added to scripts that previously specified this option.

Import Records and Export Records script steps

During conversion, field-based options for the Import Records script step are converted to calculation-based options in FileMaker Pro 7. If you are importing from an ODBC source, query syntax is displayed in the Specify ODBC Query dialog box when the script runs.

After conversion, if you import from an XML source, be aware that field-based calculations are evaluated when the script runs and the results (which can include user name and password) can appear in the HTTP edit box. To suppress the display of this information, use the No dialog option with the Import Records script step.

The Import Records and Export Records script steps that were web-compatible in FileMaker Pro Unlimited are not web-compatible in FileMaker Pro 7. Instead, a web-compatible subset of script steps are available for web-published database hosted by FileMaker Pro 7 (FileMaker Networking) or FileMaker Server 7 Advanced.

Print Setup and Print script steps (Windows)

Printer settings may not convert properly in Windows. After conversion, check settings in scripts that use the Print Setup and Print script steps. For more information, see “Printer settings (Windows)” on page 40.

Miscellaneous scripting changes

The following changes have been made to scripting functionality.

Script behavior changes due to multi-table support

Because previous versions of FileMaker Pro supported only one table per file, by default scripts were executed on the "current" table, which was the table from the current file. FileMaker Pro 7 supports multiple tables per file and each layout is specifically associated with a "base table." (When you define the layout, the base table is set in the Show records from tablename menu.)

Each window is associated with a specific layout and because multiple windows may be open, if a script doesn't reference a window in a Select Window script step or a layout in a Go To Layout script step, the script executes on the current window (window in the foreground), which determines the associated table. If there is no window currently open, the script opens a window for the current table.

For more information, see "Functions that are dependent on context" on page 52.

Script names have been shortened

In FileMaker Pro 7, script names are limited to 100 characters. During conversion, any script names over this limit are truncated. Shortened names have no effect on script functionality.

If a script name contains more than 100 characters and is referenced externally (such as from an AppleScript or via a URL using the -script query string parameter), you should update the external references to the script to reflect the new script name or refer to the script ID.

Focus when displaying sorted data in found sets

In previous versions of FileMaker Pro, the focus of sorted data in a found set was the first record in the found set. In FileMaker Pro 7, the current record is the focus, so the data displayed after a sort operation called by a script could differ from the data displayed by previous versions.

To simulate the behavior of previous versions of FileMaker Pro, a Go To Record/Request/Page [First] script step is now placed after every Sort script step when a file is converted.

Sorting empty found sets

In previous versions of FileMaker Pro, using the Sort Records, Omit Records, or Omit Multiple Records script steps to sort an empty found set returned error code 3, "Command is unavailable." In FileMaker Pro 7, sorting an empty found set no longer returns an error.

Locking records when scripts run

In FileMaker Pro 7, making a field active does not lock its record. For example, if you used the Go To Field script step in previous versions of FileMaker Pro to lock records, records won't be locked when your script executes in FileMaker Pro 7. After conversion, you can use the Open Record/Request and Commit Records/Requests script steps in such scripts to achieve the result from previous versions of FileMaker Pro.

Example

```
Open Record/Request
Set Field [x, "abc"]
Set Field [y, "abc"]
Set Field [z, "abc"]
Set Field [xxx, "abc"]
Set Field [yyy, "abc"]
Set Field [zzz, “abc”]
Commit Records/Requests[without dialog]
```

For more information, see “Open Record/Request script step” in the FileMaker Pro 7 Help.

Using scripts to create related records in portals

In FileMaker Pro 7, when you create a related record in a portal via a script, you must create the related record first by entering a non-related value before specifying a related value.

Example

Suppose you have two related tables: TableA and TableB. TableB has a layout with a portal showing related records from TableA and you want to use a script to create a new record in the portal and enter a related value from TableB.

In previous versions of FileMaker Pro: You could use the following script to achieve this:

```
Go to Portal to Portal Row [Last]
Set Field ["TableA::data_field"; "TableB::data_field"]
```

In FileMaker Pro 7: This script is invalid as no reference exists to the related data you are attempting to enter until after the related record has been created. This script will create a new related record, but the record will not contain the related data you want to enter. The following script could be used instead:

```
Go to Portal to Portal Row [Last]
Set Field ["TableA::data_field"; "null"]
# 'null' is temporary data
Go to Field [TableA::data_field]
Set Field ["TableA::data_field"; "TableB::data_field"]
```

This script creates the related record with non-related data (“null”) first, then replaces the null value with the related data you specified in TableB (TableB::data_field).

Pause/Resume time is converted to seconds

Previous versions of FileMaker Pro allowed you to specify a pause time length in hours, minutes, and/or seconds. During conversion, pause time durations are converted using the Time function, which returns time values in hours, minutes, and seconds.

Script steps that reference files

If your database contains any script steps that reference files on remote computers (for example, Open File or Open Define Database), make sure that the file references are in FileMaker network path (FMNET) format (fmnet:/*host IP address/filename*).

Capturing error codes in scripts

If you have used the Status(CurrentError) function to capture error codes in scripts, be aware that some codes from previous versions of FileMaker Pro have been reassigned to a different error condition or made obsolete. For a complete list of FileMaker Pro 7 error codes, see “Get(LastError) function” in the FileMaker Pro 7 Help.

Some script step options convert to calculations

Many script steps in FileMaker Pro 7 now allow you to derive script step options from the result of a calculation. For example, the options for the Send Mail script step can now be text, a field value, or the result of a calculation. During the conversion of these script steps, the text and field options convert to calculations that return the required text or field value. Though formatted as a calculation, these options return the same results, and their script steps should function the same as they did in the previous version.

Performing a script when a file opens

Although startup scripts are typically performed the first time you open a file, these scripts can also be triggered when you open a window for a hidden file previously opened via a script or relationship.

To ensure that a startup script is performed just once per database session, preface your startup script with a conditional script step, and use a global field to record this condition once the script has been performed. Reset this condition with a script that runs when you close the file.

Performing a script when a file closes

In the File Options dialog box, you can specify a script to run when the file is closed. This script is performed when the last open window of the file is closed. The script will run even if the file itself remains open, for example if it is referenced by another file with open windows. The closing script will be performed each time the last open window of the specified file is closed.

ScriptMaker support improved in multi-user/multi-session situations

ScriptMaker has been revised to better support multi-user/multi-session execution in FileMaker Pro and in files published to the web with FileMaker Server Advanced. Web compatibility of script steps can now be shown in the Edit Script dialog box. For more information, see FileMaker Pro 7 Help, the *FileMaker Instant Web Publishing Guide*, and the *FileMaker Server Advanced Custom Web Publishing Guide.*

Ensuring user access to subscripts in web-compatible scripts

A user’s ability to execute scripts and subscripts is now affected by data access and design privileges for scripts. After conversion, users may not have privileges to run all subscripts in scripts they are authorized to execute. You may need to consider setting main scripts to run with Full Access privileges so subscripts can be executed on behalf of the user logged in, or you may need to change user privileges so individuals can run appropriate subscripts.

The scripts steps in which access might be restricted to subscripts are:

Import Records
Export Records
Open File
Close File
Print
Send Mail
Send DDE Execute
Execute SQL
Send Event

For more information, see “Access privileges” on page 29 and “About accounts, privilege sets, and extended privileges” in the FileMaker Pro 7 Help.

Changes to databases that are published to the web

Support for web publishing has changed significantly in FileMaker Pro 7.

The following issues apply to files published using Instant Web Publishing or Custom Web Publishing:

- The FileMaker Pro Web Companion plug-in is no longer supported. Instant Web Publishing and Custom Web Publishing settings don’t convert and must be set up in the converted file.
- The URL syntax has changed for web-published solutions.
- If you manually created a link to access a FileMaker solution over the web, you must update the link after converting and sharing the file over the web.
- If a script name contains more than 100 characters and is referenced externally (such as from an AppleScript or via a URL using the `-script` query string parameter), you should update the external references to the script to reflect the new, shorter script name. For more information, see “Script names have been shortened” on page 60.
- Plug-ins written to work with FileMaker Pro 6 Unlimited (and earlier) and FileMaker Pro 7 do not automatically work when using web publishing with FileMaker Server Advanced and the Web Publishing Engine. Check with the plug-in vendor to see if a plug-in supports these environments. For more information about conversion issues and plug-ins, see “Plug-ins” on page 73.
- ScriptMaker has been revised to better support multi-user/multi-session execution in FileMaker Pro and FileMaker Server Advanced. For more information, see “ScriptMaker support improved in multi-user/multi-session situations” on page 62.
- More than 70 web-compatible script steps are supported by FileMaker Pro 7 and a few script steps, such as the Import Records and Export Records script steps, are no longer supported (see “Import Records and Export Records script steps” on page 59). Select Indicate web compatibility in the Edit Script dialog box to see which scripts are web-compatible in FileMaker Pro 7.
- The Web Security Database is no longer supported. If you relied on it for web-based security, you must set up the accounts, passwords, and privilege sets for users in your converted database files. For more information, see “Defining accounts and privilege sets for files protected by the Web Security Database” on page 67.

  For information about how web publishing privileges convert, see “Groups” on page 30.

- In previous versions of FileMaker Pro, the Status(CurrentWebSharing) function returns 1 when the Web Companion plug-in is enabled and active. In FileMaker Pro 7, this function converts to the following formula, which returns 1 if the script or calculation containing this formula is being evaluated for a user accessing the file via the web:

  (Position(Get(ApplicationVersion); "Web"; 1; 1) > 0)

  The results of calculations based on Status(CurrentWebSharing) will return slightly different information in converted files that now use the formula above. Calculations that relied on Status(CurrentWebSharing) were typically trying to determine if the database was being published to the web in an indirect manner; the results of this function indicated whether or not the database had been opened with a password that allowed data export and whether the Web Companion plug-in was enabled. The formula can determine specifically if a request to execute a script came via a web user request or another user action.
- Folders containing static content (such as images stored by reference in a container field) might need to be relocated when moving databases from FileMaker Pro to FileMaker Server Advanced. Folders located under the Web folder need to be moved to the appropriate location on the IIS or Apache HTTP server. For more information, see the *FileMaker Server Advanced Web Publishing Installation Guide*.

Instant Web Publishing

Setup has changed

The setup and user access of files published to the web with Instant Web Publishing has changed in FileMaker Pro 7. Therefore, conversion is not possible and you need to set up web publishing again. For more information, see the *FileMaker Instant Web Publishing Guide*.

Custom Web Publishing

The following changes have been made to Custom Web Publishing. For more information, see the *FileMaker Server Advanced Custom Web Publishing Guide*, included with FileMaker Server Advanced.

FileMaker Server Advanced-based versus client- based

You need FileMaker Pro 7 to create the databases and set up the extended privileges for Custom Web Publishing. You need FileMaker Server 7 Advanced and the Web Publishing Engine to host databases and make them available on the web.

CDML is no longer supported

In previous versions of FileMaker Pro, Custom Web Publishing required that databases use format files written in the FileMaker CDML mark-up language, which dynamically generated HTML-based pages for web users. FileMaker Server 7 Advanced supports Custom Web Publishing of FileMaker Pro 7 databases with server-based XSLT style sheets, which filter and format XML data. XML can generate HTML, which can be viewed using a web browser or transformed to XML grammar for use by other programs and applications.

You can use the FileMaker CDML Converter, included with FileMaker Server Custom Web Publishing, to change CDML format files to XSLT stylesheets after you convert your databases to FileMaker Pro 7.

Inbound links need to be modified

Pages within your web site that reference (link to) Custom Web Published solutions need to be updated after file conversion, including URLs that directly reference the published database. Links must be updated to use the new URL syntax and the new request query parameters. Be sure to check scripts in other FileMaker Pro databases that launch a browser to access a Custom Web Published database or that use XML Import via HTTP to bring data from a web-published database into a local database.

Movies in FileMaker Pro 3.x or earlier

If a movie file that was imported using FileMaker Pro 3.x or earlier is encountered during conversion, it is changed to an embedded image in the converted file.

Schedule settings in FileMaker Server

When you open converted files using FileMaker Server 7, no schedule information defined in previous versions of FileMaker Server is retained. You must manually redefine schedules in FileMaker Server 7.

Runtime solution files bound by the FileMaker Developer Tool

Converted files bound by a previous version of FileMaker Developer (for example, as a runtime solution or set to Kiosk mode), must be rebound using FileMaker Developer 7. Contact the solution developer for rebound copies of your solution files.

Important You should not convert a bound runtime solution unless you have the master password, because you won't be able to make modifications to the file after conversion. For more information, see "Access privileges" on page 29.

Chapter 5
Defining accounts and privilege sets for files protected by the Web Security Database

The Web Security Database released with previous versions of FileMaker Pro is not supported by FileMaker Pro 7. User definitions and field restrictions formerly stored in the Web Security Database must be set up in converted database files you plan to publish on the World Wide Web. There are two ways to transfer user information stored in your Web Security Database:

- You can manually create accounts and assign access privileges in a converted file.
- You can use ScriptMaker to create accounts in a converted file based on information in an existing Web Security Database and then assign privileges manually. In general, if you need to create accounts for more than 25 users and you are comfortable using ScriptMaker, it makes sense to automate account creation using a script.

Considering security for FileMaker Pro 7 databases

FileMaker Pro 7 introduces many changes and improvements to database access and security. For information about how database access has changed in FileMaker Pro, see “Access privileges” on page 29. Before you transfer information from your Web Security Database, consider whether you can take advantage of security improvements offered by FileMaker Pro 7.

To evaluate current security options:

1. Learn about security options and recommendations for FileMaker Pro 7. For information, see:
   - “Protecting databases with accounts and privilege sets” in FileMaker Pro 7 Help.
   - the *FileMaker Security Guide*
2. Examine the user permission settings in your current Web Security Database and plan for setting up equivalents in FileMaker Pro 7.

   Previous versions of FileMaker Pro defined user permissions by password and group. In FileMaker Pro 7, all access permissions are defined in accounts and privilege sets. For more information about how to plan file access in FileMaker Pro 7, see the next section.
3. Think about user access privileges that you will need going forward and determine if security improvements in FileMaker Pro 7 allow you to define them now.

Planning secure access for files published on the web

For each database file you plan to convert:

1. Analyze the list of users and user permissions defined for the file in your Web Security Database.

   An easy way to list user records is to view all records as a table. Open the Web Users_ file, choose View menu> View as Table, and resize columns to display all information for each user.

   Tip You can print the list by choosing File menu > Print.
2. Analyze all of the field restrictions defined for the file in your Web Security Database.

Again, you can view field restriction records as a table and print it. Open the Web Fields_ file, choose View menu > View as Table, resize columns to display field restriction settings for each file defined in Web Fields_, and choose File menu > Print.

3. Examine the existing user records in your Web Security Database.
   - Review existing user records and delete any that are no longer in use.
   - Determine why user records were set up this way and how they are being used by the database. For example, are settings for multiple users identical? If so, the collection of users represents a "group" in previous versions of FileMaker Pro, which can be translated into a privilege set shared by separate users in FileMaker Pro 7.
   - Decide how many unique accounts you will need.
   - Consider the number of accounts you had and decide if it makes sense to create them manually in converted files or to use ScriptMaker to create accounts. If you need to create accounts for more than 25 users and you are comfortable using ScriptMaker, it makes sense to automate account creation using a script.

   Note ScriptMaker can create accounts and assign privilege sets for you, but you must create privilege sets in FileMaker Pro 7 manually before you run the script.

4. Examine the existing field restrictions defined for each file in your Web Security Database.

   Field restriction settings need to be defined as access privileges in converted databases.

5. Make a comprehensive list of the new privilege sets you need for all users.
   - Identify the specific access rules needed for each privilege set. The new access privileges in FileMaker Pro 7 offer an integrated security method, without requiring maintenance of the separate Web Security Database. You can restrict access by table, by record, by layout, on value lists, on scripts, on how files are shared (extended privilege sets), or by certain features such as printing, exporting data, or requiring users to change passwords regularly (other privileges). For more information, see "Creating new privilege sets" in FileMaker Pro 7 Help.

   Note Access privileges apply to one file, so all tables need to be located in one file. If you have converted a relational file, you may want to combine tables into one file prior to setting up accounts and privilege sets.

   - Try to come up with privilege sets that can be shared by users who have the same access restrictions. You will need separate privilege sets if you need to restrict access to certain areas of a database for some users and not for others.

   Note The field restrictions Don't Search and Exact Update that were available in the Web Security Database are not supported in FileMaker Pro 7.

6. Create a comprehensive list of all users that assigns the required privilege sets to each user account (user account and privilege worksheet).

   Important Because this worksheet contains complete user log in information, it's important to keep it confidential.

7. Decide whether you are going to create accounts manually (see the next section) or use a script to create accounts in your converted database file (see "Using ScriptMaker to create new accounts").

Creating new accounts manually

To create new accounts manually:

1. Complete the tasks listed in “Planning secure access for files published on the web” above.
2. Follow the instructions in this guide to convert your database file.

   For more information, see “Converting your database files” on page 17.
3. Create the required privilege sets you identified during planning.

   For more information, see “Creating new privileges” in FileMaker Pro 7 Help.
4. Create an account for each user listed in your user account and privilege worksheet.

   For more information, see “Creating accounts” in FileMaker Pro 7 Help.
5. Test user accounts and privilege sets before making the converted file available to users. See the *FileMaker Instant Web Publishing Guide* for testing suggestions.

Using ScriptMaker to create new accounts

Note A script can create accounts in your database file and assign privilege sets to them, but you must create privilege sets manually.

To use a script to transfer account information from your Web Security Database:

List users and assign privilege sets.

Complete the tasks listed in “Planning secure access for files published on the web” on page 67.

Convert your database and your Web Security Database to FileMaker Pro 7 format.

Follow the instructions in this guide to convert your file and the Web Security Database. For more information, see “Converting your database files” on page 17.

Set up your database to accept user information from the Web Security Database.

Before you can bring over account information stored in your Web Security Database, you need to make the following changes to your converted database file:

- Define a file reference to the Web Users_.fp7 file. This file reference will appear as a table occurrence in the relationships graph in your converted file.
- Add a layout named Web Users_ that shows records from the Web Users_.fp7 table occurrence. This layout helps you verify the information the script transfers from Web Users_.fp7.
- Create the privilege sets in your database file that you will need to assign to the new accounts after you run the script.

1. Open your converted database file, choose File menu > Define > Database, then click the Relationships tab to display the relationships graph.
2. Click Add Table .
3. In the Specify Table dialog box, click the File list, choose Add File Reference, locate and select Web Users_.fp7, click Open, and then click OK to close the Specify Table dialog box.

   A table occurrence for Web Users_.fp7 appears in the relationships graph.
4. Click OK to close the Define Database dialog box.

5. Use the New Layout/Report Assistant to create a new layout in your database file. Click Move All to create fields on the layout for all fields in Web Users_.fp7.

   For information about creating layouts, see “Creating a layout” in FileMaker Pro 7 Help.

6. Create the privilege sets you identified during planning.

For more information, see “Creating new privileges” in FileMaker Pro 7 Help.

Create a script to transfer user information from Web Users_.fp7

Write a script to create accounts in your database file based on the user records in Web Users_.fp7.

For more information about scripting, see “Creating and editing scripts using ScriptMaker” in FileMaker Pro 7 Help.

Example script

The following script navigates to the first record in Web Users_.fp7, then creates an account in your database file for every user record in Web Users_.fp7. When information from all records has been transferred, the script returns to the layout displayed before it ran and informs you that it is finished.

Note This script assigns the Read Only Access privilege set to all accounts. If a large percentage of users share the same privilege set, you could name that privilege set in the script and manually modify privileges for the accounts that require a different privilege set after you run the script. Or, you could modify the script adding If and Else script steps to assign the correct custom privilege sets to each account. If you plan to assign custom privilege sets using a script, you must define them before you run this script.

```
Go to Layout ["Web Users_" (Web Users_)]
Go to Record/Request/Page
   [First]
Loop
   Add Account [Account Name: Web Users_::User Name; Password:
     Web Users_::Password; Privilege Set: "[Read-Only Access]"]
   Exit Loop If [Get (RecordNumber) = Get (FoundCount)]
   Go to Record/Request/Page
     [Next; Exit after last]
End Loop
Go to Layout [original layout]
Beep
Show Custom Dialog [Message: "Done."; Buttons: "OK"]
```

Run the script to transfer user information from Web Users_.fp7

1. Run the script to create accounts in your database file.

   Note Account names must be unique. If you have duplicate account names defined in your Web Security Database, FileMaker Pro displays an error when this script runs and does not create duplicate accounts.

2. If you used the script above without modifying it to assign specific privilege sets, use the user account and privilege worksheet you created earlier to assign the correct privilege set to each account.

   For more information, see “Editing existing accounts” in FileMaker Pro 7 Help.

3. Test user accounts and privilege sets before making the converted file available to users. See the *FileMaker Instant Web Publishing Guide* for testing suggestions.

Tip The example provided here shows one method of using ScriptMaker to automate account creation. Depending on how your database is used, you might consider letting users create their own accounts via a script when they log in the first time. By using the Add Account script step and global text fields on a designated layout, you can create a script that allows users to add an account assigned to a default privilege set you define. Afterwards, you could reassign privilege sets, if needed. For more information about scripting, see “Creating and editing scripts using ScriptMaker” in FileMaker Pro 7 Help.

Chapter 6
Considerations for associated files and interfaces

If the database you are converting uses associated files and technologies (such as plug-ins), be aware of the following changes to how these technologies interact with FileMaker Pro 7.

Security improvements require reconfiguration of associated files

External interfaces, Apple events, and ActiveX are directly affected by the new security model in FileMaker Pro 7 and must be reconfigured after file conversion to connect to a FileMaker database using both account and password information. For more information, see “Access privileges” on page 29 and “About accounts, privilege sets, and extended privileges” in FileMaker Pro 7 Help.

Starting FileMaker Pro using a URL

You use a URL to start FileMaker Pro and open a shared remote database (for example, in a hyperlink on a web page or in a FileMaker Pro script that includes the Open URL script step). The syntax for starting FileMaker Pro via URL has changed in FileMaker Pro 7 and now requires you provide both account and password information.

Example

In previous versions of FileMaker Pro	In FileMaker Pro 7
`FMP5://secret@192.168.10.0/` `My+Addresses.fp5`	`FMP7:MyAccount:MyPassword@192.168.10.0/` `My+Addresses.fp7`

For more information, see “Opening shared databases remotely using a URL” in FileMaker Pro 7 Help.

References to objects whose names were changed during conversion

During conversion, object names (such as relationship names) that use reserved words, or field names that use invalid characters are modified slightly so they will evaluate properly in FileMaker Pro 7. If renamed objects are referenced by files that interact with a converted database, the object name must be modified in the reference as well. For more information, see “Reserved words” on page 47 and “Invalid characters and words in field names used in calculations” on page 47.

Plug-ins

Ensure external functions and plug-ins have been verified for use with FileMaker Pro 7, FileMaker Server 7, or the FileMaker Server Advanced Web Publishing Engine. The same plug-ins that work with FileMaker Pro 6 in Mac OS X or Windows 2000 should be compatible with FileMaker Pro 7, though they will not take advantage of new features in version 7.

If you acquired plug-ins that were developed by a third-party vendor, contact the vendor for detailed compatibility information, or to find out if the vendor has a new version that takes advantage of FileMaker Pro 7 features.

If you are a plug-in developer, verify plug-in behavior and functionality with FileMaker Pro 7.

- Plug-ins that depend on more than the plug-in API compatibility may not function properly with FileMaker Pro. For example, in some cases plug-ins may attempt to use undocumented interfaces and make assumptions about behaviors that have changed in this version. For information about developing plug-ins for FileMaker Pro 7, see the *FileMaker Developer Developer's Guide*.
- Plug-ins written to work with FileMaker Pro 6 Unlimited (and earlier) and FileMaker Pro 7 do not automatically work when using web publishing with FileMaker Server Advanced and the Web Publishing Engine. Check with the plug-in vendor to see if a plug-in supports these environments.
- Accelerated evaluation in FileMaker Pro 7 could produce errors in files created with previous versions of FileMaker Pro if a plug-in expects complete evaluation. For more information, see "Accelerated calculation evaluation" on page 50.
- For information about where to install plug-ins required for web publishing in FileMaker Pro 7, see the *FileMaker Server Advanced Web Publishing Installation Guide.*

LDAC and RDAC changed to ODBC/JDBC and FileMaker Server Advanced

Sharing information with other applications (Windows)

The Local Data Access Companion (LDAC) and the Remote Data Access Companion (RDAC) are no longer supported. In addition, database access is now controlled through accounts, privilege sets, and extended privilege sets rather than groups and passwords.

If a database you are converting used LDAC or RDAC to share information from an ODBC data source, sharing settings and user group and password settings need to be redefined manually in FileMaker Pro 7 for use with ODBC/JDBC.

Note FileMaker Pro 7 support for ODBC/JDBC allows access to your database on the local computer. If you used RDAC to access information from data sources on computers over a network, you now need FileMaker Server 7 Advanced. For more information, see the *FileMaker Server Administrator's Guide*.

After conversion:

- Enable the database for sharing via ODBC/JDBC (Edit menu > Sharing > ODBC/JDBC).
- Make sure user accounts that access the database have the extended privilege Access via ODBC/JDBC.
- If your database was previously protected with a password, you need to update the application using ODBC/JDBC to access your database using an account name and password.

For more information, see "Access privileges" on page 29 and "Sharing FileMaker Pro data via ODBC or JDBC (Windows)" in the FileMaker Pro 7 Help or the *FileMaker ODBC and JDBC Developer's Guide*.

Accessing ODBC data sources

Drivers used with previous versions of FileMaker Pro to access information from a data source should work properly with FileMaker Pro 7. However, users won't be able to use updated ODBC drivers for FileMaker Pro 7 unless you redefine your data sources to use your new driver. For more information, see "Accessing ODBC data sources using FileMaker Pro" in the FileMaker Pro 7 Help or see the *Installing FileMaker ODBC and JDBC Client Drivers* guide. If your database uses a script to import information from an ODBC source, see "Import Records and Export Records script steps" on page 59.

ActiveX library renamed

ActiveX solutions should convert and function properly to FileMaker Pro 7. The only significant change in FileMaker Pro 7 is that the ActiveX library has been renamed from FMPRO50Lib to FMPRO70Lib. After conversion, you must manually create an instance of FileMaker Pro in Visual Basic as follows:

```
Dim FMProApp As FMPRO70Lib.Application
```

Note To support the new access privilege model in FileMaker Pro 7, the `Documents.Open(filename As String, password As String)` method has been changed to `Documents.Open(filename As String, accountName As String, password As String)`. If the accountName and password parameters are empty, the file is opened by the Guest account.

Apple event changes in FileMaker Pro 7 (Mac OS)

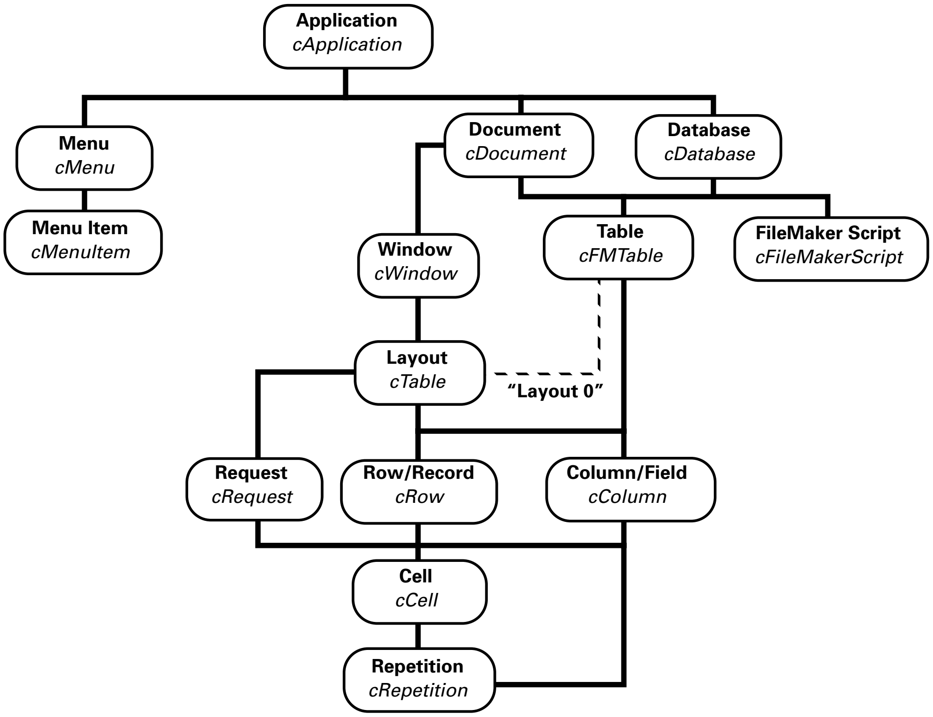
FileMaker Pro 7 has changed its Apple event support to accommodate multiple database tables per file and other new features. In general, because all FileMaker Pro 6 and earlier files are converted as single table files, an AppleScript created to work with FileMaker Pro 6 files will continue to work once those files have been converted to FileMaker Pro 7. However, you will probably need to modify existing AppleScripts as database files are modified to use multiple tables and other new features in FileMaker Pro 7.

For a complete description of Apple event support in FileMaker Pro 7, see the FileMaker Pro Apple Events Reference database, located in FileMaker Pro 7/English Extras/Apple Events/ Apple Events Reference.fp7.

The following is a summary of the major changes to Apple event support in FileMaker Pro 7.

New object model

FileMaker Pro 7 uses the following object model for Apple events:



FileMaker Pro provides *default containment*, a set of default conditions to make scripts shorter and to determine which objects to use if none are specified.

When this object has not been specified	FileMaker Pro substitutes this object
layout	current table
table	table 1
record 1 of document 1	record 1 of table 1 of document 1

Changes to functionality

New "table" object

A new object, `table`, has been added to the object model to support the multiple database tables per file that are possible in FileMaker Pro 7. This new table object has the constant value `cFMTable`.

The fields in the table object `cFMTable` correspond to the actual fields defined in the table. Fields are expressed as belonging to the object `cColumn.` The records `cRow` correspond to the total number of records in the same table. However, the fields in the layout object `cTable` correspond only to the fields `cColumn` defined in that layout and the records `cRow` correspond to the number of records in the found set for that layout.

Earlier versions of FileMaker Pro (version 6.x and earlier) defined a special layout, `layout 0`, a layout with a 0 index, to give the scripter access to the single table for a database. FileMaker Pro 7 still supports the layout 0 concept for backward compatibility, but layout 0 can only be used to represent table 1 of the database.

Using 0 as the index for layouts generates an object specifier to table 1 of a FileMaker Pro database. The value of the `pKind` property of a `cFMTable` object is `table`, and `view` for `cTable` (layout). To access fields of other tables, use the table object `cFMTable` instead. For clarity and forward compatibility, it is recommended that you convert existing scripts to use the table object instead of layout 0.

The `cColumn` elements of a `cTable` object correspond to the fields appearing on a FileMaker Pro layout, while the `cColumn` elements of a `cFMTable` object correspond to the fields as they appear in the Define Database dialog box for that table. To view fields for a table, choose File menu > Define > Database, click the Fields tab, and select the table from the table pop-up menu.

Note `cTable`, the constant value for layout objects used in the previous versions of FileMaker Pro, is still supported. The mapping of the layout object to `cTable` has not changed.

New "window" object

FileMaker Pro 7 supports the ability to view multiple windows per document. When you reference a window by index (`window 1`, `window 2`, and so on), then windows will be considered to be organized according to their order on screen. For example, if `window 1` is the foreground window, `window 2` is the window behind the foreground window, and so on, regardless of the document to which they are related.

To reference windows according to their relation to a specific document, use the syntax `window 1 of document 1`, where window 1 of document 1 is the first window created for document 1.

It is not necessary to specify the window object if you are using the current record and current layout properties. For example, a statement such as `tell application "FileMaker Pro" to get current record` will return a reference to the current record from the foreground window.

New "repetition" object

A single cell can contain multiple repetitions. The `repetition` object provides more control over data accessed from the database.

`Repetition 1 of cell 1 of window 1` represents the first cell displayed on the window's current layout. For example, if a cell has five repetitions, and that cell's layout format is set to display only repetition 2 to 4, then `Repetition 1 of cell 1 of window 1` will get the data of repetition 2. To set data from repetitions that are not currently visible, use the syntax `Repetition 1 of cell 1 of table 1`, which in this case will return the data of repetition 1.

Field order changes

For Apple events, the field order of a table `cFMTable` is the field creation order. (To view the field creation order for a table, choose File menu > Define > Database, click the Fields tab, select the table from the table pop-up menu, and choose View by creation order.) In contrast, the fields order of a layout `cTable` is the graphical order in which the fields appear on the layout, moving from top left to bottom right.

If an application issues a `Get Data` event on a `record of layout`, `request of layout`, or `layout` statement, the data will be returned using the graphical order. And if an application issues a `Get Data` event on a `record of table` or `table` statement, the data will be returned using the creation order. If the field type is date, time, timestamp, or number, the data will be returned as formatted on the layout (as set in the Date, Time, and Number Format dialog boxes).

When using the `Set Data` event with a record, request, table, or layout object, the data should be supplied in the same order it would be retrieved if using a `Get Data` event on that object.

Printing changes

The Apple events printing behavior for FileMaker Pro 7 has been changed to completely suppress the user interface. To provide support for printing, three optional parameters have been added to the printing command:

```
from page, to page, with copies
```

If any of these optional parameters are not specified with the command, FileMaker Pro 7 uses the default values saved with the file.

Index